IBM

VTAM

LY30-5601-1

Diagnosis

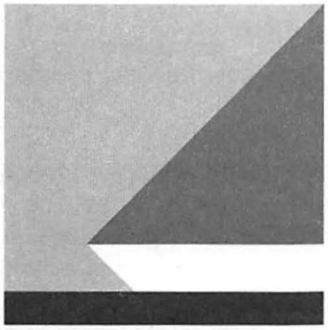
Version 3 Releases 1, 1.1, 1.2, and 2

LY30-5601-1

VTAM

Diagnosis

Version 3 Releases 1, 1.1, 1.2, and 2



File Number
S370/4300/30XX-50

Program Numbers
5665-313 (MVS/370)
5665-289 (MVS/XA)
5664-280 (VM/SP)
5666-313 (VSE)

First Edition (July 1988)

This major revision replaces SC23-0116-3. This edition applies to the Advanced Communications Function for VTAM Version 3 Release 1 (running under VSE), Version 3 Release 1.1 (running under MVS and VM), Version 3 Release 1.2 (running under VM and VSE), and Version 3 Release 2 (running under MVS, VM, and VSE). VTAM is an IBM licensed program that can be ordered by the following program numbers:

MVS/370 (program number 5665-313)
MVS/Extended Architecture (program number 5665-289)
VM/System Product (program number 5664-280)
VSE/Advanced Functions (program number 5666-313)
VSE/System Package (program number 5666-313).

The licensed program described in this licensed document, and all licensed material available for it, are provided by IBM under terms of the Agreement for IBM Licensed Programs. Changes are made periodically to the information herein; before you use this document in connection with the operation of IBM systems, consult the latest *IBM System/370, 30XX and 4300 Processors Bibliography*, GC20-0001, for the editions that are applicable and current.

Any reference to an IBM licensed program in this document is not intended to state or imply that only IBM's program may be used.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not constitute or imply a grant of any license or immunity under any patents, patent applications, trademarks, copyrights, or other similar rights of IBM or of any third party, or any right to refer to IBM in any advertising or other promotional or marketing activities. IBM assumes no responsibility for any infringement of patents or other rights that may result from use of the subject matter disclosed in this document or for the manufacture, use, lease, or sale of machines or programs described herein.

Licenses under IBM's utility patents are available on reasonable and nondiscriminatory terms and conditions. IBM does not grant licenses under its appearance design patents. Inquiries relative to licensing should be directed in writing to the IBM Director of Commercial Relations, International Business Machines Corporation, Armonk, New York, 10504.

It is possible that this material may contain reference to, or information about, IBM products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such products, programs, or services in your country.

Books are not stocked at the address given below. If you want more IBM books, ask your IBM representative or write to the IBM branch office serving your locality.

A form for your comments is provided at the back of this document. If the form has been removed, you may address comments to IBM Corporation, Department E15, P.O. Box 12195, Research Triangle Park, North Carolina 27709, U.S.A. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

About This Book

This book is intended to help customers diagnose VTAM problems. It contains diagnostic procedures and information, which must not be used as programming interfaces. Any references to programming interfaces are for diagnostic purposes only.

Who Should Use This Book

System programmers should use this book to analyze a VTAM problem, classify it as a specific type, and then describe it to an IBM Support Center. You should be familiar with VTAM's service aids, and you should know how to report problems to the IBM Support Center. You should also be familiar with the information in the following books:

- *Network Program Products: General Information*, GC30-3350
- *Programming System General Information Manual*, G229-2228.

