

**Comm-Pro Associates
Host Network Access Support**

**HNAS
Changes & New Features
Reference**

All VnRnMn Releases

(Information is also available in Appendix D of the HNAS Guide and Reference Manual)

Please see next page for important
information concerning your
Host NAS distribution materials.

This file revised November 8, 2012 9:33 am, includes maintenance thru 2400111.

General Information

Comm-Pro Associates is the designer and developer of the X.25 Host Network Access Support host resident Program Product (Commonly referred to as **Host NAS** or **HNAS**). The Host NAS product provides support for X.25 connectivity over router based networks using IBM's X.25 through TCP/IP (**XTP**) or Cisco's X.25 over TCP/IP (**XOT**) transport protocols. The HNAS implementation was designed to avoid application changes by providing a transparent migration from IBM 3745 NSPI based communication controllers to the HNAS router based solution. The product offering provides a robust suite of non-SNA NPSI type support. Please refer to Chapter 1 (Introduction) for a complete list of products and Applications supported.

Contact Information

Phone: (661) 284-3650

Fax: n/a

E-mail: support@comm-pro.com

FTP: www.comm-pro4ftp.com
(Userid required, available upon request)

WWW: www.comm-pro.com

Shipping Address: 25852 McBean Parkway #611
Santa Clarita, CA USA 91355-3705

For additional documentation and up-to-date information, please refer to member @README in the Comm-Pro distribution macro library. See our WEB site for the latest information.

Important Notes

1) Please refer to the optional README/@README file included with the HNAS product distribution media (separate file or HNASMAC macro member) for additional product information and documentation not included in this manual. Additional information can also be located on our web site (Please refer to Contact Information section for contact details).

(C) Copyright Comm-Pro Associates 2006

HNAS Summary of Changes & New Features (by VnRnMn level)

This Summary of Changes & New Features List by Release section serves as a quick reference to identify product features and enhancements as provided for recent HNAS VnRnMn release levels. Please refer to the specific 'HNAS VnRnMn - New Features Overview' section for a description of the summary items.

Additional information concerning operational changes for HNAS VnRnMn product refresh or upgrade activity can be found in Chapter 5 (Migration section) of the HNAS Guide and Reference Manual.

Host NAS Summary of Changes & New Features

This page left intentionally blank.

HNAS V2R4M0 Release Summary:

Functional Enhancements

- Improvements in product installation for organizations operating in **non-SMP/E** environments
- **SMP/E** installation and maintenance is now fully integrated into the standard documentation
- **ISARX25 Datafono** support integrated into the standard HNAS product (subscription option)
- **BUILD CONCMDQ=ddname (CMDLIST)** support allows console command list and control for various activities
- **BUILD PRTSWLST=(options)** provides improved SYSPRINT automation and controls
- **PARM='EOTKEY=dd...dd'** start parameter allows the TRIAL PERIOD EXPIRATION DATE (EOT-DATE) for a trial distribution to be extended using the key provided by the **dd...dd** digits
- **PARM='EOMKEY=dd...dd'** start parameter allows the MAINTENANCE/USE ANNIVERSARY DATE (EOMDATE) for a permanent distribution to be extended using the key provided by the **dd...dd** digits
- **PARM='EOMKEY=dd...dd'** start parameter also allows a trial distribution to be converted to a permanent distribution dynamically using the key provided by the **dd...dd** digits
- **PARM='PARMFILE=pfddname'** start parameter allows all or some HNAS start parameters to be provided in a sequential file
- **REMOTE TYPE=DMY** added permitting future expansion of some console command parms

Configuration Enhancements

- **ALRMFLTR=** table has been expanded from 16 to 128 entries allowing fine tuning of filters
- **ALRMFLTR=** now supports (FC) and (FU) message ID suffixes to allow forced conditional (subject to SHOW state) and forced unconditional (SHOW state ignored) SYSCONS display
- **CONCMDQ=** now includes the new **EXEC ddname** command list support
- **CONCMDQ=ddname** can now be specified to provide a dataset command list (provides consistency with the SCHEDULE=ddname operand)
- **EXEC=** operand has been added as an alias for CONCMDQ= (provides name symmetry with the EXEC command like the SCHEDULE= operand and SCHEDULE command)
- **EXEC=** operand has been modified to decode the new NOPURGEONERROR and PURGEONERROR keywords
- **FASTRUN AMNF** generation support now propagates VTAM parameters from the MCH to all GATE control session SLU APPL statements for the MCH
- **FASTRUN AMNF** generation support now propagates VTAM parameters from the MXT associated with a PVC SLU to the SLU APPL statement (MCH VTAM parameters used when MXT not coded)
- **INIT={ACTIVE|IDLE}** is now valid for TYPE=MCH REMOTE definition statements
- **INIT={ONLINE|OFFLINE}** is now accepted as alternate values for ACTIVE|IDLE, respectively
- **LUNAME=** operand has been modified to accept a **GENTYPE** value which is used to control hex versus decimal SLU name generation: LUNAME=(sluname-{A|}|pluname/pfxlu-{H|D}/sfxst/cnt,...)
- **OPTIONS=ALRMSGTXT={LONG|SHORT}** now supported on BUILD
- **OPTIONS=CUD0SELECTSLU** (CUD0 LU selection) now supported for the TYPE=MCH|MXT REMOTE
- **OPTIONS=DELAYBINDRESP** now supported for the TYPE=MCH REMOTE
- **OPTIONS=IDTST** (Datafono M message validation) now supported for TYPE=DFX REMOTE
- **OPTIONS=NOCLOSEONTAPFAILURE** now supported for the TYPE=XOT|XTP REMOTE
- **OPTIONS=PVCRECONTMR=secs, PVCSETUPTMR=secs** and **PVCSETUPREJ=code** now supported for the TYPE=MCH|MXT REMOTE
- **OPTIONS=RESETINO** now supported for the TYPE=MCH REMOTE
- **OPTIONS=SVCCALLTMR=secs** (T21=) now supported for the TYPE=MCH|MXT REMOTE
- **OPTIONS=(TIMESTAMP=time-format{+D})** on BUILD now provides SYSPRINT timestamp and datestamp controls
- **PARM= (EXEC)** operand now supports abbreviated start parameters reducing the likelihood that the operands 100 character limit will be exceeded

- **PARSE** definition statement support added permitting user defined CDF parsing characters
- **PRTSWLST**={LOOP|STOP,SWITCHAFTERINIT,SWITCHAT *time*,DYNAMIC=*x*} automation
- **PULSE**={*hh:mm:ss,hh:mm:ss,seconds*} operand added for heartbeat message support
- **REMOTE TYPE=DMY** permits predefined **PING (XOT)** parameters (all Call Request values)
- **REMOTE TYPE=SVC** added supporting new **SVC0|5**={(...<*rmtname*>)} enhancement (see below)
- **RTEIN= SKIP** and **CLEAR** parameters available for filter list placeholders and call clear control
- **RTEIN=** and **RTEOUT=** operands now includes basic support for DTE address 'wildcards' filtering
- **RTEOUT**={*rmtname/dteaddr/desctxt,...*} operand now accepts a descriptor text field
- **SCHEDULE**={*ddname(hh:mm:ss,cmd,...,hh:mm:ss,cmd)*} operand support added to allow scheduled console commands
- **sluname INIT** value now allowed for the LUNAME=, PVC=, SVC0=, SVC4= and SVC5= operands: syntax is *sluname*-{**A|I**} where A is for active (online) and I is for idle (offline)
- **SOCLMT=limit** operand now generates an error message if the sum of the TYPE=XOT REMOTE VCLMT= values (for which the LOCAL is HOME) is greater than *limit*-2 (eliminates runtime ABEND)
- **SVC0|5**={(*sluname*|*Xidnum*|*dteaddr*|<*rmtname*>)} MXT|SVC per *dteaddr*, *dteaddr* via *mxtname*
- **TRCLMT**=count expanded from 32767 to 1048575 entries permitting extended tracing capabilities
- **TRCTRAP=ALRMLIST=(NASnnnni' data'/data-offset,...)** operand now accept *data* as quoted string
- See [Console and Trace Enhancements](#) for **PARM=** Start Parameter Changes & Improvements
- **SVC0|5**={(*sluname+gluname*/...)} generic SLU name support added to MCH

Alert/Alarm Message and Clear/Reset Code Enhancements

- **NASCnnni** console command response messages were added improving command decode status
- **NASC052W** console command response message issued if EXEC LIST is specified without command list (*ddname* or (*cmd1*,...,*cmdn*) <- [NASC052W retired by APAR 2400098](#)
- **NASC053W** console command response message issued if EXEC *ddname* identifies an empty command list file
- **NASC100E** console command response message has been modified to more completely identify why a command that operates on the LNM=, RNM=, LUNM= or ID= modifiers is aborted when none of these modifiers are specified.
- **NAS003ni** messages identify the type of HNAS shutdown as well as the actual end
- **NAS020ni** messages generated for **PRTSWLST=** SYSPRINT switching support
- **NAS011ni** messages generated for **SHOW {ON|OFF|ERR}** console command activity
- **NAS0130W** messages generated for **SHOW {MORE|LESS}** console command activity
- **NAS0210I** messages generated for **PRNT NEXTPRSW|SWITCHNOW & SWITCHAT***type* activity
- **NAS021ni** messages generated for **PRNT {ON|OFF}** console command activity
- **NAS022ni** messages generated for **PRNT {OPEN|CLSOPN}** console command activity
- **NAS0230W** messages generated for **PRNT {DATE ON|OFF ...}** console command activity
- **NAS0299I** messages generated for **PULSE** heartbeat message
- **NAS0310W** messages generated for **VARY *mchname* |FORCE|OFF|ON** events
- **NAS1n11E** messages generated when incorrectly formatted table is specified for the USSTAB=, LOGTAB=, or OPTIONS=NRITAB= operand
- **NAS1nnni** configuration messages updated for new CDF parameters and options
- **NAS1154E** message generated when SCHEDULE= queue limit exceeded
- **NAS1301D** message is now issued when SLU initial value (-{**A|I**}) is omitted
- **NAS1311S** message is now issued when the sum of the TYPE=XOT REMOTE VCLMT values (for which the LOCAL is HOME) is greater than SOCLMT=*limit* value minus 2 (NEEDED=*count* added)
- **NAS1391I** messages now issued to display MXT operand overrides
- **NAS2109S** message is now issued when an **unexpected TCPIP interrupt** is presented and the DBUG TCP start parameter is in effect
- **NAS2110S** message is now issued when an **invalid TCPIP reply ID** is detected and the DBUG TCP start parameter is in effect

- **NAS2152E** message is now issued when a CANCEL command timeout occurs
- **NAS2502E** message now displays CLOSEONTAPFAILURE when a router contact is lost and the CLOSEONTAPFAILURE option is in effect
- **NAS2505E** message was added and displays NOCLOSEONTAPFAILURE when a router contact is lost and the NOCLOSEONTAPFAILURE option is in effect
- **NAS251nM** messages that are issued for **MON TAP** now provide information similar to **PING** console command activity
- **NAS2511M** message that is issued for **MON TAP** now contains a retry count
- **NAS261nM** messages now issued for **PING** console command activity
- **NAS261nM** messages updated for **PING** console command to display a sequence number so that termination messages can be easily correlated to initiation messages
- **NAS261nM** termination messages issued for **PING** console command now display the target IP and DTE addresses
- **NAS3796I** messages generated for start of a GATE FC session initiated by an inbound call
- **NAS3797I** messages generated for GATE control session activation (bound)
- **NAS5705W** messages for RESET SCHEDULED now provides **DIAGX=0000|0001** values
- **NAS5723W** messages issued for IDTST= ID String length exception
- **NAS5724W** messages issued for IDTST= 'M' messages response not received in 'nn' seconds
- **NAS5725W** messages issued for IDTST= Bad ID condition
- **NAS5726W** messages issued for EMSGE option (Datafono) 16 sec PLU response timer expires
- **NAS7701W** message now displays optional **RTEIN=CLEAR** forced clear filtering events
- **NAS7730I** message showing inbound call request packet data when TRCMCH ICR On set with ALRMSGTXT=SHORT
- **NAS7731I** message showing outbound call request packet data when TRCMCH OCR On set with ALRMSGTXT=SHORT
- **NAS92nns** messages added for additional authorization processing (please see HNAS Messages and Codes documentation for details)
- **NAS9205I** message withheld until today's date is within 60 days of the EOTDATE
- **NAS9206I** message withheld until today's date is within 60 days of the EOMDATE
- **Clear Request** diagnostic codes **D8 (217)** and **D9 (218)** now contain **DIAGX=** validity codes
- **Clear Request** diagnostic codes **DF (223)** **DIAGX=8|9** codes added for Datafono

Console, Trace and Debug Enhancements (Commands, Start Parameters...)

- **# text** console input echoed after the HNASCMD-> prefix (example, HNASCMD-> # *text*) comments
- **All** commands now echo their invocation parameters following the HNASCMD-> prefix
- **All** commands now echo their execution parameters following the HNASXEQ-> prefix
- **All** commands that force another command to be executed now use a LIFO rather than FIFO enqueue so that the propagated command is executed immediately after the original command
- **LCLCONS** or **rconname** now accepted as the first argument for the **ALARM**, **EXEC**, **SCHEDULE**, **SMSG** and **TRCCONS** commands
- **LCLCONS**, **NASUTIL**, **PING**, **TIMER**, **XOTUTIL**, **XTPUTIL** or **pcename** now accepted as the first argument for the **DPCE**, **STATS**, **TRCBFR**, **TRCDATA**, **TRCDISP**, **TRCIO** and **TRCPCE** commands
- **HNASCMD->** command prefix can now be replaced with **EXECCMD->**, **SCHDCMD->**, **PRXYCMD->** or **TRAPCMD->** depending on how the command was scheduled for execution
- **ALARM FILTER=** command now supports (FC) and (FU) message ID suffixes to allow forced conditional (subject to SHOW state) and forced unconditional (SHOW state ignored) SYSCONS display
- **ALARM LOG=?** command now displays date and time of last alert message log entry
- **ALARM LOG=?** now executed automatically during HNAS SHUTDOWN providing alarm log history
- **ALARM MSGTXT={?|LONG|SHORT}** command support has been added to allow the BUILD OPTIONS=ALRMSGTXT= operand to be displayed and/or toggled.
- **ALARM PULSE={?|*(hh:mm:ss, hh:mm:ss,seconds)}** command support added to allow dynamic heartbeat message control

- **ALARM LCLCONS|rconname action** command support added to allow one console to modify alarm parameters for another console
- **BFR** (internal) trace record now provides buffer allocation and release timestamp and tracestamp.
- **BPM** parm allows all modifiers to be reset before command execution, right side modifiers permitted
- **DLU** command now provides SESSINIT connect status, call set-up direction and improved filtering
- **DLU** command now provides header display information compatible with XOT and XTP resources
- **DLU** command now displays a 'P' in place of the leading zero for a PVC VCN value
- **DLU** command now displays the new LUTO column to show the active LU timer in effect (blank implies that no timer is running)
- **DNAS** command now displays additional environment & runtime info, automatic execution at startup
- **DNAS** command now supports **DNAS JCL|DDNAMES** to display the running HNAS JCL
- **DNAS** now executed automatically during HNAS SHUTDOWN providing HNAS information
- **DNAS** now displays the 'installed under' system and date stamp in addition to the 'created under' and 'running under' system and date stamp
- **DNWDF** command added to display active NEWDEFN CDF image maintained in memory
- **DPCE** command now provides header display information compatible with XOT and XTP resources
- **DRMT** command now displays **[NO]CLOSEONAPFAILURE** option for XOT|XTP REMOTEs
- **DRMT** command now displays **[NO]CUD0SELECTSLU** option for MCH|MXT REMOTEs
- **DRMT** command now displays **[NO]DELAYBINDRESP** option for MCH REMOTEs
- **DRMT** command now displays **[NO]IDTST** option for DFX REMOTEs
- **DRMT** command now displays **PVCRECONTMR=, PVCSETUPTMR=** and **PVCSETUPREJ=** options for MCH|MXT REMOTEs
- **DRMT** command now displays **[NO]RESETINO** option for MCH REMOTEs
- **DRMT** command now displays **SVCCALLTMR=** option for MCH|MXT REMOTEs
- **DRMT** command now displays **SVCO|5** generic SLU name if one is present for MCH REMOTEs
- **DVC** command now provides SESSINIT connect status, call set-up direction and improved filtering
- **DVC** command now provides header display information compatible with XOT and XTP resources
- **DVC** command now displays a 'P' in place of the leading zero for a PVC VCN value
- **DVC** command now displays the new VCTO column to show the active VC timer in effect (blank implies that no timer is running)
- **ECHOXEQ** command added to control how much HNASXEQ> data is echoed back to operator
- **EXEC ddname** command allows a list of console command in a HNAS start job ddname to be run
- **EXEC ddname** command now stops queuing commands when an embedded EXEC command or the new END command is decoded
- **EXEC ddname** command now allows multiple commands to be specified on the same record within the command list file
- **EXEC (command1,command2,command3,...)** allows multiple console commands on a single line
- **EXEC LIST ddname** command added to view the content of externally defined command lists
- **EXEC LIST** command has been modified to issue NASC052E error message if **ddname** is omitted
- **EXEC STOP** command added to terminate PAUSE command delay mode or CMDLIST execution
- **EXEC ddname** command modified to allow CMDLIST comments that start with # to be displayed
- **EXEC LCLCONS|rconname action** command support added to allow one console to queue commands and alter queued command processing for another console
- **EXEC** command has been modified to decode the new NOPURGEONERROR and PURGEONERROR keywords
- **HELP** command now displays common command parameters after header message
- **HELP** command new **SHOWALL** keyword displays complete help text for all commands
- **MMEM EOTKEY=dd...dd** command allows the TRIAL PERIOD EXPIRATION DATE (EOTDATE) for a trial distribution to be extended using the key provided by the **dd...dd** digits
- **MMEM EOMKEY=dd...dd** command allows the MAINTENANCE/USE ANNIVERSARY DATE (EOMDATE) for a permanent distribution to be extended using the key provided by the **dd...dd** digits
- **MMEM EOMKEY=dd...dd** command also allows a trial distribution to be converted to a permanent distribution dynamically using the key provided by the **dd...dd** digits

- **MON TAP** command NAS251*nM* monitor messages can now be filtered using the ALRMFLTR= operand so that they can be optionally routed to SYSCONS using the (FC) or (FU) suffixes
- **MON TAP** command now accepts **PKTDATA|MAXDATA|MINDATA|NODATA** as arguments to control the type of information displayed in the NAS251*nM* monitor messages
- **MRMT LOGTAB=*lgtbname*-R** command now permits the named LOGON table to be reloaded into memory
- **MRMT LUNAME=|SVC0=|SVC4=|SVC5=*sluname*-{A|I}** command now permits manipulation of SLU state (A is for active (online) and I is for idle (offline)) - VARY command is executed
- **MRMT LUNAME=*** command now permits manipulation of asterisk (*) for GATE (LLC4) support
- **MRMT *rmtname* IPADDR=*aaa.bbb.ccc.ddd* dynamic TYPE=XOT REMOTE** IP address changes
- **MRMT OPTIONS=[NO]CLOSEONTAPFAILURE** command allows a customer to toggle this option
- **MRMT OPTIONS=[NO]CUD0SELECTSLU** command allows a customer to toggle this option
- **MRMT OPTIONS=[NO]DELAYBINDRESP** command allows a customer to toggle this option
- **MRMT OPTIONS=[NO]IDTST** command allows a customer to toggle this option
- **MRMT OPTIONS=NRITAB=*nrtbname*-R** command now permits the named NRI table to be reloaded into memory
- **MRMT OPTIONS=PVCRECONTMR=*secs*, PVCSETUPTMR=*secs* and PVCSETUPREJ=*code*** allows a customer to delete, change or add these options for TYPE=MCH|MXT REMOTES
- **MRMT OPTIONS=[NO]RESETINO** command allows a customer to toggle this option
- **MRMT OPTIONS=SVCCALLTMR=*secs*** allows a customer to delete, change or add this option
- **MRMT SVC0|5={(*sluname+gluname|...*)}** command now allows generic SLU name to be added for TYPE=MCH REMOTE
- **MRMT USSTAB=*ustbname*-R** command now permits the named USS table to be reloaded into memory
- New **PARM='DEBUG TCP'** start parameter allows certain TCPIP 198 ABENDs to be bypassed, replaced by new NAS2109S and NAS2110S alarm messages - HNAS execution continues
- **PAUSE *seconds*** command suspends execution of console command following PAUSE command
- **PING** command now produces NAS261*nM* monitor messages that can be filtered using the ALRMFLTR= operand so that they can be optionally routed to SYSCONS using the (FC) or (FU) suffixes
- **PING (XOT) *dmyname*** allows predefined pinging xot call request values to be employed (all fields)
- **PRNT** command **PRTSWLST=(?|*|list)** displays, clears, switches, updates SYSPRINT controls
- **PRNT** command now support the **NEXTPRSW/SWITCHNOW** to force a switch to next ddname
- **PRNT** command now support the **RSMEPRSW** parameter to restart SYSPRINT switching
- **PRNT** command **TIMESTAMP=(*time-format*,+D|-D)** updates TIMESTAMP time and date controls
- **PRNT ON|OFF** command now forces NAS021*ni* alarm message to be issued
- **PRNT DATE ON|OFF, QLLC ON|OFF ...** command now forces NAS0230W alarm message to be issued
- **QUIT - QE [ddname](*cmd1,cmd2...cmdn*)** command executes command list before shutdown
- **QUIT** command now unconditionally executes **DNAS** and '**ALARM LOG=?**' as part of the shutdown process
- **SCHEDULE {ddname}(*hh:mm:ss,cmd,....,hh:mm:ss,cmd*)|LIST|SUSP|RSME|PRG}** console command support added to allow scheduled console commands to be added, deleted or displayed
- **SCHEDULE LCLCONS|*rconname action*** command support added to allow one console to schedule commands and alter scheduled command processing for another console
- **SHOW MSG ON|OFF** command added to manipulate the **SHOWMSG** start parameter
- **SHOW ON|OFF|ERR** command now forces NAS011*ni* alarm message to be issued
- **SHOW MORE|LESS** command now forces NAS0130W alarm message to be issued
- **SHOWMSG** parameter allows configuration and alarm message multiple blanks compression
- **MSG LCLCONS|*rconname 'text'*** command support added to allow the target console to be identified by PCE name
- **SNAP ALL|*trgtlist*** command support added to allow specific HNAS storage areas to be dumped in the SYSPRINT file
- **TCPIP** external interrupt table entries extended to allow logging of additional diagnostic information making HNAS TCPIP tracing less of a requirement for problem diagnosis

- **TRCCONS LCLCONS|*rconname* {ON|OFF}** command support added to allow the target console to be identified by PCE name
- **TRCLU, TRCMCH, TRCMCHX & TRCVC** provide **HNASXEQ->** modifiers and arguments in effect
- **TRCMCH ICR OCR ICLR OCLR** now supported on **PARM=** Start Parameter
- **TRCMCH ICR OCR ICLR OCLR** now have *local* or *global* significance (Default|LCL|GBL argument)
- **TRCSUBR *eventlist*** now supported on **PARM=** Start Parameter and as console command
- **TRCTRAP ALRMLIST=(NAS*nnnni*'*data*/*data-offset*,...)** command, refined trap message filtering
- **TRCTRAP ALRMLIST=(NAS*nnnni*' *data*'/*data-offset*,...)** command (*data* as quoted string)
- **TRCTRAP TRAPACTION=(EXEC=*ddname*)** option support command list execution upon trap
- **TRC*type*** console commands (i.e. TRCMCH ICR OCR ...) now accept multiple arguments
- **VARY *mchname* {OFF|ON}** command allows MCHs to be disabled or enabled
- **VARY *sluname* {OFF|ON}** command permits or restricts SLUs for inbound or outbound calls
- **VARY *sluname* FORCE** command has been modified to close the TCPIP socket associated with the SLU even if the SLU itself is inactive (normally only PVCs can be in this state)

HNAS V2R3M0 Release Summary:

Functional Enhancements

- SMP/E Product Installation Supporting HNAS private global zone
Common global zone support was added on 03-31-2004.
- Extended Diagnostic Reason Code Support (Improved session Call and Clear Troubleshooting)
- CART= WTO Support for Session Manager (Netview, TDSLink) Operator Consoles (ALRMCART=)
- Improved Console and SYSPRINT output filtering using new PRNT *type* and SHOW CONS settings
- SYSPRINT Dataset Recording Improvements
- SYSPRINT Log Filtering
- Start parameter follower support
- XOT TAP= support now provides calling/called address, facilities and call user data parameters
- TRCTRAP= support provides automation tools for improved debugging and trace suspension
- NETVIEW routing support, asynchronous alarm messages/synchronous console command output
- TRCTRAP= enhancement provides optional SNAP dynamic dump of control blocks for trap cond.
- NETVIEW routing enhancement for asynchronous alarm messages/synchronous console command output that allows WTO routing code to be configured (OPTIONS=WTOROUTCDE(*type*)=*code* on BUILD, where *type*=ALRM|CONS and *code*=0-128)

Configuration Enhancements

- Callout Session Connect Balancing Extension (OPTIONS=BALANCERTEOUT)
- Backup Router Support for LLC0/5 Callout
- CUD=, FAC=, DCEADDR= and DTEADDR= TAP parameters added for TYPE=XOT REMOTE
- OPTIONS=INHIBITBIDREJ - optional BID unconditional accept logic
- OPTIONS=NORTRBIDREJ - optional BID 0814 sense reject logic
- OPTIONS=ONEPIUINB - option to force one PIU per Bracket
- OPTIONS=REQSESSDELAY=*value* - option for request session delay timer value
- OPTIONS=RESIDSTART=*decimal-start-value* - option to assign start values for GATE Resource ID
- INIT={({ACTIVE|IDLE},DELAYTIME=*minutes*,RETRYLMT=*count*) added for LOCAL definitions
- HNAS CDF Configuration Error Summary Improvements
- HNAS CDF Configuration Error Processing, NAS1999W|E|S improves error message identification
- FASTRUN VTAM operand generation enhancement for QLLC TYPE=SPU REMOTE resources
- FASTRUN AMNF VBUILD statement name from BUILD APPLNAME= rather than NASNAME=
- IDLETO=*minutes* added for non TYPE=XOT REMOTE level definition statements
- The configuration process will now accept a pound sign (#) in cc-1 improving CDF comments
- Start Parameter PFXWTO CONS causes console output to be prefixed with the NASNAME=*value*
- Start Parameter PFXWTO *text* allows user defined alert message and console output prefix name
- OPTIONS=PFXDCEADDR appends DCEADDR= in front of GATE outbound *calling* DTE address
- OPTIONS=LLC0CTCPCHK/LLC5CTCPCHK DATE like control LLC0|5 Call Accept CTCP sessions
- CONLMT=0 can now be coded on BUILD to disable Remote Console access
- Bidirectional Twoway (Callin/Callout) VC support for PCNE/PAD (SVC0=/SVC5=) resources
- XOT PVC= support now includes mxtname assignment for user defined window/packet size values
- OPTIONS=REUSEBSYSPU supports special QLLC call-to-call (already connected VC) condition
- SVC0=, SVC4= and SVC5= parms now allow SLUname generation using prefix and suffix values
- TRAN= parm now accepts NPSIEVEN, NPSIODD, NPSIMARK, NPSISPACE for NPSI tables
- RTEIN= now supports calling (=>S) source address in addition to the existing (=>T) target address
- OPTIONS=BALANCERTEIN supports LLC-n RTEIN= MCH level round robin session balancing

Alert/Alarm Message and Clear/Reset Code Enhancements

- Message Filtering enhancement for NAS*innns* messages
- Julian Date (*nnn*) Now Available in Alert Message Entries

- NAS2021W SERVER INITIALIZATION FAILED, LOCAL VARIED OFFLINE message added
- NAS2601E SOCKET POOL DEPLETED/NAS2602I SOCKET POOL RESTORED messages added
- NAS8000I Starting Session message modified to improve QLLC call setup status notification
- NAS7718T, NAS7719T and NAS7798T Inbound/Outbound call request packet content trace messages via TRCMCH ICR|OCR
- NAS0050A, NAS0060W and NAS0070W alert messages provided for TRCTRAP support
- NAS3703W NOTIFY and NAS3704W CLEANUP (REQSESS related) messages added
- NAS7715W Call Request Failed message now contains extended diagnostic event reason codes
- NAS7713W SECOND CLEAR FROM ip-addr(port)... message added improving event reporting
- Clear Diagnostic codes
 - Added diagnostic code 221 x'DD' - Timeout. CTCP did not UNBIND after receiving clear
 - Added diagnostic code 222 x'DE' - Second clear received from router for VC
- NAS7797W CALL FROM... XOT TCP/IP message added improving event reporting
- NAS3797I LU sluname RECEIVED BIND FROM PLU pluname PVC messages added
- NAS4706W LU sluname REJECTING BIND FROM PLU pluname SENSE=xxxx message added
- NAS0001W PARMLIST OMITTED message added indicating PARM= values omitted
- Clear diagnostic code 211 x'D3' logic added to address non recoverable packet Reset conditions
- NAS3705W LU lu-name REJECTING message added indicating that PIU is being rejected
- NAS7774W PVCSETUP FAILED message when remote issues socket close in response to setup
- NAS7718T, NAS7719T and NAS7798T Inbound/Outbound pvc setup content trace messages
- HNAS PVC Setup Status Codes were reassigned improving router setup retry processing
- NAS7795T Inbound/Outbound clear request packet trace messages via TRCMCH ICLR|OCLR
- NAS251xM TAPping (Keep Alive) monitor messages via MONTAP/MON TAP
- NAS4710W lu-nm LU st-addr SENDING DIAG PKT :text BFR NEXT alert message added
- NAS3705I LU lu-name REJECTING cmd #seq SENSE=bbbbbbbbb BID alert message added
- NAS4707W sluname GENERATING ERR/INFO PACKET FOR CTCP alert message added
- NAS4708W GATE FC CTL SES LU luname CLEARED BY CTCP alert message added
- NAS4709W REMOTE LU LUIQ TIMEOUT,LUIQ BFR alert message added
- Clear diagnostic code 212 x'D4' logic added to report VC session already active conditions
- NAS3797I messages generated for GATE control session activation (bound)

Console and Trace Enhancements

- Error Reporting Enhancement
- Resource Name follower supersedes RNM=, LNM= or LUNM= Console Command Modifiers
- DMAP APAR command executed at startup, now generates APAR ID summary output
- DNAS command now displays APAR through information as well as sorted APAR list
- TRCLU command improvements for TYPE=SPU resources
- DLU command now displays SLU association with TYPE=SPU REMOTE definitions
- DPCE command now displays PCE association with TYPE=SPU REMOTE definitions
- DPCE, some TRCtypes and STATS commands now accept various arguments relative to PCE
- DLU, DVC and DPCE commands now accept TRACE=YES|NO arguments
- New PING XOT Command Allows Addressing of Routers IP Address, Protocol and MCH
- MRMT Expanded REMOTE LUNAME= Config. Parameter/Resource Reassignment (QLLC LLC3)
- MLCL & MRMT commands have been modified to permit updating of the INIT= operand
- VARY command has been modified to allow a LOCAL to be varied offline or online like REMOTEs
- DTRC command now displays the trace table wrap count, WRAPCNT=*wrap-cnt* in the trace header
- DLU, DMCH, DPCE and DVC commands now accept SHOWtype arguments for improved status
- TRCLU trace entries now provide keywords (BIND, SDT, UNBIND) to improve search capabilities
- DMCH command now accepts FMT3 as an argument providing MCH and LU association displays
- PFXWTO *name* option allows user defined alarm messages & console output prefix name
- TRCMCH ICR and OCR trace option now produces formatted call request packet message content
- ALRMFLTR console command now appends instead of replacing messages from list array
- MLCL *lcl-name* RTEIN= and RTEOUT= operands now permits insertion/deletion of table entries
- DNAS command now produces a list of missing *apar-ids* as well as individually specified *apar-ids*

- New DNAS USERMODS command produces a list custom-user-mods on system, as appropriate
- DNAS command now provides HNASOBJX and HNASOBJX distribution USERMOD indicator
- MRMT expanded allowing PCNE/GATE/PAD (SVC0|4|5=) SLU names to be added/chgd/deleted
- SHOW MORE/SHOWMORE console/start parameter provides expanded TCP/IP event debugging
- TRCALL command modified to remove PCE trace function for option ON|OFF, reducing confusion
- DNAS command now provides distribution type (SMP/E or NON-SMP) on the header line
- VARY command has been modified to allow specific sockets or ranges to be varied offline or online
- TRCMCH ICR and OCR trace option now produces formatted PVC setup packet message content
- DVC command now displays PVC setup packet exchange status
- MRMT command now allows TRAN= parm to be updated 'on the fly' (all TRAN= values accepted)
- TRCMCH ICLR and OCLR trace option now produces NAS7795T messages for all clear packets
- MON TAP ON|OFF|ALLON|ALLOFF command produces NAS251xM KeepAlive monitor messages.
- VARY command now allows closure of active sockets (LOCAL|REMOTE) with new FORCE option
- DPARM command now accepts MODIFIERS as an argument to restrict display to modifiers only
- HELP command text localized into respective help command modules improving maintenance
- DPARM command now displays ALLON|ALLOFF instead of ON|OFF for some TRCtyp states
- PRNT QLLC console command start parameter PRNTQLLC added to filter NAS8xxx messages
- DNAS command now provides Host OS & Version information and Execution date in display output

QLLC Enhancements

- QLLC Callout via TYPE=SPU REMOTE
- QLLC Callout via TYPE=MCH REMOTE
- QLLC Callout Support - Calls initiated via Application Bind to designated Control LU
- QLLC Callout Support - Calls initiated via various Timer Control Options
- QLLC Callout Support - Calls initiated via HNAS Console Command Controls
- QLLC Callout Support - Calls initiated when HNAS is Started/Activated
- QLLC TYPE=SPU LUNAME= SLUname MXTname association (wire an SLU to an application)
(LUNAME= operand enhancement for QLLC SPU's (SLU *applid* and *mxtname* support)
- QLLC TYPE=SPU REMOTE IDLETO=*minutes* Inactivity Clear Support
- QLLC TYPE=SPU LUNAME= coding improvement for LOCADDR values (gap-count vs. commas)
- QLLC TYPE=SPU APPLNAME=ACQUIRE Printer Bind Support Improvement
- QLLC TYPE=SPU SVC3=ALLOW option improves MCH SPU coding and operation

HNAS V2R2M0 Release Summary:

- REGION Size and CPU Utilization Improvements
- High Memory Support
- Multiple 'same type' Server Support
- Multiple Stack Support
- Shared Socket Support
- Authorization Key Support (trial users distribution)
- XOT QLLC PU Type 1|2, LU Type 0|1|2|3 device Support (PUT2.1 or LU 6.2 not available)
- FASTRUN AMNF Generation
- Multiple SYSOUT Support
- Callout Session Connect Balancing
- Callout Alternate DTE Connect Retry Support
- Callout Connection Routing using Calling or Called Addresses
- Callin DTE Address Filtering Support
- Callin SLU/PLU Fixed Connection Support
- Callin Default PLU Assignment Support (SYSL= now optional)

- Gate Control Session parameter to fine tune the Activation Delay Timer
- Global Buffer List Table Size parameter to reduce memory requirements
- Extensive Console Alert Filtering/Reduction
- Alert Message Additions and Enhancements
- Improved Diagnostic capabilities, Improved Options and various other enhancements
- TAP= XOT protocol level tapping improvements
- SUBD= allowed on non-Fast Connect GATE MCH to permit CTCP selection by subaddress digits
- FASTRUN AMNF VBUILD statement name now comes from APPLNAME= rather than NASNAME= operand on BUILD definition statement (APAR 2200080).

HNAS V2R1M1 Release Summary:

- Maintenance Update
- Z/OS V1R2 (and V1R4) Support
- SLU Selection by CUD Data
- SLU to PLU Connection Wiring
- MCHTMR Support
- Minor Options/Enhancements

HNAS V2R1M0 Release Summary:

- Dynamic IP Address Assignment (Call-in) Support
- LOGTAB Support
- Performance Improvements
- Remote Console Alarm
- Monitoring, Local Console Alert Filtering
- Improved Options and various enhancements

HNAS V1R1M4 Release Summary:

Please refer to the 114 documentation manual for the list of new features.

Please refer to the specific 'HNAS VnRnMn - New Features Overview' for a brief description of all enhancements.

HNAS V2R4M0 New Features

V2R4M0 General Availability - 2006-07-31

Functional Enhancements:

- All appropriate V2R3M0 **APARs** have been incorporated as permanent fixes in HNAS V2R4M0 as of the general availability date. The HNAS product is continually updated using Standard APARs (bug fixes) and Enhancement APARs (enhancements) in an effort to provide a stable product as well as product improvements and new features for our customers. The use of enhancement APARs is a way to provide these product updates without having to wait for a new VnRnMn release (Note that some customer enhancement requests are deferred to a later release because the changes are either too big or inappropriate as an Enhancement APAR).

- **Non-SMP/E installation and maintenance improvements**

The process that we employ to generate non-SMP/E HNAS 240 edistributions has been improved. A REXX exec is provided which, when customized, generates the jobs required to allocate data sets and to install and maintain HNAS. Non-SMP/E and SMP/E product edistributions are now built using a common interface providing a uniform installation process.

- **SMP/E fully integrated in the installation and maintenance documentation**

The SMP/E installation and maintenance instructions are now fully integrated into the standard HNAS documentation series (including content improvements in the maintenance section).

- **ISARX25 Datafono Support Integrated into the Standard HNAS Product (IBM SPAIN)**

This support was previously available as a custom 230.c add-on to the HNAS 230 product. This support was developed for customers in Spain who are using the Datafono ISARX25 implementation and required a HNAS solution to migrate from their 3745 communication controllers. The Datafono option must be ordered prior to shipment of HNAS because a special distribution key is required to enable the support. The DNAS display output contains the text '**DATAFONO SUPPORT IS INCLUDED**' if the support is enabled.

The Datafono documentation and installation instructions are now included in the standard HNAS documentation series (in 230.c these instructions were in a separate add-on booklet).

With APAR 2400011 installed Datafono customers can easily install custom versions of the XAICDTFT translate table CSECT (see XAICDTFT in the index)

With APAR 2400074 the IDTST and CUD0SELECTSLU options are supported. These options provide features (message ID checking and CUD0 participation in LU selection) provided by ISARDX25.

- **BUILD CONCMDQ=ddname (CMDLIST)** support permits lists of console commands to be pre-coded in JCL DD statements outside of the CDF file. While the command list support is primarily utilized via the new **EXEC** console command there are other console processes that utilize the command lists. Following is an overview of the new CMDLIST features while a detailed description can be located under the Configuration Enhancements and Console and Trace Enhancement headings in this section:

HNAS V2R4M0 New Features

- Many (but not all) start parameters have a console command equivalent. At start-up, you have the option of specifying console command-like start parameters in the PARM= operand (or in the file identified by the PARMFIL= parameter - see below) or by specifying the equivalent console command in the BUILD EXEC= (or CONCMDQ=) operand. Prior to the introduction of the PARMFIL= parameter, the 100 character JCL limit imposed by z/OS on the EXEC PARM= operand made it necessary to use the BUILD EXEC= operand in conjunction with the EXEC PARM= operand so that the 100 character limit would not be exceeded. Now that you can specify all start parameters in a parameter file, the need for listing parameter equivalent console commands in the BUILD EXEC= operand becomes less necessary although still equally valid.
- A list of commands can be executed via the operator console by entering **EXEC ddname**. This command can also be provided at startup via CONCMDQ=('EXEC ddname'). Equivalently, you can also specify CONCMDQ=ddname. The difference is that the ddname identified in the CONCMDQ=('EXEC ddname') is not processed until the EXEC command is executed while the ddname identified in CONCMDQ=ddname is processed when the CDF is scanned.
- A list of commands can be executed via the new **TRCTRAP TRAPACTION=(EXEC=ddname)** option.
- **QUIT** command now accepts **Exec** as a new keyword follower. **QE {ddname}(cmd1, ..., cmdn)** causes command lists to be executed before HNAS is shutdown.

Note: Currently **BUILD CONCMDQ=(EXEC ddname)** and command line '**EXEC ddname**' are restricted to a single 'EXEC ddname' reference. CONCMDQ=(EXEC ddname1, EXEC ddname2, ...) or a command line of 'EXEC ddname1 EXEC ddname2 ...' is not supported as a concatenation. When multiple EXEC ddname statements are decoded in a single list, all but the first are ignored. If you want command lists to be concatenated, you can either concatenate the command lists in JCL, merge the command list files or include the next command list as the last command in the current command list. For example, you could end the command list identified by ddname1 with EXEC ddname2. We plan to expand this support in a future 240 Enhancement APAR by allowing multiple EXEC queues. This will permit multiple lists to be concatenated via the CONCMDQ= operand or via command line input. CONCMDQ=(EXEC ddname1, EXEC ddname2, ...) and command line input of 'EXEC ddname1 EXEC ddname2 ...' will operate as the operator expected.

- **PARM='EOTKEY=dd...dd'** allows the TRIAL PERIOD EXPIRATION DATE (EOTDATE) for a trial distribution to be extended using a special 16 decimal digit key provided by the **dd...dd** digits. An HNAS trial distribution is normally shipped with an EOTDATE that specifies when the TRIAL AUTHORIZATION will expire. An unexpired EOTDATE is required to use HNAS. In the past, the only way to extend the EOTDATE was to order and install a new refresh distribution. The new EOTKEY= parameter now allows the EOTDATE to be extended without the need of a new refresh distribution. The EOTKEY=dd...dd digits are provided by Comm-Pro in a special file that is sent as an e-mail attachment or is downloaded from our FTP server. The EOTKEY=dd...dd string can then be cut and pasted to the PARM= operand (or MMEM console command - see page ChangesNewFeatures-43). The EOTKEY file has the following format:

```
EOTKEY=4961000737880526
HNAS EOTKEY CREATED AT 08:17:02 ON 2010/11/29
TRIAL PERIOD EXPIRATION DATE IS 2011/01/28
CUSTID=SFD_99999
CUSTINFO=COMM-PRO ASSOCIATES
```


ETKYDC=0201101281199999

The DNAS display for a trial distribution has the following form (for example):

```

HOST NAS INFORMATION FOLLOWS
  HNAS VERSION=V2R4M0 DIST=SMP/E                               1
  HNAS PROGRAM RUNNING UNDER z/OS 01.11.00                   2
  HNAS PRODUCT INSTALLED UNDER z/OS 01.11.00                 3
  HNAS PRODUCT CREATED UNDER z/OS 01.11.00                   4
  DNAS COMMAND ENTERED AT 18:54:01 ON 2010/12/01              5
  HNAS PROGRAM STARTED AT 18:54:01 ON 2010/12/01              6
  HNAS PRODUCT INSTALLED AT 08:12:00 ON 2010/11/29            7
  HNAS PRODUCT CREATED AT 08:19:12 ON 2010/11/29             8
  HNAS PRODUCT CREATED WITH MAINTENANCE THROUGH APAR 2400106  9
  MOST RECENT MAINTENANCE APPLIED IS APAR 2400106            10
  AUTH=032D SHIPID=1100000011199999                          11
  CUSTID=SFD_99999                                           12
  CUSTINFO=COMM-PRO ASSOCIATES                               13
  TRIAL PERIOD EXPIRATION DATE IS 2010/12/31                  14
  DATAFONO SUPPORT IS INCLUDED                              15
                                                                16
                                                                17
  APARID MAINTENANCE STATUS                                  18
  ALL MAINTENANCE ON THROUGH MOST RECENT APAR 2400106       19
  
```

If EOTKEY=4961000737880526 (for example) is used to extend the EOTDATE, DNAS display lines 11, 14 and 16 will be modified as follows:

```

  AUTH=060D SHIPID=1100000011199999 ETKYID=1100000011199999  11
  TRIAL PERIOD EXPIRATION DATE IS 2011/01/28*                  14
  EOTKEY=4961000737880526 IS IN EFFECT                        16
  
```

Note: EOTKEY= parameter logic was introduced into 240 via APAR 2400106.

- **PARM='EOMKEY=dd...dd'** allows the MAINTENANCE/USE ANNIVERSARY DATE (EOMDATE) for a permanent distribution to be extended using a special 16 decimal digit key provided by the **dd...dd** digits. An HNAS permanent distribution is normally shipped with an EOMDATE that specifies when the MAINTENANCE/USE license will expire. An unexpired EOMDATE is required to use HNAS. In the past, the only way to extend the EOMDATE was to order and install a new refresh distribution. The new EOMKEY= parameter now allows the EOMDATE to be extended without the need of a new refresh distribution. The EOMKEY=dd...dd digits are provided by Comm-Pro in a special file that is sent as an e-mail attachment or is downloaded from our FTP server. The EOMKEY=dd...dd string can then be cut and pasted to the PARM= operand (or MMEM console command - see page ChangesNewFeatures-43). The EOMKEY file has the following format:

```

EOMKEY=4962030747980516
HNAS EOMKEY CREATED AT 16:00:19 ON 2010/11/28
MAINTENANCE/USE ANNIVERSARY DATE IS 2010/12/31
CUSTID=SFD_99999
CUSTINFO=COMM-PRO ASSOCIATES
EMKYDC=0201012311199999
  
```

Note: EOMKEY= parameter logic was introduced into 240 via APAR 2400095.

HNAS V2R4M0 New Features

- **PARM='EOMKEY=dd...dd'** also allows a trial distribution to be converted to a permanent distribution dynamically using the key provided by the **dd...dd** digits.

If EOMKEY=4962030747980516 (for example) is used to convert a trial distribution to a permanent distribution, the DNAS display will reflect this by changing DNAS display records 11, 14 and 16 above as follows:

```
AUTH=000 SHIPID=1100000011199999 EMKYID=1100000011199999      11
MAINTENANCE/USE ANNIVERSARY DATE IS 2010/12/31*                14
EOMKEY=4962030747980516 IS IN EFFECT                          16
```

Note that AUTH=000 on DNAS display record 11 above reflects the new permanent status.

Note: EOMKEY= trial conversion logic was introduced into 240 via APAR 2400106.

-
- **PARM='PARMFILE=pfdname'** allows all or some HNAS start parameters to be provided in a sequential file. This means that you can specify all HNAS start parameters without regard to the 100 character limit imposed by the JCL scanner for the EXEC PARM= operand. Please refer to Chapter 2 of the HNAS Guide and Reference manual for more information.

Note: PARMFILE= parameter logic was introduced into 240 via APAR 2400094.

- **BUILD PRTSWLST=(controls)** are now provided in HNAS permitting automatic SYSPRINT switching when the current SYSPRINT log file becomes full, when the designated time occurs or when forced by the console operator. Please refer to the PRTSWLST= entry provided in Configuration Enhancements section for additional information.
- **REMOTE TYPE=DMY** was added to permit future expansion of some console command parameters. PING (XOT) dmyname is the first command to use this remote type.

Note: Please refer to the sections (Configuration Enhancements, Alert/Alarm Message and Clear/Reset Code Enhancements, Console and Trace Enhancements and Enhancements - Expanded Content) in this 240 New Features document for additional information regarding these and other new or enhanced features.

Configuration Enhancements:

Various **CDF Options** and **Parameters** were added or changed to allow the user greater flexibility when defining resources in V2R4M0:

- **ALRMFLTR=** table has been expanded from 16 to 128 entries allowing fine tuning of alert messages filtering for SYSCONS and SYSPRINT message routing or purging.
- **ALRMFLTR=** now supports (FC) and (FU) message ID suffixes which allow alarm messages to be forced conditionally to SYSCONS (subject to SHOW state) or forced unconditionally to SYSCONS (SHOW state ignored). For example, if ALRMFLTR=(...,NAS2020I(FC),NAS0299I(FU),...) is specified, NAS2020I alarm messages will be sent to SYSCONS except when SHOWOFF is effect while NAS0299I alarm messages will be sent to SYSCONS regardless of the SHOW state.

Note: (FC) and (FU) message ID suffix support was introduced into 240 via APAR 2400021.

- **CONCMDQ=** can now include the new **EXEC *ddname*** command which permits a list of console commands to be pre-coded in JCL DD statements outside of the CDF file. Please refer to the **EXEC *ddname*** command for additional information.
- **CONCMDQ=*ddname*** can now be specified to provide a dataset command list (provides consistency with the SCHEDULE=*ddname* operand).

Syntax: CONCMDQ={*ddname*|(*cmd*,...,*cmd*)}

(*cmd*,...,*cmd*) is an inline command list (no change).

ddname identifies a file that contains a list of commands. A *ddname* command list can contain the following records:

Comments are allowed and start with an asterisk (*), semi-colon (;) or pound sign (#) in record column 1. Comments can also appear on a command line but must start with a semi-colon after the command. A single command can be specified on a single record as follows:

cmd ; *comment*

Multiple commands can be specified on a single record as follows:

(*cmd* , ... , *cmd*) ; *comment*

Note: The new CONCMDQ=*ddname* operand support was introduced into 240 as Enhancement APAR 2400080.

- **EXEC=** operand has been added as an alias for CONCMDQ= (provides name symmetry with the EXEC command like the SCHEDULE= operand and SCHEDULE command).

Note: The new EXEC= operand support was introduced into 240 as Enhancement APAR 2400080.

- **EXEC=** operand has been modified to decode the new NOPURGEONERROR and PURGEONERROR keywords. One of these keywords may be entered as the first value in an inline command list (**NOPURGEONERROR|PURGEONERR,*cmd1*,...,*cmdn***) or as the first non-comment record in a *ddname* command list.

PURGEONERROR (the default) specifies that the remaining commands in an active command list will be purged if an error occurs for any command in the command list.

NOPURGEONERROR specifies that the remaining commands in an active command list will still be executed even if an error occurs for any command in the command list.

Notes: 1) The PURGEONERROR|NOPURGEONERROR options are valid for the EXEC= operand on the BUILD definition statement as well as the EXEC console command.

2) The PURGEONERROR|NOPURGEONERROR options are not included in the command list but are decoded and acted upon immediately when parsed.

3) Although the PURGEONERROR|NOPURGEONERROR options should be specified as the first command list element, the parser will accept the keywords anywhere in the list. However, the last occurrence of either option is the value that will be used.

4) The PURGEONERROR|NOPURGEONERROR option remains in effect until changed by

HNAS V2R4M0 New Features

the EXEC console command. To view the current option in effect, specify the following:

```
EXEC LIST
```

(command list omitted) which will yield the following display:

```
OPTSONLY COMMAND QUEUE COUNT: 0000/0512
PURGEONERROR <- IN EFFECT
```

- 5) If the PURGEONERROR|NOPURGEONERROR option is specified and the EXEC *cmdlist* LIST command is entered, the value of the specified option will be listed first regardless of where it was specified in the command list. For example:

```
EXEC (DLP,NOPURGEONERROR,DPARM,DLU) LIST
```

will yield the following display:

```
INLINE COMMAND QUEUE COUNT: 0014/0512
NOPURGEONERROR <- SPECIFIED
=> DLP
=> DPARM
=> DLU
```

Note: The new NOPURGEONERROR|PURGEONERROR keyword processing was introduced into 240 by Enhancement APAR 2400098.

- **FASTRUN CONCMDQ** support provides the ability to execute a string of console commands (CONCMDQ=*cmdvalues*) during the FASTRUN process. This can be very useful to set, display and test various console commands sequences after the CDF scan without actually starting HNAS (HNAS doesn't initiate any conversations with VTAM resources or TCPIP Stack processes in FASTRUN mode).
- **FASTRUN AMNF** generation support now propagates VTAM parameters from the MCH to all GATE control session SLU APPL statements for the MCH. Prior to this change, the VTAM parameters for GATE control session SLUs defaulted to DLOGMOD=INTERACT,EAS=1 only.

Note: GATE control session SLU VTAM parameter change was introduced into 240 as Enhancement APAR 2400029.

- **FASTRUN AMNF** generation support now propagates VTAM parameters from the MXT associated with a PVC SLU to the SLU APPL statement (MCH VTAM parameters used when MXT not coded). Prior to this change, the VTAM parameters for PVC SLUs came from the MCH only.

Note: PVC SLU VTAM parameter change was introduced into 240 as Enhancement APAR 2400029.

- **INIT={ACTIVE|IDLE}** is now valid for TYPE=MCH REMOTE definition statements. INIT=ACTIVE is the default. INIT=IDLE will cause the MCH to be initially inactive. No calls to or from this MCH can take place until it is activated via the new VARY MCH ON console command (see below) or the MRMT INIT=ACTIVE console command which invokes VARY MCH ON.
- **INIT={ONLINE|OFFLINE}** is now allowed as alternate values for ACTIVE|IDLE, respectively. The **DPCE** console command display for a LOCAL or REMOTE in IDLE state will depict OFLN as the current socket state.

Note: INIT=ONLINE|OFFLINE support was introduced into 240 as Enhancement APAR 2400014.

- **LUNAME=(...,sluname-{A|I}),...** is now allowed to specify an initial state for the named SLU.

-A (or omitted) is for active and -I is for inactive or idle with active being the default.

Note: *sluname*-{A|I} support was introduced into 240 as Enhancement APAR 2400028.

- **LUNAME=** operand has been modified to accept a **GENTYPE** value which is used to control hex versus decimal SLU name generation: LUNAME=(*sluname*-{A|I}/*pluname*/*pxlu*-{H|D}/*sfxst*/*cnt*,...)

When *pxlu*-D is specified, SLU names are generated in ascending decimal order starting with the *sfxst* value for number of SLUs identified by the *cnt* value.

When *pxlu*-H or *pxst* (without the -H) is specified, SLU names are generated in ascending hexadecimal order starting with the *sfxst* value for number of SLUs identified by the *cnt* value. This is how SLU names have always been generated prior to APAR 2400035.

Note: *pxlu*-{H|D} support was introduced into 240 as Enhancement APAR 2400035.

- **OPTIONS=ALMSGTXT={LONG|SHORT}** has been added to the BUILD definition statement. This OPTION allows you to select the long (default) or short format for alarm messages. Heretofore, only long format alarm messages were provided which can produce multiple lines of output for each alarm. Short format alarm messages consolidate information on a single line. The new OPTIONS=ALMSGTXT= operand effects various error and informational alarms.

For callin/callout informational messages (**NAS7nnnl**), see their description later in this chapter.

For TCPIP (**NAS2nnns**), VC (**NAS5nnns**) and QLLC (**NAS8nnns**) alarm messages that have the following LONG format:

```
NAScnnns comp=iii.iii.iii.iii(port) SOCKID=sockid PCEID=pceid NAME=compname
NAScnnns information text
```

The new SHORT format will appear as follows:

```
NAScnnns comp=iii.iii.iii.iii(port) NAME=compname information text
```

As you can see, the SOCKID= and PCEID= fields have been removed for the SHORT format so the alarm message can fit on a single line.

For these messages, **c** is the component ID (2|5|8), **nnn** is the message number, **s** is the message severity (I|W|E|S) and **comp** is the component type (SERVER or CLIENT).

Note: BUILD OPTIONS=ALMSGTXT={SHORT|LONG} support was introduced into 240 as Enhancement APAR 2400068.

- **OPTIONS=DELAYBINDRESP** has been added to the TYPE=MCH REMOTE statement. This OPTION causes HNAS to delay the response to a BIND that triggers an LLC0 or LLC5 callout operation until the call succeeds (call accept received, +RSP) or fails (clear or is timeout, -RSP). The sense data for a -RSP will be 0801C3D9 - resource not available.

HNAS V2R4M0 New Features

- **OPTIONS=CUD0SELECTSLU** has been added to the TYPE=MCH REMOTE definition statement. This OPTION allows the CUD0 byte from an incoming Call Request packet to participate in LU name selection. This provides a capability found in ISARDX25 (IBM Spain) 37xx support.

Note: OPTIONS=CUD0SELECTSLU support was introduced into 240 as Enhancement APAR 2400074.

- **OPTIONS=IDTST** has been added to the TYPE=DFX REMOTE definition statement. This OPTION implements the ISARX25 Datafono IDTST option which ensures that PLU messages are being sent to the correct SLU.

Note: OPTIONS=IDTST support was introduced into 240 as Enhancement APAR 2400074.

- **OPTIONS=NOCLOSEONTAPFAILURE** has been added to the TYPE=XOT|XTP REMOTE definition statement. This OPTION allows active sockets to remain active when router contact is lost after 2 consecutive TAP (Keep Alive) failures. This option was added for customers who would like to activate TAP (XOT Keep Alive) processing without taking a REMOTE down (out of service) if a TAP Contact Lost failure occurs.

Note: OPTIONS=NOCLOSEONTAPFAILURE support was introduced into 240 as Enhancement APAR 2400055.

- **OPTIONS=PVCRECONTMR=secs, PVCSETUPTMR=secs and PVCSETUPREJ=code** have been added to the TYPE=MCH|MXT REMOTE definition statement.

PVCRECONTMR= specifies how often HNAS tries to re-establish a session between a PVC VC and the PLU. The VTAM session with the PLU is established after the PVC session between HNAS and the router is established by PCV SETUP packets. If the PVC's VTAM session is ended (PLU UNBIND, Notify or TPEND) then the value specified by this operand controls how often HNAS will try to reconnect the session with the PLU. If PVCRECONTMR= is omitted for a TYPE=MCH REMOTE, a default value of 60 seconds will be used. If PVCRECONTMR= is omitted for a TYPE=MXT REMOTE that is associated with a PVC, no default value is set so that the root MCH value will be used.

PVCSETUPTMR= specifies how often HNAS should send PVC SETUP packets to establish a PVC session between HNAS and the router. The option only has meaning when the PVC definition in the HNAS CDF identifies the router to be used for the session (see PVC= operand). If PVCSETUPTMR= is omitted for a TYPE=MCH REMOTE, a default value of 60 seconds will be used. If PVCSETUPTMR= is omitted for a TYPE=MXT REMOTE that is associated with a PVC, no default value is set so that the root MCH value will be used.

PVCSETUPREJ= specifies a PVC SETUP reject status code that will be used any time HNAS receives a SETUP for the PVC. Since a status code greater than X'0F' causes the router to stop sending SETUP packets for the PVC, this option may be used to shut down PVC SETUP initiation by the router. If PVCSETUPREJ= is omitted, SETUP packets are processed normally (status code to connect or reject the session generated by HNAS). This parameter was initially implemented for PVC setup testing but also includes benefits as described above.

Note: OPTIONS=PVCRECONTMR=,PVCSETUPTMR=,PVCSETUPREJ= support was introduced into 240 as Enhancement APAR 2400059.

- **OPTIONS=RESETINO** has been added to the TYPE=MCH REMOTE definition statement. This OPTION specifies that HNAS is to terminate the session with the PLU when a RESET is received from the remote.

Note: OPTIONS=RESETINO support was introduced into 240 as Enhancement APAR 2400081.

- **OPTIONS=SVCCALLTMR=secs** has been added to the TYPE=MCH|MXT REMOTE definition statement to specify how long HNAS should wait for a response to a transmitted Call Request packet before it assumes that the call has failed because there was no response. Note that T21= can be specified as an alternate name for the SVCCALLTMR= suboperand.

Note: OPTIONS=SVCCALLTMR= support was introduced into 240 as Enhancement APAR 2400069. Prior to the APAR a fixed 30 second timer was used.

- **OPTIONS=(TIMESTAMP=format-*{+D|-D}*)** was added to the BUILD definition statement to enhance SYSPRINT timestamp resolution.

Note: Please refer to the **Enhancements - Expanded Content** heading at the end of this section for additional information regarding this new or enhanced feature.

- **PARM= (EXEC)** operand now supports abbreviated start parameters reducing the likelihood that the **PARM=** operands 100 character limit will be exceeded. For example, **TLU MXDT** can now be specified instead of **TRCLU MAXDATA** which conserves 5 characters of PARM= operand space.

Note: Abbreviated PARM= operand support was introduced into 240 via APAR 2400048.

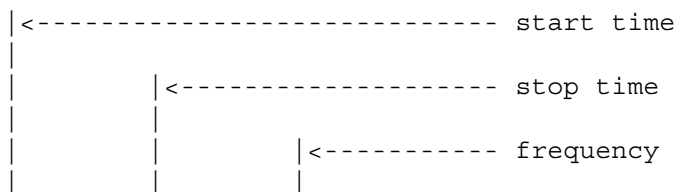
- **PARSE** definition statement has been added to allow customers to specify their own parsing characters that will override established HNAS parsing characters for the CDF scan. The PARSE definition statement must be the first definition statement in the CDF and must precede any CDF records that utilize the special parsing characters that it defines. After the PARSE definition statement is processed, the override parsing characters will go into effect turning the established parsing characters into non-parsing characters. They will simply become data characters.

Note: Please refer to the **Enhancements - Expanded Content** heading at the end of this section for additional information regarding this new or enhanced feature.

- **PRTSWLST=({LOOP|STOP},SWITCHAFTERINIT,SWITCHAT *time*, {ddname1|DYNAMIC=*outclass*},...,{ddnamen|DYNAMIC=*outclass*})** was added to the BUILD definition statement to provide automatic SYSPRINT switching when the current SYSPRINT log file becomes full or when the designated action occurs. You may specify static DDNAMEs and/or request DYNAMIC DDNAME allocation. The DDNAMEs you specify or request dynamically are used sequentially. The default SYSPRINT file is always used initially (ddname=SYSPRINT).

Note: Please refer to the **Enhancements - Expanded Content** heading at the end of this section for additional information regarding this new or enhanced feature.

- **PULSE=(*hh:mm:ss, hh:mm:ss, seconds*)** operand added to the BUILD definition statement for heartbeat message support. The syntax and suboperand meanings is as follows:



HNAS V2R4M0 New Features

V V V

PULSE= (hh:mm:ss, hh:mm:ss, seconds)

HNAS will issue the following message at the frequency given by the *seconds* value within the interval specified by the start and stop times:

```
NAS0299I HNAS PULSE TAKEN AT hh:mm:ss ON yyyy/mm/dd
```

If the stop time is less than the start time, the interval wraps through midnight. If the start time and end time are equal, pulsing will be continuous.

The purpose of the message is to provide an indication that HNAS is being dispatched on a regular basis. The SYSPRINT log will contain the NAS0299I message. The message can also be sent to SYSCONS if ALRMFLTR=(...,NAS0299I(FU),...) is also specified.

Note: PULSE= operand support was introduced into 240 via APAR 2400021.

- **PVC=(...,sluname{-{A|I}},...)** is now allowed to specify an initial state for the named SLU.

-A (or omitted) is for active and -I is for inactive or idle with active being the default.

Note: *sluname*{-A|I} support was introduced into 240 as Enhancement APAR 2400028.

- New **TYPE=DMY REMOTE** was added to improve automation of console command capabilities as well as improvements for the HNAS **PING (XOT) dmyname** console command operation.
- **RTEIN=** processing allows **SKIP/dce-addr** or **CLEAR/dce-addr** to be specified as entries in the RTEIN= list. If SKIP is selected by a DTE address the RTEIN= entry is skipped. If CLEAR is selected by a DTE address then the inbound call is cleared with DIAG=202. The **NAS7701W** alert message has been modified to contain the string 'RTEIN CLEAR' when a RTEIN CLEAR causes the clear.
- The **RTEIN=** and **RTEOUT= LOCAL** parameters now include support for DTE address calling or called address 'wildcards' filtering.

@ as the first *dteaddr* character tells HNAS to accept any DTE address whose trailing digits match the *dteaddr* digits that follow the @. For example, @6789 will allow DTE addresses of 123456789, 16789, 3216789, etc. to be considered a match. Use of this 'match at end' feature changes the operation of the STRIPRTEIN option -- please see Chapter 4 for details.

* within the *dteaddr* will be considered a wildcard character. For example, 12**56 will allow DTE addresses of 120056, 120156, 121056, 120256, etc. to be considered a match. The @ and * characters can be used together.

- **RTEOUT=** processing has been modified to allow a descriptor text value (*desc txt*) to be used in place of the *dteaddr* value in the NAS7717W alarm message. Like *dteaddr*, *desc txt* is limited to 15 characters. This will preserve the format of the NAS7717W message. The new syntax is RTEOUT=(*rmtname/dteaddr*{T|S}=*desc txt*,...). *desc txt* can be enclosed in quotes. For example, RTEOUT=(XOTCLNT1/1234T='RMT USER #1',...). When *desc txt* is provided, the NAS7717W message will display DTE IDNT desc txt instead of DTE ADDR dteaddr.
- **SCHEDULE={ddname}(hh:mm:ss,cmd,....,hh:mm:ss,cmd)** operand support added to allow scheduled console commands.

As an aid to problem diagnosis, sometimes it is necessary to start/stop traces as well as issue display commands at specific times. Prior to the new SCHEDULE operand, this action could only be done manually via operator intervention. The new SCHEDULE= operand was added to allow commands to be scheduled for automatic execution at defined times.

Syntax: SCHEDULE={*ddname*|(*hh:mm:ss,cmd*,...,*hh:mm:ss,cmd*)}

(*hh:mm:ss,cmd*,...,*hh:mm:ss,cmd*) is an inline schedule list.

ddname identifies a file that contains a list of times and commands. A *ddname* schedule list can contain the following records:

Comments are allowed and start with an asterisk (*) or semi-colon (;) in record column 1. Comments can also appear on a command line but must start with a semi-colon after the command. A single time and command can be specified on a single record as follows:

hh:mm:ss , *cmd* ; *comment*

Multiple times and commands can be specified on a single record as follows:

(*hh:mm:ss* , *cmd* , ... , *hh:mm:ss* , *cmd*) ; *comment*

Note: The new SCHEDULE= operand support was introduced into 240 as Enhancement APAR 2400064.

- **SOCLMT=limit** operand now generates an error message if the sum of the TYPE=XOT REMOTE VCLMT= values (for which the LOCAL is HOME) is greater than *limit*-2. This eliminates a runtime ABEND during HNAS initialization.

Note: The SOCLMT= operand change was introduced into 240 as Enhancement APAR 2400077.

- **SVC0|4|5=(...,sluname[-{A|I}],...)** is now allowed to specify an initial state for the named SLU.

-A (or omitted) is for active and -I is for inactive or idle with active being the default.

Note: *sluname*-{A|I} support was introduced into 240 as Enhancement APAR 2400028.

- **SVC0|5={... (sluname/{Xidnum|dteaddr|<rmtname>}...)** permits an MXT|SVC per *dteaddr* and a *dteaddr* via *rmtname*. New **REMOTE TYPE=SVC** was added to provide a smaller MXT footprint without PADPARAM=*value* support.

Note: Please refer to the **Enhancements - Expanded Content** heading at the end of this section for additional information regarding this new or enhanced feature.

- **SVC0|5=(...,sluname+gluname[-{A|I}],...)** is now allowed to specify a generic SLU name (**gluname**) to be associated with a primary HNAS SLU name (**sluname**) for callout resources acquired by a PLU. The **gluname**, in conjunction with VTAM tables, allows a PLU to acquire an HNAS resource using a generic name.

Note: *gluname* support was introduced in 240 as Enhancement APAR 2400084.

- **TRCLMT=count** expanded from 32767 to 1048575 entries permitting extended tracing capabilities.

HNAS V2R4M0 New Features

Note: Expanded support was introduced into 240 as Enhancement APAR 2400052.

- **TRCTRAP ALRMLIST=(NASnnnni' data'/data-offset,...)** operand now accepts **data** as quoted string (see TRCTRAP console command enhancement below for more information).

See [Console and Trace Enhancements](#) for **PARM=** Start Parameter Changes & Improvements

Alert/Alarm Message and Clear/Reset Code Enhancements:

- **NASCnnni** console command response messages were added to improve console command decode processing. Please refer to the 'new for 240' references on the specific entries in the Console Messages section of the Messages and Codes guide for additional information.

- **NASC052W** console command response message issued if EXEC LIST is specified without command list (*ddname* or (*cmd1*,...,*cmdn*)).

NASC052W EXEC COMMAND LIST WAS NOT PROVIDED, REQUIRED FOR EXEC COMMAND

Note: NASC052W error message was introduced into 240 as Enhancement APAR 2400034.

Note: NASC052W error message was retired from 240 as part of Enhancement APAR 2400098.

- **NASC053W** console command response message issued if EXEC *ddname* identifies an empty command list file.

NASC053W EXEC *ddname* FILE IS NULL, EXEC COMMAND ABORTED

Note that a command list file is considered empty if it contains no commands or display comments (# text) but may contain non-display comments (* text').

Note: NASC053W error message was introduced into 240 as Enhancement APAR 2400034.

- **NASC100E** console command response message has been modified to more completely identify why a command that operates on the LNM=, RNM=, LUNM= or ID= modifiers is aborted when none of these modifiers are specified.

NASC100E ID= AND OTHER MODIFIERS OMITTED, REQUIRED FOR *cmdname* COMMAND EXECUTION
NASC100E ID= IS REQUIRED WHEN NO OTHER RESOURCE IDENTIFICATION IS SPECIFIED
NASC100E IF COMMAND OPERATES ON LNM=, RNM= OR LUNM=, ENSURE THAT RESOURCE NAME
NASC100E IS SPECIFIED BEFORE ANY COMMAND OPERANDS (EXAMPLE: V LUNM=*sluname* OFF)

Note: The new NASC100E message change was introduced into 240 by Enhancement APAR 2400098.

- **NAS003ni** alert messages now identify the type of shutdown as well as the actual end status.

NAS0030I SHUTDOWN STARTED, RELEASING RESOURCES

is issued when QS, QQ/*password* or QE *ddname* are entered (no SNAP dump is taken). This is a fast shutdown.

NAS0031I SHUTDOWN STARTED, FORMATTING STORAGE, RELEASING RESOURCES

is issued when QY/*password* is entered (SNAP dump is taken). The SNAP dump extends the shutdown processing and adds to SYSPRINT content.

NAS0032I SHUTDOWN STARTED, FORCED ABEND REQUESTED

is issued when QA/*password* is entered (ABEND dump is taken). The ABEND dump extends the shutdown processing and adds to SYSPRINT content. It should only be used when requested by Comm-Pro.

NAS0035I SHUTDOWN COMPLETED, RC=xx

is issued when shutdown completes.

- **NAS01nni** alarm messages are now generated when the SHOW ON|OFF|ERR ... MORE|LESS ... console command is executed.

For ON

NAS0110I SYSCONS ERROR AND INFO ALARM OUTPUT ENABLED BY *consname*

For OFF

NAS0111W SYSCONS NON-FORCED ALARM OUTPUT DISABLED BY *consname*

For ERR

NAS0112I SYSCONS ERROR ALARM OUTPUT ENABLED BY *consname*

For MORE|LESS

NAS0130W SYSCONS ALARM OPTIONS MODIFIED BY *consname*

Note: These messages are only issued if a change is effected by the command, for example from the ON to OFF state. The messages are withheld if the command does not effect a change, for example when ON is specified and the ON state is already active (effectively, a no operation).

Note: The new NAS01nni alarm messages were added to 240 as part of Enhancement APAR 2400017.

- **NAS020ni** alert messages are now generated when the new PRTSWLST= SYSPRINT switching support is enabled providing SYSPRINT status notification:

NAS0205I *sysprint* LOGGING TERMINATED

This message is issued after the SYSPRINT log file identified by DDNAME *sysprint* has been closed during SHUTDOWN processing. This message is issued unconditionally even when PRTSWLST= support is not being used.

NAS0207W ALL PRTSWLST DDNAMES HAVE BEEN USED, LOGGING TERMINATED

This message is issued after the last SYSPRINT log file in the PRTSWLST= operand list has been closed and the stop action is in effect (PRTSWLST=(STOP,...) was specified). The PRNT OPEN

HNAS V2R4M0 New Features

ddname console command will have to be used to manually open a new SYSPRINT log file.

Note: This message will not be issued if the LOOP action is in effect (PRTSWLST=(LOOP,...) was specified).

NAS0208I *sysprint* LOG FILE HAS BEEN ACTIVATED FOR LOGGING

This message is issued when the SYSPRINT log file identified by DDNAME *sysprint* has been opened automatically as the next log file in the PRTSWLST= operand list.

NAS0209E *sysprint* LOG FILE COULD NOT BE OPENED

This message is issued when the SYSPRINT log file identified by DDNAME *sysprint* could not be opened. If alternate SYSPRINT file(s) have been defined via the PRTSWLST= operand on the BUILD definition statement, a new SYSPRINT log file will be opened automatically. Otherwise, the PRNT OPEN *ddname* console command will have to be used to manually open a new SYSPRINT log file.

- **NAS02nni** alarm messages are now generated when the PRNT ON|OFF ... DATE ON|OFF ... console command is executed.

For ON

NAS0210I SYSPRINT LOGGING ENABLED BY *consname*

For OFF

NAS0211W SYSPRINT LOGGING DISABLED BY *consname*

For DATE ON|OFF, QLLC ON|FF, TCP ON|OFF ...

NAS0230W SYSPRINT LOGGING OPTIONS MODIFIED BY *consname*

Note: These messages are only issued if a change is effected by the command, for example from the ON to OFF state. The messages are withheld if the command does not effect a change, for example when ON is specified and the ON state is already active (effectively, a no operation).

Note: The new NAS02nni alarm messages were added to 240 as part of Enhancement APAR 2400017.

- **NAS0210I** alert messages are now generated when a PRTSWLST= event occurs:

NAS0210I PRTSWLST SWITCHNOW REQUESTED BY *consname*

NAS0210I PRTSWLST SWITCHAFTERINIT ACTION INITIATED

NAS0210I PRTSWLST SWITCHATHh00 ACTION INITIATED

NAS0210I PRTSWLST SWITCH ON FULL DATASET INITIATED

NAS0210I PRTSWLST SWITCH ON PRTLMT INITIATED

These messages are issued when the a switch occurs in the PRTSWLST= operand of the BUILD definition statement. The text indicates why the SYSPRINT log file switch was requested. *consname* identifies the name of the console that issued the PRNT command. For the local console (SYSCONS), this is WACONPCE. For a remote console, this is RCONnnnn where nnnn is the remote console number.