How To Use This Book

Follow these steps to diagnose a problem:

1. Determine which licensed program is causing the problem in your network. You can get help with this step from *Network Program Products Problem Determination* (SC30-3250), from your branch office, or from the IBM Support Center (Level 1).
   - If the problem is in VTAM, TSO/VTAM, or VSCS, you are using the right book.
   - If the problem is in the hardware or in some other licensed program, you are using the wrong book. You should be using the *Diagnosis Guide* for that product.
2. Use Chapter 1, "Documentation to Collect for All Problem Types" to collect the information you will need to report to IBM for all problems, regardless of type.
3. Use external symptoms to determine the type of problem. Chapter 2, "Classifying the Problem's Symptoms" describes these symptoms.
4. Follow the procedure for your type of problem to find its source (if possible), and to continue collecting the documentation that the IBM Support Center needs to solve the problem.
   - Chapter 3, "Procedures for Specific Problem Types" gives diagnosis procedures for VTAM problems.
   - Chapter 4, "MVS Procedures for for TSO/VTAM Problems" gives procedures for TSO/VTAM problems.
   - Chapter 5, "VM Procedures for VSCS Problems" gives procedures for VSCS problems.

   Chapters 6, 7, and 8 describe the VTAM service aids you can use to collect the problem documentation you need. Go to these chapters to collect output from traces and dumps.

   The appendixes contain information you may need to refer to when you are diagnosing problems.

Note: Appendix C, "Network Flows" describes aspects of the internal processing of VTAM. The information in that appendix is provided for problem diagnosis purposes only. Customers are strongly discouraged against using the information in that appendix for purposes other than those for which it is intended.

Notes:

a. You may need to see *VTAM Operation* for the exact syntax for some commands.

b. Operating system books can assist you in getting output from operating system service aids.

c. If you have access to a software support data base, you can search for your problem in that data base and apply the recommended correction, if there is one.

5. After collecting the necessary documentation, report the problem to the IBM Support Center.

   You can find information on how to work with the IBM Support Center, and how to carry out their suggestions, in the *Programming System General Information Manual*, G229-2228.

6. If the problem has not been reported before, Level 1 may forward the problem to Level 2. Keep all of your supporting documentation on hand to help the Level 2 representative solve your problem.

Technical terms used in this book are defined in the glossary, beginning on page /GLOSS/.

Symbols Used in This Book

The following symbols are used in this book to indicate information that pertains to a specific operating system:

MVS/XA Indicates information that applies to MVS/XA only.

MVS/370 Indicates information that applies to MVS/370 only.

MVS Indicates information that applies to both MVS/XA and MVS/370, but not to VM or VSE.

VM Indicates information that applies to VM only.

VSE Indicates information that applies to VSE only.

Note: Because VSE/Advanced Functions is one of the licensed programs in VSE/System Package, no distinction is made between the two.

These symbols precede unique information. If a piece of information applies to more than one operating system, but not to all of them, more than one symbol may precede the information. For example:

> MVS VSE *This function locates the resource in this network or another network associated with a given symbolic name or network address.*

The above information applies to MVS/XA, MVS/370, and VSE.

Information that applies to *all* operating systems is *not* denoted by a symbol.

Artwork Used in This Book

Figure 1 shows the conventions used in this book to illustrate the parts of a network.