- **NAS022ni** alert messages are now generated when the new **PRNT {OPEN|CLSOPN}** console command is used to open or close SYSPRINT files:

```
NAS0220W sysprint LOG FILE OPENED BY consname
```

This message is issued when the **PRNT {OPEN|CLSOPN}** *sysprint* console command is used to open a new SYSPRINT file.

sysprint is the new SYSPRINT DDNAME that was opened. *consname* identifies the name of the console that issued the PRNT command. For the local console (SYSCONS), this is WACONPCE. For a remote console, this is RCON*nnnn* where *nnnn* is the remote console number.

```
NAS0221W sysprint LOG FILE CLOSED BY consname
```

This message is issued when the **PRNT {CLOSE|CLSOPN newddname}** console command is used to close an old SYSPRINT file.

sysprint is the old SYSPRINT DDNAME that was closed. *consname* identifies the name of the console that issued the PRNT command. For the local console (SYSCONS), this is WACONPCE. For a remote console, this is RCON*nnnn* where *nnnn* is the remote console number.

- **NAS0299I** messages generated for **PULSE** heartbeat message. See description of new PULSE= operand above for additional information.
- **NAS0310W** alert message is issued when the VARY console command is used to change a resource state.

```
NAS0310W VARY ACTION=action TYPE=RMT RNM=rmtname BY consname
NAS0310W VARY ACTION=action TYPE=LCL LNM=lclname BY consname
NAS0310W VARY ACTION=action TYPE=LU LUNM=sluname BY consname
```

The **action** value can be ON or OFF. **rmtname**, **lclname** or **sluname** identify the name of RMT, LCL or LU resource be varied ON or FF. **consname** identifies the name of the console that issued the VARY command. For the local console (SYSCONS), this is WACONPCE. For a remote console, this is RCON*nnnn* where *nnnn* is the remote console number.

- **NAS1n11E** messages are now issued when an incorrectly formatted table is specified for the USSTAB=, LOGTAB=, or OPTIONS=NRITAB= operand. For example:

```
NAS1311E REMOTE mchname LOGTAB=ISTINCDT IS NOT A VALID LOGON TABLE, REQUIRED
```

Note: New NAS1n11E message support was introduced into 240 as Enhancement APAR 2400032.

- **NAS1154E** message is issued when SCHEDULE=*ddname* queue limit is exceeded.

```
NAS1154E BUILD SCHEDULE ddname COMMAND QUEUE LIMIT count/limit
EXCEEDED, REQUIRED
```

The **ddname** value identifies the DDNAME of the file that is too large. **count** represents the number of bytes required for the file and **limit** represents the maximum number of bytes allowed for any file.

Note: The new SCHEDULE= operand support was introduced into 240 as Enhancement APAR

HNAS V2R4M0 New Features

2400064.

- **NAS1301D** message is issued when an SLU initial value is omitted for an SLU named in the LUNAME=, PVC=, SVC0=, SVC4= or SVC5= operand. For example:

```
NAS1301D REMOTE operand SLU INIT VALUE (-{A|I}) OMITTED FOR ONE OR MORE
NAS1301D          ENTRIES STARTING WITH operand XXXX, -A (ACTIVE) ASSUMED
```

Note: *sluname*-{A|I} support was introduced into 240 as Enhancement APAR 2400028.

- **NAS1311S** message is now issued when the sum of the TYPE=XOT REMOTE VCLMT values (for which the LOCAL is HOME) is greater than SOCLMT=*limit* value minus 2. The new NEEDED=*count* value was added to show how many additional sockets are required.

```
NAS1311S REMOTE XOTCNOT2 HOME XOTSRVR LIMIT REACHED
          SOCCNT=00303 SOCLMT=00602 NEEDED=00301
```

This message is issued when NEEDED+SOCCNT>SOCLMT. SOCCNT includes 2 reserved sockets.

Note: The NAS1311S message change was introduced into 240 as Enhancement APAR 2400077.

- **NAS1nnni** configuration messages updated in an effort to provide improved configuration error messages troubleshooting as well as adding messages for new 240 CDF parameters and options
- **NAS2109S** alarm message is now issued when an **unexpected TCPIP interrupt** is presented and the DBUG TCP start parameter is in effect.

```
NAS2109S CLIENT=010.117.056.100(04545) SOCKID=0001 PCEID=000C NAME=R1CNIN
NAS2109S UNEXPECTED TCPIP INTERRUPT, IGNORED
NAS2109S IPARM=0001850200000024001300FAD5C1E2E3F0E3E2D6
          0000D740004EA808004EAB9000000000000508C00
          0000000001000000001348352413483533
```

When DBUG TCP is not active, the following 0198 ABEND will occur when an unexpected TCPIP interrupt is presented:

```
HALT AT LOC 80065CB4 IN NASUTIL : TCPIP INTERRUPT UNEXPECTED
```

- **NAS2110S** alarm message is now issued when an **invalid TCPIP reply ID** is detected and the DBUG TCP start parameter is in effect.

```
NAS2110S SERVER=172.029.127.220(01998) SOCKID=0000 PCEID=0009 NAME=LXOT
NAS2110S INVALID TCPIP INTERRUPT REPLY ID, IGNORED FOR SELECT
NAS2110S EXPECTED=0000001D PRESENTED=0000001C
NAS2110S IPARM=000185020000001C001300FAD5C1E2E3F0E3E2D6
          0000D740004E8C18004E8FA0000000000005187E0
          00000000000000001209175512101732
```

When DBUG TCP is not active, the following 0198 ABEND will occur when an invalid TCPIP reply ID is presented:

```
HALT AT LOC 80070952 IN NASTCP : TCPIP REPLY ID FAILURE
```

- **NAS2152E** alarm message is now issued when a CANCEL command timeout occurs.

```
NAS2152E CLIENT=010.117.056.100(04545) SOCKID=0001 PCEID=000C NAME=R1CNIN
NAS2152E CANCEL REQUEST INTERRUPT LOST, NORMAL COMPLETION ASSUMED FOR CONNECT
```

- **NAS2502E** message now displays **CLOSEONTAPFAILURE** when a router contact is lost and the CLOSEONTAPFAILURE option is in effect. This is the default option when NOCLOSEONTAPFAILURE is not specified in the CDF.
- **NAS2505E** message was added and displays **NOCLOSEONTAPFAILURE** when a router contact is lost and the NOCLOSEONTAPFAILURE option is in effect.

Note: [NO]CLOSEONTAPFAILURE support was introduced into 240 as Enhancement APAR 2400055.

- **NAS2511M** message that is issued for **MON TAP** now contains a retry count (*dddd*). The count is reset when a successful TAP response is received. The count indicates the number of unsuccessful TAP attempts.

```
NAS2511M CLIENT=010.117.056.100(02704) SOCKID=0001 PCEID=000B NAME=R1CNIN
NAS2511M XOT TAP TIMEOUT, RESPONSE NOT RECEIVED FOR CONNECTION SETUP (dddd)
```

Note: Support was introduced into 240 as Enhancement APAR 2400055.

- **NAS251nM** messages that are issued for **MON TAP** console command activity are normally logged in SYSPRINT only can now (optionally) be displayed on the SYSCONS using the ALRMFLTR= operand (FC) or (FU) suffixes.

```
NAS2513M CLIENT=010.117.056.100(02190)
      SOCKID=0002 PCEID=000B NAME=R1CNIN
NAS2513M XOT TAP SEQUENCE IS STARTING,
      TRANSMITTING CALL REQUEST
NAS2513M PKT=0000001010010B000001000000C8D5C1E2E3C1D7
```

```
NAS2515M CLIENT=010.117.056.100(02190)
      SOCKID=0002 PCEID=000B NAME=R1CNIN
NAS2515M XOT TAP SEQUENCE IN PROGRESS,
      RECEIVED CLEAR REQUEST
NAS2515M PKT=000000051001130342
```

```
NAS2513M CLIENT=010.117.056.100(02190)
      SOCKID=0002 PCEID=000B NAME=R1CNIN
NAS2513M XOT TAP SEQUENCE IN PROGRESS,
      TRANSMITTING CLEAR CONFIRM
NAS2513M PKT=00000003100117
```

```
NAS2517M CLIENT=010.117.056.100(02190)
      SOCKID=0002 PCEID=000B NAME=R1CNIN
NAS2517M XOT TAP SEQUENCE COMPLETED SUCCESSFULLY
```

HNAS V2R4M0 New Features

- **NAS251nM** messages that are issued for **MON TAP** can now provide information similar to **PING** console command activity as requested by user via the **PKTDATA|MAXDATA|MINDTA|NODATA** arguments. For **MAXDATA**:

```
NAS2513M CLIENT=010.117.056.100(02190)
          SOCKID=0002 PCEID=000B NAME=R1CNIN
NAS2513M XOT TAP SEQUENCE IS STARTING,
          TRANSMITTING CALL REQUEST
NAS2513M DTEADDR=ddd...ddd DCEADDR=ddd...ddd
NAS2513M FAC=xx...xxx
NAS2513M CUD=xx...xxx
```

Note: NAS251nM DTEADDR=, etc. display support was introduced into 240 as Enhancement APAR 2400044.

- **NAS261nM** messages are now issued for **PING** console command activity will be logged in **SYSPRINT** and (optionally) displayed on the **SYSCONS** using the **ALRMFLTR=** operand (FC) or (FU) suffixes.

```
NAS2613M PINGING SOCKET=aaa.bbb.ccc.ddd(port) FOR consname
NAS2613M          DTEADDR=ddd...ddd DCEADDR=ddd...ddd
NAS2613M          FAC=xx...xxx
NAS2613M          CUD=xx...xxx
```

```
NAS2611M PING TIMED OUT FOR consname
```

```
NAS2615M PING COMPLETED FOR consname, CLEAR REQUEST
          RECEIVED, CAUSE/DIAG=ccdd
```

```
NAS2617M PING COMPLETED FOR consname, CALL ACCEPT
          RECEIVED
```

- **NAS261nM** messages have been updated for **PING** console command to display a sequence number so that termination messages (NAS2611M, NAS2615M, NAS2617M) can be easily correlated to initiation messages (NAS2613M). In addition, termination messages now display the target IP and DTE addresses. The new message formats are as follow:

```
NAS2613M PINGING SOCKET=aaa.bbb.ccc.ddd(port) FOR consname (seqno)
NAS2613M          DTEADDR=ddd...ddd DCEADDR=ddd...ddd
NAS2613M          FAC=xx...xxx
NAS2613M          CUD=xx...xxx
```

```
NAS2611M PING CALL TIMED OUT FOR SOCKET=aaa.bbb.ccc.ddd(port)
          DTEADDR=ddd...ddd (seqno)
```

```
NAS2615M PING CALL CLEARED (cc/dd) FOR SOCKET=aaa.bbb.ccc.ddd(port)
          DTEADDR=ddd...ddd (seqno)
```

```
NAS2617M PING CALL ACCEPTED FOR SOCKET=aaa.bbb.ccc.ddd(port)
          DTEADDR=ddd...ddd (seqno)
```

Note: PING NAS261nM message support was updated to include the correlation sequence number (**seqno**) and IP and DTE addresses in the response messages effective with APAR 2400039.

- **NAS1391I** messages now issued to display MXT operand overrides. When a user associates an MXT with a PVC= or SVC0/3/5= operand entry, he is asking HNAS to override operands on the root MCH with those from the MXT. In some cases, this can lead to confusion if one does not remember to review the HNAS documentation to see which operands can be overridden based on the associated resource type.

The configuration logic has been modified to display, for each SLU or SPU defined in the PVC= or SVC0/3/5= operands with an associated MXT (or SVC), those MXT operands that will override operands of the same name on the root MCH. Not all MXT operands are valid overrides for all SLUs or SPUs. For example, IDLETO= is a valid override when specified for an MXT that is associated with a SVC0/5= SLU but is not used for a PVC SLU.

MXT override messages are listed at the end of the CDF scan in NAS1391I messages. Only those MXT operands that are specified and, of these, only those that are valid as overrides for an SLU or SPU are displayed. If an MXT operand is not listed, it is because it was either omitted from the MXT (and hence cannot override the MCH operand) or, if specified, is not valid as an override for the SLU or SPU resource. See expanded content for additional information.

Note: NAS1391I message support was introduced into 240 as Enhancement APAR 2400018.

- **NAS3796I** alert message has been added to indicate that a Fast Connect GATE CTCP has sent HNAS a call accept packet to the named HNAS SLU. This indicates the start of a GATE FC session initiated by an inbound call.

```
NAS3796I FC LU luname ON MCH mchname REC'D CALL ACCEPT FROM pluname FOR  
SESSION WITH ip-addr(port)
```

- **NAS3797I** alert message is now generated when a GATE control session is bound. Previously this alert was only used for BINDs received for PVC LUs. This will eliminate the requirement to view the HNAS DLU command display output to confirm that the GATE control session is active.

```
NAS3797I LU lu-nm RECEIVED BIND FROM PLU plu-nm
```

Note: This logic change was introduced into 240 under APAR 2400005.

- **NAS5705W** RESET SCHEDULED alert message has been modified to provide **DIAGX=0000|0001** values in an effort to better define the cause of the PVC reset. The HNAS Reset Diagnostic code table entry 196 (C4) now contains definitions for these new values.

Note: This logic change was introduced into 240 under APAR 2400053.

- **NAS5723W** messages issued for IDTST= ID String length exception, sample follows:

```
NAS5723W DATAFONO IDTST LU lu-name REC'D ID STRING W LENGTH > 11
```

Note: This logic change was introduced into 240 as part of Enhancement APAR 2400074.

- **NAS5724W** messages issued for IDTST= 'M' messages response not received in 'nn' seconds, sample follows:

```
NAS5724W LU lu-name DID NOT RECEIVE 'M' MSG SYNC RESP FROM REMOTE
```

HNAS V2R4M0 New Features

Note: This logic change was introduced into 240 as part of Enhancement APAR 2400074.

- **NAS5725W** messages issued for IDTST= Bad ID condition, sample follows:

```
NAS5725W LU lu-name DISCARDING MSG W BAD ID FROM PLU plu-name
```

Note: This logic change was introduced into 240 as part of Enhancement APAR 2400074.

- **NAS5726W** messages issued when EMSGE option was specified for a Datafono session and the 16 second PLU response timer expired, sample follows:

```
NAS5726W EMSGE MESSAGE SENT TO LU lu-name (16 SEC PLU DELAY)
```

Note: This logic change was introduced into 240 as part of APAR 2400097.

- **NAS7701W** alert message has been modified to contain the string 'RTEIN CLEAR' when a RTEIN CLEAR causes the clear. This change was implemented to support the new **RTEIN=** processing which allows **SKIP/dce-addr** or **CLEAR/dce-addr** to be specified as entries in the RTEIN= list.

```
NAS7701W CALL FROM iii.iii.iii.iii(port) CAN'T ROUTE CALLED ADDR  
ldddddd ddddddd [RTEIN CLEAR]
```

- **NAS7716W** this GATE call request failure alert message will display as a single line alert if OPTIONS=ALRMSGTXT=SHORT is coded on BUILD (APAR 2400068).
- **NAS7717W** alert message will now display the descriptor text value (*desctxt*) instead of the *dteaddr* value when *desctxt* is coded on the RTEOUT=(*rmtname/dteaddr/desctxt,...*) operand.

```
NAS7717W LU sluname CALL TO DTE IDNT desctxt VIA REMOTE rmtname FAILED
```

This call request failure alert will display as a single line alert message if OPTIONS=ALRMSGTXT=SHORT is coded on BUILD (APAR 2400068).

- **NAS7718T** this trace message and 3 **NAS7798T** messages are recorded in SYSPRINT when TRCMCH ICR ON is set and a call request is received. When OPTIONS=ALRMSGTXT=SHORT is coded on BUILD (APAR 2400068) then the NAS7718T and NAS7798T messages are replaced with a **NAS7730I** message which is sent to the operator's console and SYSPRINT. This allows job log monitoring programs to see inbound call activity.
- **NAS7719T** this trace message and 3 **NAS7798T** messages are recorded in SYSPRINT when TRCMCH OCR ON is set and a call request is sent by HNAS. When OPTIONS=ALRMSGTXT=SHORT is coded on BUILD (APAR 2400068) then the NAS7718T and NAS7798T messages are replaced with a **NAS7731I** message which is sent to the operator's console and SYSPRINT. This allows job log monitoring programs to see outbound call activity.
- **NAS7730I** this single line message records an inbound call request on the operator's console and SYSPRINT when TRCMCH ICR ON is active and OPTIONS=ALRMSGTXT=SHORT is coded on BUILD.
- **NAS7731I** this single line message records an outbound call request on the operator's console and SYSPRINT when TRCMCH OCR ON is active and OPTIONS=ALRMSGTXT=SHORT is coded on BUILD.

- **NAS7798T** these messages follow NAS7718T or NAS7719T messages when TRCMCH ICR/OCR ON is active an OPTIONS=ALRMSGTXT=LONG (the default) is coded on BUILD. If ALRMSGTXT=SHORT is coded these messages do not appear in SYSPRINT.
- **NAS92nns** messages added for additional authorization processing (please see HNAS Messages and Codes documentation for details).
- **NAS9205I** message is now withheld until today's date is within 60 days of the EOTDATE. In addition, the message severity is changed from 'I' to 'S' as today's date gets closer to the EOTDATE. Specifically, the message ID is generated as follows:

NAS9206I, when TODAYSDATE >= EOTDATE-60
NAS9206W, when TODAYSDATE >= EOTDATE-45
NAS9206E, when TODAYSDATE >= EOTDATE-30
NAS9206S, when TODAYSDATE >= EOTDATE-15

Note: The new NAS9205I message change was introduced into 240 by Enhancement APAR 2400106.

- **NAS9206I** message is now withheld until today's date is within 60 days of the EOMDATE. In addition, the message severity is changed from 'I' to 'S' as today's date gets closer to the EOMDATE. Specifically, the message ID is generated as follows:

NAS9206I, when TODAYSDATE >= EOMDATE-60
NAS9206W, when TODAYSDATE >= EOMDATE-45
NAS9206E, when TODAYSDATE >= EOMDATE-30
NAS9206S, when TODAYSDATE >= EOMDATE-15

Note: The new NAS9206I message change was introduced into 240 by Enhancement APAR 2400098.

- HNAS **Clear Request** diagnostic codes **D8 (217)** and **D9 (218)** now provide special **DIAGX=** values in alert messages that replace validity checks in V2R4M0. Please contact customer support if one of these CAUSE/DIAGX combinations is seen.

D8 (217) Clear diagnostic code - New **DIAGX= codes 17-1A:**

17 VTAM SEND routine entered with RPL already busy.
18 VTAM MODCB macro failed.
19 SEND routine passed QLLC message with invalid FID byte.
1A SEND routine passed invalid QLLC LUSTAT request.

D9 (218) Clear diagnostic code - New **DIAGX= codes 05-0B:**

05 VTAM RECEIVE routine entered with RPL already busy.
06 Normal flow request value from exit routine invalid.
07 VTAM RECEIVE routine entered, required buffer not present.
08 VTAM RECEIVE completion entered, RPL flags do not indicate reason.
09 VTAM RECEIVE completion routine entered, LUHNFREQ indicates active command.
0A VTAM RECEIVE completion routine entered, RPL does not show completed command.
0B VTAM RECEIVE RESPONSE completion routine entered RPL does not show cmd for response

- HNAS **Clear Request** diagnostic codes **DF (223)** now provide new **DIAGX= codes 08|09** values:

HNAS V2R4M0 New Features

- 08 'M' message sync response not received from remote.
- 09 PLU sent data to pseudo leased LU with no VC session.

Note: This logic change was introduced into 240 as part of APAR 2400097.

Console, Trace and Debug Enhancements (Commands, Start Parameters...):

- When # is the first character of a console input record, the record is not processed as a command but is simply echoed after the HNASCMD-> prefix (for example, HNASCMD-> # *text*). The comment record may be specified like any other console input, that is, via remote command line input, the modify interface (/F *jobname*,# *text*), the CONCMDQ=(...,# *text*,...) operand or embedded in an EXEC *cmdlist* file.

Note: Console # comment processing was added to 240 as part of Enhancement APAR 2400017.

- All** commands now echo their invocation parameters following the HNASCMD-> display prefix and their execution parameters following the HNASXEQ-> display prefix. The execution parameters include specified and default modifiers and parameters. For example, if **TRCMCH MCH1 ON ICR** is entered, the following lines are echoed back to the console operator:

```
HNASCMD-> TRCMCH MCH1 ON ICR          <- echo entered command
HNASXEQ-> TRCMCH RNM=MCH1 ON GBL ICR ON <- show execution parameters
```

For additional information on these trace commands, please see Console Subsystem documentation or command HELP displays.

Note: HNASXEQ-> support was introduced into 240 as Enhancement APAR 2400014.

- All** commands that force another command to be executed now enqueue the second command in a LIFO rather than FIFO fashion so that the propagated command is executed immediately after the original command (for example, DMCH FMF3 -> DLU, MLCL *lclname* INIT={ACTIVE|IDLE} -> VARY *lclname* {ON|OFF}, MLCL *rmtname* INIT={ACTIVE|IDLE} -> VARY *rmtname* {ON|OFF})

Note: LIFO console command enqueue support was introduced into 240 as Enhancement APAR 2400028.

- LCLCONS** or *rconname* are now accepted as the first argument for the **ALARM**, **EXEC**, **SCHEDULE**, **SMSG** and **TRCCONS** commands. **LCLCONS** is the alias for the local console PCE (WACONPCE). *rconname* is the PCE name for a target remote console. Console PCE names can be displayed using the DPCE ID= RNM= TYPE=CONS command. These new arguments are provided for commands that operate on console PCEs.

- Notes:**
- 1) When the **LCLCONS** argument is specified, it directs command processing to the local console. When **LCLCONS** and *rconname* are omitted for the local console, **LCLCONS** is assumed.
 - 2) When the *rconname* argument is specified, it directs command processing to the named remote console. When **LCLCONS** and *rconname* are omitted for a remote console, *rconname* is forced to the name of the remote console.
 - 3) Console names can be provided by issuing the DPCE ID= RNM= TYPE=CONS command. Only consoles that are in the active state can be targeted by the

rconname parameter.

- 4) The LCLCONS and *rconname* arguments are valid for privilege consoles only when specified for the EXEC or SCHEDULE command. These commands are normally non-privileged while the ALARM command and all trace commands are privileged.
- 5) When a command is executed, the normal console command echo prefix is HNASCMD->. This prefix is changed when the command is the result of an EXEC command, SCHEDULE command, comes from another console or is the result of a TRAP. In these cases, the HNASCMD-> echo prefix is replaced with EXEC CMD->, SCHDCMD->, PRXYCMD-> or TRAPCMD->, respectively, so that these commands can be identified in the SYSPRINT log file.

Note: LCLCONS and *rconname* argument support was introduced into 240 with Enhancement APAR 2400075.

- **LCLCONS, NASUTIL, PING, TIMER, XOTUTIL, XTPUTIL** or *pcename* are now accepted as the first argument for the **DPCE, STATS, TRCBFR, TRCDATA, TRCDISP, TRCIO** and **TRCPCE** commands. LCLCONS, NASUTIL, PING, TIMER, XOTUTIL, XTPUTIL are the aliases, respectively, for the local console PCE (WACONPCE), the HNAS utility PCE (WANASPCE), the PING PCE (WAPNGPCE), the timer PCE (WATMRPCE), the XOT utility PCE (XOTUTIL) and the XTP utility PCE (XTPUTIL). *pcename* is the PCE name for a target PCE. PCE names can be displayed using the DPCE ID= RNM= command. These new arguments are provided for commands that operate on PCEs.

Notes: 1) *pestatic* = LCLCONS|NASUTIL|PING|TIMER|XOTUTIL|XTPUTIL

Alternate names for the *pestatic* argument are as follows:

```

WACONPCE = LCLCONS
WANASPCE = NASUTIL
WAPNGPCE = PING
WATMRPCE = TIMER
XOTUTIL  = XOTUTIL
XTPUTIL  = XTPUTIL
    
```

- 2) When the *pestatic* argument is specified, it directs command processing to named static PCE.
- 3) When the *pcename* argument is specified, it directs command processing to the named PCE.
- 4) When *pestatic* and *pcename* are omitted, the command operates off of the RNM= or ID= modifier. When the RNM= modifier is used, command processing is directed at the named REMOTE. When the ID= modifier is used, command processing is directed at the identified PCEs.
- 5) PCE names can be provided by issuing the DPCE ID= RNM= command.

Note: LCLCONS, NASUTIL, PING, TIMER, XOTUTIL, XTPUTIL and *pcename* argument support was introduced into 240 with Enhancement APAR 2400075.

- **HNASCMD->** command prefix can now be replaced with EXEC CMD->, SCHDCMD->, PRXYCMD-> or TRAPCMD-> depending on how the command was scheduled for execution.

HNAS V2R4M0 New Features

EXECCMD-> => command came from CONCMDQ= operand or EXEC command.
SCHDCMD-> => command came from SCHEDULE= operand or SCHEDULE command.
PRXYCMD-> => command came from another console.
TRAPCMD-> => command came from a trapped event.

This change requires the use of bits 0 and 1 in the command string length byte for all queued commands which restricts each queued command string length to 63-bytes.

Note: HNASCMD-> overlay support was introduced into 240 with Enhancement APAR 2400075.

- **ALARM FILTER=** now supports (FC) and (FU) message ID suffixes to allow forced conditional (subject to SHOW state) and forced unconditional (SHOW state ignored) SYSCONS display. For example, if ALARM FILTER=(...,NAS2020I(FC),NAS0299I(FU),...) is specified, NAS2020I alarm messages will be sent to SYSCONS except when SHOWOFF is effect while NAS0299I alarm messages will be sent to SYSCONS regardless of the SHOW state.
- The **ALARM LOG=?** command now displays date and time of last alert message log activity for each alarm message entry. This information is very useful when immediate access to SYSPRINT is not available.

ALARM LOG=?

```
ALARM_ID  TOTAL_CT    LOG_DATE  LOG_TIME    (0013/0256)
NAS0001I  00000001  2006/02/21  17:08:03
NAS0070W  00000001  2006/02/21  17:08:00
:
```

- The **ALARM LOG=?** command is now executed when HNAS is SHUTDOWN using the Q (QUIT) command. This is done unconditionally for any QUIT command follower. If QE *ddname* is entered, the ALARM LOG=? command is executed before the command list identified by *ddname*.
- The **ALARM MSGTXT={?|[LONG|SHORT]}** command support has been added to allow the BUILD OPTIONS=ALRMSGTXT= operand to be displayed and/or toggled. This OPTION allows you to select the long (default) or short format for alarm messages. Heretofore, only long format alarm messages were provided which can produce multiple lines of output for each alarm. Short format alarm messages consolidate information on a single line. This option applies to all alarm messages including informational alarms (see NAS773xl messages above for additional information).

When ALARM MSGTXT=? is entered, the current ALRMSGTXT= operand value is displayed.

When ALARM MSGTXT=LONG is entered, the alarm message format is set to long (old format).

When ALARM MSGTXT=SHORT is entered, the alarm message format is set to short (new format).

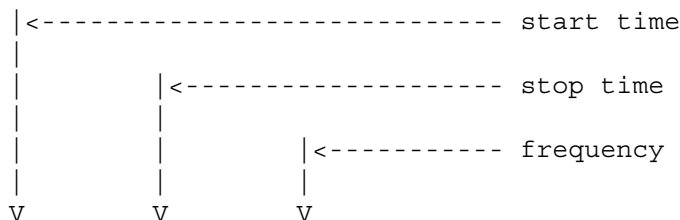
Note: ALARM MSGTXT= command support was introduced into 240 as Enhancement APAR 2400068.

- The **ALARM PULSE={?}|*(*hh:mm:ss, hh:mm:ss,seconds*)}** command support has been added to allow dynamic heartbeat message control.

When ALARM PULSE=? is entered, the current PULSE= operand values are displayed.

When ALARM PULSE=* is entered, the current PULSE= operand values are reset. Pulsing is stopped.

When ALARM PULSE=(*hh:mm:ss, hh:mm:ss, seconds*) is entered the syntax meaning is as follows:



ALARM PULSE=(*hh:mm:ss, hh:mm:ss, seconds*)

Note: Prior to updating the PULSE= operand values that are currently in effect, the values are copied into a temporary workarea. This allows you to modify one or all PULSE= suboperands before the values are returned to the permanent savearea. For example if you only want to change the PULSE frequency, enter ALARM PULSE=(*,,newseconds*). The start and end times will remain the same.

HNAS will issue the following message at the frequency given by the *seconds* value within the interval specified by the start and stop times:

```
NAS0299I HNAS PULSE TAKEN AT hh:mm:ss.uu ON yyyy/mm/dd
```

If the stop time is less than the start time, the interval wraps through midnight. If the start time and end time are equal, pulsing will be continuous.

The purpose of the message is to provide an indication that HNAS is being dispatched on a regular basis. The SYSPRINT log will contain the NAS0299I message. The message can also be sent to SYSCONS if ALRMFLTR=(*...,NAS0299I(FU),...*) is also specified.

Note: ALARM PULSE= command support was introduced into 240 via APAR 2400021.

- The **ALARM LCLCONS|*rconname* action** command support added to allow one console to modify alarm parameters for another console.

```

action = {ON|OFF} {TEST='text'}
          {FILTER={?|*|({A|P|S}, id1({A|P|S|FC|FU}), ..., idn({A|P|S|FC|FU}))}}
          {LIMITS={?|*|@| (tsec, ict, dct, act, wct, ect, sct, uct)}}
          {LOG={?|CLEAR|RESET}}
          {MSGTXT={?|SHORT|LONG}}
          {PULSE={?|*| (hh:mm:ss, hh:mm:ss, seconds)}}
          {SHOW}
  
```

Note: The ALARM LCLCONS|*rconname* action change was introduced into 240 with Enhancement APAR 2400075.

- **BFR** (internal) trace record now provides buffer allocation and release timestamp and tracestamp which will aid in diagnosing some types of problems. It is advantageous to know when a buffer was allocated and when it was released. Using the allocation timestamp and release timestamp, it can be determined how long a buffer has been in use. The current trace table pointer is also saved at allocation and release so these events can be correlated with trace activity.

HNAS V2R4M0 New Features

Note: This change was introduced into 240 as Enhancement APAR 2400051.

- **DLU** console command now displays the new LUTO column to show the active LU timer in effect (blank implies that no timer is running).

DLAY - QLLC DACTLU/ACTLU wait state, ACTLU issued on timer expiration.

REXP - QLLC PIU sent to SLU, response expected.

RQSD - REQSHUT sent to VTAM, SHUTDOWN expected.

RQSE - REQSESS sent to VTAM, BIND expected.

Note: The DLU LUTO change was introduced into 240 with Enhancement APAR 2400078.

- **DVC** console command now displays the new VCTO column to show the active VC timer in effect (blank implies that no timer is running).

CALL - SVC Call Request sent to DTE, response expected.

CLR - SVC Clear Request sent to DTE, response expected.

CLSD - QLLC SPU close wait state, open performed on timer expiration.

DISC - QLLC QDISC sent to SPU, response expected.

EMSG - Datafono EMSGE sent to DTE, response expected.

ICOL - QLLC SPU input collection in progress.

MSYN - Datafono MULTISYNC sent to DTE, response expected.

OMTR - QLLC SPU output metering in progress.

RCWT - PVC reconnect wait state (PVCRECONTMR=), REQSESS issued on timer expiration.

REXP - QLLC PIU sent to SPU, response expected.

RQSD - REQSHUT sent to VTAM, SHUTDOWN expected.

RQSE - REQSESS sent to VTAM, BIND expected.

STLG - REQSESS wait state (REQSESSDELAY=), REQSESS issued on timer expiration.

STMD - QLLC QSM sent to SPU, response expected.

STUP - PVC Setup sent to DTE, response expected.

SUWT - PVC Setup wait state (PVCSETUPTMR=), Setup issued on timer expiration.

TEST - QLLC QTEST sent to SPU, response expected.

XID - QLLC QXID sent to SPU, response expected.

Note: The DVC VCTO change was introduced into 240 with Enhancement APAR 2400078.

- The **DLU** and **DVC** console commands will now display the VC setup direction in the new **SESSINIT** column (old CID column) as **INBOUND** or **OUTBOUND**. This works for SVCs and PVCs. Blanks are displayed in the SESSINIT column if no VC connection is active. The CID can be displayed if the SHOWCID operand is provided for DLU and DVC.
- The **DLU** and **DVC** console commands will now display a 'P' in place of the leading zero for a PVC VCN value. The VCN for a PVC can range from 0 to 255 while the VCN display ranges from 0 to 4095 to accommodate both PVCs and SVCs.

Note: The PVC change for the DLU and DVC console commands was introduced into 240 as Enhancement APAR 2400028.

- The **DLU**, **DPCE** and **DVC** console commands will now display header information that is appropriate for both XOT and XTP resources. In the past, only XTP header information was displayed. For example, the DVC header will now look as follows:

IFN|

HNAS V2R4M0 New Features

MPID RPID VCN SESSINIT SLUNAME VCOPT VCST VCTO LLC CLGADDR

The MPID column identifies the PCE ID for the MCH that owns a VC. The IFN|RPID column identifies either the Interface Number for an XTP MCH or the PCE ID for the XOT REMOTE over which the VC connection is active.

Note: This new display header support was introduced into 240 as Enhancement APAR 2400014.

- The **DNAS** command has been modified to display additional customer specific content. These include the **DNAS COMMAND ENTERED AT hh:mm:ss ON 2006/07/31** entry depicting the time the DNAS command was entered, **CUSTID='cid_cust#'** display line, **CUSTINFO='text'** display line containing up to 32 characters of customer account information while the **MAINTENANCE/USE ANNIVERSARY DATE IS yyyy/mm/dd** display line contains the HNAS license maintenance/use coverage period. **DATAFONO SUPPORT IS INCLUDED** will display only when the distribution is enabled for Datafono support.

```
HNAS --> VERSION=V2R4M0 DIST=NON-SMP
        ASMDATE=2006/07/31 ASMHOST=ZOS
        RUNNING UNDER z/OS 01.04.00
        DNAS COMMAND ENTERED AT 11:22:12 ON 2006/07/31 <-- new line
        HNAS PROGRAM STARTED AT 11:22:08 ON 2006/07/31
        HNAS PRODUCT CREATED AT 06:36:58 ON 2006/07/31
        CREATED WITH MAINTENANCE THROUGH APAR 2400000
        MOST RECENT MAINTENANCE APPLIED IS APAR 2400000
        SHIPID=0000000000199999 AUTH=00
        CUSTMAC=                                <-- if present
        CUSTOBJ=                                <-- if present
        CUSTID=cid_cust#                        <-- if present
        CUSTINFO=optional cust information      <-- if present
        MAINTENANCE/USE ANNIVERSARY DATE IS 2007/07/31 <-- new line
        DATAFONO SUPPORT IS INCLUDED           <-- if present

        APARID    MODULE    (APPLIED MAINTENANCE)
        ALL MAINTENANCE ON THROUGH MOST RECENT APAR 2400000
```

- The **DNAS** console command now supports **DNAS JCL|DDNAMES** parameter to display the running HNAS JCL DDnames. This display is very useful when DDname information is required by remote console users who do not have access to SYSPRINT. Please refer to the sample DNAS JCL output and description located in the Console SubSystem documentation.
- The **DNAS** console command has been modified to display the 'installed under' system and date stamp in addition to 'created under' and 'running under' system and date stamp. The new display appears as follows:

```
HOST NAS INFORMATION FOLLOWS
ASMDATE=2010/09/16 ASMHOST=ZOS                <--- deleted line
HNAS VERSION=V2R4M0 DIST=NON-SMP
HNAS PROGRAM RUNNING UNDER z/OS 01.10.00
HNAS PRODUCT INSTALLED UNDER z/OS 01.10.00      <-- new line
HNAS PRODUCT CREATED UNDER z/OS 01.11.00
DNAS COMMAND ENTERED AT 07:33:27 ON 2010/09/16
HNAS PROGRAM STARTED AT 07:33:26 ON 2010/09/16
HNAS PRODUCT INSTALLED AT 07:29:00 ON 2010/09/16 <-- new line
HNAS PRODUCT CREATED AT 07:08:18 ON 2010/09/16
```

HNAS V2R4M0 New Features

```
HNAS PRODUCT CREATED WITH MAINTENANCE THROUGH APAR 240BETA
MOST RECENT MAINTENANCE APPLIED IS APAR 2400103
AUTH=000 SHIPID=1100000011199999
CUSTID=SFD_99999 <-- if present
CUSTINFO=COMM-PRO ASSOCIATES <-- if present
MAINTENANCE/USE ANNIVERSARY DATE IS 2010/09/31
DATAFONO SUPPORT IS INCLUDED <-- if present
CUSTMAC= <-- if present
CUSTOBJ= <-- if present
```

ALL MAINTENANCE ON THROUGH MOST RECENT APAR 2400103

Note: Support was introduced into 240 as Enhancement APAR 2400103.