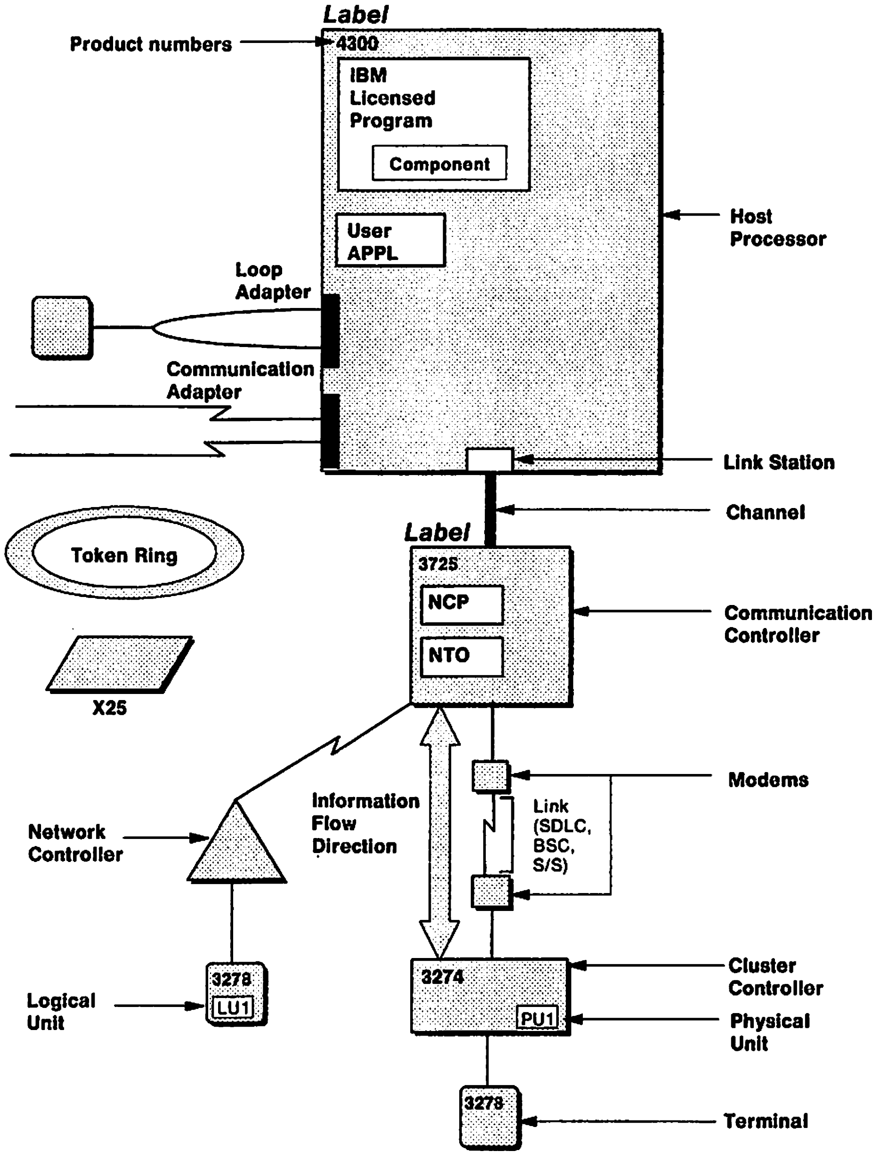


Figure 1. Conventions Used in Network Illustrations

What Is New in This Book (Fifth Edition)

For MVS:

- LU 6.2 support has been added to this book, including new VTAM internal trace records and dump records for LU 6.2.
- A new controller, the 3720 Communication Controller with an attached disk, allows NCP, MOSS, and CSP dumps to be transferred to the disk and transferred back to the host when convenient. 3720 disk support is available as a PTF in V3R1.1 and as part of the base code in V3R2.
- A new controller, the 3745 Communication Controller, allows bus-switching between central control units (CCUs) for twin-CCU 3745s, and allows the recording of 3745 network management vector transport (NMVT) alerts in LOGREC. 3745 support is available as a PTF in V3R1.1 (the 3720 disk support PTF is a prerequisite) and as part of the base code in V3R2.
- The tracing of CPWAIT, CPPOST, CPPURGE, DISP, SCHD, and QUE trace records has been streamlined.
- The SSCP and PSS VTAM internal trace options have each been split into two options (SSCP and NRM, and PSS and ESC).
- Information about VTAM V3R1 for MVS has been deleted. For information about VTAM V3R1 MVS, see the previous edition of the *VTAM Diagnosis Guide* (SC23-0116).

For VM:

- LU 6.2 support has been added to this book, including new VTAM internal trace records and dump records for LU 6.2.
- Local area network (LAN) support has been added to this book, including new VTAM internal trace records.
- X.25 support has been added to this book, including new VTAM internal trace records.
- A new controller, the 3720 Communication Controller with an attached disk, allows NCP, MOSS, and CSP dumps to be transferred to the disk and transferred back to the host when convenient. 3720 disk support is available as a PTF in V3R1.1 and as part of the base code in V3R1.2 and V3R2.
- A new controller, the 3745 Communication Controller, allows bus-switching between central control units (CCUs) for twin-CCU 3745s, and allows the recording of 3745 network management vector transport (NMVT) alerts in LOGREC. 3745 support is available as part of the base code in V3R2.
- New VTAM exit data trace entries have been added — DTIVRECX, DTIVRESX, and DTIVSEND.
- The tracing of CPWAIT, CPPOST, CPPURGE, DISP, SCHD, and QUE trace records has been streamlined.
- The SSCP and PSS VTAM internal trace options have each been split into two options (SSCP and NRM, and PSS and ESC).
- The VSCS DATA, FRE, and GET trace options have been added.

- New problem diagnosis procedures for VSCS have been added, including:
  - General procedures to diagnose logical unit hang problems (page 94)
  - Procedures for determining device characteristics (page 97)
  - Procedures for Resolving VM/SP logo problems on an SNA device (page 113).
- VTAM V3R1.2 and later can now use the alert function to notify an operator of an actual or impending loss of availability of a resource. VTAM builds a network management vector transport (NMVT) for the alert. The NMVT can be traced with the PIU option of the VIT.
- Revisions have been made to VSCS storage management to allow the swapping and releasing of VSCS dynamic storage.
- Information about VTAM V3R1 for VM has been deleted. For information about VTAM V3R1 VM, see the previous edition of the *VTAM Diagnosis Guide* (SC23-0116).

For VSE:

- Local area network (LAN) support has been added to this book, including new VTAM internal trace records.
- X.25 support has been added to this book, including new VTAM internal trace records.
- A new controller, the 3720 Communication Controller with an attached disk, allows NCP, MOSS, and CSP dumps to be transferred to the disk and transferred back to the host when convenient. 3720 disk support is available as part of the base code in V3R2.
- A new controller, the 3745 Communication Controller, allows bus-switching between central control units (CCUs) for twin-CCU 3745s, and allows the recording of 3745 network management vector transport (NMVT) alerts in LOGREC. 3745 support is available as part of the base code in V3R2.
- The format of the message numbers has been changed so that message numbers now begin with "IST" instead of the number "5".
- The tracing of CPWAIT, CPPOST, CPPURGE, DISP, SCHD, and QUE trace records has been streamlined.
- The DSCD, NRSP, OPER, and RCE entries for the VTAM internal trace have been included.
- The SSCP and PSS VTAM internal trace options have each been split into two options (SSCP and NRM, and PSS and ESC).

Where To Find More Information

Figure 2 on page xviii shows the books in the VTAM V3R2 library, arranged according to related tasks. For a description of these books, see the "Bibliography" on page 671. The bibliography also lists the titles and order numbers of books related to this book or cited by name in this book.

Evaluation and Education

Network Program Products
General Information
GC30-3350

Network Program Products
Bibliography and
Master Index
GC30-3353

Planning

Network Program Products
Planning
SC30-3351

Network Program Products
Storage Estimates
SC30-3403

Installation and Resource Definition

VTAM
Installation and
Resource Definition
SC23-0111

Network Program Products
Samples
SC30-3352

Customization

VTAM
Customization
LY30-5614

Operation

VTAM
Operation
SC23-0113

VTAM
Messages and Codes
SC23-0114

Diagnosis

VTAM
Diagnosis
LY30-5601

VTAM Data Areas
for MVS, LY30-5592
for VM, LY30-5593
for VSE, LY30-5594

Writing Application Programs

VTAM
Programming
SC23-0115

VTAM
Programming for LU 6.2
SC30-3400

VTAM
Directory of Programming
Interfaces for Customers
GC31-6403

(For Reference Summary)

VTAM
Reference Summary
LY30-5600

Figure 2. The VTAM V3R2 Library

Chapter 1. Documentation to Collect for All Problem Types

Chapter 1. Documentation to Collect for All Problem Types

The following chart describes documentation you should have for all types of problems.