- The **DNWDF** console command has been added to display the NEWDEFN CDF that is maintained in memory while HNAS is running. The NEWDEFN CDF can be modified by the MLCL and MRMT commands. The NEWDEFN CDF is written to disk when HNAS is SHUTDOWN. The GENNWDF start parameter is required for NEWDEFN CDF support. Please refer to the DNWDF command description located in the Console SubSystem documentation for more information.

- The **DRMT** command now displays **[NO]CLOSEONTAPFAILURE** option for XOT|XTP REMOTES.

Note: Support was introduced into 240 as Enhancement APAR 2400055.

- The **DRMT** command now displays **[NO]CUD0SELECTSLU** option for MCH|MXT REMOTES.

Note: Support was introduced into 240 as Enhancement APAR 2400074.

- The **DRMT** command now displays **[NO]DELAYBINDRESP** option for MCH REMOTES.

- The **DRMT** command now displays **[NO]IDTST** option for DFX REMOTES.

Note: Support was introduced into 240 as Enhancement APAR 2400074.

- The **DRMT** command now displays **PVCRECONTMR=**, **PVCSETUPTMR=** and **PVCSETUPREJ=** options for MCH|MXT REMOTES.

Note: Support was introduced into 240 as Enhancement APAR 2400059.

- The **DRMT** command now displays **[NO]RESETINO** option for MCH REMOTES.

Note: Support was introduced into 240 as Enhancement APAR 2400081.

- The **DRMT** command now displays **SVCCALLTMR=** option for MCH|MXT REMOTES.

Note: Support was introduced into 240 as Enhancement APAR 2400069.

- The **DRMT** command now displays the generic SLU name if one was specified for the **SVC0|5=** operand for MCH REMOTES (e.g., **SVC0=(...,sluname+gluname/...)**).

Note: *gluname* support was introduced into 240 as Enhancement APAR 2400084.

- The **DRMT** command has been modified to display the time and date when a table specified for the LOGTAB=, USSTAB= and OPTIONS=NRITAB= operands is loaded into memory. This means that you will be able to tell if a table was loaded during the CDF scan (HNAS start time) or later on using the MRMT command (see below). For example:

```
HNASCMD-> DRMT MCH1 USSTAB
HNASXEQ-> DRMT RNM=MCH1 USSTAB
RMTNAME   OPERAND   TYPE
MCH1      008242B4   MCH
           USSTAB=ZZUSST1 (000D8D30) LOADED AT 08:53:29 ON 2007/04/09
```

Note: The DRMT support described above was introduced into 240 with APAR 2400032.

- The **ECHOXEQ** command added to control how much HNASXEQ> data is echoed back to console operator.

ECHOXEQ MAX causes all applicable command modifiers and parameters to be echoed on multiple lines if necessary with the HNASXEQ-> prefix for each line. Data at the end of one line is continued on the next with an ellipsis (...) suffix as an indication that more information exists.

ECHOXEQ MIN causes all applicable command modifiers and parameters to be echoed on a single line. If more information exists than can be displayed on one line, an ellipsis (...) will terminate the line. ECHOXEQ MIN is the default.

ECHOXEQ NO causes the echoing of command modifiers and parameters to be inhibited.

- The **EXEC *ddname*** command now allows you to specify a DDNAME that exists in the HNAS start job which points to a file that contains a list of console commands. In the initial implementation, the DDNAMEs you that provide in the HNAS start job are arbitrary: For example:

```
//CMDLIST1 DD DSN=COMM1.V2R4M0.TEMPMAC(CMDLIST1),DISP=SHR
//CMDLIST2 DD DSN=COMM1.V2R4M0.TEMPMAC(CMDLIST2),DISP=SHR
...
```

Note: Please refer to the **Enhancements - Expanded Content** heading at the end of this section for additional information regarding this new or enhanced feature.

- The **EXEC *ddname*** command now stops queuing commands when an embedded EXEC command or the new END command is decoded. This will reduce the likelihood of receiving the following error message:

```
NASC054E EXEC CMDLIST1 FILE EXCEEDS QUEUED COMMAND LIMIT, EXEC COMMAND ABORTED
```

If more than 512 bytes of command data and data length bytes are listed in the command list identified by the *ddname* argument, the NASC054E message will be generated. Prior to APAR 2400034, this would occur even if the command list contains an embedded EXEC command (which ends the current command list) listed early in the command list file. The problem occurred because the decode of the embedded EXEC command was not performed until after the entire command list was read into memory. Some users like the ability to list many commands in an EXEC command list beyond the EXEC command for use at a later time. The old logic prevented large numbers of saved commands to be listed in a command list file. The new logic will allow any number of commands to be saved after the END or embedded EXEC command.

Note: EXEC command list END statement and suspend queuing support was introduced into 240

HNAS V2R4M0 New Features

as Enhancement APAR 2400034.

- The **EXEC *ddname*** console command now allows multiple commands to be specified on a single record within the command list file identified by *ddname*.

Multiple commands can be specified on a single record as follows:

```
( cmd , ... , cmd ) ; comment
```

A single command can still be specified on a single record as follows:

```
cmd ; comment
```

Note: This new EXEC support was introduced into 240 as Enhancement APAR 2400064.

- The **EXEC (*command1,command2,command3*,...)** console commands can now be entered on a single line without having to invoke a ***ddname***. Prior to the 240 EXEC command only multiple non display type commands could be entered on a single line. For example, if EXEC (DNAS,DPARM) were entered, the DNAS and DPARM commands would be executed consecutively.
- The **EXEC LIST *ddname*** command is now available that allows the console operator to view externally defined command lists (command lists specified via CMDLIST JCL DD statements) to determine what command sequences are associated with each command list.
- The **EXEC LIST** command has been modified to issue the new NASC052E error message (see above) if ***ddname*** is omitted.

Note: NASC052E error message was introduced into 240 as Enhancement APAR 2400034.

- The **EXEC STOP** command is now available to terminate PAUSE command delay mode or CMDLIST execution.
- The **EXEC *ddname*** command was modified to allow CMDLIST statements of the form **# *text*** to be displayed as comments following the HNASCMD-> prefix. Input records that start with the character **#** are not processed as commands but are simply displayed.
- The **EXEC LCLCONS|*rconname action*** command support added to allow one console to queue commands and alter queued command processing for another console.

```
action = (cmd1 , ... , cmdn) <- Execute inline command list  
         ddname           <- Execute dataset command list  
         LIST ddname     <- List dataset command list  
         STOP             <- Stop command list execution
```

Note: The EXEC LCLCONS|*rconname action* change was introduced into 240 with Enhancement APAR 2400075.

- The **EXEC** command has been modified to decode the new NOPURGEONERROR and PURGEONERROR keywords. One of these keywords may be entered as the first value in an inline command list (**NOPURGEONERROR|PURGEONERR,*cmd1*,...,*cmdn***) or as the first non-comment record in a *ddname* command list. For detailed usage information, please see description of the NOPURGEONERROR|PURGEONERROR options for the EXEC= operand on page ChangesNewFeatures-17.

- The **HELP** command now displays common command parameters after the HELP header message. For example, when HELP ? is entered the following is displayed:

```

HNASCMD-> HELP ?
HNASXEQ-> HELP HELP
COMMAND      DESCRIPTION (* => PRIVILEGED)
              COMMON PARAMETERS FOR ALL COMMANDS: cmdname parm
              parm=? to display help for cmdname.
              BPM to bypass permanent modifiers for cmdname.

HELP         DISPLAY CONSOLE COMMAND HELP INFORMATION
             ENTER> HELP {SHOWALL| command}
    
```

General notes for the HELP command:

- 1) HELP with no arguments will display a one line description for each command.
- 2) HELP *command* will display a verbose description for the given *command*.
- 3) HELP SHOWALL will display a verbose description for all commands.

- The **HELP** command now accepts **SHOWALL** as a new keyword argument. **HELP SHOWALL** will display the complete help text for every HNAS command (verbose). HELP (with no argument) will continue to display a one-line description for each command.
- The **MMEM EOTKEY=dd...dd** command allows the TRIAL PERIOD EXPIRATION DATE (EOT-DATE) for a trial distribution to be extended using a special 16 decimal digit key provided by the **dd...dd** digits. An HNAS trial distribution is normally shipped with an EOTDATE that specifies when the TRIAL AUTHORIZATION will expire. A unexpired EOTDATE is required to use HNAS. In the past, the only way to extend the EOTDATE was to order and install a new refresh distribution. The new MMEM EOTKEY=**dd...dd** command now allows the EOTDATE to be extended without the need of a new refresh distribution or without having to stop and re-start HNAS to specify the EOTKEY=**dd...dd** start parameter. The EOTKEY=**dd...dd** start parameter is also provided so that the EOTDATE can be extended without having to issue the MMEM EOMKEY=**dd...dd** command each time HNAS is stopped and re-started. Please see PARM='EOTKEY=**dd...dd**' description on page ChangesNewFeatures-15 for additional information.:

Note: EOTKEY= parameter logic was introduced into 240 via APAR 2400106.

- The **MMEM EOMKEY=dd...dd** command allows the MAINTENANCE/USE ANNIVERSARY DATE (EOMDATE) for a permanent distribution to be extended using a special 16 decimal digit key provided by the **dd...dd** digits. An HNAS permanent distribution is normally shipped with an EOMDATE that specifies when the MAINTENANCE/USE license will expire. A unexpired EOMDATE is required to use HNAS. In the past, the only way to extend the EOMDATE was to order and install a new refresh distribution. The new MMEM EOMKEY=**dd...dd** command now allows the EOMDATE to be extended without the need of a new refresh distribution or without having to stop and re-start HNAS to specify the EOMKEY=**dd...dd** start parameter. The EOMKEY=**dd...dd** start parameter is also provided so that the EOMDATE can be extended without having to issue the MMEM EOMKEY=**dd...dd** command each time HNAS is stopped and re-started. Please see PARM='EOMKEY=**dd...dd**' description on page ChangesNewFeatures-15 for additional information.

Note: MMEM EOMKEY= command logic was introduced into 240 via APAR 2400095.

HNAS V2R4M0 New Features

- **PARM='EOMKEY=dd...dd'** also allows a trial distribution to be converted to a permanent distribution dynamically using the key provided by the **dd...dd** digits.

Note: EOMKEY= trial conversion logic was introduced into 240 via APAR 2400106.

-
- The **MON TAP** command NAS251 *nM* monitor messages can now be filtered using the **ALRMFLTR=** operand so that they can be optionally routed to SYSCONS using the (FC) or (FU) suffixes. For example specifying **ALRMFLTR=(...,NAS251*M(FU),...)** on BUILD in the CDF or issuing the **ALARM FILTER=(...,NAS251*M(FU),...)** console command will route the NAS251 *nM* monitor messages to SYSCONS as well as SYSPRINT.
- The **MON TAP** command now accepts **PKTDATA|MAXDATA|MINDATA|NODATA** as arguments to control the type of information displayed in the NAS251 *nM* monitor messages.

For PKTDATA (the default), PKT=xx...xxx data is displayed after the NAS251 *nM* message header.

For MINDATA, DTEADDR=dd...dd is displayed after the NAS251 *nM* message header.

For MAXDATA, DTEADDR=dd...dd is displayed after the NAS251 *nM* message header and for the NAS2513M message, additionally the FAC= and CUD= values.

For NODATA, only the NAS251 *nM* message header is displayed.

Note: MONTAP PKTDATA|MAXDATA|MINDATA|NODATA argument support was introduced into 240 as Enhancement APAR 2400044.

- The **MRMT LUNAME=|SVC0=|SVC4=|SVC5=*sluname*-{A|I}** command now permits manipulation of SLU state (A is for active (online) and I is for idle (offline)). This command will now propagate the **VARY *sluname* {ON|OFF}** command based on the SLU state provided.

Note: *sluname*-{A|I} support was introduced into 240 as Enhancement APAR 2400028.

- The **MRMT** command has been modified to allow an asterisk to be removed from or added to the *pluname* suboperand of the **GATE LUNAME=** operand for a **TYPE=MCH|XTP REMOTE** definition statement. The presence or absence of the asterisk dictates whether a REQSESS request is presented to VTAM to solicit a BIND from the CTCP application. If an asterisk is present, HNAS will issue a REQSESS request to the application named *pluname*. If the asterisk is absent, HNAS will simply wait for the application to BIND the SLU.

To remove, enter: **MRMT {RNM=} *mchname* LUNAME=*sluname/pluname***

To add, enter: **MRMT {RNM=} *mchname* LUNAME=*sluname/pluname****

Note that you can also use this command to change the PLU name.

Note: The MRMT support described above was introduced into 240 with APAR 2400008.

- The **MRMT *rmtname* IPADDR=*aaa.bbb.ccc.ddd*** command has been modified to allow the IP address to be modified dynamically for **TYPE=XOT REMOTE**s. This ability was previously reserved for **TYPE=DMY REMOTE**s only. The remote must be **OFFLINE** in order to alter the IP address.

Please refer to the **Enhancements - Expanded Content** heading at the end of this section for additional information regarding this new or enhanced feature.

Note: The MRMT support described above was introduced into 240 with APAR 2400010.

- **MRMT OPTIONS=[NO]CLOSEONTAPFAILURE** command allows a customer to toggle this option for TYPE=XOT|XTP REMOTEs.

Note: MRMT OPTIONS=[NO]CLOSEONTAPFAILURE command support was introduced into 240 as Enhancement APAR 2400055.

- **MRMT OPTIONS=[NO]CUD0SELECTSLU** command allows a customer to toggle this option for TYPE=MCH|MXT|XTP REMOTEs.

Note: MRMT OPTIONS=[NO]CUD0SELECTSLU command support was introduced into 240 as Enhancement APAR 2400074.

- **MRMT OPTIONS=[NO]DELAYBINDRESP** command allows a customer to toggle this option for TYPE=MCH|XTP REMOTEs.

- **MRMT OPTIONS=[NO]IDTST** command allows a customer to toggle this option for TYPE=DFX REMOTEs.

Note: MRMT OPTIONS=[NO]IDTST command support was introduced into 240 as Enhancement APAR 2400074.

- **MRMT OPTIONS=PVCRECONTMR=secs, PVCSETUPTMR=secs and PVCSETUPREJ=code** command allows a customer to delete, change or add these options for TYPE=MCH|MXT REMOTEs.

Note: MRMT OPTIONS=PVCRECONTMR=,PVCSETUPTMR=,PVCSETUPREJ= command support was introduced into 240 as Enhancement APAR 2400059.

- **MRMT OPTIONS=[NO]RESETINO** command allows a customer to toggle this option for TYPE=MCH|XTP REMOTEs.

Note: MRMT OPTIONS=[NO]RESETINO command support was introduced into 240 as Enhancement APAR 2400081.

- **MRMT OPTIONS=SVCCALLTMR=secs** allows customer to delete, change or add this option for TYPE=MCH|MXT REMOTEs.

Note: MRMT OPTIONS=SVCCALLTMR= command support was introduced into 240 as Enhancement APAR 2400069.

- The **MRMT** command has been modified to allow a generic SLU name to be specified for a TYPE=MCH REMOTE. Syntax is **MRMT SVC0|5=sluname+gluname|...**

Note: *gluname* support was introduced into 240 as Enhancement APAR 2400084.

- The **MRMT** command has been modified to allow the tables specified for the LOGTAB=, USSTAB= and OPTIONS=NRITAB= operands to be reloaded into memory. This means that changes to tables that are made while HNAS is running can be put in to effect without having to stop and restart HNAS.

Syntax: MRMT rmtname LOGTAB=*lgtbname*-R

HNAS V2R4M0 New Features

```
MRMT rmtname USSTAB=ustbname-R
MRMT rmtname OPTIONS=NRITAB=nrtbname-R
```

When the -R follower is omitted, old logic is used. That is, the given *lgtbname*, *ustbname* or *nrtbname* is compared against all LOGTAB=, USSTAB= or NRITAB= names, respectively, currently in use. If the name found, the given LOGTAB=, USSTAB= or NRITAB= is copied from the *found* REMOTE to the *specified* REMOTE. If the name is not found, the given LOGTAB=, USSTAB= or NRITAB= is LOADED into memory.

When the -R follower is specified, the given table is unconditionally loaded into memory even if it is already present. The new version of the table is then propagated to all other REMOTES that share the table.

Note: The MRMT support described above was introduced into 240 with APAR 2400032.

General Notes for APAR 2400032:

1) Because the transmission of a USSMSG in a USSTAB can span task dispatches (and TCPIP interrupts), HNAS will refuse to reload a USSTAB if it is currently being used for a USSMSG transmission. In this case, the following message is issued:

```
NASC031W USSTAB ISTINCDT IS CURRENTLY IN USE, RETRY MRMT COMMAND LATER
```

This is the case because of a potential ABEND that can result due to an addressing exception if the new table alters USSMSGs currently being transmitted.

Further, if you attempt to load a table that is not appropriate for the specified operand, for example a USS table is specified for a LOGON table (e.g., LOGTAB=ISTINCDT), the following message is issued:

```
NASC030E LOGTAB ISTINCDT HAS INVALID FORMAT, MRMT COMMAND ABORTED
```

2) The following message is also issued if you specify the wrong table in the CDF, for example a USS table is specified for a LOGON table (e.g., LOGTAB=ISTINCDT):

```
NAS1311E REMOTE mchname LOGTAB=ISTINCDT IS NOT A VALID LOGON TABLE, REQUIRED
```

***** THIS NEW MESSAGE WILL PREVENT HNAS FROM STARTING**

Prior to APAR 2400032, you would not have known that a bad table was specified until it was accessed during call setup.

For a bad USSTAB=, the following message is issued and the call is cleared:

```
NAS5702E LU sluname ADDRESSES INV USSTAB
```

For a bad LOGTAB= value, no action is taken (treated as if no LOGTAB= value was specified). In this case USSTAB= is used. If the USSTAB= operand is (also) missing, the following message is issued and the call is cleared:

```
NAS5703E LU sluname MISSING USSTAB
```

- **PARM='DEBUG TCP'** start parameter allows certain TCPIP 198 ABENDs to be bypassed, replaced by new NAS2109S and NAS2110S alarm messages - HNAS execution continues. When this option

is specified, the following 198 ABENDs are bypassed:

```
HALT AT LOC 80065CB4 IN NASUTIL : TCPIP INTERRUPT UNEXPECTED
HALT AT LOC 80070952 IN NASTCP : TCPIP REPLY ID FAILURE
```

- A new **PAUSE seconds** command is now available that suspends or delays execution of the console command following the PAUSE commands by the seconds value. This command is useful in delaying command execution in the new CMDLISTs or in the existing CONCMDQ=list execution after initial activity has subsided.
- **PING** command now produces NAS261 nM monitor messages that can be filtered using the ALRMFLTR= operand so that they can be optionally routed to SYSCONS using the (FC) or (FU) suffixes. For example specifying ALRMFLTR=(...,NAS261*M(FU),...) on BUILD in the CDF or issuing the ALARM FILTER=(...,NAS261*M(FU),...) console command will route the NAS261 nM monitor messages to SYSCONS as well as SYSPRINT.

Note: PING NAS261 nM message support was introduced into 240 with APAR 2400022.

- The **PING (XOT)** command has been modified allowing all special ping XOT Call Request values to be coded on a TYPE=DMY REMOTE. This allows the user to issue **PING dmyname** so that the ping can be associated with predefined XOT Call Request packet values for router or interface level PINGing.

The syntax is: [RNM=dmyname] PING [RNM=] dmyname [parms]

You can set a permanent TYPE=DMY REMOTE with the left side RNM= modifier and override it with a right side *dmyname* (the RNM= keyword is optional on the right). The *parms* operand are existing PING parameters that can be entered to override values on the dummy. For example, you can predefine an IPADDR on the dummy and override when the PING command is entered via PING *dmyname* 1.2.3.4 (for example). Note also that you can use MRMT to modify any supported dummy operand. The TYPE=DMY REMOTE now accepts PROTOCOL=XOT|XTP so you don't have to specify a protocol on the PING command although you can override it with PING *dmyname* XTP (for example). This operand can also be changed with the MRMT command.

- The **PRNT** console command now supports the **{PRTSWLST|SWITCH}=(?|*|/list)** parameter which is used to display, clear, switch and/or update the actions and active *ddnamei*|DYNAMIC list as initially specified by the PRTSWLST= operand on the BUILD definition statement. Please refer to the PRNT command description located in the Console SubSystem documentation for more information.
- The **PRNT** console command now supports the **NEXTPRSW** parameter, or equivalently the **SWITCHNOW** parameter, which is used to force a switch to the next *ddnamei*|DYNAMIC SYSPRINT log in the BUILD PRTSWLST= operand. Please refer to the PRNT command description located in the Console SubSystem documentation for more information.
- The **PRNT** console command now supports the **RSMEPRSW** parameter which is used to restart SYSPRINT switching in the BUILD PRTSWLST= operand after all datasets have been used and the STOP action is in effect. Please refer to the PRNT command description located in the Console SubSystem documentation for more information.
- The **PRNT** console command now supports the **TIMESTAMP=(time-format,+D|-D)** parameter which is used to updates TIMESTAMP time and date controls.

HNAS V2R4M0 New Features

- The **PRNT ON|OFF ... DATE ON|OFF ...** console command now generates a NAS01 *nni* alarm message when a state change is effected by the command.

Note: The new PRNT alarm messages were added to 240 as part of Enhancement APAR 2400017.

- The **QUIT** command now accepts **E** a new keyword follower. **QE {ddname}(cmd1,...,cmdn)** will cause the command list identified by *ddname* or supplied by the specified (*cmd1*,...,*cmdn*) list to be executed before HNAS is shutdown. In addition, QUIT help has been enhanced.
- The **QUIT** shutdown process now executes **DNAS** and **ALARM LOG=?** console commands unconditionally when any QUIT command is entered. If QE *ddname* entered, the target list is appended to the forced commands.
- **SCHEDULE {ddname}(hh:mm:ss,cmd,...,hh:mm:ss,cmd)|LIST|SUSP|RSME|PRG** console command added to allow scheduled console commands to be added, deleted or displayed.

Syntax: SCHEDULE {*ddname*|(hh:mm:ss,cmd,...,hh:mm:ss,cmd)|LIST|SUSP|RSME|PURGE}

(*hh:mm:ss,cmd,...,hh:mm:ss,cmd*) is an *inline* schedule list. An *inline* schedule list is added to the end of an existing schedule list if one is active.

ddname identifies a file that contains a list of times and commands. A *ddname* schedule list is added to the end of an existing schedule list if one is active. A *ddname* schedule list can contain the following records:

Comments are allowed and start with an asterisk (*) or semi-colon (;) in record column 1. Comments can also appear on a command line but must start with a semi-colon after the command. A single time and command can be specified on a single record as follows:

hh:mm:ss , cmd ; comment

Multiple times and commands can be specified on a single record as follows:

(*hh:mm:ss , cmd , ... , hh:mm:ss , cmd*) ; *comment*

LIST [ddname] is used to display a schedule list. If *ddname* is not specified, the active (current) schedule list is displayed. If *ddname* is specified, the identified schedule list is displayed but the active schedule list remains unaffected.

SUSP causes the active schedule list to be suspended so that no commands will be scheduled. The list itself is maintained.

RSME causes the active schedule list that was suspended to be resumed so that commands will be scheduled again.

PRG causes the current schedule list to be cleared. No commands are left to be scheduled.

Note: Please refer to the **Enhancements - Expanded Content** heading at the end of this section for additional information regarding this new or enhanced feature.

Note: The new SCHEDULE command support was introduced into 240 as Enhancement APAR 2400064.

- The **SCHEDULE LCLCONS|*rconname* *action*** command support added to allow one console to schedule commands and alter scheduled command processing for another console.

```

action = (time1,cmd1,..) <- Append inline schedule list to active schedule list
          ddname           <- Append dataset schedule list to active schedule list
          LIST ddname     <- List dataset schedule list
          LIST              <- List active schedule list
          SUSP              <- Suspend active schedule list
          RSME              <- Resume active schedule list
          PRG               <- Purge active schedule list
  
```

Note: The SCHEDULE LCLCONS|*rconname* *action* change was introduced into 240 with Enhancement APAR 2400075.

- The **SHOW CMSG ON|OFF** command has been added to manipulate the **SHOWCMSG** start parameter.

Note: The SHOW CMSG ON|OFF console command support was introduced into 240 with APAR 2400036.

- The **SHOW ON|OFF|ERR ... MORE|LESS ...** console command now generates a NAS02*nni* alarm message when a state change is effected by the command.

Note: The new SHOW alarm messages were added to 240 as part of Enhancement APAR 2400017.

- New **SHOWCMSG** parameter allows alarm messages containing variable length text data to be compressed (multiple consecutive blanks are removed).

Prior to APAR 2400036, alarm messages that contain names or other variable text data can cause multiple blanks to appear in the alarm messages. Removing multiple blanks makes messages look cleaner in SYSPRINT log and on SYSCONS. If the SHOWCMSG parameter is not specified, no compression is performed. All messages appear as they always have. This will allow customers to control the compression process for those who use message filtering tools that depend on fixed message offsets. Consider the following message for example:

NAS3799I Without Compression (SHOWCMSG OFF):

```

0           1           2           3           4           5           6
0123456789012345678901234567890123456789012345678901234567890123456789
|
NAS3799I LU MCH1PVC  ENDING  SESSION ON MCH MCH1      RMT  CAUSE/DIAG=
000/130 (00/82) DIAGX=0000
|
01234567890123456789012345
7           8           9
  
```

NAS3799I With Compression (SHOWCMSG ON):

```

0           1           2           3           4           5           6
0123456789012345678901234567890123456789012345678901234567890123456789
|
NAS3799I LU MCH1PVC ENDING SESSION ON MCH MCH1 RMT CAUSE/DIAG=000/130
(00/82) DIAGX=0000
  
```

HNAS V2R4M0 New Features

```
|           |
012345678901234567
7           8
```

Note: While message compression does use additional CPU cycles, the affect appears to be minimal. However, HNAS environments with excessive alert message activity should consider measuring the effect upon CPU load with compression enabled (SHOWCMSG ON) and disabled (SHOWCMSG OFF) to see if compression is causing an unacceptable percentage of CPU load. Our in-house testing revealed a CPU seconds difference per WTO for SHOWCMSG ON versus SHOWCMSG OFF to be an additional 0.000011048 seconds per WTO or approximately 0.002846% additional overhead per WTO.

Note: The SHOWCMSG support described above was introduced into 240 with APAR 2400036.

- **SMSG LCLCONS|rconname 'text'** command support added to allow the target console to be identified by PCE name.

Note: The SMSG LCLCONS|rconname 'text' change was introduced into 240 with Enhancement APAR 2400075.

- **SNAP ALL|trgtlist** command support added to allow specific HNAS storage areas to be dumped in the SYSPRINT file.

```
trgtlist={ALOG|ALARMLOG} <- alarm log
          {BFR|BFRSTOR}   <- buffer pool
          {CNFG|CONFIG}   <- configuration area
          {CVT}           <- communications vector table
          {DNAS|CONSDNAS} <- DNAS module (NASIDENT)
          {MCH|MCHSTOR}   <- MCH area (MCH,VCB,LUB)
          {MLOG|MAINTLOG} <- maintenance log
          {PCE|PCESTOR}   <- dynamic PCE area
          {TINT|TCPINTBL} <- TCPIP interrupt table
          {TRC|TRACE}     <- internal trace table
          {VINT|VTAMINTBL}<- VTAM interrupt table
          {WA|XFNASWA}    <- workarea
          {WAX}           <- workarea extension
```

Note: The new SNAP command support was introduced into 240 as Enhancement APAR 2400080.

- **TCPIP** external interrupt table entries extended to allow logging of additional diagnostic information making HNAS TCPIP tracing less of a requirement for problem diagnosis. New values are TCPIP target/class (command/socket ID), RC/ERNO, start/end times.

- The **TRCCONS LCLCONS|rconname {ON|OFF}** command support added to allow the target console to be identified by PCE name.

Note: The TRCCONS LCLCONS|rconname {ON|OFF} change was introduced into 240 with Enhancement APAR 2400075.

- The **TRCMCH {ICR|ICRF|OCR|ICLR|OCLR}** followers are now allowed as a start parameters to enable *global* Call and Clear request trace logging. **TRCMCH GBL {ICR|ICRF|OCR|ICLR|OCLR}** are the equivalent console commands.

The **TRCMCH** console command supports the ICRF, ICR, OCR, ICLR and OCLR arguments to

allow tracing of Inbound Call failures (ICRF), all Inbound Calls (ICR), all Outbound Calls (OCR), all Inbound Clears (ICLR) and all Outbound Clears (OCLR). These operators can have *local* or *global* significance depending on whether the LCL or GBL argument is also specified.

If LCL is specified, the ICRF, ICR, OCR, ICLR and/or OCLR trace flags are maintained in the MCH control blocks identified by the RNM=*mchname* or ID=*loid-hiid* command modifiers. RNM= or ID= must be set if LCL is specified. *Local* ICRF, ICR, OCR, ICLR and/or OCLR trace entries are only logged if the MCH is also being traced (TRCMCH ON is in effect for the MCH).

If GBL is specified, the ICRF, ICR, OCR, ICLR and/or OCLR trace flags are maintained in the HNAS common area (XFNASWA) and thus affect all MCHs. RNM= and ID= are ignored if GBL is specified. *Global* ICRF, ICR, OCR, ICLR and/or OCLR trace entries are only logged if all MCHs are also being traced (TRCMCH ALLON is in effect).

If you want ICRF, ICR, OCR, ICLR and/or OCLR to have *global* significance, you should enter GBL immediately after the TRCMCH command (e.g., TRCMCH GBL ICR ... OCLR). GBL is assumed if RNM= and ID= are not set (both are null). LCL is assumed if GBL is omitted and RNM= or ID= are set.

- The **TRCSUBR** start parameter and console command now allow an event list to be provided in addition to the normal ON|OFF arguments. The syntax is **TRCSUBR {ON|OFF} {eventlist}** where **eventlist**={CONS|MCH|NETV|PCE|TCP|VTAM}. **eventlist** values are provided so that subroutine call traces can be filtered by the event currently being processed. This reduces the number of unwanted trace entries being logged.

History: When TRCSUBR is in effect, every subroutine within HNAS logs a number of trace entries. Some are very useful but others are simply unwanted 'noise'. What is necessary to eliminate unwanted TRCSUBR entries is the ability to filter subroutine calls based on the event(s) being processed. For example, the TRCPCE command is used to log TCP/IP related events. To coordinate TCP/IP subroutine calls with these events requires filtering TRCSUBR traces for TCP/IP related calls only. Currently, HNAS waits on the following 6 events:

```
TCP   - TCP/IP interrupt completions
VTAM  - VTAM interrupt completions
MCH   - REMOTE TYPE=MCH service
NETV  - NETVIEW interrupt completions
CONS  - CONSOLE interrupt completions
PCE   - Miscellaneous task service
```

The TRCSUBR start parameter and console command will now accept one or more of these events to be specified so that subroutine call traces are logged only when the selected event(s) are being processed. This means that only subroutine calls associated with the selected event(s) will generate trace entries. HELP for the TRCSUBR command will now display the following:

```
TRCSUBR  *SUBROUTINE CALL TRACE CONTROL

ENTER> TRCSUBR {ON|OFF} {eventlist}

eventlist = {ALLEVENTS|NOEVENTS} or
            {CONS|NETV|TCP|VTAM|MCH|PCE}
```

General notes for the TRCSUBR command:

- 1) Enter ON to initiate global subroutine call tracing.

HNAS V2R4M0 New Features

- 2) Enter OFF to terminate global subroutine call tracing.
- 3) Enter ALLEVENTS to set the subroutine call trace event filter to all events.
- 4) Enter NOEVENTS to reset the subroutine call trace event filter to no events.
- 5) Enter CONS, NETV, TCP, VTAM, MCH and/or PCE to set the subroutine call trace filter to specific events. You may enter NOEVENTS followed by a list of events to first clear then repopulate the subroutine call trace event filter.
- 6) ON is assumed if TRCSUBR is entered with no argument.
- 7) If NOEVENTS is in effect when ON is set, ALLEVENTS is forced.
- 8) Tracing requires additional CPU cycles.

Note: TRCSUBR *eventlist* logic was introduced into 240 via APAR 2400108.

- Enhancements to **TRCTRAP ALRMLIST=(NASnnnni/data/data-offset,...)** now provides the ability to not only restrict a trap 'hit' to a specific message ID but also to data within the message. When the *data* suboperand is provided, the trap is not sprung when a message ID match occurs unless the message also contains the text *data* you provide (*data* is not enclosed in quotes and no spaces are permitted). The *data* is used to further restrict the message search to specific data content. For example, the DIAG= value in a NAS7715W message. The *data-offset* value, which can range from 1 to 128, identifies where in the message the *data* can be found. The *data-offset* allows the *data* search to start at a specific offset within the message which reduces CPU utilization when looking for a match. The *data-offset* value is relative to the first character of a message, that is, the beginning of the message ID (NASnnnni). If no *data-offset* is given, the *data* search starts at the beginning of the message. If a *data-offset* value plus the *data* length exceeds the message length, a 'hit' is assumed for the message ID only.
- The *data* suboperand of the **TRCTRAP ALRMLIST=(NASnnnni' data'/data-offset,...)** command will now accept a quoted string. This is required so that a forward slash (/) within the data is treated as part of the data and not as a suboperand delimiter.

History: If *data* contains a forward slash (/), the slash will be treated as a delimiter making the data that follows it being treated as the *data-offset* suboperand. This can cause the entire ALRMLIST value to be ignored and an error condition to be set. For example, when the following TRCTRAP console command is entered:

```
TRCTRAP ALRMLIST=(NAS3799I/DIAG=000/195)
```

the following error message is issued and the request is rejected:

```
NASC532E PARAMETER DATA INVALID: IAG=000/195)...,  
TRCTRAP COMMAND ABORTED
```

This is occurring because the *data* value DIAG=000/195 contains a forward slash making the 195 that follows the slash treated as a *data-offset* value. Since 195 is too large for a *data-offset* value, the command is rejected.

The TRCTRAP CDF operand processor and the TRCTRAP console command processor have been modified to accept the *data* suboperand of the ALRMLIST operand as a quoted string so that ALL data within the string is treated as data. This includes spaces, forward slashes and so on. For the ALRMLIST operand described above, the following is now allowed:

TRCTRAP ALRMLIST=(NAS3799I/'DIAG=000/195')

In this case, the forward slash in DIAG=000/195 is no longer treated as an suboperand separator but as part of the message data.

Note1: If *data* does not contain spaces or a forward slash, it does not have to be specified within quotes, but can be if you wish. ABCDEF and 'ABCDEF' are treated the same.

Note2: *data* may not include embedded quotes. For example, ABC'DEF or ABC"DEF or 'ABC'DEF' or 'ABC"DEF' are not permitted.

Note: TRCTRAP quoted '*data*' logic was introduced into 240 via APAR 2400108.

- The **TRCTRAP TRAPACTION=(EXEC=*ddname*)** option now supports command list execution upon trap action filter condition. This allow for automation of various console command actions for debugging without operator intervention.
- The **TRCtype** console commands have been modified to accept multiple arguments for the same command invocation. For example, TRCMCH ICR ICRF OCR ... OCLR will now be valid. In the past, you were required to enter each option via a separate command. For example TRCMCH ICR ... TRCMCH OCLR. This syntax is still valid but the multiple argument syntax is also accepted.
- The **VARY *mchname* {OFF|ON}** command has been added to the VARY command. This permits or restricts MCH resources to be used for inbound or outbound calls. An inactive MCH will be skipped when processing the RTEIN= operand.
- The **VARY *sluname* {OFF|ON}** command has been added to the VARY command. This permits or restricts SLU resources to be used for inbound or outbound calls.
- The **VARY *sluname* FORCE** command has been modified to close the TCP/IP socket associated with the SLU even if the SLU itself is inactive (normally only PVCs can be in this state). Prior to this change, the command would have been rejected if the SLU was inactive with the following error message:

```
NASC015E FUNCTION ALREADY INACTIVE, VARY COMMAND ABORTED
```

Note that this message can still be issue for VARY *sluname* FORCE if the SLU does not have an active TCP/IP socket connection or for VARY *sluname* OFF is the SLU is inactive.

This new action for VARY *sluname* FORCE eliminates the need to issue the DLU *sluname* command to obtain the *rpid* value for the SLU followed by the VARY ID=*rpid* FORCE command to close the associated TCP/IP socket.

Note: The new VARY *sluname* FORCE change was introduced into 240 by Enhancement APAR 2400098.

HNAS V2R4M0 New Features

Enhancements - Expanded Content:

- The **EXEC *ddname*** command allows you to specify a DDNAME that exists in the HNAS start job which points a file that contains a list of console commands.