Note: Documentation for the NetView™ program[1] is included in the chart.

Documentation	Description
Component ID	**V3R1** **Component** **ID** VTAM (VSE) 5666-31301-H50 **V3R1.1** **Component** **ID** VTAM (MVS/370) 5665-31301-112 VTAM (MVS/XA) 5665-28901-113 VTAM (VM) 5664-28001-311 **V3R1.2** **Component** **ID** VTAM (VM) 5664-28001-312 VTAM (VSE) 5666-31301-H52 **V3R2** **Component** **ID** VTAM (MVS/370) 5665-31301-204 VTAM (MVS/XA) 5665-28901-205 VTAM (VM) 5664-28001-320 VTAM (VSE) 5666-31301-G69 Information about the release level of each component is contained in an Access Method Support Vector List pointed to by the access method control block (ACB). See *VTAM Programming* for more information about the ACB.
Program Update Tape (PUT) and Program Temporary Fix (PTF)	Supply a list of any PUTs and PTFs that have been applied to your system. Also, supply a list of changes that have been applied to the hardware, such as REAs and ECs. If you have identified a module as the source of the problem, supply the PTF *eyecatcher* if the module has one. The PTF eyecatcher is the latest PTF number that has been applied to a module. It follows the module ID in a dump.

[1] NetView is a trademark of International Business Machines Corporation.

Documentation	Description
List of Changes to the Network Configuration	List any application programs, new devices, or new levels of the operating system you have added.
Version/Release Number	VTAM V3R1, V3R1.1, V3R1.2, or V3R2
System Console/ Hard-Copy Log	Shows all messages sent to or commands received from the operator. May provide clues as to when the system began to have problems. (VTAM problems may not be apparent at the time they occur.)
NetView or NCCF Hard-Copy Log (if using NetView or NCCF)	Contains messages routed to NetView or to NCCF that are associated with an operator terminal.
NetView or NLDM File (if using NetView or NLDM)	Contains session trace data for specified resources and session awareness data for all active sessions.
Application Program Log (if appropriate)	Some user-written operator application programs produce an application program log.
VTAM Definition Library	This is a set of definition statements for resources in the VTAM network, such as the application programs and network nodes. The VTAM definition library also contains the start options used to initialize VTAM, unless they were entered by the system operator. Detailed information about the VTAM definition library is in *VTAM Installation and Resource Definition*.
MVS VM LOGREC	This is a data set (MVS) or file (VM) that contains records of various types of system failures, both hardware and software. For hardware failures, LOGREC entries contain sense and status information about the device causing the failure. For software failures, LOGREC entries contain information such as the PSW, abend code, failing module name (when possible), and the general registers at the time of failure. LOGREC entries are written each time VTAM produces a supervisor call (SVC) dump.
VSE SYSREC	The VSE recorder file (SYSREC) contains error records for devices owned by VTAM. SYSREC records are printed with the Environmental Recording. Editing, and Printing (EREP) program.

Documentation	Description
Link-Edit Map	MVS If a VTAM load module is involved in a problem, an XREF map of the load module is needed to show the location of other VTAM modules within that load module. To get an XREF map, use the service aid LIST (AMBLIST) with the control statement LISTLOAD and the parameter OUTPUT=XREF. This produces a listing showing the module (CSECT) names and their location within the load module. Specific information on how to use the LIST service aid can be found in the *Service Aids* book for your operating system. VM Save the VTAM LKEDIT, VSCS LKEDIT, and VTAMMAP files that are produced by the installation procedure for VTAM and VSCS. VSE Most VTAM modules are link-edited into multiple CSECT phases. In some cases, it may be necessary to determine where within the phase a failing module is loaded. To do this you must have the link-edit map for the phase. Save the output from the initial installation steps for VTAM, as well as Maintain System History Program (MSHP) linkage editor output from the application of maintenance to the system.
Symptom String	Some VTAM routines provide a symptom string after a failure. MVS VM The symptom string is put in the system diagnostic work area (SDWA), which is printed by EREP as part of the LOGREC entries. For more information on LOGREC, see: MVS *SYS1.LOGREC Error Recording* VM *OLTSEP and Error Recording Guide.* VSE The symptom string is put in the first records at the beginning of the SVC dump, usually on the first page.
MVS Link Pack Area (LPA) Map	Contains module names and starting addresses of modules in SYS1.LPALIB. To get an LPA map, use the IBM service aid LIST (AMBLIST) with the control statement LISTLPA. Specific information on how to use the LIST service aid can be found in the *Service Aids* book for your operating system. When used with a link-edit map and a dump, an LPA map will enable you to identify a module that is found at a specific address within the link pack area.

What is the Problem Related to?

Symptoms are often related to a particular device, command, or update to the system. If you suspect this is so, you should tell the Support Center of this relationship. Here are some possible relationships and the kind of documentation you should have for each one:

APAR or PTF Number

If the problem appeared after you applied an APAR fix, supply the APAR number. If the fix was a PTF, supply the PTF number. The format of APAR numbers and PTF numbers in each operating system is shown below:

Operating System	APAR	PTF
MVS	OYnnnnn OZnnnnn	UYnnnnn UZnnnnn
VM	VMnnnnn	UVnnnnn
VSE	DYnnnnn	UDnnnnn

Device Type

If the problem is associated with the use of a particular type of terminal or other hardware unit, supply that device type (such as 3278 Model 2). If the problem is associated with a particular type of communication link, supply appropriate link characteristics, such as SDLC, BSC, SNA or non-SNA. Also, identify any recent microcode activity on the control units involved.

Operator Command

If the problem is associated with a particular VTAM operator command, supply the full command name (such as VARY). Also, note any command operand (such as INACT) or a network node type (such as CDRM) that has been definitely associated with the problem.

Terminal User Action

If the problem is associated with a particular terminal user action, such as IBMECHO, USS LOGON, or pressing the CLEAR key, describe the user action (or sequence of actions).

VTAM Application Program

If the problem is associated with a VTAM application program that is an IBM licensed program (such as CICS or TSO), supply the name of the licensed program.

Hardware Error Condition

Sometimes it is immediately apparent that a problem is related to a specific hardware error condition. The hardware error might have been detected and reported in several ways:

- By an operating system message
- By a VTAM or application program message
- By the system operator
- By a terminal user (an indicator of the error status appears in the operator information area, at the bottom of the terminal screen)
- By a VTAM buffer filling up with identical information from one device
- Through LOGREC
- Through SYSREC or RMS.

If a hardware error occurred, note the failure condition that accompanied it, such as UNIT CHECK or TIMEOUT

Note: The IBM Support Center cannot help you with hardware problems. Try to solve the problem locally. Use NetView or NPDA if you have it installed, or use the system console message, which identifies the affected subarea. If you still need help, contact your branch office.

Coding Change

The problem may have occurred after a coding change in:

- VTAM network definitions
- Macro usage
- Start options
- User-coded exit routines
- Job control statements.

If so, supply the macro, the option, and the value specified. Examples are:

- *luname* LU PACING=1 on an NCP LU definition statement
- SIMLOGON OPTCD=(ANY,Q) on an *API* macro.