In the initial implementation, the DDNAMEs you that provide in the HNAS start job are arbitrary: For example:

```
//CMDLIST1 DD DSN=COMM1.V2R4M0.TEMPMAC(CMDLIST1),DISP=SHR
//CMDLIST2 DD DSN=COMM1.V2R4M0.TEMPMAC(CMDLIST2),DISP=SHR
//CMDLIST3 DD DSN=COMM1.V2R4M0.TEMPMAC(CMDLIST3),DISP=SHR
```

These command list files can be executed using the new EXEC console command as follows:

```
EXEC CMDLIST1
EXEC CMDLIST2
EXEC CMDLIST3
```

If a *ddname* is specified that does not exist in the start JCL, the following message is issued:

```
NASC051E EXEC baddnm INVALID, CANNOT BE OPENED
```

When a valid command list *ddname* is specified and it has been read successfully, the following message is generated:

```
NASC055I EXEC CMDLIST1 FILE HAS BEEN READ, IT WILL NOW BE EXECUTED
```

A command list file can contain another EXEC *ddname* statement. When the embedded EXEC command is decoded, the balance of the current command list will be ignored and the new command list will then be executed. This allows you to chain command lists. If a command list contains an EXEC command that points at itself (e.g. EXEC CMDLIST3 is contained in CMDLIST3), the current command list (CMDLIST3 in this case) will be terminated and the following message will be issued:

```
NASC050E EXEC CMDLIST3 INVALID, CANNOT POINT AT ITSELF
```

You can also chain command lists via JCL by concatenating the command list dataset names as long as the concatenation forms a sequential dataset. Note that in the initial development, command lists are always sequential datasets or members of partitioned datasets. The EXEC command does not currently allow you to specify a DDNAME and MEMBER name if DDNAME references a PDS.

```
//CMDLISTS DD DSN=COMM1.V2R4M0.TEMPMAC(CMDLIST1),DISP=SHR
//          DD DSN=COMM1.V2R4M0.TEMPMAC(CMDLIST2),DISP=SHR
//          DD DSN=COMM1.V2R4M0.TEMPMAC(CMDLIST3),DISP=SHR
```

```
EXEC CMDLISTS
```

Again, all DDNAMEs and the members they reference are totally arbitrary. The only requirement is that the *ddname* you provide for the new EXEC command **MUST** exist in the HNAS start JCL.

Note that if DISP=SHR is specified for all command list *ddnames* above, it will allow you to modify the command lists when HNAS is running.

The commands you provide in a command list file are completely arbitrary. The EXEC command processor does not check the commands for validity. Validity checking is provided by the individual command processors.

Note that the command list file can contain comments that start with an asterisk (*) or semi-colon (;) in column 1. Note also that each individual record in a command list file can contain a comment that follows the command text if the first character is a semi-colon (just like the CDF parsing). Command text is assumed to start with the first non-blank character on a line and end with the last non-blank character on a line (prior to the comment start ; if it is present). Listed below is a sample command list file:

```
* THIS IS A SAMPLE CONSOLE COMMAND LIST FILE. COMMENTS CAN
* BE SUPPLIED THAT START WITH '*' OR ';' IN RECORD COLUMN 1.
* COMMENTS ON INDIVIDUAL LINES CAN BE SUPPLIED THAT START
* WITH ';'. CONSOLE COMMANDS WILL BE EXTRACTED FROM EACH
* RECORD FROM THE FIRST NON-BLANK CHARACTER TO THE LAST
* NON-BLANK CHARACTER ON EACH LINE. THE SEARCH FOR THE
* FIRST NON-BLANK CHARACTER IS EXECUTED FOR NON-COMMENT
* RECORDS ONLY (*|; IS NOT IN CC1). THE SEARCH FOR THE
* LAST NON-BLANK CHARACTER IS AFFECTED AFTER THE LINE
* COMMENT STARTING DELIMITER (;) IS FOUND.
* |<-- FIRST NON-BLANK
* |
* | |<-- LAST NON-BLANK
* | |
* V V
  DNAS                               ; FIRST QUEUED COMMAND
  DPARM EXEC                         ; SECOND QUEUED COMMAND
```

This file would be processed as though CONCMDQ=(DNAS,DPARM) were specified.

Currently the BUILD CONCMDQ= operand allows a maximum of 512 bytes for queued commands. This is also true for commands that are provided by the EXEC ddname command. Note that the 512 byte maximum includes a length byte for each queued command. For example: If DNAS and TRCMCR ICR were provided in the CONCMDQ= operand or in a command list, the total number of bytes that will be saved is 5 for DNAS (length byte = 4) and 11 for TRCMCH ICR (length byte = 10). If you provide enough data in a command list that exceeds the 512 byte limit, the following messages will be issued:

```
NASC054W EXEC XXXXXXXX FILE EXCEEDS QUEUED COMMAND LIMIT,
          COMMAND LIST IGNORED
```

- The **MRMT rmtname IPADDR=aaa.bbb.ccc.ddd** command has been modified to allow the IP address to be modified dynamically for TYPE=XOT REMOTEs. This ability was previously reserved for TYPE=DMY REMOTEs only.

In order to allow this change, the target TYPE=XOT REMOTE must be in the offline state. This can be done in the CDF by specifying INIT=IDLE or during execution using the VARY FORCE command. For example, assume that the TYPE=XOT REMOTE named R1CNIN was configured with IPADDR=10.117.56.100 and INIT=ACTIVE. To change the IP address from its current value of 10.117.56.100 to 10.117.56.201, for example, the following command sequence would have to be entered:

HNAS V2R4M0 New Features

```
RNM=R1CNIN          <- set remembered RMT name
VARY FORCE NOPR      <- vary R1CNIN offline
MRMT IPADDR=10.117.56.201 <- change IP address
VARY ON NOPR        <- vary R1CNIN back online
```

Notes: 1) The commands above could have been entered with a temporary REMOTE name as a right side modifier, e.g., VARY R1CNIN FORCE NOPR

2) If the MRMT command above is issued while the REMOTE is online, the command will be rejected and the following message issued:

```
NASC022W RESOURCE ONLINE, MUST BE OFFLINE (USE VARY)
```

3) If the MRMT command attempts to change a dynamic IP address (IPADDR=DYNAMIC), the command will be rejected and the following message issued:

```
NASC312E RNM=rmtname CONFIG INVALID, REQUIRED
```

- **NAS1391I** messages now issued to display MXT operand overrides. When a user associates an MXT with a PVC= or SVC0/3/5= operand entry, he is asking HNAS to override operands on the root MCH with those from the MXT. In some cases, this can lead to confusion if one does not remember to review the HNAS documentation to see which operands can be overridden based on the associated resource type.

The configuration logic has been modified to display, for each SLU or SPU defined in the PVC= or SVC0/3/5= operands with an associated MXT (or SVC), those MXT operands that will override operands of the same name on the root MCH. Not all MXT operands are valid overrides for all SLUs or SPUs. For example, IDLETO= is a valid override when specified for an MXT that is associated with a SVC0/5= SLU but is not used for a PVC SLU.

MXT override messages are listed at the end of the CDF scan in NAS1391I messages. Only those MXT operands that are specified and, of these, only those that are valid as overrides for an SLU or SPU are displayed. If an MXT operand is not listed, it is because it was either omitted from the MXT (and hence cannot override the MCH operand) or, if specified, is not valid as an override for the SLU or SPU resource. Consider the following configuration example:

```
MCH1      REMOTE TYPE=MCH
          IDLETO=2
          PAD=INTEG
          PADPARM= (1/1,
                   2/1,
                   13/7,
                   21/0,
                   110/0)
          PVC= (3,
               MCH1P001/5/255/01/SERIAL0-1/R2CNOT1/MXT0,
               MCH1P002/5/255/02/SERIAL0-1//MXT2,
               MCH1P003/5/255/03/SERIAL0-1)
          SVC0= (2,
                MCH10011/1234-X1234-<MXT1>T01/MXT0
                MCH10012/1234-<MXT0>-X1234T1)
          SVC5= (2,
```

```

                MCH15001/1234-X1234-<MXT1>T01/MXT0
                MCH15002/1234-<MXT0>-X1234T1)
SVC3=(1,Q1990100//MXT0)
:
MXT0  REMOTE  TYPE=MXT
        CUD=(01111111)
        DCEADDR=1111
        DTEADDR=9999
        FAC=(420808
            430404)
        IDLETO=15
        PADPARAM=(1/1,
                2/1,
                3/2,
                4/0,
                5/2,
                7/2,
                12/1,
                13/7,
                21/0,
                110/0)
MXT1  REMOTE  TYPE=MXT
        CUD=(01222222)
        DCEADDR=2222
        FAC=(0101420A0A430202)
        PADPARAM=(1/1,
                2/1,
                7/2,
                12/1,
                13/7,
                21/0,
                110/0)
MXT2  REMOTE  TYPE=MXT
        CUD=(01222222)
        DCEADDR=2222
Q1990100 REMOTE  TYPE=SPU
        LUNAME=(,
        Q1990102/201/201/1/PVCMXT,
        Q1990103)
        DTEADDR=20361234
        DCEADDR=20364321
        FAC=420707430707
        CUD=C3000000
        IDBLK=199
        IDNUM=00001
:

```

Based on this configuration, the following messages are generated at the end of the CDF scan.

```

NAS1391I THE FOLLOWING OPERANDS FOR SLU MCH1P001 (DEFINED ON MCH MCH1
NAS1391I AS PVC ENTRY 000) COME FROM MXT MXT0      RATHER THAN THE MCH
NAS1391I --> FAC=420808430404
NAS1391I THE FOLLOWING OPERANDS FOR SLU MCH1P002 (DEFINED ON MCH MCH1
NAS1391I AS PVC ENTRY 001) COME FROM MXT MXT2      RATHER THAN THE MCH

```

HNAS V2R4M0 New Features

```
NAS1391I --> NO OVERRIDES
NAS1391I THE FOLLOWING OPERANDS FOR SLU MCH10011 (DEFINED ON MCH MCH1
NAS1391I AS SVC0 ENTRY 0000) COME FROM MXT MXT0      RATHER THAN THE MCH
NAS1391I WHEN CONNECTION IS NOT ESTABLISHED VIA A <RMTNAME>
NAS1391I --> CUD=01111111
NAS1391I --> DCEADDR=1111
NAS1391I --> DTEADDR=9999
NAS1391I --> FAC=420808430404
NAS1391I --> IDLETO=00015
NAS1391I THE FOLLOWING OPERANDS FOR SLU MCH10011 (DEFINED ON MCH MCH1
NAS1391I AS SVC0 ENTRY 0000) COME FROM MXT MXT1      RATHER THAN THE MCH
NAS1391I WHEN CONNECTION IS ESTABLISHED VIA <MXT1>
NAS1391I --> CUD=01222222
NAS1391I --> DCEADDR=2222
NAS1391I --> FAC=0101420A0A430202
NAS1391I THE FOLLOWING OPERANDS FOR SLU MCH10012 (DEFINED ON MCH MCH1
NAS1391I AS SVC0 ENTRY 0001) COME FROM MXT MXT0      RATHER THAN THE MCH
NAS1391I WHEN CONNECTION IS ESTABLISHED VIA <MXT0>
NAS1391I --> CUD=01111111
NAS1391I --> DCEADDR=1111
NAS1391I --> DTEADDR=9999
NAS1391I --> FAC=420808430404
NAS1391I --> IDLETO=00015
NAS1391I THE FOLLOWING OPERANDS FOR SPU Q1990100 (DEFINED ON MCH MCH1
NAS1391I AS SVC3 ENTRY 0000) COME FROM MXT MXT0      RATHER THAN THE MCH
NAS1391I WHEN THEY ARE OMITTED FOR THE SPU
NAS1391I --> CUD=01111111
NAS1391I --> DCEADDR=1111
NAS1391I --> DTEADDR=9999
NAS1391I --> FAC=420808430404
NAS1391I --> IDLETO=00015
NAS1391I THE FOLLOWING OPERANDS FOR SLU MCH15001 (DEFINED ON MCH MCH1
NAS1391I AS SVC5 ENTRY 0000) COME FROM MXT MXT0      RATHER THAN THE MCH
NAS1391I WHEN CONNECTION IS NOT ESTABLISHED VIA A <RMTNAME>
NAS1391I --> CUD=01111111
NAS1391I --> DCEADDR=1111
NAS1391I --> DTEADDR=9999
NAS1391I --> FAC=420808430404
NAS1391I --> IDLETO=00015
NAS1391I --> PADPARAM=001/001 002/001 003/002 004/000 005/002 007/002 ..
NAS1391I THE FOLLOWING OPERANDS FOR SLU MCH15001 (DEFINED ON MCH MCH1
NAS1391I AS SVC5 ENTRY 0000) COME FROM MXT MXT1      RATHER THAN THE MCH
NAS1391I WHEN CONNECTION IS ESTABLISHED VIA <MXT1>
NAS1391I --> CUD=01222222
NAS1391I --> DCEADDR=2222
NAS1391I --> FAC=0101420A0A430202
NAS1391I --> PADPARAM=001/001 002/001 007/002 012/001 013/007 021/000 ...
NAS1391I THE FOLLOWING OPERANDS FOR SLU MCH15002 (DEFINED ON MCH MCH1
NAS1391I AS SVC5 ENTRY 0001) COME FROM MXT MXT0      RATHER THAN THE MCH
NAS1391I WHEN CONNECTION IS ESTABLISHED VIA <MXT0>
NAS1391I --> CUD=01111111
NAS1391I --> DCEADDR=1111
NAS1391I --> DTEADDR=9999
NAS1391I --> FAC=420808430404
```

```
NAS1391I --> IDLETO=00015
NAS1391I --> PADPARAM=001/001 002/001 003/002 004/000 005/002 007/002 ...
```

General Notes:

- 1) The NAS1391I message for SLU MCH1P001 above only displays the FAC= operand for MXT0 even though other operands were specified. This is because only the FAC= operand is used as a PVC SLU override operand.
- 2) The NAS1391I message for SLU MCH1P002 shows 'NO OVERRIDES' because the FAC= operand for was omitted for MXT2. Note that no NAS1391I message is issued for SLU MCH1P003 since no MXT was associated with this SLU.
- 3) The NAS1391I messages for the SVC3= operand above list MXT overrides when operands are omitted from the SPU. The order is MXT first, SPU then MCH. This only applies when SPU connections are established via the MCH and not by IDBLK/IDNUM matching which is MCH independent.
- 4) The SVC0= and SVC5= operands above are identical except for the SLU names. Note that the NAS1391I messages for the SVC0= and SVC5= operands are the same except that the PADPARAM= operand is listed for SVC5= SLUs (when specified on the MXT) while it is not for SVC0 SLUs. This is because only SVC5= SLUs use the PADPARAM= operand.
- 5) The NAS1391I messages are generated only when an SLU or SPU has an associated MXT as provided in the PVC= or SVC0/3/5= operand. If you wish to withhold the NAS1391I messages from the SYSPRINT log, specify ALRMFLTR=(...,NAS1391I(P),...) on the BUILD definition statement.
- 6) If you plan on filtering NAS1391I messages from SYSPRINT, we highly recommend that you first run a FASTRUN pass without filtering so that these important messages are at least displayed once for your perusal.

Note: NAS1391I message support was introduced into 240 as Enhancement APAR 2400018.

- **OPTIONS=(TIMESTAMP=*format*-{+D|-D})** was added to the BUILD definition statement to enhance SYSPRINT timestamp resolution.

Specify **TIMESTAMP=HH:MM:SS[.T[H[M[I]]]]** to set the HNAS SYSPRINT log timestamp resolution to seconds, tenths of seconds (T), hundredths of seconds (H), thousandths of seconds (M) or ten-thousandths of seconds (I). STD and EXT are abbreviations for HH:MM:SS and HH:MM:SS.THMI, respectively.

Append +D to the **TIMESTAMP** format to cause the Julian Date to prefix the timestamp for each SYSPRINT record (treated the same as PRNTDATE ON).

- **PARSE** definition statement has been added to allow customers to specify their own parsing characters that will override established HNAS parsing characters for the CDF scan. The **PARSE** definition statement must be the first definition statement in the CDF and must precede any CDF records that utilize the special parsing characters that it defines. After the **PARSE** definition statement is processed, the override parsing characters will go into effect turning the established parsing characters into non-parsing characters. They will simply become data characters.

HNAS V2R4M0 New Features

The following parsing override characters are currently provided:

- **_** Comment start character

The semi-colon (;) is normally used to indicate the start of a comment that follows configuration data on a CDF record. This character can also occur in record column 1 when the entire record is to be treated as a comment, however, an asterisk (*) in record column 1 is normally used for this purpose and cannot be overridden. To override the semi-colon as the comment start character, specify **OPTIONS=(...,COMCHAR=*comchar*,...)** on the PARSE definition statement.

- Continuation start character

The plus-sign (+) is normally used to indicate that data on the current CDF record is to be continued on the next CDF record. To override the plus-sign as the continuation start character, specify **OPTIONS=(...,CONCHAR=*conchar*,...)** on the PARSE definition statement.

- Suboperand separator character

The forward-slash (/) is normally used to separate suboperands for an operand entry that accepts multiple suboperands. For example, `SVC0=(...,sluname/dteaddr/mxtname,...)`. To override the forward-slash as the suboperand separator character, specify **OPTIONS=(...,SEPCHAR=*sepchar*,...)** on the PARSE definition statement.

- Forced space character

The blank () is normally used to provide spacing. When CDF records are continued using the plus-sign (+) or the **OPTIONS=CONCHAR=*comchar*** override character, the CDF parser removes all blanks between the last non-blank character on a record and the continue character. If you want HNAS to leave spaces between the last non-blank character and the continue character on a record, you can specify a forced space character that you can use to provide a place holder for blanks that are not to be deleted. To supply a forced space character, specify **OPTIONS=(...,FSPCHAR=*fspchar*,...)** on the PARSE definition statement. For example, if **OPTIONS=FSPCHAR=~** is specified, you can prevent blanks from being deleted on a continuation record as follows:

```
BUILD CONCMDQ=('TRCMCH~~~+
                ~~ICR')
```

This would be processed as though `CONCMDQ=('TRCMCH ICR')` were entered.

comchar, ***conchar***, ***sepchar*** and ***fspchar*** in the above descriptions is a single EBCDIC character without framing quotes ("").

- **PRTSWLST={({LOOP|STOP},SWITCHAFTERINIT,SWITCHAT *time*,
{*ddname1*|DYNAMIC=*outclass*},...,{*ddnamen*|DYNAMIC=*outclass*})**

was added to the BUILD definition statement to provide automatic SYSPRINT switching when the current SYSPRINT log file becomes full or when the designated action occurs.

You may specify static DDNAMEs and/or request DYNAMIC DDNAME allocation. The DDNAMEs you specify or request dynamically are used sequentially. The default SYSPRINT file is always used initially (`ddname=SYSPRINT`).

If you specify **DYNAMIC=** (no output class) or **DYNAMIC** (no equal sign), class A is assumed. You can also specify just one DYNAMIC value with LOOP which will cause a new dynamically allocated

SYSOUT file to be used when the PRTLMT is reached, the specified switch action event occurs or when the PRNT NEXTPRSW console command is entered. For example, PRTSWLST=(LOOP,SWITCHAFTERINIT,SWITCHATMIDNIGHT,DYNAMIC).

The dynamic datasets are allocated with a DDNAME of DYN#xxxx where xxxx=0001 for the first dynamically allocated dataset and is incremented by one as new datasets are allocated. If you issue a DNAS JCL command, you will see the dynamically allocated DDNAMEs listed. This DDNAME naming convention was chosen so the dynamic DDNAMEs would be easy to remember.

Note: PRTSWLST=DYNAMIC support was introduced into 240 as Enhancement APAR 2400001.

LOOP, STOP, SWITCHAFTERINIT, SWITCHAT hh 00, SWITCHAT6AM, SWITCHATMIDDAY, SWITCHAT6PM and SWITCHATMIDNIGHT are reserved keywords that can be specified anywhere in the PRTSWLST= operand list. They are NOT treated as DDNAMEs.

If LOOP is specified as a PRTSWLST= suboperand, DDNAMEs in the list are used in a round robin fashion. When the default SYSPRINT file becomes full, it is closed and *ddname1* is then used. When *ddname1* becomes full, *ddname2* is then used. This continues until *ddnamen* (the last DDNAME in the list) is used, in which case the *ddname1* file is reused and the process repeats. Note that *ddnamei* can be SYSPRINT in which case it will be reused.

If STOP is specified as a PRTSWLST= suboperand, SYSPRINT logging stops when *ddnamen* becomes full.

If SWITCHAFTERINIT is specified as a PRTSWLST= suboperand, logging in the default SYSPRINT dataset stops and logging in the *ddname1* dataset starts after the NAS0001I INITIALIZATION COMPLETE message is issued. This option was added so that customers can send us an abbreviated HNAS log file if we need to look at the CDF scan and DNAS console command output.

If SWITCHAT hh 00 is specified (military time) as a PRTSWLST= suboperand, logging in the current PRTSWLST= dataset stops and logging in the next PRTSWLST= *ddnamei* dataset starts at the hour specified by hh every day. Note that hh can be 01 to 24|00 (00 is treated the same as 24).

If SWITCHAT6AM (=SWITCHAT0600) is specified as a PRTSWLST= suboperand, logging in the current PRTSWLST= dataset stops and logging in the next PRTSWLST= *ddnamei* dataset starts at 6AM every day.

If SWITCHATMIDDAY (=SWITCHAT1200) is specified as a PRTSWLST= suboperand, logging in the current PRTSWLST= dataset stops and logging in the next PRTSWLST= *ddnamei* dataset starts at 12PM every day.

If SWITCHAT6PM (=SWITCHAT1800) is specified as a PRTSWLST= suboperand, logging in the current PRTSWLST= dataset stops and logging in the next PRTSWLST= *ddnamei* dataset starts at 6PM every day.

If SWITCHATMIDNIGHT (=SWITCHAT2400) is specified as a PRTSWLST= suboperand, logging in the current PRTSWLST= dataset stops and logging in the next PRTSWLST= *ddnamei* dataset starts at 12AM every day.

Note: If the STOP action is in effect, logging is terminated when a switch occurs. Switching can occur based on a PRTSWLST= action like SWITCHATMIGNIGHT or when the current PRTSWLST= *ddnamei* dataset becomes full. For this reason, we recommend that the LOOP action be used when SWITCHAFTERINIT or any of the SWITCHAT actions is specified.

HNAS V2R4M0 New Features

For example: PRTSWLST= (LOOP , SWITCHATMIDNIGHT , SYSPRNT1 , SYSPRNT2 , SYSPRNT3) .

The default SYSPRINT DDNAME should not be included in the list unless you want it to be reused. If a list entry is reused after it was detected full and DISP=MOD is specified, it will appear full again immediately when the first record is written because new data is added to the end of the file. DISP=OLD will cause the old data to be purged so that new data is logged at the beginning of the file when it is reused.

In all cases, an alert message is issued when a SYSPRINT switch takes place and, in the case of the STOP action, when the last PRTSWLST= *ddname* file is used. This will indicate that SYSPRINT logging has been terminated. If you wish to restart SYSPRINT logging, you will manually have to enter the PRNT OPEN *ddname* command or the new PRNT RSMEPRSW command.

Alarm messages **NAS0207W**, **NAS0208I** and **NAS0209E** are now generated when PRTSWLST= SYSPRINT switching support is enabled. Please refer to the Alert/Alarm Message and Clear/Reset Code Enhancements area of this section for a description of the messages.

- **SCHEDULE {*ddname*(*hh:mm:ss,cmd,...,hh:mm:ss,cmd*)|LIST|SUSP|RSME|PRG}** console command added to allow scheduled console commands to be added, deleted or displayed.

Syntax: SCHEDULE {*ddname*(*hh:mm:ss,cmd,...,hh:mm:ss,cmd*)|LIST|SUSP|RSME|PURGE}

(*hh:mm:ss,cmd,...,hh:mm:ss,cmd*) is an *inline* schedule list. An *inline* schedule list is added to the end of an existing schedule list if one is active.

ddname identifies a file that contains a list of times and commands. A *ddname* schedule list is added to the end of an existing schedule list if one is active. A *ddname* schedule list can contain the following records:

Comments are allowed and start with an asterisk (*) or semi-colon (;) in record column 1. Comments can also appear on a command line but must start with a semi-colon after the command. A single time and command can be specified on a single record as follows:

hh:mm:ss , cmd ; comment

Multiple times and commands can be specified on a single record as follows:

(*hh:mm:ss , cmd , ... , hh:mm:ss , cmd*) ; *comment*

LIST [*ddname*] is used to display a schedule list. If *ddname* is not specified, the active (current) schedule list is displayed. If *ddname* is specified, the identified schedule list is displayed but the active schedule list remains unaffected.

SUSP causes the active schedule list to be suspended so that no commands will be scheduled. The list itself is maintained.

RSME causes the active schedule list that was suspended to be resumed so that commands will be scheduled again.

PRG causes the current schedule list to be cleared. No commands are left to be scheduled.

General Notes:

1) In order to provide symmetry between SCHEDULE and EXEC command processing, the EXEC command was also modified to allow multiple commands to be specified on a single record within a command list file as follows:

(cmd , ... , cmd) ; comment

A single command can still be specified on a single record as follows:

cmd , comment

2) For both the EXEC and SCHEDULE commands, you may enclose a *cmd* within single or double quotes (e.g., *cmd*, '*cmd*' or "*cmd*" are valid). If single quotes are part of a *cmd*, it can be enclosed in double quotes (e.g., SMSG '*text*' or "SMSG '*text*'" are valid). The enveloping single or double quotes are removed from each *cmd* before the values are saved.

3) For both the EXEC and SCHEDULE commands, leading and trailing blanks are removed from each *cmd* (and time for the SCHEDULE command) before the values are saved.

4) For both the EXEC and SCHEDULE commands, an END command will signal the end of data in a file. In addition, for EXEC, an embedded EXEC command and for SCHEDULE, an embedded SCHEDULE command will also signal the end of data in a file. Any records that follow these statements will be ignored.

5) For both the EXEC and SCHEDULE commands, the CONCMDQ queue is used to save commands. For the EXEC command, *cmd* values are enqueued to CONCMDQ in a first in, first out (FIFO) manner. For the SCHEDULE command, *cmd* values are enqueued to CONCMDQ in a last in, first out (LIFO) manner. LIFO enqueue is used because scheduled commands must be executed at the specified time rather than after any other queued commands.

WARNING: If a command is currently running when a scheduled command is enqueued, the current command is aborted in favor of the scheduled command (the same action takes place when a command is entered by a console operator). If a command list is running, the list is interrupted while the scheduled command executes. The command list is then restarted with the next queued command after the scheduled command executes.

6) Multiple commands can be scheduled for execution at the same time. For example:

(12:00:00,ALARM LOG=?,12:00:00,DNAS)

WARNING: Because scheduled commands are enqueued to CONCMDQ in a LIFO manner, commands with the same schedule time are executed in the reverse order from how they are listed in the schedule list. In the list above, DNAS will be executed before ALARM LOG=? at approximately noon each day. You should consider this when listing commands in a schedule list.

A simple solution to ensure processing order (avoiding reverse coding) is to sequentially list the commands and step by 1 second as depicted in the following sample:

(12:00:00,ALARM LOG=?,12:00:01,DNAS)

7) Wildcards (*) may be specified for any time digit except the low order seconds digit. When a wildcard character is supplied for a time digit (*hh:mm:ss*), it will be replaced with the corresponding digit from the current time then the resulting time will be tested against the current time.

Examples:

HNAS V2R4M0 New Features

If (**:**:00,ALARM LOG=?) is a scheduled command, the ** will be replaced by the current hours and minutes values so that ALARM LOG=? will be executed every 60 seconds.

If (**:30:00,ALARM LOG=?) is a scheduled command, the ** will be replaced by the current hours value so that ALARM LOG=? will be executed at half past every hour.

If (1*:00:00,ALARM LOG=?) is a scheduled command, the * will be replaced by the low order hours digit so that ALARM LOG=? will be executed on the hour between 10:00 and 19:00 each day.

If (12:**:00,ALARM LOG=?) is a scheduled command, the ** will be replaced by the current minutes value so that ALARM LOG=? will be executed once per minute between 12:00 and 12:59 each day.

You can specify a wildcard for any time except the low order second digit (*hh:mm:s**) because a scheduled command could be executed every second which could impact HNAS performance.

8) For the SCHEDULE command when an *inline* list or *ddname* list is given, entries are added to the end of the active SCHEDULE list in effect. If the active schedule list contains another SCHEDULE command, the specified schedule list will replace the active (RESIDENT) list. This allows you to chain schedule lists (see Note 9 examples).

9) For the EXEC command, an embedded EXEC command allows you to chain command lists. For the SCHEDULE command, an embedded SCHEDULE command allows you to chain schedule lists.

For example, if you want to run a different schedule list each day, you could structure them as follows (assume HNAS is started on Monday):

```
BUILD SCHEDULE=DD4MON
```

```
//DD4MON DD DSN=hlq.HNASMAC (DD4MON) , DISP=SHR
//DD4TUE DD DSN=hlq.HNASMAC (DD4TUE) , DISP=SHR
//DD4WED DD DSN=hlq.HNASMAC (DD4WED) , DISP=SHR
//DD4THU DD DSN=hlq.HNASMAC (DD4THU) , DISP=SHR
//DD4FRI DD DSN=hlq.HNASMAC (DD4FRI) , DISP=SHR
//DD4SAT DD DSN=hlq.HNASMAC (DD4SAT) , DISP=SHR
//DD4SUN DD DSN=hlq.HNASMAC (DD4SUN) , DISP=SHR
```

Each schedule list file will end with a SCHEDULE command that points at the next file which will take effect at midnight.

The DD4MON file will end with the following statement: (00:00:00,SCHEDULE DD4TUE)

The DD4TUE file will end with the following statement: (00:00:00,SCHEDULE DD4WED)

:

The DD4SUN file will end with the following statement: (00:00:00,SCHEDULE DD4MON)

In this way, you can operate with a different set of scheduled commands each day and because DISP=SHR is specified for each *ddname*, you can make changes to any of the inactive list files while HNAS is running. Note also that you can alter the normal schedule list flow by simply entering a new SCHEDULE command from the operator console.

As another example, suppose you want to run different schedule lists during the day and night, you

could structure them as follows (assume HNAS is started during the day):

```
BUILD SCHEDULE=DD4DAY
```

```
//DD4DAY DD DSN=hlq.HNASMAC(DD4DAY),DISP=SHR
//DD4NIT DD DSN=hlq.HNASMAC(DD4NIT),DISP=SHR
```

As before, each schedule list file will end with a SCHEDULE command that points at the other file which will take effect at the specified time.

The DD4DAY file will end with the following statement: (00:00:00,SCHEDULE DD4NIT)

The DD4NIT file will end with the following statement: (06:00:00,SCHEDULE DD4DAY)

10) When a scheduled command is executed, the normal console command echo prefix is changed from HNASCMD-> to SCHDCMD-> so that scheduled commands can be identified in the SYSPRINT log file.

11) When SCHEDULE=schdlist is decoded during HNAS activation, scheduled commands are suspended until after the **NAS0001I INITIALIZATION COMPLETE** message is issued and after first CONCMDQ=cmdlist processing.

Note: The new SCHEDULE command support was introduced into 240 as Enhancement APAR 2400064.

- **SVC0|5**={...(sluname/{Xidnum|dteaddr|<rmtname>}...)} permits an MXT|SVC per dteaddr and a dteaddr via rmtname. New **REMOTE TYPE=SVC** was added to provide a smaller MXT footprint without PADPARAM=value support.

This enhancement is provided so that customers can associate a separate TYPE=MXT REMOTE or the new TYPE=SVC REMOTE with each callout and callin DTE address when multiple DTE addresses are specified for an SVC0= or SVC5= operand entry. Prior to this enhancement, an LLC0 callout resource with multiple destination addresses was specified as follows:

```
MCH1 REMOTE TYPE=MCH|XTP
      SVC0=(n,sluname/dteaddr1-dteaddr2-dteaddr3{O|T}[applid]
          /mxtname/cud,...
```

When the SLU is bound by a PLU, a Call Request packet is generated using dteaddr1 as the *called* DTE address. If this call fails (Clear received or no response to Call Request packet), a new call is then placed to dteaddr2 and, if necessary, to dteaddr3.

When a dteaddr*n* value is specified, the MXT identified by mxtname above becomes the TYPE=MXT REMOTE that is associated with the SLU for the entire LU/VC session. The dteaddr*n* value is used as the *called* DTE address while the named MXT supplies the *calling* DTE address (DCEADDR= operand), facilities data (FAC= operand), call user data (CUD= operand) and for PAD calls, PAD parameters (PADPARAM= operand). When multiple dteaddr*n* values are coded, all outbound Call Request packets will carry the same *calling* DTE address, facilities data and call user data because there is only one MXT pointer.

HNAS has been modified to allow the name of an MXT (short for a TYPE=MXT REMOTE) or the new SVC (short for a TYPE=SVC REMOTE) to be specified in place of the dteaddr*n* suboperands.

HNAS V2R4M0 New Features

When an MXT or SVC is specified as a *rmtnamei* in place of a *dteaddr* value, it becomes the REMOTE that is associated with the SLU for the duration of the LU/VC session. In this case, it will be the REMOTE that provides the *called* DTE address (DTEADDR= operand), *calling* DTE address (DCEADDR= operand), facilities data (FAC= operand) and call user data (CUD= operand). These four operands provide all the information necessary to create a unique Call Request packet.

For PAD calls that require PADPARAM= values to override those on the root MCH, an MXT rather than an SVC should be used in place of a *dteaddr* value. This is because the SVC REMOTE does not support the PADPARAM= operand.

If PADPARAM= values are not required, we recommend using an SVC rather than an MXT for call processing. The SVC control block uses less memory than the MXT. The SVC control block is X'178' (376) bytes in length while the MXT control block is X'480' (1152) bytes in length.

Note that when an MXT or SVC is specified in place of a *dteaddr* value, the MXT identified by *mxt-name* will be ignored for the duration of the LU/VC session. However, if specified, this MXT will be used to supply VTAM parameters for the SLU during AMNF generation via the FASTRUN process. If this MXT is not specified, VTAM parameters come from the root MCH.

If the direction indicator at the end of the *dteaddr* list is a 'T' instead of an 'O' (two-way support) and a *dteaddr* list entry is a *rmtnamei* instead of a *dteaddr*, the named REMOTE will be used for inbound as well as outbound connections. If the direction indicator is an 'I' and a *dteaddr* list entry is a *rmtnamei* instead of a *dteaddr*, the named REMOTE will be used for inbound connections only. For inbound calls, the DTEADDR= value is used like a *dteaddr* value from *dteaddr* list. If the *calling* DTE address from the inbound Call Request packet matches the DTEADDR= operand value, the SLU will be allocated for the inbound call.

The new syntax for the REMOTE SVC0= and SVC5= operands is as follows:

```
MCH1 REMOTE TYPE=MCH|XTP
          SVCx=(n,sluname/dteid1-dteid2-dteid3{T|O|I}[applid]
              /mxtname/cud,...
```

The new syntax for the MRMT SVC0= and SVC5= parameters is as follows:

```
MRMT mchname SVCx=sluname/dteid1-dteid2-dteid3{T|O|I}[applid]/mxtname/cud
```

Where: *dteidi* = *dteaddr* (*dd...dd*) (for T|O|I)
 = *Xidnumi* (*Xdd...dd*) (for T|I)
 = *<rmtnamei>* (for T|O|I)
 x = 0 for SVC0 or 5 for SVC5

CDF example to specify a *dteaddr*, *Xidnum* and *<rmtname>* value for the same SLU:

```
MCH1 REMOTE TYPE=MCH
          SVC0=(20,
              MCH10011/1234-X1234-<SVC1>T01/MXT0,
              :
          )
SVC1 REMOTE TYPE=SVC
          DTEADDR=10981000,DCEADDR=20351000
          FAC=420909430707,CUD=C0000000
```

```

MXT0 REMOTE TYPE=MXT
      DTEADDR=10991000 ,DCEADDR=20361000
      FAC=420707430303 ,CUD=C0000000
  
```

```

MXT1 REMOTE TYPE=MXT
      DTEADDR=10991001 ,DCEADDR=20361001
      FAC=420808430303 ,CUD=C0000000
  
```

MRMT example to modify the *Xidnum* and *<rmtname>* value while leaving the *dteaddr* value alone:

```
MRMT MCH1 SVC0=MCH10011/-X6789-<MTX1>
```

Note that when a hyphen is entered without a preceding value, it acts as a place holder preserving the value of the skipped *dteidi* list entry (1234 in this case).