Documentation for Hardware Failures

If you think your problem is related to a hardware failure, use the following tools to collect information about the hardware failure:

- SDLC link level 2 test (For more information on the LL2 test, see "Modify SDLC Link Level 2 Test" on page 271.)
- NCP intensive mode error recording (For more information on intensive mode recording, see "Modify NCP Intensive Mode Recording" on page 271 and the *NCP, SSP, and EP Diagnosis Guide*.)
- NetView or NPDA, if you use it in your system
- [illegible] LOGREC (or similar operating system facilities)
- VSE SYSREC (or similar operating system facilities).

Recommendations

For your problem to be serviced more efficiently:

- When VTAM messages are symptoms of a problem and your installation has written its own version of a VTAM message, supply the original VTAM message when you report the problem.
- When VTAM or operating system messages are symptoms of a problem, follow the recommendations in the appropriate message book.
- Use the available formatting routines for dump and trace output instead of unformatted hexadecimal output.

  MVS VM For machine-readable dumps, use the interactive problem control subsystem (IPCS), if you have it installed, to format and print the output. You can use IPCS to look at the dump online.

  You can also use the following utilities to format and print a dump:

  – [illegible] AMDPRDMP
  – [illegible] IKJEFT01
  – [illegible] PRTDUMP
  – [illegible] DOSVSDMP.

  For VTAM and NCP trace data, use one of the following to format and print the output:

  – [illegible] TAP or PRDMP
  – [illegible] TAP or TRAPRED
  – [illegible] TAP or TPRINT.

  See "Printing Trace Records" on page 223 for more information about these trace formatters.

- Keep on hand supporting documentation that may apply to most problems. This includes:
  - Listings of the VTAM tables your installation has defined, such as USS and LOGMODE
  - Listings of VTAM application programs and VTAM exit routines
  - A table of destination subareas, explicit route numbers, virtual route numbers, paths, and transmission groups
  - Outboard recorder (OBR) records and miscellaneous data records (MDRS) describing network hardware errors. (For more information on these records, see "Hardware Error Recording" on page 300.)
  - A table associating session types, COS names, and COS tables
  - A listing of VTAM components, as follows:

    MVS A System Modification Program (SMP) configuration data set (CDS) for VTAM and TSO/VTAM components (see the *SMP* book for information about SMP)

    VM Save the VTAM LKEDIT, VSCS LKEDIT, and VTAMMAP files that are produced by the installation procedure for VTAM and VSCS.

    VSE A Maintain System History Program (MSHP) RETRACE listing for VTAM components (see the *MSHP* book for information about MSHP)

- Keep a current list of the network's physical configuration.

- If you encounter problems during network activation after installing VTAM or changing the network, simplify your operating procedures to a step-by-step approach. Use appropriate trace facilities and the VTAM DISPLAY command to monitor the process.

- MVS VM If your configuration is using SNA network interconnection and you are using alias names, keep a list of the alias names defined to each name translation program.

- If you are using NetView or NLDM, collect session-related data for same-network and cross-network sessions. To identify the first occurrence of a problem, NetView or NLDM should be running all the time.

  NetView and NLDM collect two kinds of session data: session awareness data and session trace data.

  NetView and NLDM collect session awareness data for all active sessions. Session awareness data includes:

  - Session type
  - Names of session partners
  - Session activation status
  - IDs of subarea physical units contained in the explicit route assigned to the session
  - Transmission group numbers
  - Addresses and network IDs of SSCPs that own links in the transmission groups.

Session trace data is recorded only for sessions with a resource for which a session monitor or NLDM session trace has been started. It includes:

- Session activation parameters
- VTAM PIU data
- NCP data.

Chapter 2. Classifying the Problem's Symptoms

Chapter 2. Classifying the Problem's Symptoms

This book contains diagnosis procedures for **VTAM**, **TSO/VTAM** (**MVS** only), and **VSCS** (**VM** only). For other types of problems (such as operating system, NCP, application program, and hardware problems) see the list of appropriate books in "Bibliography" on page 671.