DRMT example to review MRMT change:

```
DRMT MCH1 SVC0=MCH10011
```

```

RMTNAME  OPERAND  TYPE
MCH1     0084F4E8  MCH
          SVC0=0020
          MCH10011/1234-X6789-<MXT1>T01/MXT0
  
```

- Notes:**
- 1) Up to 3 *dteidi* values may be specified for any direction identifier (T|O|I). *Xidnumi* values are only used for inbound connections while *dteaddr* and *<rmtnamei>* values can be used for inbound and outbound connections. For inbound connections, the SLU is allocated if a match occurs for any *dteidi* value in the list.
 - 2) The *rmtnamei* in the *dteidi* list must be enclosed in non-equivalence symbols <>. The MXT|SVC addressed by a *<rmtnamei>* operand is used for the duration of the call. Note that in this case, an MXT name coded after the *dteidi* list (*/mxtname*) is not used. It is only used when a *dteaddr* value is specified as a *dteidi* list entry.
 - 3) A *dd...dd* value prefixed with an 'X' is used as an IDNUM value to be matched against IDNUM value that is carried in the CUD field of an inbound Call Request packet. If a match occurs, the SLU is allocated for the inbound call. The *Xdd...dd* value can contain up to 14 paired hex digits.
 - 4) A *dd...dd* value without the 'X' represents a DTE address that is used as the *called* DTE address in an outbound Call Request packet OR is matched against the *calling* DTE address from an inbound Call Request packet. If a match occurs, the SLU is allocated for the inbound call. The *dd...dd* value can contain up to 15 decimal digits.
 - 5) Multiple *dteidi* values may also be specified when the direction identifier is I. In this case, inbound *dteaddr* or *Xidnumi* matching will look at up to 3 values. If *<rmtnamei>* is specified as a *dteidi* value, the *dteaddr* will come from the DTEADDR= operand on the named MXT|SVC.
 - 6) *Xidnum* values are not supported in the DTEADDR= operand on a TYPE=MXT REMOTE. An *Xidnum* value may only be specified as a *dteidi* value.

HNAS V2R4M0 New Features

HNAS V2R3M0 New Features

V2R3M0 General Availability - 03-22-2004

Functional Enhancements:

- All appropriate V2R2M0 **APARs** have been incorporated as permanent fixes in HNAS V2R3M0 as of the general availability date.

- **SMP/E Product Installation Support Implemented for HNAS V2R3M0 edistributions**

SMP/E product installation now available for HNAS 230 edistributions. The initial SMP/E product implementation was designed for installation using HNAS private global zone support although common shared global zone support was added to 230 effective 03-31-2004.

The HNAS 230 product can now be installed using SMP/E or the default TSO Transmit format installation method. If you require SMP/E please indicate as such when you order the product.

- **Extended Diagnostic Event Reason Code support.**

Extended diagnostic event reason codes are now displayed in various alert/alarm messages, trace entries and some console display messages. These extended codes will make it easier to define session connect and disconnect causes as well as improve general debugging.

- **CART= WTO support for Session Manager (Netview, TDSLink) Operator Consoles.**

The CART= operand is now supported on the WTO macro to pass a token from MODIFY command back to the MODIFY sender in order to associate console commands with responses. This is used by Netview, TDSLink and other host products to solicit specific information from the HNAS console subsystem.

The new ALRMCART= operand has been added to the BUILD definition statement so that alarm messages can be routed to a specific component like Netview or TDSLink. The ALRMCART= token is passed to the WTO service routine via the CART= operand for alarm messages only.

Refer to the "**NETVIEW** routing support for asynchronous alarm messages as well as synchronous console command output..." later in this section for additional Netview support services information.

- **Improved Console and SYSPRINT output filtering.**

Using new **PRNT type** and **SHOW CONS** start parameters/console commands, user can now control which specific types of output are directed at the SYSCONS or SYSPRINT interfaces.

- **SYSPRINT dataset recording improvements.**

HNAS logic enhanced to better handle SYSPRINT *ddname* assignment and dataset full conditions.

HNAS V2R3M0 New Features

The PRTLMT= operand on the BUILD definition statement limits the number of records that can be written to SYSPRINT while an HNAS ABEND EXIT eliminates SB37 SYSPRINT dataset full conditions. <See Problem Log Ref: 2002346A if additional information is required>.

Future enhancements will allow automatic switching of SYSPRINT datasets based on a list provided in the Configuration Data File. When a dataset becomes full, the next *ddname* in the SYSPRINT list will be used to continue the logging of events and or trace information.

- **SYSPRINT log filtering improvements.**

New HNAS start parameters and PRNT console command arguments have been added so that various informational messages (severity code of I) can be filtered from the SYSPRINT log based on the component type. HNAS starts with all component logging enabled. To turn off informational message logic for a particular component, the following start parameters may be used:

```
PRNTCNFG OFF - Inhibit logging of informational configuration messages.
PRNTLU OFF   - Inhibit logging of informational LU messages.
PRNTSYS OFF  - Inhibit logging of informational system messages.
PRNTTCP OFF  - Inhibit logging of informational TCP/IP messages.
PRNTVC OFF   - Inhibit logging of informational VC messages.
PRNTXOT OFF  - Inhibit logging of informational XOT messages.
PRNTXTP OFF  - Inhibit logging of informational XTP messages.
PRNTVTAM OFF - Inhibit logging of informational VTAM messages.
```

To turn on or off informational message logic for a particular component using the PRNT console command, you may enter the following:

```
PRNT CNFG ON|OFF - Toggle logging of informational configuration messages.
PRNT LU ON|OFF   - Toggle logging of informational LU messages.
PRNT SYS ON|OFF  - Toggle logging of informational system messages.
PRNT TCP ON|OFF  - Toggle logging of informational TCP/IP messages.
PRNT VC ON|OFF   - Toggle logging of informational VC messages.
PRNT XOT ON|OFF  - Toggle logging of informational XOT messages.
PRNT XTP ON|OFF  - Toggle logging of informational XTP messages.
PRNT VTAM ON|OFF - Toggle logging of informational VTAM messages.
```

- **Start parameter follower improvements.**

All HNAS start parameters may now accept ON and OFF as followers. ON is the default follower if none is specified after a parameter. For example, TRCPRNT and TRCPRNT ON are treated the same. OFF turns off the function that the parameter sets. For example, TRCVC OFF turns off the default TRCVC MINDATA setting.

- **XOT TAP= support improvements.**

XOT TAP= support now provides calling/called address, facilities and call user data parameters allowing **greater control** over **routing** and **filtering** of the special XOT call request packet used for tapping control. The tapping XOT calls can now be **selectively cleared** (providing the necessary tapping response) using the Cisco '**x25 route dteaddr clear**' configuration option. The tapping DTEADDR=, DCEADDR=, FAC= and CUD= fields are supported on the TYPE=XOT REMOTE for remotes with tapping enables (TAP=nn option).

- **TRCTRAP= diagnostic automation tools**

TRCTRAP= support provides automation tools for improved debugging and trace suspension.

Sometimes it is difficult to capture the cause of a problem in the HNAS internal trace table when tracing has to be stopped manually. In many cases, the only way to resolve a problem is to run HNAS traces. If a problem is intermittent, TRCPRNT is not a viable option because it can create an inordinate amount of SYSPRINT and consume valuable CPU cycles. Internal tracing is the next best option but it must be stopped as soon as possible after the problem has occurred. This may be difficult and, in some cases, may even be impossible.

HNAS has been modified to accept a new configuration operand and console command that will provide the ability to suspend internal tracing (and TRCPRNT) based on alarm message ID match, an input packet match or an output packet match. Multiple alarm IDs and packets may be specified so that a collection of values may be monitored (trapped)...

Note: Please refer to the Enhancements - Expanded Content heading at the end of this section for additional information regarding this new or enhanced feature.

This new feature was introduced into 230 as Enhancement APAR 2300045/2300047.

- **NETVIEW** routing support for asynchronous alarm messages as well as synchronous console command output can be provided using the PFXWTO, PFXWTO CONS and SHOWCONS start parameters or the PFXWTO ON, PFXWTO CONS and SHOW CONS console commands in conjunction with the Netview ASSIGN command as follows:

Use the PFXWTO start parameter or the PFXWTO ON console command to cause the NASNAME= operand value to prefix all alarm messages issued by HNAS.

Use the PFXWTO CONS start parameter or the PFXWTO CONS console command to cause the NASNAME= operand value to prefix all console command output when SHOWCONS is in affect.

Use the SHOWCONS start parameter or the SHOW CONS console command to cause all console command output to be routed to SYSCONS in addition to SYSPRINT.

To allow Netview to filter on the NASNAME= operand value, you need to issue the following Netview ASSIGN commands:

For alarm messages: `ASSIGN MSG=nasname, PRI=opername`

For console output: `ASSIGN MSG=nasname, COPY=opername`

Some improved features for this support was introduced into 230 as Enhancement APAR 2300006 and 2300026.

- **TRCTRAP= diagnostic enhancement**

Some times a customer is unable to capture the cause of certain problems when systems operation is automated/unattended. Customers have reported that it is difficult and sometimes impossible to trap

HNAS V2R3M0 New Features

an error and SHUTDOWN HNAS before debugging control block information is lost. This may be the case even when operations personnel are present and is certainly the case when no one is around.

To avoid the need for an operator initiated SHUTDOWN or a forced ABEND, the TRCTRAP= operand logic has been modified to allow an action operand (,TRAPACTION=) as follows:

```
TRCTRAP= (ALRMLIST= ( . . . ) ,  
          RCVLIST= ( . . . ) ,  
          XMTLIST= ( . . . ) ,  
          TRAPACTION= { ALL | NONE | ( { SUSP | NOSUSP } , { SNAP | NOSNAP } ) } )
```

The action taken when a TRCTRAP 'hit' occurs currently defaults to TRAPACTION=SUSP which causes HNAS tracing and SYSPRINT trace logging to be suspended. The new TRAPACTION=SNAP operand will allow a snapshot of all HNAS control blocks and trace entries to be logged in SYSPRINT when a 'hit' occurs. If SUSP is required in addition to SNAP, both must be specified so that both functions are performed. For example:

```
TRCTRAP=(TRAPACTION=(SUSP,SNAP)) or simply TRCTRAP=(TRAPACTION=ALL).
```

If either SUSP or SNAP is specified by itself, only that action is performed on a hit. If the TRAPACTION= suboperand is omitted, SUSP is assumed which provides downward compatibility with operation under implementation of Enhancement APAR 2300045. If you wish to define trap filters but postpone any action on a trap hit until trace trapping is started or resumed using the TRCTRAP RSME, TRCTRAP RSMESNAP or TRCTRAP RSMEALL console commands, specify TRAPACTION=(NOSUSP,NOSNAP) or simply TRAPACTION=NONE.

Note: The TRCTRAP console command has been modified to accept TRAPACTION={({SUSP|NOSUSP},{SNAP|NOSUSP}) as well as TRAPACTION={ALL|NONE} as an argument in order to provide equivalent functions for the TRCTRAP= configuration operand.

This new feature was introduced into 230 as Enhancement APAR 2300094.

- **WTO** routing codes can now be configured so that alarm messages can be routed to the **NETVIEW** NETLOG without also going to SYSLOG when SYSLOG is defined as a HARDCOPY console. This is accomplished by specifying **OPTIONS=WTOROUTCDE(ALRM)=value** (we recommend *value=11* => programmer information) on the BUILD definition statement and coding the following in the **CONSOLxx** member in the SYS1.PARMLIB library for the SYSLOG HARDCOPY console:

```
ROUTCODE(1-10,12-128)
```

This will route all WTOS except ROUTCODE=11 to the HARDCOPY console but will require a system IPL to activate. To accomplish the same thing immediately, the following system command can be used:

```
VARY SYSLOG,HARDCPY,DROUT=(11)
```

This drops ROUTCDE=11 from the HARDCOPY ROUTCODE list but will only last until the system is re-IPLed. So the permanent change to CONSOLxx is required.

Note: All HNAS WTOS use a default routing code of 8 (teleprocessing control). This routing code along with 1,2,3,4,7,10 and 42 cannot be dropped from the HARDCOPY ROUTCODE= list. For this

reason we recommend specifying a WTOROUTCDE= value that is unused and not one of those listed above.

Note: To route error alarm messages, the HNAS SHOWERR parameter must be in effect. This parameter can be specified as an EXEC start parameter (PARM='...,SHOWERR,...') or via the SHOW ERR console command. Note that SHOWERR is a default HNAS start parameter so that technically it does not have to be specified in the PARM= operand.

To route all alarm messages (error and informational alarms), the HNAS SHOWON parameter must be in effect. This parameter can be specified as an EXEC start parameter (PARM='...,SHOWON,...') or via the SHOW ON console command.

The HNAS PFXWTO parameter can be used to prefix all HNAS alarm messages with either the BUILD NASNAME= operand value (specify PFXWTO with no follower) or with any text string (specify PFXWTO followed by a *text* string). For example, specify EXEC PARM='...,PFXWTO *text*,...' or issue the PFXWTO *text* console command. If the PFXWTO parameter is used, the MSG= operand for the ASSIGN command or the MSGID= operand for the IF/THEN clause above will have to be changed to specify the WTO prefix text (either the NASNAME= operand value or *text*).

Note: WTOROUTCDE(ALRM)= support was introduced into 230 under APAR 2300163.

Configuration Enhancements:

- Various **CDF Options** and **Parameters** were added or changed to allow the user greater flexibility when defining resources in V2R3M0:

- **Callout Session Connect Balancing Extension**

Consecutive RTEOUT= entries with the same selection address can be used as a round robin group. Outbound calls will be balanced across the group. OPTIONS=BALANCERTEOUT is required.

- **Backup Router Support for LLC0/5 Callout**

When an LLC0/5 callout fails (timeout or clear) HNAS will try to find an alternate path for the call by continuing to process the DTE address from the SVC0/5 entry in the RTEOUT= list. If another match in RTEOUT= is located the call will be tried on the associated router. After the entire RTEOUT= list has been processed the process is repeated for the second and third DTE address in the SVC0/5 entry (if present).

- **CUD=, FAC=, DCEADDR= and DTEADDR= (new) allowed on TYPE=XOT REMOTES**

These operands allow the fields of a Call Request packet to be provided by the user for the router TAP request. When these operands are omitted, the TAPPING Call Request packet will contain no called and called DTE address, no facilities data and default call user data of 01000000HNASTAP.

HNAS V2R3M0 New Features

- **OPTIONS=INHIBITBIDREJ** added for the TYPE=MCH|XTP REMOTE definition statement. This option sets **never reject BID** mode.
- **OPTIONS=NORTRBIDREJ** added for the TYPE=MCH|XTP REMOTE definition statement. This option will cause HNAS to **not generate 0814 for a BID reject**.
- **OPTIONS=ONEPIUINB** added for the TYPE=MCH|XTP REMOTE definition statement. This option will **force one PIU per bracket, inbound** (towards application) mode.
- **OPTIONS=REQSESSDELAY=*value*** added for the TYPE=MCH|XTP REMOTE definition statement. The *value* you specify is used to provide a delay between the UNBINDing of an LU and the REQSESS that solicits a subsequent BIND. The delay is needed by some applications to give them time to recycle their LU(s). In the past no delay was enforced which, in some cases, caused the REQSESS PIU to be rejected. You may code a *value* between 0 and 254. If omitted, 2 seconds is assumed.
- **OPTIONS=RESIDSTART=*decimal-start-value*** added for the TYPE=MCH|XTP REMOTE definition statement. This option allow users to **specify the decimal GATE resid (resource id) start value**. 'X'FFFF' =>omitted (use default based on LCN0USED, PVC count).
- **INIT=({ACTIVE|IDLE},DELAYTIME=*minutes*,RETRYLMT=*count*)** added for the TYPE=XOT|XTP LOCAL definition statement.

Where: *minutes* can range from 0 to 62 and *count* can also range from 0 to 62. The INIT= operand values can also be modified using the MLCL.

The DELAYTIME value is used to provide a delay after a TCPIP BIND failure before the BIND is retried. If 0 is specified, no delay is provided. The RETRYLMT value is used to count the number of BIND failures before the LOCAL is taken offline (INIT=IDLE forced). If 0 is specified, the BIND is retried indefinitely. If the INIT= operand is omitted from a LOCAL, INIT=(ACTIVE,DELAY-TIME=5,RETRYLMT=0) is assumed.

Note that when the BIND ends successfully, you will continue to get the following standard alert:

```
NAS2202I SERVER=ipaddr(port) SOCKID=xxxx PCEID=xxxx NAME=rmtname  
NAS2020I SERVER INITIALIZATION COMPLETE
```

If all BINDs fail and a RETRYLMT value is specified, you will get the following new alert:

```
NAS2202W SERVER=ipaddr(port) SOCKID=xxxx PCEID=xxxx NAME=rmtname  
NAS2020W SERVER INITIALIZATION FAILED, LOCAL VARIED OFFLINE
```

In all cases the NAS0001I HOST NAS INITIALIZATION COMPELE, ALL FUNCTIONS00 READY message will be withheld until all LOCALs come active or a failing LOCAL is taken offline. If INIT=IDLE is specified, the LOCAL is offline when HNAS is started and no BIND is attempted.

Additional information on this support is provided in Chapter 5, section Migration - V2R3M0, under Item (29).

- **IDLETO=*minutes*** added for non TYPE=XOT REMOTE definition statement. This support was previously provided as a global value on the BUILD definition statement.
- The HNAS CDF **configuration error summary** has been modified to display the count of default messages issued. In prior releases, both default and warning message counts were combined because the HNAS return code for their respective message severity was the same.
- The HNAS **configuration error processing** has been modified to display a common message ID (**NAS1999W|E|S**) marker whenever a configuration error message is generated that sets RC= 4|8|12. This will allow error messages to be quickly located in the HNAS SYSPRINT log using find or search tools (i.e. find 'NAS1999'). In addition, HNAS will now withhold generating configuration default messages (NAS1xxxD) if any configuration error message has been previously generated. Since HNAS processing will terminate after the CDF scan is completed due to RC>4, no default action can be taken, hence, no reason to generate default messages. The HNAS CDF scan no longer generates a RC-4 return code for (D)efault messages which further reduces completion code viewing requirements.
- The **FASTRUN** process will now propagate VTAM operands that are specified on a TYPE=MXT REMOTE definition statement if that MXT is associated with a SLU in the LUNAME= operand list on a TYPE=SPU REMOTE definition statement. If no MXT is associated with an SLU entry, the VTAM operands are taken from the root TYPE=SPU REMOTE definition statement as in previous releases. In this way, the generated APPL statements for each SLU can have different VTAM operands (for example, MODETAB, DLOGMOD, etc.).
- The **FASTRUN** process will now allow the name for the AMNF VBUILD statement to be specified using the APPLNAME= operand rather than NASNAME= operand on BUILD definition statement.

Note: This new feature was introduced into 230 as Enhancement APAR 2300007.

- The **configuration** process will now accept a pound sign (#) in record column one (1) to identify a comment line that will be logged in the HNAS SYSPRINT file. This will allow you to place block comments in front of definition statements or operands that will be listed in SYSPRINT when the CDF is scanned. Contrast this to standard CDF comment record that starts with an asterisk (*) in record column one which will not be listed in SYSPRINT.
- The new **PFXWTO CONS** start parameter causes HNAS console output to be prefixed using the existing **NASNAME=** value. This enhancement provides a unique message id that can be used for customer automation processes. This processing requires that the **SHOW CONS ON** option must also be in affect. This processing does not affect alarm output.

Note: This new feature was introduced into 230 as Enhancement APAR 2300006.

- The **PFXWTO text** start parameter may be specified to cause the *text* value to be appended to the beginning of each alarm WTO in lieu of the NASNAME= operand value. This same text is used to prefix console output if the PFXWTO CONS option is also in affect. Up to 8 non-blank characters

HNAS V2R3M0 New Features

may be specified.

Specify **PFXWTO** *text* **PFXWTO OFF** **PFXWTO CONS** if you only want console output to be prefixed with the *text* value.

Note: This new feature was introduced into 230 as Enhancement APAR 2300026.

In 220 and earlier versions, **PRNT** is a default state (not a start parameter) that enables global SYSPRINT logging. The state can be altered using the **PRNT ON** or **PRNT OFF** console command.

- **OPTIONS=PFXDCEADDR** added for the TYPE=MCH|XTP REMOTE definition statement. This option specifies that the value coded for the **DCEADDR=** operand is to prefix (appended in front of) the **calling** DTE address in a GATE **outbound** Call Request packet when a **calling** DTE address (or subaddress) is provided by the CTCP. If no **calling** DTE address is supplied by the CTCP (field is null), PFXDCEADDR is treated as REPDCEADDR. TYPE=MCH and GATE=GENERAL must also be specified.

Note: This new feature was introduced into 230 as Enhancement APAR 2300052.

- **OPTIONS=LLC0CTCPCHK** and **OPTIONS=LLC5CTCPCHK** added for the TYPE=MCH REMOTE definition statement. This new option allows control of Call Accept logic for LLC0 and LLC5 sessions using a CTCP (as is done with DATE). When the LLC0 or LLC5 check option is set, inbound LLC0/5 calls will be cleared with a diagnostic of 254 if the first CTCP in the LUNAME= list is not active (the call request is not sent to the CTCP). These options when used in conjunction with a Transpac option will retry the call request using different addresses. This allows sessions to be switched to different MCHs.

Note: This new feature was introduced into 230 as Enhancement APAR 2300060.

- **CONLMT=0** is now allowed on the BUILD definition statement. This new specification allows remote console access to be barred and also suppresses default console configuration messages that are only applicable when remote console access is allowed. Prior to APAR 2300061, if CONPSWD= is omitted, the following configuration warning message is generated:

```
NAS1101W BUILD CONPSWD OMITTED, REMOTE CONSOLE ACCESS WILL  
NOT BE ALLOWED
```

With APAR 2300061 applied, when CONPSWD= omitted but CONLMT=0 coded, this warning message is suppressed. This support was added so that a customer, who may not want remote console access, no longer has to omit the CONPSWD= operand. This new option also provides users with a means of disabling remote console access without having to remove the sample CONSOLE MCH definition in the CDF template that is typically provided by Comm-Pro or their HNAS Business partners in sample CDF definitions. While start parameter RMTCONS OFF will also disable remote console access, some customers will want to disable it in the CDF via CONSLMT=0.

Note: This new feature was introduced into 230 as Enhancement APAR 2300061.

- Bidirectional **Twoway** (Callin/Callout) VC support can now be enabled for PCNE and PAD resources via **SVC0=** or **SVC5=** TYPE=MCH REMOTE definition statements.

The SVC0= and SVC5= operand logic has been modified to accept a T value (for Tway) following the DTE address or IDNUM value so that the same SLU can be used for callin and callout connections. When a T is detected, the ACB for the SLU is opened so that it can be acquired (bound). If an inbound call arrives before the SLU is bound and the calling DTE address matches one of the called DTE addresses associated with the SLU or the CUD field contains an IDNUM value that matches one of *Xidnum* values associated with the SLU (there can be up to 3 DTE addresses and/or *Xidnum* values specified for a twoway SLU), the SLU will be allocated to the inbound VC.

If a BIND arrives prior to an inbound call, the SLU will be processed in the same fashion as an SLU defined with the O (for callout) option. In other words, if the SLU is bound (acquired) before any inbound Call Request is received, the DTE addresses in the list will be used (successively) for outbound call attempts as the called DTE address. Any *Xidnum* values in the list will be skipped.

Note: This logic change was introduced into 230 under APAR 2300083.

- **PVC= *mxtname*** support added to the TYPE=MCH REMOTE definition statement so that user defined packet and window size values can be provided in the HNAS PVC Setup packet when PVC= *rmtname* option is enabled.

Note: This new feature was introduced into 230 as APAR 2300107.

- **SVCi=(*pfxlu*,*sfxst*,*vclmt*)** coding support added to the TYPE=MCH|XTP REMOTE definition statement so that user can control the generation of SLU names based on a prefix value (*pfxlu*), suffix start value (*sfxst*) and SLU count value (*vclmt*).

Currently, the first SVCi= entry is the *vclmt* and the subsequent entries are SLU names. Missing SLU names are replaced by default SLU names which are created using the first 4-characters of the REMOTE name followed by a 0|4|5 for the LLC type followed by the SVCi= index number (001, 002, 003, etc.).

Primary: SVCi=(*vclmt*,*slunm1*,*slunm2*,...)

Alternate: SVCi=(*pfxlu*,*sfxst*,*vclmt*)

The primary and alternate specifications will both be accepted. The alternate specification is appropriate when no additional suboperands need to be associated with each SLU name. For the alternate specification:

The *pfxlu* value must be the first SVCi= suboperand and may be any valid assembler language symbol up to 7-characters in length starting with either an alpha character (A,B,C,...,Z) or an accepted special character (@, #, \$ or %). This suboperand is REQUIRED to indicate that the alternate specification is being used.

The *sfxst* value must be the second SVCi= suboperand and must be a hexadecimal number (without the framing characters X") between 0 and F when the *pfxlu* length is 7, between 0 and FF when *pfxlu* length is 6, ..., between 0 and FFFFFFFF when the *pfxlu* length is 1. If *sfxst* is omitted (,), a default value of 0 will be used.

The *vclmt* value must be the third SVCi= suboperand and must be a decimal number between 0 and 511 (the SVCi= array size). This suboperand is REQUIRED. The SLU names that HNAS generates from the *pfxlu*, *sfxst* and *vclmt* values will always be 8-characters in length with zero (0) pad characters added as required between the last *pfxlu* character and the suffix value.

HNAS V2R3M0 New Features

Examples:

If SVC0=(TT,1,3) is specified, the generated SLU names would be TT000001, TT000002, TT000003.

If SVC0=(TTTTT,0,3) is specified, the generated SLU names would be TTTTT000, TTTTT001, TTTTT002.

Note: This new feature was introduced into 230 as APAR 2300138.

- **TRAN=NPSIEVEN|NPSIODD|NPSIMARK|NPSISPACE** values are now accepted in order to allow standard NPSI translate tables to be specified in addition to standard HNAS translate tables. Prior to APAR 2300152, NPSI translate tables could only be included in HNAS by renaming the MCHTTBL1 member in the HNASOBJX library to MCHTTBLS then re-linking HNAS. This allowed the standard NPSI translate tables to be used in lieu of the standard HNAS translate tables using the existing TRAN= operand values of EVEN|ODD|MARK|SPACE. The new TRAN= values now permit NPSI as well as HNAS translate tables to be coresident and CDF configurable. NPSI tables are more appropriate and are even required for some host applications like TPE.

Note: This new feature was introduced into 230 as APAR 2300152.

- Allow **RTEIN=** processing to be controlled by source calling address (**=>S**) in addition to the existing target called DTE address (**=>T**) in a manner similar to that used for the RTEOUT= operand. Please refer to remote type LOCAL RTEIN= parameter for additional information.

Note: This new feature was introduced into 230 as APAR 2300176.

- **OPTIONS=BALANCERTEIN** enhancement to RTEIN= processing allows inbound calls to be distributed across multiple MCHs using a round robin session connect balancing scheme. Please refer to remote type LOCAL OPTIONS=BALANCERTEIN and RTEIN= operand for additional information.

Note: This new feature was introduced into 230 as APAR 2300176.

Alert/Alarm Message and Clear/Reset Code Enhancements:

- **Message filtering** enhancement.

Informational messages can now be filtered from the HNAS SYSPRINT log file (SYSOUT) in addition to the system console log. In conjunction with the SHOWERR start parameter (or SHOW ERR console command), you may now specify the following options to prevent certain informational messages from being written to the HNAS SYSPRINT log file.

<u>Message ID</u>	<u>Component</u>	<u>Start Parameter</u>	<u>Console Command</u>
NAS0xxxI	System (General)	PRNTSYS OFF	PRNT SYS OFF
NAS1xxxI	Configuration	PRTNCNFG OFF	PRNT CNFG OFF
NAS2xxxI	TCP/IP	PRNTTCP OFF	PRNT TCP OFF

NAS3xxxI	VTAM	PRTNVTAM OFF	PRNT VTAM OFF
NAS4xxxI	LU	PRTNLU OFF	PRNT LU OFF
NAS5xxxI	VC (Common)	PRTNVC OFF	PRNT VC OFF
NAS6xxxI	XTP	PRTNXTP OFF	PRNT XTP OFF
NAS7xxxI	XOT	PRTNXOT OFF	PRNT XOT OFF
NAS8xxxI	VC (QLLC)	PRTNVC OFF	PRNT VC OFF
NAS9xxxI	System (Auth)	PRTNSYS OFF	PRNT SYS OFF

- **Julian Date (nnn)** added to **Alert** Message prefix for improved identification of the day the event occurred.
- **NAS2021W Alert** Message was added to improve notification regarding server notification. The message will be generated when HNAS cannot connect to the stack after the retries that are specified in the REMOTE INIT operand have been reached. See section Configuration Enhancements, **INIT=({ACTIVE|IDLE})** description in this document for additional information.
- **NAS2601E** and **NAS2602I Alert** Messages were added to improve XOT socket availability status notification. These messages extend HNAS awareness of socket availability beyond the stack to the router. The socket slowdown messages are as follows:

NAS2601E SOCKET POOL DEPLETED, SOCCNT=dddd SOCLMT=dddd ',

This message is issued when the number of sockets in use (SOCCNT) reaches 80% of the maximum number of sockets for a server (SOCLMT)

NAS2602I SOCKET POOL RESTORED, SOCCNT=dddd SOCLMT=dddd ',

This message is issued when the number of sockets in use (SOCCNT) returns to 60% of the maximum number of sockets for a server (SOCLMT)

- **NAS8000I Alert** Message was modified to improve QLLC call setup status notification.

```
NAS8000I CLIENT=iii.iii.iii.iii(port) SOCKID=sockid
          PCEID=pceid NAME=rmtname
NAS8000I VC vcb STARTING SESSION ON MCH mchname (count)
NAS8000I direction CLGADDR=lddd...ddd CLDADDR=lddd...ddd
```

This message is issued when a QLLC session starts for the VC associated with the MCH named **mchname**. The VCB address **vcb** and active VC connect counter **count** are also displayed. The **count** value identifies the number of virtual circuits connected to the denoted MCH. The **direction** value identifies the call as being INBOUND to HNAS or OUTBOUND from HNAS. The Calling DTE address (CLGADDR) and Called DTE address (CDLADDR) are also displayed. For both, **lddd...ddd** is the decimal DTE address (**ddd...ddd**) prefixed with the hexadecimal single digit address length (**l**).

Note: This enhancement was introduced into 230 as Enhancement APAR 2300039.

HNAS V2R3M0 New Features

- **NAS7718T, NAS7719T and NAS7798T Special Trace Alert** Message added to provide formatted content of inbound and outbound call request packets. The special **NASinnnt** trace entry alert message types are only written to sysprint not SYSCONS (system operator console).

Note: This new feature was introduced into 230 as Enhancement APAR 2300046.

- **NAS3703W (NOTIFY) and NAS3704W (CLEANUP) VTAM Alert Messages** were added to improve diagnostic capabilities of REQSESS related error conditions by providing sense code information. Please refer to the VTAM alert messages section for a description and layout of these messages.

Note: This new feature was introduced into 230 as Enhancement APAR 2300054.

- **NAS7715W (CALL REQ TO MCH FAILED)** alert message now contains the extended diagnostic event reason codes (DIAGX=xxxx) values to improve diagnostic capabilities. This message wasn't included in the group of alert messages that were provided with this support when HNAS 230 was initially released.

Note: This new feature was introduced into 230 as Enhancement APAR 2300053.

- **NAS7713W SECOND CLEAR FROM ip-addr(port) ON rmt-name MCH mch-name LU lu-name** alert message added improving event reporting and problem determination.

Note: This new feature was introduced into 230 as Enhancement APAR 2300085.

- **Clear Request diagnostic codes** were added to improve event reporting and fault isolation:

Clear diagnostic code **221 x'DD'** - Timeout. CTCP did not UNBIND after receiving clear

Clear diagnostic code **222 x'DE'** - Second clear received from router for VC

Note: These new diagnostic codes introduced into 230 under APAR 2300085.

- **NAS7797W CALL FROM ip-addr. PCE HAS VC AT vc-addr LCST=st LU=lu-name LU ADDR=lu-addr** alert message added indicating that HNAS may have failed to properly terminate an XOT TCP/IP session with a router. Please contact customer support if this alarm is received.

Note: This new feature was introduced into 230 as Enhancement APAR 2300092.

- **NAS3797I LU sluname RECEIVED BIND FROM PLU pluname PVC** messages added so that users could easily identify when an LU with a PVC session receives a BIND from the PLU. Because PVCs are always active, HNAS tries to establish a session with the PLU using a REQSESS macro (asks PLU for a BIND) issued by timer logic. There can be many hours between the NAS3798I session starting message and the time when the PLU sends a BIND to start the session.

Note: This new feature was introduced into 230 as Enhancement APAR 2300118.

- **NAS4706W** LU sluname REJECTING BIND FROM PLU pluname SENSE=xxxx message was added for all LLC-n types so that users could easily identify the cause of a BIND failure. This message is issued when HNAS rejects a bind command sent by the PLU. The sense code indicating the cause of the reject.

Note: This new feature was introduced into 230 as Enhancement APAR 2300118.

- **Clear Request diagnostic code 211 x'D3'** was added to mirror NPSI's handling of a SIGNAL PIU sequence. When HNAS receives a SIGNAL PIU a +RSP will be returned to the PLU. For LLC5 sessions an IOB (indication of break) Q packet will be sent to the remote. For other LLC types SIGNAL does not cause X25 side activity. This processing conforms to NPSI operation.

Note: This new diagnostic code was introduced into 230 under APAR 2300140.

- **NAS3705W** LU *lu-name* REJECTING message added indicating that PIU is being rejected. Up until now there was no alert message issued to denote when the PIU is rejected so it is difficult to determine the reason for the session end.

Note: This new feature was introduced into 230 as Enhancement APAR 2300144.

- **NAS7774W** PVCSETUP FAILED - REMOTE rmt-nm CLOSED TCP SESSION... message added indicating that the remote responded to a HNAS PVC setup packet with a socket close (FIN).

Note: This new feature was introduced into 230 as Enhancement APAR 2300151.

- **NAS7718T, NAS7719T and NAS7798T Special Trace Alert** Message added to provide formatted content of inbound and outbound PVC Setup packets. The special **NASinnnt** trace entry alert message types are only written to sysprint not SYSCONS (system operator console).

Note: This new feature was introduced into 230 as Enhancement APAR 2300151.

- HNAS **PVC Setup Status Codes** were reassigned improving router setup retry processing. Additional information is available in Chapter 5 - Migration, under the 230 section concerning this topic.

Note: This new feature was introduced into 230 as Enhancement APAR 2300151.

- **NAS7795T Special Trace Alert** Message added providing an alert message trace record for all inbound and outbound clear request packets. Logging for these new message types is enabled using the new TRCMCH ICLR|OCLR option. The special **NASinnnt** trace entry alert message types are only written to sysprint not SYSCONS (system operator console).

Note: This new feature was introduced into 230 as Enhancement APAR 23000153.

- **NAS251xM Monitor Alert** Messages added providing information on TAP Keep Alive activity for REMOTE TYPE=XTP|XOT) resource. Console command **MON TAP {ON|OFF|ALLON|ALLOFF}** or Start Parameter **PARM=MONTAP** enable generation of these message types. TAP status log entries are written to SYSPRINT only.

HNAS V2R3M0 New Features

Note: This new feature was introduced into 230 as Enhancement APAR 2300155.

- **NAS4710W** lu-nm LU lu-addr SENDING DIAG PKT *text* BFR NEXT trace record now provided as an alert message. Message is generated when HNAS sends a diagnostic packet to the CTCP. BFR NEXT indicates the buffer that triggered the error is displayed in SYSPRINT while the error indication **text** values are shown below:

INVALID CTCP GATE CMD BYTE	- HNAS has received an invalid GATE message from a GATE CTCP control session PLU.
FMD INV ON FC GATE CTL SES	- HNAS has received an FMD PIU from a GATE FC PLU. FC GATE control session do not send FMD PIUs.
INVALID CTCP DATA SES CMD BYTE	- HNAS has received an invalid GATE message from a GATE CTCP data session PLU.
GT FC LN ER	- HNAS has received a PIU from a GATE FC PLU that is too short to be valid.
GT DS LN ER	- HNAS has received a PIU from a GATE PLU that is too short to be valid.

The most likely cause of NAS4710W is that the PLU is not a GATE CTCP.

Note: This new feature was introduced into 230 as Enhancement APAR 2300160.

- **NAS3705I** LU lu-name REJECTING cmd #seq SENSE=bbbbbbbb BID alert message is now issued when HNAS rejects a PIU from the PLU SENSE=0813xxxx or 0814xxxx. These values indicate that the PLU and the HNAS SLU are attempting to start a bracket at the same time. These rejects can be considered normal so the severity code is 'I' (follows 'NAS3705').

OPTIONS=(INHIBITBIDREJ,NORTRBIDREJ) may be coded to reduce the number of NAS3705I alerts with SENSE=0813/0814.

Note: This new feature was introduced into 230 as Enhancement APAR 2300177.

- **NAS4707W** sluname GENERATING ERR/INFO PACKET FOR CTCP alert message is issued when HNAS receives an invalid packet on a GATE control or data session LU. In addition to the alert message an ERROR/INFORMATION REPORT message (not a DIAGNOSTIC message) will be sent to the CTCP on the control session LU.

Note: This new feature was introduced into 230 as Enhancement APAR 2300178.

- **NAS4708W** GATE FC CTL SES LU luname CLEARED BY CTCP alert message is issued when HNAS receives a CLEAR packet on an FC control session LU a NAS4708W alert message will be issued and a CLEAR-CONFIRM will be returned to the CTCP. Data session LUs will not be affected.

Note: This new feature was introduced into 230 as Enhancement APAR 2300178.

- **NAS4709W** REMOTE rmt-nm lu-nm LU lu-addr LUIQ TIMEOUT, LUIQ BFR CT=xxxx message is issued when HNAS has a message from the remote that has been waiting for delivery to the PLU for a period of 4 minutes. The alert indicates that there is something wrong with the VTAM session between the HNAS SLU and the PLU. After the ALERT the LU's VTAM session is ended (PLU will receive a NOTIFY). If there is an associated VC session it is cleared with DIAG=212.

Clear Request diagnostic code 212 x'D4' was added to report this condition.

Note: This new feature was introduced into 230 as Enhancement APAR 2300180.

- **NAS3797I** alert message is now generated when a GATE control session is bound. Previously this alert was only used for BINDs received for PVC LUs. This will eliminate the requirement to view the HNAS DLU command display output to confirm that the GATE control session is active.

```
NAS3797I LU lu-nm RECEIVED BIND FROM PLU plu-nm
```

Note: This logic change was introduced into 230 under APAR 2300201.

Console and Trace Enhancements:

- **Error reporting enhancement** provides more targeted information when a command is rejected due to a syntax, specification or consistency error.
- A **resource name** may now be specified on the right side of a command without a command modifier like RNM=, LNM= or LUNM=. The console command parser will associate the specified name with the appropriate command modifier based on the type of resource specified. This enhancement is provided to further simplify console command syntax.
- The **DMAP APAR** command has been modified so that it automatically executes at initialization time *with no delays*. The output of the command is logged in the HNAS SYSPRINT so that maintenance can be viewed using an SDSF panel.

Additionally, during the initialization pass, the DMAP APAR command creates a table that is sorted in APAR ID order so that it can be displayed using the new **DNAS APAR** command. Note that you can still use the DMAP APAR command to display APARs but command output is in module name order rather than APAR ID order.

Note: This new feature was introduced into 230 as APAR 2300004 and 220 as APAR 2200079.

- The **DNAS command** has been modified to display the APAR=*apar-id* of the latest maintenance applied by the customer in addition to the *apar-id* that was incorporated when the HNAS distribution was created. The AUTH=*month-count* value was also added to the DNAS display output to provide the number of months established for the trial period.

Additionally, a new argument (APAR) has been added to the DNAS command so that a table, created at initialization time by the **DMAP APAR** command, can be displayed. The **DNAS APAR** command will display all APARs on the HNAS system in APAR ID order while the DMAP APAR command displays all maintenance in module name order.

HNAS V2R3M0 New Features

Note: This new feature was introduced into 230 as APAR 2300004 and 220 as APAR 2200079.

- The **TRCLU command** will now start or stop tracing for all SLUs associated with a TYPE=SPU REMOTE definition statement that is identified by the RNM= command modifier. This is an extension to the logic that exists in V2R2M0 for TYPE=XTP|MCH REMOTE definition statements.
- The **DLU command** will now display all SLUs associated with a TYPE=SPU REMOTE definition statement that is identified by the RNM= command modifier. This is an extension to the logic that exists in V2R2M0 for TYPE=XTP|MCH REMOTE definition statements.
- The **DPCE command** will now display the PCE for a TYPE=SPU REMOTE definition statement that is identified by the RNM= command modifier. This is an extension to the logic that exists in V2R2M0 for TYPE=XTP|XOT|MCH REMOTE definition statements.
- The **DPCE, TRCDISP, TRCIO, TRCBFR, TRCDATA and STATS commands** have been modified to accept **CONS|TCP|TMR|UTIL|XOT|XTP** as a command argument so that only PCE resources of the specified type will be operated on.
- The **DLU, DVC and DPCE commands** have been modified to accept **TRACE=YES|NO** as a command argument so that only resources that *are or are not* being traced will be displayed. This is useful for finding all resources that are actively being traced.
- The HNAS **PING command** has been added to the console subsystem that will allow the console operator to check on the existence of a router on the IP network as well as a specific MCH on the router. The HNAS PING command is not a traditional IP PING command but rather an **X.25 PING**. The HNAS PING command requires an IP address to identify a router and an X.25 DTE address to identify the MCH link. These two pieces of information allow HNAS to establish an IP connection to the router and then an X.25 (XOT) connection via the MCH link. An indication of success or failure for the PING request is reported back to the console operator.

Note: Please refer to the Enhancements - Expanded Content heading at the end of this section for additional information regarding this new or enhanced feature.

- The **MRMT command** was enhanced to allow users to modify QLLC (LLC3) SLU definitions in the LUNAME= operand of TYPE=SPU REMOTE definition statements via privileged local (SYSCONS) or remote console access.

Note: Please refer to the Enhancements - Expanded Content heading at the end of this section for additional information regarding this new or enhanced feature.

- The **MRMT command** has been modified to permit updating of the INIT= operand of a TYPE=XOT|XTP REMOTE definition statement. Enter:

```
RNM=rmtname MRMT INIT={ACTIVE|IDLE}
```

If the ACTIVE|IDLE value is changed, a V RMT ON|OFF command is propagated.

- The **MLCL command** has been modified to permit updating of the INIT= operand of a TYPE=XOT|XTP LOCAL definition statement. Enter:

```
LNM=lclname MLCL INIT={ACTIVE|IDLE},DELAYTIME=minutes,RETRYLMT=count
```

Where: *minutes* can range from 0 to 62 and *count* can also range from 0 to 62.

Note: For the console command, the INIT= argument does not use opening and closing parens.

If the ACTIVE|IDLE value is changed, a V LCL ON|OFF command is propagated.

- The **VARY command** has been modified to allow a LOCAL to be varied offline or online just like a REMOTE. The VARY command now has the following format:

```
LNM=lclname V LCL {ON|OFF} or RNM=rmtname [RMT] {ON|OFF}
```

If you vary a LOCAL (server) offline when it was previously online, subsequent calls to its HOME IP address will be cleared by the router because the server will not be LISTENing on its IP address. If you vary the LOCAL back online, subsequent calls will then be accepted. This is a nice feature for quiescing REMOTE access to HNAS without stopping it. Active REMOTE connections will remain active until terminated by either end.

- The **DTRC command** has been modified to display the WRAPCNT=*wrap-cnt* counter in the trace table header. The wrap count identifies the number of time the trace table has wrapped.
- The **DLU, DMCH and DPCE commands** now provide support for the **SHOWNAME|SHOWADDR** parmlist options while the **DVC command** now provides support for the **SHOWNAME|SHOW-ADDR|SHOWIPAD|SHOWCGAD|SHOWCDAD|SHOWVCST|SHOWLUST** parmlist options. These new **SHOWtype** arguments provide for improved console command usability and resource status.
- The **TRCLU command** now produces trace entries providing keywords (BIND, SDT, UNBIND, etc.) improving search capabilities.

Note: This new feature was introduced into 230 as Enhancement APAR 2300010.

- The **DMCH command** now accepts **FMT3** as an argument which will cause the **DLU** command to be executed after the DMCH command completes. This will provide a display of the selected MCHs and all LUs that are associated with those MCHs. This command is currently provided under Local console operation. Remote console support will be included in HNAS 231.

Note: This new feature was introduced into 230 as Enhancement APAR 2300016.

HNAS V2R3M0 New Features

- The **PFXWTO command** now accepts *name* as an argument which will allow alarm messages and console output to be prefixed with a user defined name other than the current NASNAME= option. This new feature can also be enabled as a parm list option (PARM='PFXWTO *name*').

Note: This new feature was introduced into 230 as Enhancement APAR 2300026.

- The **TRCMCH ICR and TRCMCH OCR commands** (and **CONCMDQ=**) trace logic was enhanced to provide improved formatting of inbound and outbound call request packet content. The information is provided in special NAS7718T, NAS7719T and NAS7798T alert trace messages that are written to sysprint only.

Note: This new feature was introduced into 230 as Enhancement APAR 2300046.

- The **MLCL command** has been modified to permit an insert/delete index to be specified for the RTEIN= and RTEOUT= operands.

Prior to enhancement APAR 2300056, if a new entry was to be added to the RTEIN= or RTEOUT= operand, it could only be added to the end of each operand's list. If the associated DTE address is a superset of a previous entry, the new entry would never be used. For example, assume the following RTEIN= operand list:

```
RTEIN= (ABCD/47113456 ,
        EFGH/23456 ,
        IJKL/345689)
```

When `MLCL RTEIN=XYZA/2345678` is entered, the RTEIN= operand list will look as follows:

```
RTEIN= (ABCD/47113456 ,
        EFGH/23456 ,
        IJKL/345689 ,
        XYZA/2345678)
```

Because DTE address 23456 occurs earlier in the RTEIN= list (for MCH EFGH), the new entry for MCH XYZA will not be accessed.

To correct this anomaly, the MLCL command has been modified so that a new entry can be inserted at a specific position in the RTEIN= and RTEOUT= operand lists. In this way the new entry can be tested before the old entry that has a subset of the new DTE address. Using the example above, to insert XYZA/2345678 as the second entry in the RTEIN= operand list, prefix the MCH name (XYZA) with a decimal entry number value as follows:

```
MLCL RTEIN=2,XYZA/2345678      <- comma before XYZA is optional
```

This command changes the RTEIN= operand list to look as follows:

```
RTEIN= (ABCD/47113456 ,
        XYZA/2345678 ,
        EFGH/23456 ,
        IJKL/345689)
```

Note also that logic has been added that allows you to delete an entire entry, not just its DTE address. To remove the entry for MCH EFGH in the RTEIN= list above, enter the following com-

mand:

```
MLCL RTEIN=3,*          <- comma before * is optional
```

This command changes the RTEIN= operand list to look as follows:

```
RTEIN=(ABCD/47113456,
        XYZA/2345678,
        IJKL/345689)
```

Note: The RTEIN/RTEOUT insert/delete index logic was introduced in 230 as Enhancement APAR 2300056.

- The **DNAS command** has been modified to display a list of missing *apar-ids* if the 'latest' *apar-id* is different than the *apar-id* that was incorporated when the HNAS distribution was created. For example, if you were shipped HNAS with APAR 2300059 incorporated and you then installed APAR 2300062 separately, the following would be displayed if you did not also install the 'pre-reqs' for APAR 2300062:

```
HNAS --> VERSION=V2R3M0 HOST=OS390|ZOS ASMDATE=08/02/04
          CREATED AT 14:11:31 ON 07/23/2004
          CREATED WITH MAINTENANCE THROUGH APAR 2300059
          MOST RECENT MAINTENANCE APPLIED IS APAR 2300062
          SHIPID=0000000000099999 AUTH=00
          APARID  MODULE      (MISSING MAINTENANCE)
          2300060 MISSING
          2300061 MISSING
```

If there is no missing maintenance, that is, if there are no holes in the *apar-id* values from the shipped *apar-id* value to the latest *apar-id* value (all pre-reqs were also installed), the following is displayed:

```
HNAS --> VERSION=V2R3M0 HOST=OS390|ZOS ASMDATE=08/02/04
          CREATED AT 14:11:31 ON 07/23/2004
          CREATED WITH MAINTENANCE THROUGH APAR 2300059
          MOST RECENT MAINTENANCE APPLIED IS APAR 2300062
          SHIPID=0000000000099999 AUTH=00
          APARID  MODULE      (MISSING MAINTENANCE)
          NOTHING MISSING
```

Note: The missing *apar-id* logic is only executed when DNAS is entered with no arguments.

Additionally, the DNAS APAR command now accepts an *apar-id* value to restrict the displayed applied maintenance to the specified *apar-id only*. For example: **DMAP APAR 2300062** will produce the following display:

```
HNAS --> VERSION=V2R3M0 HOST=OS390|ZOS ASMDATE=08/02/04
          CREATED AT 14:11:31 ON 07/23/2004
          CREATED WITH MAINTENANCE THROUGH APAR 2300059
          MOST RECENT MAINTENANCE APPLIED IS APAR 2300062
          SHIPID=0000000000099999 AUTH=00
          APARID  MODULE      (APPLIED MAINTENANCE)
          2300062 CONSDNAS
```

HNAS V2R3M0 New Features

CONSHelp
NASCONS

Note: This new feature was introduced into 230 as APAR 2300062.

- The **DNAS command** now accepts **USERMODS** as an argument which will cause the command to list any custom-user-mods applied to the system. The list includes the Problem Number or unique Customer Fix ID as well as the modules affected by the custom enhancement. Prior to this new display users had to search through the DNAS APAR or DMAP APAR display output for usermods intermingled in with standard APARs. Please refer to the console command description for additional information.

Note: This new feature was introduced into 230 as APAR 2300065.

- The **DNAS APAR** and **DMAP APAR commands** now optionally display APARID's HNASMACX and/or HNASOBJX when CustomUserMods are included in the HNAS product distribution.

The HNAS distribution process has been modified to generate an 'APARID' of HNASMACX and/or HNASOBJX if a custom MACLIB and/or OBJLIB has been included in the distribution. These APARIDs are contained in the NASEND module and will be displayed in the HNAS SYSPRINT when the DMAP APAR console command is executed when HNAS is started. These APARIDs will also display whenever the DNAS APAR console command is executed. These APARIDs are intended to provide an additional reminder that custom modifications are on the HNAS system and, hence, caution should be exercised when applying maintenance. Please refer to the DNAS console command description for additional information.

Note that the majority of customers don't have custom enhancements (CustomUserMods) although this enhancement APAR was developed to improve their support services.

Note: This new feature was introduced into 230 as APAR 2300069.

- The **MRMT command** processor has been modified to allow PCNE/GATE/PAD SLU names to be added, deleted or changed for the SVC0=/SVC4=/SVC5= operands on TYPE=MCH|XTP REMOTE's in a manner similar to what is now done for the LUNAME= operand on a TYPE=SPU REMOTE.

Note: This logic change was introduced into 230 under APAR 2300084.

- The **SHOW MORE command** (and **SHOWMORE start parameter**) were added providing expanded TCP/IP stack event debugging.

Note: This logic change was introduced into 230 under APAR 2300088.

- The **TRCALL command** was modified to remove the PCE trace function for the ON|OFF arguments. The processing for the ON and OFF arguments is now treated identically to the STRT and STOP arguments, respectively. The PCE trace function is now provided by the new **TRCPCE** command.

Note: This logic change was introduced into 230 under APAR 2300110.

- The **DNAS command** has been modified to display the distribution type (SMP/E or NON-SMP) on the header line. In addition, up to 2 new lines can be displayed to show the name of an optional custom MACLIB and/or OBJECT library that was used to create a non-SMP distribution.

```
HNAS --> VERSION=V2R3M0 HOST=OS390 | ZOS ASMDATE=02/24/05 DIST=NON-SMP
          CUSTMAC=COMM1.TEST.HNASMAC
          CUSTOBJ=COMM1.TEST.HNASOBJ
          CREATED AT 14:11:31 ON 02/26/2004
          CREATED WITH MAINTENANCE THROUGH APAR 2300000
          MOST RECENT MAINTENANCE APPLIED IS APAR 2300113
          SHIPID=9999999999999999 AUTH=00
          APARID   MODULE   (APPLIED MAINTENANCE)
          2300113  CONSDPRM
                   CONSLNCT
```

Note: This logic change was introduced into 230 under APAR 2300113.

- The **VARY command** has been modified to allow specific sockets or ranges of sockets to be varied offline or online. Please refer to the Console Subsystem guide for additional information and command syntax.

Note: This logic change was introduced into 230 under APAR 2300123.

- The **TRCMCH ICR** and **TRCMCH OCR command** trace logic was enhanced to provide improved formatting of inbound and outbound PVC setup packet content. The information is provided in special NAS7718T, NAS7719T and NAS7798T alert trace messages that are written to sysprint only.

Note: This new feature was introduced into 230 as Enhancement APAR 2300151.

- The **DVC command** has been modified to show whether or not a PVC setup exchange has occurred (display field VCOPT **D**=setup not complete, **I**=pvc received reset link/device inoperative and **N**=setup completed. Additional information is available in the DVC section under VCOPT in the HNAS Console subsystem guide.

Note: This new feature was introduced into 230 as Enhancement APAR 2300151.

- The **MRMT command** has been modified to permit updating of the TRAN= operand of a TYPE=MCH|XTP REMOTE definition statement. Enter:

```
RNM=rmtname MRMT TRAN={NO | USER | EVEN | ODD | MARK | SPACE |
                        NPSIEVEN | NPSIODD | NPSIMARK | NPSISPACE }
```

Note: This new feature was introduced into 230 as Enhancement APAR 2300152.

- The **TRCMCH ICLR** and **TRCMCH OCLR commands** were implemented to produce trace messages for all inbound and outbound clear request packet activity. The information is provided in the NAS7795T alert message (written to SYSPRINT only).

HNAS V2R3M0 New Features

Note: This new feature was introduced into 230 as Enhancement APAR 2300153.

- The **MON[ITOR] command** has been modified to accept **TAP {ON|OFF|ALLON|ALLOFF}** as arguments for the purpose of start and stopping *local* (individual REMOTE TYPE=XTP|XOT) or *global* (all REMOTE TYPE=XTP|XOT) TAP (Keep Alive) monitoring. TAP status log entries are written to SYSPRINT only and are identified by alert message IDs of the form NAS251xM.

Note: This new feature was introduced into 230 as Enhancement APAR 2300155.

- The **VARY command** has been modified to accept the new FORCE option which allows active remote client sockets to be closed when a LOCAL or REMOTE is varied offline. When FORCE is not specified with OFF (or INACT), the LOCAL or REMOTE is marked offline but any active client sockets are left active. They will be closed at end of session and will then remain closed until the LOCAL or REMOTE is varied back online. Please refer to the Console Subsystem guide for additional information and command syntax.

Note: This logic change was introduced into 230 under APAR 2300156.

- The **DPARM command** has been modified to accept the new MODIFIERS option which restricts the DPARM display to command modifiers only. This provides symmetry with DPARM EXEC command which restricts the DPARM display to HNAS start parameters only. DPARM ALL (or DPARM with no arguments) will continue to display both command modifiers and start parameters. In addition, when modifiers are displayed, their accepted values will also be displayed.

Note: This logic change was introduced into 230 under APAR 2300161.

- The **HELP command** text has been moved from the CONSHelp module to each command processor module so that when a change is made for a command, the help text can also be changed in the same module without affecting CONSHelp.

Note: This logic change was introduced into 230 under APAR 2300161.

- The **DPARM EXEC** command has been modified to display ALLON|ALLOFF instead of ON|OFF for TRCBFR, TRCDATA, TRCDISP, TRCIO, TRCLU, TRCMCH, TRCMCHX and TRCVC to more properly describe *global* trace states

- The **PRNT QLLC command** (and start parameter **PRNTQLLC**) were added so that NAS8xxxx QLLC alert messages could be filtered independently from PRNTVC NAS5xxxx and NASAxxxx VC alarm messages.

Note: This logic change was introduced into 230 under APAR 2300167.

- The **DNAS command** has been modified to display the Host OS and Version information and the execution start date and time. Please refer to the Chapter 5 migration section for a complete description of the display output changes.

Note: This logic change was introduced into 230 under APAR 2300168.

QLLC Enhancements:

• QLLC Callout via TYPE=SPU REMOTE

HNAS will initiate an SVC connection to an SPU that is defined by a TYPE=SPU REMOTE definition statement when OPTIONS=CLOTINITYP=BIND|TIMER|CONSOLE is specified. The DTEADDR=, DCEADDR=, FAC= and CUD= operands on the same TYPE=SPU REMOTE definition statement provide, respectively, the called DTE address, calling DTE address, facilities and call user data for the outbound Call Request packet.

For OPTIONS=CLOTINITYP=BIND, an Outbound QLLC call is initiated when any SLU defined in the LUNAME= operand for the SPU is bound (acquired).

For OPTIONS=CLOTINITYP=TIMER, an Outbound QLLC call is initiated automatically when the OPTIONS=MCHTMR=*value* expires for the *first* defined MCH.

For OPTIONS=CLOTINITYP=CONSOLE, an Outbound QLLC call is initiated when the VARY ACT console command is entered for the named SPU.

For all CLOTINITYP values, an established call remains active until cleared by the remote SPU.

OPTIONS=CLOTFAILRTYLMT=*value* may also be specified to limit the number of callout retries in the event of a call failure. A default *value* of 3 is assumed if omitted. If a zero *value* is specified, failed calls will be retried indefinitely until a connection is established.

OPTIONS=CLOTCONLMT=*value* may also be specified to limit the number of callout attempts after a successful connection has been cleared. A default *value* of 0 is assumed if omitted. If a zero *value* is specified or is set by default, HNAS will attempt to reestablish the connection to the SPU after a forced delay based on the CLOTINITYP setting. This operand is provided to limit the use of X.25 network facilities for an SPU. When the number of call attempts reaches the CLOTCONLMT *value*, a subsequent call can only be initiated via the VARY ACT console command.

• QLLC Callout via TYPE=MCH REMOTE

HNAS will initiate an SVC connection to all SPUs that are defined in an SVC3= operand on a TYPE=MCH REMOTE definition statement if they are associated with a callout DTE address (O specified as DTE address delimiter). If the HNAS call completes normally, the connection remains in affect until cleared by the remote SPU. Following a clear, HNAS will attempt to reestablish the connection to the SPU after a forced delay.

Outbound QLLC calls are initiated automatically when the OPTIONS=MCHTMR=*value* expires for

HNAS V2R3M0 New Features

an MCH. Initiation of outbound QLLC calls will continue until a Call Accept packet is received. An accepted call remains active until cleared by the remote SPU.

- **QLLC Callout Summary:**

- **QLLC Callout via Application Bind**

HNAS will initiate a QLLC SVC connection for an SPU when an Application Bind is received on *any* one of the SLUs identified in the SPU's LUNAME= operand if, and only if, OPTIONS=CLOTINITYP=BIND is specified for the SPU. PLU BINDs that are received before the SPU is connected are queued for delivery after the SPU and its SLUs come active. After the SPU connection is established, an XID request/response exchange is performed followed by an ACTPU and ACTLU for all defined SLUs in the LUNAME= operand. Any BINDs that have been queued for transfer are then passed to the target SLUs. In the case of a queued BIND, no SLU input and hence no REQSESS is required to establish a SLU/PLU session. The queued BIND already does this. If the SPU is already connected when a BIND is received, the BIND is treated in the normal fashion and is immediately passed to the remote SLU.

- **QLLC Callout via Timer Control**

HNAS can initiate a QLLC SVC connection based on a timer. There are two (2) methods for timer initiated callout.

1) If OPTIONS=CLOTINITYP=TIMER is specified for an SPU, a callout is initiated when the MCHTMR= value expires for the *first* MCH defined in the CDF. This MCH does not have to be associated with any SPU. It is used solely to provide an event that initiates SPU callout processing.

2) If an SPU is identified in the SVC3= operand for an MCH and the SVC3= entry has a callout DTE address (identified by the character 'O' following the DTE address), HNAS will initiate a connection for the SPU when the MCHTMR= value expires for *this* MCH.

Note: OPTIONS=CLOTINITYP=TIMER callout initiation takes precedence over SVC3= callout initiation. That is, HNAS will look for all SPU with OPTIONS=CLOTINITYP=TIMER before it looks for any identified in an SVC3= operand.

- **QLLC Callout via HNAS Console Control**

HNAS will initiate an QLLC SVC connection for an SPU when the following HNAS Console Command is executed if OPTIONS=CLOTINITYP=CONSOLE|TIMER|BIND is specified:

RNM=*spuname* VARY ACT

In the case of OPTIONS=CLOTINITYP=CONSOLE, this command is the *only* way to initiate an SPU callout.

In the case of OPTIONS=CLOTINITYP=TIMER|BIND, this command is the *only* way to initiate an SPU callout after the call connection limit (OPTIONS=CLOTCONLMT=*value*) has been reached.

- **QLLC TYPE=SPU LUNAME= SLUname MXTname association (applid and mxtname support)**

(LUNAME= operand enhancement for QLLC SPUs (SLU *applid* and *mxtname* support)

An APPLNAME= index and MXT name may now be associated with the QLLC SLUs identified in the LUNAME= operand on a TYPE=SPU REMOTE definition statement. The ***applid*** and ***mxtname*** for LLC3 SLUs provide the same function as they do for LLC0 and LLC5 SLUs (via the SVC0= and SVC5= operands). The ***applid*** allows an SLU to be 'wired' (dedicated) to a specific host application. The ***mxtname*** is used to override operands that are specified on the root REMOTE statement and, for the FASTRUN process, to supply overriding VTAM parameters that are propagated to the generated APPL statement for the SLU in the HNAS AMNF. The LUNAME operand for a TYPE=SPU REMOTE definition statement now has the following form:

LUNAME=(...,*sluname* / *rpccnt* / *spccnt* / *applid* / *mxtname*,...)

- sluname*** is the SLU name (no change).
- rpccnt*** is a receive pacing count value (no change).
- spccnt*** is a send pacing count value (no change).
- applid*** is an APPLNAME= operand index value (new).
- mxtname*** is a TYPE=MXT REMOTE name (new).

Note that the MRMT console command process has been modified to allow all suboperands of the LUNAME= operand on a TYPE= REMOTE to be added, changed or deleted.

- **QLLC SPU IDLETO= Inactivity Clear Support.**

You may now specify the IDLETO=*minutes* operand on a TYPE=SPU REMOTE definition statement to force HNAS to clear an SPU call if no VC or LU activity occurs within the *minutes* interval. This operand is provided to limit the use of X.25 network facilities for an idle SPU.

- **QLLC SPU LUNAME= LOCADDR Coding Improvements.**

The **LUNAME=** operand that is coded for a TYPE=SPU REMOTE definition statement may now contain a 'gap count' rather than consecutive commas to separate SLU names and their related parameters when the SLU LOCADDR values are non-sequential. For example:

coding

```

1          2          3456789          1
LUNAME=( , slunm2, slunm3, , , , , , slunm10)
0 <- LOCADDR values

```

is equivalent to coding

```

1          2          34-9          1
LUNAME=( , slunm2, slunm3, 6, slunm10)
0 <- LOCADDR values

```

This logic was added to HNAS 230 APAR 2300033 on 05-13-2004.

- **QLLC SPU APPLNAME=ACQUIRE Printer Bind Support Improvement.**

HNAS V2R3M0 New Features

The **APPLNAME=** operand on a TYPE=SPU REMOTE definition statement has been modified to accept ACQUIRE as a keyword (reserved) value. When this APPLNAME= operand entry is selected via a 'hard wired' *applid* value (see LUNAME= operand) or via USSTAB processing, the ACB for an SLU is opened and a SETLOGON request is passed to VTAM but a REQSSESS request is withheld. This operation conditions the SLU to accept an application BIND but does not solicit the BIND. The SLU remains passive. This support is primarily used for printer LUs that are acquired based on need via requests from other LUs.

Note: This new feature was introduced into 230 as Enhancement APAR 2300030.

- **QLLC SPU OPTIONS=REUSEBUSYSPU support added.**

The **OPTIONS=** operand on a TYPE=SPU REMOTE definition statement has been modified to accept REUSEBUSYSPU as a suboperand value. This new operand allows the SPU to be allocated to a new call when it is already connected. This may be required if the QLLC PAD fails to notify HNAS of an SPU disconnect. When the REUSEBUSYSPU option is effect, the old VC call is cleared and the new VC call is allowed to proceed. When the REUSEBUSYSPU option is not in effect, the new VC call is cleared and the old VC call is allowed to continue. In the case, the following alarm message is generated:

```
NAS8102W PU FOR DTEADDR=DDDDDDDDDDDDDDDD IDBLK/IDNUM=xxxxxxx  
ON MCH mchname  
NAS8102W WAS IN USE WHEN SELECTED
```

Note: This new feature was introduced into 230 as APAR 2300116.

Enhancements - Expanded Content:

- Console command **MRMT** was enhanced to allow users to modify QLLC (LLC3) SLU definitions in the LUNAME= operand of TYPE=SPU REMOTE definition statements via privileged local or remote console access as follows:

```
MRMT RNM=spuname LUNAME=(locaddr,  
                           {slunamei}*  
                           [/{rcvpaci}*]  
                           [/{sndpaci}*]  
                           [/{applidi}*]  
                           [/{mxtnamei}*],...  
                           :  
                           {slunamen}*  
                           [/{rcvpacn}*]  
                           [/{sndpacn}*]  
                           [/{applidn}*]  
                           [/{mxtnamen}*])
```

Where: **locaddr** represents the starting position in the LUNAME= operand list where SLU name replacement, addition or deletion will start. The number of **slunamei** that follow the **locaddr** value cannot exceed the extent of LUNAME= operand list which is current set to a maximum of 255 entries.

To delete an SLU entry in the LUNAME= operand list, enter an asterisk in place of the SLU name.

To change an SLU name in the LUNAME= operand list, enter a new SLU name for the existing **locaddr** entry.

To add an SLU name to the LUNAME= operand list, enter a new SLU name for a new **locaddr** entry. Note that adding SLUs to a LUNAME= operand on any TYPE=SPU REMOTE definition statement requires that HNAS be configured for dynamic SLU support. This is done by creating a dynamic SLU pool using the **OPTIONS=LUDRPOOLCNT=count** operand on the BUILD definition statement. You can reserve up to 65534 dynamic SLUs for this purpose depending on the amount of available memory. Each SLU takes approximately X'638' bytes.

In addition to changing or adding an SLU name, you may also change, add or delete(*) the receive pacing count (**rcvpaci**), send pacing count (**sndpaci**), the APPLNAME= index (**applidi**) and/or the name of a TYPE=MXT REMOTE (**mxtnamei**) for an LUNAME= operand entry. If you wish to modify **rcvpaci**, **sndpaci**, **applidi** or **mxtnamei** without changing the **slunamei** for a **locaddr** entry, omit the **slunamei** value (*I* immediately follows **locaddr**) and specify the other suboperands as follows:

MRMT LUNAME=(locaddr,rcvpaci/sndpaci/applidi/mxtnamei,...).

Note: The MRMT command **will not** change or delete an existing LUNAME= operand entry if the SLU is not idle. To make an active SLU idle, you must issue the VTAM **V NET,INACT,slunamei** command. An SLU on an SPU that does not have a VC connection is, by default, idle.

The MRMT command processes SLU names left to right in the LUNAME= operand list. If an error occurs, it is reported and subsequent command processing is aborted. This means that it is possible for the MRMT to make some but not all of the changes you requested. To see all the changes that were made before the error occurred, issue the **DRMT LUNAME=** command after the MRMT command.

Examples:

Assume that for **RNM=spuname DRMT LUNAME=** the following display is presented:

```
RMTNAME  OPERAND  TYPE
spuname  008AB254  SPU
          LUNAME=*****/ / / /
          Q1990102/ / / /
          Q1990103/ / / /
```

This SPU has SLUs defined at LOCADDR=2 and 3. To add an SLU for LOCADDR=5, enter **MRMT LUNAME=(5,Q1990105)** (the SLU name(s) you enter are your choice). Following this command, enter **DRMT LUNAME=** which will produce the following display:

```
RMTNAME  OPERAND  TYPE
spuname  008AB254  SPU
          LUNAME=*****/ / / /
          Q1990102/ / / /
          Q1990103/ / / /
          *****/ / / /
          Q1990105/ / / /
```

HNAS V2R3M0 New Features

To delete the SLU for LOCADDR=3, enter **MRMT LUNAME=(3,*)**. Following this command, enter DRMT LUNAME= which will produce the following display

```
RMTNAME  OPERAND  TYPE
spuname  008AB254  SPU
          LUNAME=*****/ / / /
          Q1990102/ / / /
          *****/ / / /
          *****/ / / /
          Q1990105/ / / /
```

To add an SLU for LOCADDR=1 and 4, enter **MRMT LUNAME=(1,Q1990101,,,Q1990104)**. Following this command, enter DRMT LUNAME= which will produce the following display:

```
RMTNAME  OPERAND  TYPE
spuname  008AB254  SPU
          LUNAME=Q1990101/ / / /
          Q1990102/ / / /
          *****/ / / /
          Q1990104/ / / /
          Q1990105/ / / /
```

To add pacing counts for LOCADDR=1, 2 and 3 while also adding a new SLU name for LOCADDR=3, enter **MRMT LUNAME=(1,/2/3,/2/3,Q1990103/2/3)**. Following this command, enter DRMT LUNAME= which will produce the following display:

```
RMTNAME  OPERAND  TYPE
spuname  008AB254  SPU
          LUNAME=Q1990101/002/003/ /
          Q1990102/002/003/ /
          Q1990103/002/003/ /
          Q1990104/ / / /
          Q1990105/ / / /
```

To delete SLUs for LOCADDR=4 and 5, enter **MRMT LUNAME=(4,*,*)**. Following this command, enter DRMT LUNAME= which will produce the following display:

```
RMTNAME  OPERAND  TYPE
spuname  008AB254  SPU
          LUNAME=Q1990101/002/003/ /
          Q1990102/002/003/ /
          Q1990103/002/003/ /
```

- The HNAS **PING command** has been added to the console subsystem that will allow the console operator to check on the existence of a router on the IP network as well as a specific MCH on the router. The HNAS PING command is not a traditional IP PING command but rather an X.25 PING. The HNAS PING command requires an IP address to identify a router and an X.25 DTE address to identify the MCH link. These two pieces of information allow HNAS to establish an IP connection to the router and then an X.25 (XOT) connection via the MCH link. An indication of success or failure

for the PING request is reported back to the console operator.

The format of the PING command is as follows:

```
[IPADDR=a.b.c.d]
[CLGADDR=ddd.ddd]
[CLDADDR=ddd...ddd] PING [XTP|XOT] a.b.c.d ddd...ddd [ifn] [vcn]
```

If you set the IPADDR= and CLDADDR= modifiers, they become the default IP address and called DTE address for the PING command. They will be used if PING is entered with no arguments. If PING is entered with no arguments and IPADDR= and/or CLDADDR= is not set, the command will be rejected. XOT is the default REMOTE type unless you specifically enter XTP. You can always override the default IP address and/or called DTE address by specifically entering them as PING command arguments. If you set the default IP address and only want to change the called DTE address, enter `PING , ddd...ddd`

For XTP, 2 additional arguments are required: *ifn* selects the actual MCH on the IBM router and *vcn* selects the LCN for the call.

The PING command sends a Call Request packet to the target IP address carrying the called DTE address that you specify. A calling DTE address is supplied from the CLGADDR= modifier if one is given, otherwise, the calling DTE address is null. The facilities and call user data are canned as follows:

```
facilities: 080100420707430202    cud: 01000000HNASPING
```

The target IP address selects the router. The target DTE address selects the MCH serial interface on the router. Note that the router must be configured to map the target DTE address to a specific serial interface.

The PING can result in a Call Accept from the network if the call reaches its destination. Most likely, the PING will result in a Clear. The Clear cause and diagnostic codes can then be used to determine how far the actual calling function has gone. The Cisco doc and HNAS doc lists most clear reason and diagnostic codes.

- **TRCTRAP= diagnostic automation tools**

TRCTRAP= support provides automation tools for improved debugging and trace suspension.

Sometimes it is difficult to capture the cause of a problem in the HNAS internal trace table when tracing has to be stopped manually. In many cases, the only way to resolve a problem is to run HNAS traces. If a problem is intermittent, TRCPRNT is not a viable option because it can create an inordinate amount of SYSPRINT and consume valuable CPU cycles. Internal tracing is the next best option but it must be stopped as soon as possible after the problem has occurred. This may be difficult and, in some cases, may even be impossible.

HNAS has been modified to accept a new configuration operand and console command that will provide the ability to suspend internal tracing (and TRCPRNT) based on alarm message ID match, an input packet match or an output packet match. Multiple alarm IDs and packets may be specified so that a collection of values may be monitored (trapped). Once the trace and TRCPRNT activity is suspended (NAS0050A message generated), HNAS must be shutdown with `QY/password` so that the internal trace table can be formatted. Because the trace is suspended (frozen), HNAS need not be shutdown immediately but can be shutdown at your convenience. If the trap was performed with

HNAS V2R3M0 New Features

TRCPRNT enabled, the SYSPRINT file can be manually switched (see PRNT CLOSE|OPEN *dsname*|CLSOPN *dsname* console for additional information on HNAS SYSPRINT dataset switching). This will allow you to send us the closed SYSPRINT file without having to shutdown HNAS. If multiple SYSPRINT datasets are used, make sure that DISP=SHR is specified for all so that they can be accessed by ISPF or another program (like IEBCOPY or IEBGENER). Note: A second phase of this enhancement will be provided which will locate the trace data into a separate SYSOUT or DSN= file so that a shutdown is not required. The following text identifies the new HNAS BUILD TRCTRAP configuration operand and console command that may be used for trace trap processing.

CONFIGURATION ENHANCEMENT:

```
BUILD TRCTRAP=(ALRMLIST=(id1,...,idn),
               RCVLIST=(prot1 | pkt1,...protn | pktn),
               XMTLIST=(prot1 | pkt1,...protn | pktn))
```

CONSOLE COMMAND ENHANCEMENT:

```
TRCTRAP {ALRMLIST=? | * | (id1,...,idn) }
        {RCVLIST=? | * | (prot1 | pkt1,...protn | pktn) }
        {XMTLIST=? | * | (prot1 | pkt1,...protn | pktm) }
        {SHOW} {SUSP} {RSME}
```

FOR BOTH CONFIGURATION AND CONSOLE COMMAND:

idi => alarm ID (e.g, NAS1001I) that may contain asterisks (*) as wildcards (e.g., NAS1****).
A maximum of 16 *idi* entries may be specified.

proti => packet protocol (00=>XTP, 04=>XOT).

pkti => packet data from PKTTYPE for length *m* that can contain all ones as wild cards (e.g., 13FFFF).

A maximum of 512 bytes for protocol and packet data may be specified which include a length byte (*m*) for each packet. The packet data is compared only for the length (*m*) that is given. PAD sessions operating with non-space parity will encounter data mismatches with the filter list when non-qualified data is specified. You may specify the same data with different parity settings as required.

Note: A match on any value in the ALRMLIST=, RCVLIST= or XMTLIST= operands can suspend tracing. The first match in any of these operands is the winner. Multiple operands with multiple values are provided to allow for the greatest flexibility.

FOR CONSOLE COMMAND ONLY:

Enter: * as first operand value to reset operand
 ? as first operand value to display operand
 SHOW to display all operand values
 SUSP to suspend trace manually
 RSME to resume suspended trace

In addition to the list display produced by TRCTRAP ALRMLIST=? or TRCTRAP RCVLIST=? or TRCTRAP XMTLIST=? or TRCTRAP SHOW, the following trace state information is displayed:

```
TRACE STATE: IDLE          <- logging inactive
              LOGGING      <- logging active
              SUSPENDED BY COMMAND <- logging suspended
              SUSPENDED BY TRAP   <- logging suspended
```

Note: When values are specified for the ALRMLIST=, RCVLIST= and XMTLIST= command arguments, they are appended to the end of the corresponding table in the order in which the values are entered. For example:

If BUILD TRCTRAP=(ALRMLIST=(NAS8***I) was specified in the CDF and if TRCTRAP ALRMLIST=(NAS2021I,NAS2100I) then TRCTRAP ALRMLIST=(NAS2201I) are issued as console commands, the TRCTRAP ALRMLIST=? console command would produce the following display:

```
ALRMLIST=NAS8***I
          NAS2021I
          NAS2100I
          NAS2201I
```

CONFIGURATION EXAMPLES:

Trap multiple events:

```
BUILD TRCTRAP=(ALRMLIST=(NAS3701W, <- LU ACB open failed issued
                  NAS3798I), <- LU starting session issued
              RCVLIST=(041B0502, <- Reset 05/02 received
                  041309A5, <- Clear 09/A5 received
                  04F1FFFF), <- any Diag packet received
              XMTLIST=(041BFFFF, <- any Reset packet sent
                  0413FFFF)) <- any Clear packet sent
```

Trap PAD logon request:

```
BUILD TRCTRAP=(RCVLIST=(04FF6CA063F0, <- l cp (even)
                  04FFEC20E370, <- l cp (odd)
                  04FF6C206370, <- l cp (none)
                  04FFCCA0C350, <- L CP (even)
                  04FF4C2043D0, <- L CP (odd)
                  04FF4C204350)) <- L CP (none)
```

Trap QLLC Terminate-Self PIU:

```
BUILD TRCTRAP=(RCVLIST=(04FF2C0000FFFFFFFF0B8000010683)) <- TERM-SELF
```

TRCTRAP ALERT MESSAGES:

The following alerts are provided when tracing is suspended due to an alarm ID or packet match.

```
NAS0050A TRACING SUSPENDED DUE TO TRAPPED ALARM ddddddd
NAS0050A TRACING SUSPENDED DUE TO TRAPPED INPUT PACKET xx...xx
NAS0050A TRACING SUSPENDED DUE TO TRAPPED OUTPUT PACKET xx...xx
```

HNAS V2R3M0 New Features

The following alert is provided when tracing is suspended via the TRCTRAP SUSP or TRCALL SUSP command.

```
NAS0050A TRACING SUSPENDED BY COMMAND
```

The following alert is provided when tracing is resumed via the TRCTRAP RSME or TRCALL RSME command.

```
NAS0060W TRACING RESUMED BY COMMAND
```

The following alert is provided after the NAS0050A message (for TRCTRAP SUSP) or NAS0060A message (for TRCTRAP RSME) when trace trap filtering is active. The message is also issued if trace trap filtering is enabled via the TRCTRAP configuration operand.

```
NAS0070W TRACE TRAP FILTERING ACTIVE
```

The following alert is provided after the NAS0050A message (for TRCTRAP SUSP) or NAS0060A message (for TRCTRAP RSME) when trace trap filtering is inactive.

```
NAS0070W TRACE TRAP FILTERING INACTIVE
```

GENERAL NOTES (SUMMARY):

- 1) The current tracing parameters are not altered when tracing is suspended. To restart trace logging, you need to issue the TRCTRAP RSME or TRCALL RSME console command.
- 2) When tracing is suspended, TRCPRNT trace entry logging in SYSPRINT is also suspended.
- 3) Since TRCTRAP processing is global (not directly associated with enabled trace options), it is important to have the appropriate traces activated so that the events leading up to the suspended trace will be captured in the internal trace table. For example, if a RCVLIST= or XMTLIST= is provided, TRCVC MINDATA or TRCVC MAXDATA should be active.
- 4) TRCTRAP logic was added primarily as a tool for Comm-Pro in order to provide a way of simplifying problem diagnosis. In the past, Comm-Pro had to develop custom modifications to trap certain events. The TRCTRAP logic now standardizes this processing. In most cases, Comm-Pro will be advising customers on what to enter for TRCTRAP arguments rather than giving them a custom trap modification. This summary intended to provide some information about the TRCTRAP function. We do not, however, expect customers to use TRCTRAP functions without Comm-Pro involvement.

Note: This new feature was introduced into 230 as Enhancement APAR 2300045/2300047.

HNAS V2R2M0 New Features

V2R2M0 General Availability - 11-11-2002

- All appropriate V2R1Mn **APARs** incorporated as permanent fixes into this release. Please refer to the appropriate APAR Summaries for additional information.
- **High Memory Support** was added to this release to reduce low memory requirements and allow substantially larger configurations. The feature is enabled when **APFXEQ** is coded on the PARM= start parameter. This new support allows HNAS to allocate free memory from above the low memory line which is typically 16MB. You may specify up to 7 sub-pools using the APFMEMSP= operand. For example: **APFMEMSP=(229,230)**. Memory is allocated in the sub-pool from left to right, so that for this example memory in sub-pool 230 will not be used until all memory in sub-pool 229 is exhausted.
- **Multiple 'same type' Server** enhancement has been added to V2R2M0. It allows the use of the same or different IP addresses for the same type of LOCAL as well as a socket limit of up to 65535 (the theoretical limit) per LOCAL. When more than 2000 sockets are specified per LOCAL, you need to specify the SHAREPORT operand in the TCPIP PROFILE so that the stack will allow port 1998 to be used for more than 2000 connections. Within HNAS, each set of 2000 sockets is treated internally as a server. So if you code **SOCLMT=6000** on a LOCAL statement, this will create 3 servers for the same IP address and port number. The stack will present connections to the 3 servers in a round robin fashion. With this feature there is no longer a requirement to run multiple copies of HNAS if you need to support more than 2000 sockets.
- **Multiple Stack Support** enhancement has been added to V2R2M0. This allows the same HNAS address space to act as server for more than one TCP/IP Stack. The TCPNAME operand which in the past was only allowed on the BUILD definition statement may now also be coded on the LOCAL definition statement to identify a specific stack for the server.
- **Shared Socket Support (REMOTE TYPE=XOT IPADDR=a.b.c.d,PORT=1998)**. For Cisco routers, this feature eliminates the need to pre-define separate pools of inbound and outbound TCPIP sockets for a specific router in the HNAS configuration. Specifying IPADDR=a.b.c.d, PORT=1998 on a TYPE=XOT REMOTE definition statement (which identifies a specific router) creates a single pool of TCPIP sockets that can be used for both inbound and outbound sessions. The number of sockets in the pool is determined by the VCLMT= operand. A socket from the pool is allocated to a activating session on a first come, first served basis. The same socket can be used for an inbound initiated connection at one time then an outbound initiated connection at another time. Please refer to the Chapter 4 operand IPADDR= for a complete description of this support.
- **Authorization Key Support** is now required in V2R2M0. HNAS **now requires an authorization file (NASAUTH)** in order for it to operate beyond the 90 day trial period. The authorization file is specified by the **//AUTH DD DSN=QQQQ.HNASMAC(NASAUTH),DISP=SHR** statement in the HNAS start job.

For users undergoing HNAS trials, a NASAUTH file is provided with a 90 day expiration date. Customers who have completed their trial tests and have purchased HNAS will be given a permanent authorization file that will never expire. Existing customers upgrading to 220 from an earlier release will be provided with a permanent key when they acquire the upgrade.

Please refer to the HNAS Authorization Considerations content under the Chapter 2 Activation and Operation section of the HNAS Guide and Reference manual for additional information.

HNAS V2R2M0 New Features

- **Qualified Logical Link Control (QLLC/LLC3)** over XOT support has been added. Native SNA support for Type 1 PUs (maximum of 31 SLUs per PU) and Type 2 PUs maximum of 255 SLUs per PU) and LU Type 0, 1, 2 and 3 device support over QLLC (PUT2.1 or LU 6.2 not currently available). This support is new for V2R2M0 and eliminates the previous requirement to employ DLSW for QLLC line attachment. For additional information please see QLLC operand and parameter additions at the end of this document.
- **Application Major Node File generation (AMNF)** support has been added. When PARM=FASTRUN is specified and the configuration return code is 4 or less, HNAS will produce an AMNF using the SLU names that are specified in the **LUNAME, PVC, SVC0, SVC4** and **SVC5** operands from all REMOTE definition statements in the **Configuration Data File (CDF)**. The AMNF is placed in a dataset identified by the MAJNODE DD statement. The MAJNODE dataset can be a sequential file or a member of a partitioned dataset. DISP=OLD is required. Automatic AMNF generation ensures that there will never be a mismatch between the SLUs specified in the CDF and those specified in the AMNF. Application specific VTAM parameters can now be coded on the respective CDF resources. The VTAM parameters will be promoted along with the SLU names when using the AMNF generation process.
- **Multiple SYSOUT Support** has been added to this release. This allows the local or remote console operator to Close and Open (switch) the active SYSPRINT DSN=dsname to any entry defined in the HNASXEQ JOB/JCL.
- The HNAS **Timer Queue** logic was improved to reduce processing cycles for environments with large configurations (LUs, VCs and REMOTE resource definitions).
- Various **CDF Options** and **Parameters** were added or changed to allow the user greater flexibility when defining resources in V2R2M0:
 - **Callout Alternate DTE Connect Retry Support -**
SVC0=(...,lu-name/dteaddr1-dteaddr2-dteaddr3O/mxt-name,...) may be coded to enable callout connection retry using up to 2 alternate called DTE addresses across TYPE=XOT REMOTEs identified by the RTEOUT= list on a LOCAL TYPE=XOT statement. If the call to dteaddr1 fails (CLEAR or timeout response to HNAS Call Request packet), then HNAS will try the call using dteaddr2 and dteaddr3 (if necessary). This logic is available for LLC0 and LLC5 callout operation (SVC0= and SVC5=).
 - **Callout Session Connect Balancing -**
OPTIONS=BALANCERTEOUT operand has been added to the **TYPE=XOT** LOCAL definitions to enable Round Robin connection balancing across TYPE=XOT REMOTEs identified in the **RTEOUT=** list. HNAS now has the ability to provide a form of load balancing for host initiated (callout) sessions across multiple routers.
 - **Callout Connection Routing using Calling or Called Address -**
RTEOUT=(remote-name1/dteaddr1T,remote-name2/dteaddr2S,..) The RTEOUT= operand of the TYPE=XOT LOCAL statement has been modified to direct routing of HNAS call requests by called or calling address. When HNAS places a call the called and calling addresses are known. For LLC0 and LLC5 the called address comes from the SVC0= or SVC5= operands on the TYPE=MCH REMOTE statement. The Calling address comes from the DCEADDR= operand. For GATE (LLC4) the addresses are provided by the CTCPC. 'T' in the RTEOUT= operand specifies that the called address (target) is to be used for RTEOUT= entry selection. 'S' specifies that the calling address is to be used for entry selection. Note that 'T' is the default and provides downward compatibility with earlier releases.

- Callin DTE Address Filtering - LLC0 (SVC0=) and LLC5 (SVC5=) Callin DTE Address filtering for TYPE=MCH|XTP**
 REMOTEs SLUname association. HNAS now supports filtering of DTE addressed based on the number of digits coded on the SVC0= and SVC5= dteaddri suboperands. In earlier releases the inbound DTE address in the call request packet had to match the SVC0|5 dteaddri value exactly for LU selection to occur. Please refer to Chapter 4 for revised information on SVC0 and SVC5 coding rules.
- Callin SLU/PLU Fixed Connection Support - SLU/PLU fixed connection support for PCNE LLC0 (SVC0) and PAD LLC5 (SVC5) now supports coding of an MCH APPLNAME= index entry for direct mapping instead of the previous 211 requirement for a SYSL=DATA=char system select value. Please refer to Chapter 4 for revised information on SVC0 and SVC5 coding rules. The following table depicts the coding differences between 211 and the new 220 coding requirements:**

Old Method Under V2R1M1	New Method Under V2R2M0
SVC0= (2, MCH10001/123456IA, MCH10002/X001540IB) SYSL= (DATA=A/0,DATA=B/1) APPLNAME= (TSO,CICS)	SVC0= (2, MCH10001/123456I0, MCH10002/X001540I1) APPLNAME= (TSO,CICS)

- Callin Default PLU Support (SYSL= now Optional) - Callin Default PLU assignment via SVC0=/SVC5= APPLNAME= association** is now supported in HNAS. For **LLC0 and LLC5 inbound calls**, after HNAS sets the LLC TYPE and locates an LU for the call, a PLUname for the session must be determined. If the name is not provided by the **SVC0/5=** operand (SVC5=(...,LU1/I2,...) where 2 is an index in APPLNAME=) or by SYSL= then the first **APPLNAME=** entry is used as a default. See sample coding above. This default is also taken if **SYSL=** is omitted (**SYSL= was required** in previous releases and is **now optional**). Please refer to Chapter 4 for revised information on SVC0= and SVC5= coding rules.
- Callin GATEFC REQSESS Support**
Callin GATEFC logic now supports an option which causes HNAS to issue a REQSESS for each terminal session SLU to force an acquire from the target PLU. This is done to ensure that the terminal SLUs are in the active state (BIND/SDT) to accept incoming calls. This new logic is enabled when an asterisk (*) is appended to the end of a plunamei value (e.g., CSFI*) on the REMOTE LUNAME= operand.<Ref: 11-20-2002>
- OPTIONS=LCN0USED** may be specified on the **TYPE=MCH** REMOTE definition to force HNAS to start the GATE resource identifier (RESID) at zero rather than at one (plus the PVC count). **GATE=GENERAL,PVC=NONE** is also required. Some **CTCPs** (**CSFI** for example) know when LCN0 is used on an MCH so they expect the first RESID to be zero not one. Without this option a CTCP, which is configured for LCN0=USED, would think that the second SLU in the SVC4 operand was being used rather than the first. **OPTIONS=LCN0USED** ensures that HNAS and the CTCP are in synch.
- OPTIONS=MCHTMR=value** operand has been added to the **TYPE=XOT/XTP** REMOTE definitions to allow users to fine tune the activation delay for GATE Control Session and Fast Connect

HNAS V2R2M0 New Features

LU Sessions. This timer also applies to activation of the SLUs defined for callout support. In previous releases of HNAS the timer value was fixed at 60 seconds which is also the new parameters default value. <Ref: 210C002 05-10-2002>

- **OPTIONS=TCPRBLMT=value** operand was added to **TYPE=XOT/XTP** REMOTE definitions to allow users to reduce the number of staging buffers associated with each TCP/IP socket (XOT Virtual Circuit VC). In previous HNAS releases the buffer count was fixed at 7. We have observed that a value of 2 is adequate for a variety of transaction processes as long as the buffer will accommodate a complete XOT packet. Configuration environments with hundreds of sockets/VCs will benefit from the reduced memory requirements when running under the reduced buffer count. The new parameter default buffer limit value currently remains at 7 for XTP and is now 2 for XOT.
- **OPTIONS=LUBLTCNT=value** operand has been added to the BUILD definition to allow users to control the length of the Buffer List Table. The table is used to process all data packets in an m-bit chain for transfer to the application. In V2R1Mn the length was increased to 40 while earlier releases had a length of 30. <Ref: 2100024 06-22-2002>
- **HOME=value** operand has been added to the **TYPE=XOT|XTP** REMOTE definition to allow users to associate a REMOTE with a LOCAL resource. This new operand operates in conjunction with the Multiple Server Support. If a HOME values isn't specified a default will be assigned at activation.
- **SOCLMT=value** operand has been added to the **TYPE=XOT|XTP** LOCAL definition to allow users to control the number of sockets available under each LOCAL definition. The previous limitation restricted users to 2000 sockets per HNAS address space. This new operand is used in conjunction with the Multiple Server Support.
- **TAP=0** is the new default operand for **TYPE=XOT|XTP** REMOTE definitions. In previous releases the default TAP (keep-alive) value was 60. <Ref: 2100015 05-16-2002> The revised logic also restricts TAP= value parameter coding to the first **TYPE=XOT|XTP** REMOTE definition per unique IPADDR. In previous releases the CDF decode would allow TAP=value coding on all remotes even though only the TAP= parameter from the primary (first) remote was used for the keep-alive timer control.
- **OPTIONS=ECHOFAC** operand has been added to the **TYPE=MCH** REMOTE definition to enable echoing of facilities from the inbound XOT Call Request packet to the HNAS XOT Call Accept packet. Some networks don't support this feature while others can use the response to further negotiate facilities values. A default of **NOECHOFAC** is assumed if no parameter is coded.
- **SUBD=** allowed on **non-Fast Connect GATE MCHs** (CTCP selection using subaddress digits) This permits inbound GATE sessions to select a CTCP using subaddress digits in the call request packet. Previous logic allowed SUBD= only for Fast Connect GATE sessions (NPSI restriction). <Added to 220 New Features list on 11-21-2003. See APAR Ref: 2200061 dated 11-21-2003 for additional information.>
- **TAP** keep alive logic was redesigned and now schedules an XOT Call Request instead of an XOT Clear Request to ensure that a protocol level response (in this case a Clear Request) is always received from the router's XOT services. Some XOT router IOS or software levels don't always respond to an XOT Clear Request received on an idle virtual circuit but must respond to a XOT Call Request.

TAP= logic can now perform protocol level tapping with XOT Call or XOT Clear packet. See new parameter OPTION=TAPWITHCLR that was added on 2003-08-27 via APARs 2200048 and 2200052.

- The **FASTRUN** process will now allow the name for the AMNF VBUILD statement to be specified using the APPLNAME= operand rather than NASNAME= operand on BUILD definition statement (APAR 2200080).
- The **FASTRUN** start parameter option process was updated to produce a **Memory Requirements Summary Report**. This new feature is extremely useful in identifying storage requirement for various HNAS system components such as: program size, buffer, trace tables and a variety of control block types.
- **PRNTDATE** start parameter option is now supported which allows users to enable a special DATE prefix in front of the TIME prefix for SYSPRINT log entries. The **yy.ddd** format provides a 2 digit year, a period and the Julian date. This will improve SYSPRINT log viewing capabilities.
- **Several enhancements** were made in this release to improve **Alarm Event** and **Alert Message** delivery processes and provide alert event counter history:
 - **ALRMLMTS** operand has been added to the BUILD definition statement to allow alarm limits and the alarm table size to be configured (ALRMLMTS=(Time_interval, Informational_cnt, Default_cnt, Action_cnt, Warning_cnt, Error_cnt, Severe_cnt, Other_cnt, Log_size)) or bypass alarm limits (ALRMLMTS=NONE). The time value and message limit interval values were fixed in our previous release while the alarm logging table size is new to this release:
 - **ALRMLMTS** values may be modified via the **ALARM** console command (ALARM LIMITS=(Time, Icnt, Dcnt, Acnt, Wcnt, Ecnt, Scnt, Ocnt)) or disabled using the console command (ALARM LIMITS=NONE) which resets the settings to zero's effectively disabling the limits. The **alarm logging table** count may not be modified once HNAS is running, it remains fixed by the value specified in the BUILD ALRMLMTS operand.
 - **ALRMLMTS** values may be displayed via the **ALARM** console command (ALARM LIMITS=?).
 - An **alarm logging table** has been added to record alarm ID's and the number of times a particular alarm is issued. The table is ordered by alarm ID. These counters are independent of alarm limits or alarm filtering modes. An alarm logging table is generate for each console PCE (initial local alarm logging limits and table size are based on the ALRMLMTS=values defined in the CDF configuration file:
 - The **alarm logging table** may be cleared (all entries removed) or reset (all counters reset for current entries) via the **ALARM** console command (ALARM LOG=CLEAR or ALARM LOG=RESET).
 - The **alarm logging table** may be displayed to show the alarm IDs and associated counts via the **ALARM** console command (ALARM LOG=?).
- **ALRMFLTR (P)**urge option has been added which allows users to specify whether or not specific alert/event alarm messages should be recorded in the SYSPRINT file. The filtering option available in our previous HNAS release gave users the ability to prevent messages from being displayed at the operator console although they would always be logged in the SYSPRINT file. The major benefit of this new feature allows users to dramatically reduce the size of the

HNAS V2R2M0 New Features

SYSPRINT file by eliminating specific message activity. The alarm counts for all purged events are still recorded in the Alarm Logging Table summary (by type) so that users are aware of specific message activity.

- An **Initialization Complete alert message NAS0001I** is now generated once the initialization phase of the HNAS activation process completes successfully. This new alert message is independent from the HNAS and router connectivity alert messages which were previously used to determine HNAS availability. The new message now provides a more accurate representation of HNAS system availability.

```
NAS0001I HOST NAS INITIALIZATION COMPLETE, ALL FUNCTIONS READY
```

- Following are some of the **alert messages** that were added to convey additional **TCP/IP states**. Please refer to the documentation manual for a complete list of new and reclassified alert message ID's. <Ref: 2100020 05-31-2002>

```
NAS2321W (TAKESOCKET FAILED)
NAS2311W (GIVESOCKET FAILED)
NAS2301W (GETCLIENTID FAILED)
NAS2268I (ACCEPTED CONNECT PASSED)
```

- **NAS6715W and NAS7715W Alert Message** SYSPRINT entries now include a display of the buffer content for the call request packet that encountered the clear condition. The availability of the call request packet image generally eliminates the previous requirement to run debugging traces to view packet content improving fault isolation resolution times.
- **NAS2nnns and NAS2nnnI TCPIP Alert Message Enhancement** - The following HNAS TCPIP messages were modified to improve resource identification. <**06-25-2003** - Although this new feature was provided in the initial implementation of 220 we erroneously omitted the description in this section >.

From	To	Message ID
NAS2nnns->		`cmd FAILED FOR SOCK=a.b.c.d(p) SOCKID=sid RC=rc/erno'
	->NAS2nnns	`comp=a.b.c.d(p) SOCKID=sid PCEID=pid NAME=compname'
	->NAS2nnnI	`cmd REQUEST FAILED, RC=rc/erno'

Where: *cmd* = failing TCPIP command.
comp = failing component (SERVER (LOCAL) or CLIENT (REMOTE)).
a.b.c.d(p) = IP address and port number for failing component.
sid = TCPIP internal socket identifier for failing component.
pid = Process Control Element identifier for failing component.
compname = name of failing LOCAL or REMOTE component.
rc/erno = return code and error number for failing TCPIP command.

- **NAS2252E TCPIP Alert Messages** were added to provide additional diagnostic information and event activity for TCP/IP Select processing. <Added to 220 New Features list on 02-19-2004. See APAR Ref: 2200073 dated 02-16-2004 for additional information.>

```
NAS2252E SELECT REQUEST INTERRUPT LOST, RETRY WILL BE ATTEMPTED
NAS2252E SELECT REQUEST INTERRUPT LOST, SOCKET MUST BE CLOSED
NAS2252E CLIENT=iii.iii.iii.iii(port) SOCKID=sockid PCEID=pceid
NAME=compname
```

- A **NAME=pce-name** field was added to the TCP/IP SERVER and CLIENT continuation messages to convey the resource name that the process is associated with. The pce-name may refer to a REMOTE, LOCAL, or MCH resource name. This new field provides an immediate association with the alert message and the resources connected with the event.
- A new default start parameter **SHOWCNFG OFF** was added to eliminate CDF scan messages from being displayed at the master operator console when HNAS is started. Operation's environments that wish to continue seeing these messages can code **SHOWCNFG** on the start parameter to enable the messages. Note that messages are always written to the **SYSPRINT** file unless the **ALRMFLTR (P)**urge option is enabled. This parameter operates independently from the **SHOWOFF** or **SHOWERR** parameters. <Ref: 2100022 06-13-2002> This new release also allows specific start parameter values to contain a space separator character between the primary command and the action value (i.e. SHOWCNFG OFF).
- Corrective logic for the **SHOWERR** start parameter was developed to prevent HNAS from delivering Information, Default and Comment messages to the master operator console. Only error messages are now delivered to the master operator console in SHOWERR mode (which is the new default). Note that all of these messages will continue to get logged in the **SYSPRINT** file unless the **ALRMFLTR (P)**urge option is enabled.
- **Additional trace entries** and the ability to control buffer data information written to the **SYSPRINT** file continue to improve the diagnostic capabilities of this product:
 - **TRCLU** and **TRCVC** sub-operands **MINDATA** (default) **MAXDATA** and **NODATA** can now be coded on the PARM= or set via the HNAS console. The NODATA option dramatically reduces SYSPRINT output when tracing doesn't require partial 'MINDATA' or complete 'MAXDATA' buffer or trace control area data to be displayed along with the trace entries.
 - **TRCBST** start parameter and associated console command has been added to allow tracing of **alarm logging table entry** insertion.
 - **TRCCNFG** start parameter has been added to allow tracing of the entire HNAS configuration process.
- **Additional HNAS console capabilities:**
 - Console commands **DLCL** and **MLCL** were added to allow users to modify **TYPE=XOT|XTP** LOCAL definitions via the privileged local or remote console access.
 - Console command **DPARM** display output was modified to list all of the default start parameter options to improve presentation of the operational parameters.
 - Console modify command logic was added that will allow **sublist** elements to be continued so that you do not have to enter them on one line. The continuation (M-char) is a plus (+) that appears after the record text but before any comments (;). This logic became a requirement as the REMOTE **SVC0=, SVC3= and SVC5=** operand parameter strings continue to expand.
 - Console **DPARM EXEC** argument was added allowing for the displaying of start-up EXEC-PARM= parameters as well as a means of viewing the revised settings after parameter changes are made to the active HNAS environment.
 - Console **HELP** enhancement treats command ? as HELP command.

HNAS V2R2M0 New Features

- Console **VARY** and **DPCE** command has been updated to allow users to enter the actual resource name in the command syntax.
- Console **resource identifier enhancement** allows resource IDs to be entered on either side of a command: (e.g., RNAME=rmtname1 DRMT RNAME=rmtname2). The value on the left side of a command remains the default value for all related commands. The value on the right side of a command overrides the default for the current command only. If a resource ID is entered without a command, it simply sets the default value for all related commands. Please refer to the V2R2M0 documentation manual for a complete list of resource IDs that may be entered on either side of a command.
- **Qualified Logical Link Control operand and parameter** additions for V2R2M0:
 - **SVC3** and **LLC3** operands have been added to the **TYPE=MCH** REMOTE definition statement. The SVC3 operand has the same form as the SVC0 and SVC5 operands except that instead of specifying SLU names, SVC3 entries are SPU names. SPUs are defined by **TYPE=SPU** REMOTE definition statements.
 - **IDBLK** and **IDNUM** operands have been added to the **TYPE=SPU** REMOTE definition statement. HNAS will allocate an SPU to a QLLC virtual circuit based on an IDBLK and IDNUM match or based on a calling DTE address match from the SVC3 operand. This ensures that a one-to-one mapping is maintained between the logical SPU in HNAS and the real SPU in the network.
 - **LUNAME** operand has been added to the **TYPE=SPU** REMOTE definition statement. The LUNAME operand provides SLU names for the real SPU in the network. The SLU names are positional. The first SLU name is for LOCADDR=1, the second for LOCADDR=2 and so on. If the real SLU local addresses are non-contiguous, the corresponding entries in the LUNAME operand are simply skipped. For example: LUNAME=(SLU1,SLU2,,SLU4,,SLU6) identifies SLUs whose local addresses are 1,2,4 and 6, respectively. Local addresses 3 and 5 are undefined.
 - **APPLNAME** and **SYSL** operands have been added to the **TYPE=SPU** REMOTE definition statement. QLLC SLUs can select a host application via LOGTAB or USSTAB interpretation or based on text specified in the SYSL operand.
 - **CUD**, **DCEADDR**, **FAC**, **LOGTAB** and **USSTAB** operands have been added to the **TYPE=SPU** REMOTE definition statement to override values specified on the root **TYPE=MCH** REMOTE definition statement.
 - **MAXDATA** operand has been added to the **TYPE=SPU** REMOTE definition statement. This allows users to control the MAXDATA value of the SPU level. The default value is 256 (plus the RH count).
 - **OPTIONS** operand has been added to the **TYPE=SPU** REMOTE definition statement. The following connection options are currently supported:
 - **PRIMARY** - marks the SPU as the primary half session in a primary-to-secondary.
 - **SECONDARY** - marks the SPU as the secondary half session in a primary-to-secondary.
 - **PEER** - marks the SPU as a peer half session in a peer-to-peer.
- Additional information regarding product upgrade considerations is available to existing customers in the Migration section of the HNAS Guide and Reference manual.

HNAS V2R1M1 New Features

V2R1M1 General Availability - 06-01-2002

- All appropriate V2R1M0 **APARs** have been incorporated as permanent fixes
- **Z/OS V1R2 (and V1R4)** officially supported in HNAS V2R1M1. While Z/OS V1R1 is supported under HNAS V1R1M4 and V2R1M0, TCP/IP Stack design changes in Z/OS V1R2 required logic changes in HNAS that were initially attempted with APAR correction (210nnnn) and ultimately resolved in the HNAS V2R1M1 release. <Ref: 2100003, 2100011 05-24-2002>
- **SLU Selection by CUD Data** - Allow an SLU to be selected by Call User Data rather than calling DTE address when a Call Request packet is received. <Ref: 210C001 05-07-2002 Item 1>
- **SLU to PLU Connection Wiring** - Allow an SLU connection to a PLU to be 'wired' together without operator/user input. This enhancement permits a user to dedicate an SLU to a specific host application for a PCNE or PAD session by specifying a system select character after the DTE address for an SVC0|5 operand. <Ref: 210C001 05-07-2002 Item 2>
- **OPTIONS=MCHTMR=value** operand has been added to the TYPE=XOT|XTP REMOTE definitions to allow users to fine tune the activation delay for GATE Control Session and Fast Connect LU Sessions. In previous releases of HNAS the timer value was fixed at 60 seconds which is also the new parameters default value. <Ref: 210C002 05-10-2002>

This page left intentionally blank.

HNAS V2R1M0 New Features

V2R1M0 General Availability - 03-01-2002

- All appropriate V1R1M4 **APARs** have been incorporated as permanent fixes.
- The HNAS **dispatcher** has been enhanced to minimize non-productive subtask dispatches which reduces total CPU utilization.
- **Dynamic IP address** assignment (**IPADDR=DYNAMIC**) is now supported for Cisco XOT router environments. This feature eliminates the previous requirement to pre-define routers to HNAS when only inbound connections will be used and reduces the size of the HNAS CDF configuration file and memory requirements. When dynamic IP address assignment is in affect for a router, shoulder tapping (KEEP ALIVE) processing is inhibited.
- **MODIFY** interface made standard and is invoked via the **USEMDFY** start parameter.
- **TRCTASK** console command added to trace subtask WAIT and POST events.
- **TRCSUBR** trace entry expanded to log full register set in HNAS trace table.
- **USSTAB** operand can now be specified on the TYPE=MXT REMOTE definition statements to override USSTAB on the root TYPE=MCH REMOTE.
- **PADPARM** operand can now be specified on the TYPE=MXT REMOTE definition statements to override PAD parameters on the root TYPE=MCH REMOTE.
- **PADPARM**, **CUD** and **FAC** values can now be modified via the **MRMT** console command accepting full list or partial list via offset/value notation.
- **LLC0,...,LLC5** entries can now be added or deleted via the **MRMT** console command.
- A **Configuration Error Summary** is now displayed as a function of severity after the CDF is processed. The number of Informational (RC=0 level), Warning (RC=4 level), Error (RC=8 level) and Severe Error (RC=12 level) messages are listed.
- **XOT Call Accept Facilities support.** Facilities values provided in the router's XOT Call Accept packets are now processed by HNAS. This support allows the router to step down the facilities values (like packet and window sizes) originally provided in the HNAS XOT Call Request packet.
- **CONCMDQ** operand has been added to BUILD definition statement to allow HNAS console commands to be executed at HNAS startup without operator intervention. **DNAS** is the default command when the CONCMDQ operand is omitted.
- **ALRMFLTR** operand has been added to BUILD definition statement to allow HNAS console messages to be filtered by message ID.
- **ALRMFLTR** values may be added or deleted via the **ALARM** console Command (ALARM FILTER=(ALLOW|SUPPRESS,id1(A|S),...idn(A|S))).
- **ALRMFLTR** values may be displayed via the **ALARM** console command (ALARM FILTER=?).

HNAS V2R1M0 New Features

- **Alarm limits** are now enforced on all messages written to the system console. This logic was implemented to prevent excessive alarm event activity from inundating the master console log with HNAS Alert messages. Fixed values are as follows:
 - 30 - Time interval (in seconds) in which that limits are in effect.
 - 20 - maximum number of Informational messages allowed during interval.
 - 10 - maximum number of Default messages allowed during interval.
 - 20 - maximum number of User Action messages allowed during interval.
 - 25 - maximum number of Warning messages allowed during interval.
 - 30 - maximum number of Error messages allowed during interval.
 - 30 - maximum number of Severe Error messages allowed during interval.
 - 10 - maximum number of all Other messages allowed during interval.
- We can provide customized over-ride values for V2R1M0 in the event that our fixed values aren't suitable for your environment. In our upcoming V2R2M0 release the default values can be replaced with user defined values via the revised ALRMLMTS operand.
- Some new **Alert messages** were added to improve environment monitoring capabilities:
 - NAS0910I - Daily Date Cross Over event (issued at midnight) in SYSPRINT for date validation.
 - NAS5000I - Starting Console Session event for remote access monitoring and audit.
- **Remote consoles** can be conditioned to receive HNAS error messages via the **ALARM** console command. Remote consoles, like the system console, are also subject to alarm limits.
- **LOGTAB** operand has been added to the BUILD and REMOTE definition statements to allow LOG table processing in addition to USS table processing (**MCHSOL** must be selected from the APPLNAME operand list). **LOGTAB** value may be modified via the **MRMT** console command.
- **OPTIONS** operand has been added to the REMOTE definition statement to allow activation of special processing. The following options are currently supported:
 - **REPDCEADDR** - causes the calling DTE address in outgoing Call Request packets to be replaced with the value specified for the DCEADDR operand (valid for **GATE=GENERAL** only).
 - **STRIPRTEIN** - causes the called DTE address from incoming Call Request packets to be removed before the packets are passed to the host CTCP application (valid for **GATE=GENERAL** only).
 - **STRIPFAC** - causes the facilities from incoming Call Request packets to be removed before the packets are passed to the host CTCP application (valid for **GATE=GENERAL** only).
 - **OPTIONS** values may be added or deleted via the **MRMT** console command.
- **MONITOR** console command has been modified to display VC and LU connect and disconnect counts at the end of the monitor interval.
- **PACE=0** is no longer necessary to inhibit default messages when pace is omitted for TYPE=MCH resources.
- **Console Prompt** changed from '**COMMAND UNRECOGNIZED, RE-ENTER**' to '**INPUT ERROR, RE-ENTER**' to clearly distinguish from application error messages like those generated by TSO.