If any of the following caused your problem, this book may not help you:

- Did you modify an application program that has run without problems in the past?
- Did you change the processing environment? For example, did you introduce a new host processor or communication controller?
- Did you modify the operating system, or did you install a new release of the operating system?
- Did you add a new terminal to your VTAM network that had incorrect features or incorrect Request for Engineering Activity (REA) and Engineering Change (EC) level?
- Did attaching a link cause the problem?
- Did setting switches at a terminal cause the problem?
- Did initializing link parameters for a programmable controller cause the problem?
- Did you modify installation-provided VTAM tables? For example, did you modify LOGMODE, COS, or USS tables?
- Did modifying VTAM messages cause the problem?

This book does not address problems caused by these user activities. You need to resolve the problem on your own.

Note: If you cannot resolve the problem on your own, and if one of the user actions listed above resulted in one of the symptoms listed in "Symptom Listing" on page 14, follow the appropriate instructions in Chapter 3, "Procedures for Specific Problem Types" on page 19 to gather additional information, then call the IBM Support Center for assistance.

Symptom Listing

Find the symptom you are experiencing in the following list. The symptoms are listed alphabetically. Match your symptom to the appropriate VTAM problem type, and turn to the page indicated. If none of the symptoms matches yours exactly, choose the one that is most similar.

Symptom	Problem Type	See
Abend message	Abend	Page 19
Activating network nodes takes too long	Performance	Page 62
Application program cannot terminate	Wait	Page 29
Application programs and terminals cannot communicate	Wait [L]	Page 29
Application program reports an unexpected return or sense code	Incorrect Output [M]	Page 57
Batch application program fails to complete	Wait	Page 29
Book contains information that is wrong or ambiguous	Documentation	Page 66
Book is missing information	Documentation	Page 66
Books contradict each other	Documentation	Page 66
Command is not completed	Wait [IO]	Page 29
Commands cannot be entered on system console	Loop	Page 46
Commands take too long to complete	Performance	Page 62
Cursor is in the wrong position	Incorrect Output [DEF]	Page 57
Deactivating network nodes takes too long	Performance	Page 62
DTI error message	Message	Page 53
DTI message is wrong or formatted improperly	Message [IO]	Page 53
Dump file contains a VTAM, TSO/VTAM, or VSCS dump	Abend or Program Check	Page 19 Page 25
Error message	Message	Page 53
X'FFF', X'FFB', or X'FED' appears	Program Check [IO]	Page 25
Hung session	Incorrect Output	Page 57

[M] This may also be a *message* problem.

[L] This may also be a *loop* problem.

[IO] This may also be *incorrect output*.

[H] This may also be a *hardware problem*.

[DEF] This is probably an application program or VTAM definition error, such as using an incorrect LOGMODE definition.

Symptom	Problem Type	See
Hung system	Wait[L]	Page 29
IKT error message	Message	Page 53
IKT message is wrong or formatted improperly	Message[IO]	Page 53
Initialization problem (VSCS only)	Chapter 5	Page 93
Internal error message (VSCS only)	Chapter 5	Page 93
IST error message	Message	Page 53
IST message is wrong or formatted improperly	Message[IO]	Page 53
IST252I message appears	Program Check	Page 25
IST413I message appears	Abend	Page 19
Keyboard locks unexpectedly	Incorrect Output	Page 57
Logo problem	Incorrect Output	Page 57
LOGON takes too long to complete	Performance	Page 62
LOGREC entries indicate an abend	Abend	Page 19
LOGREC fills with repeated entries	Loop[H]	Page 46
Message is wrong or formatted incorrectly	Message[IO]	Page 53
Message from application program	Incorrect Output[M]	Page 57
Message is sent to the wrong console	Incorrect Output	Page 57
Message repeats continuously	Loop	Page 46
Message text does not explain a condition	Message	Page 53
Message is missing text	Message[IO]	Page 53
Output data is formatted incorrectly	Incorrect Output[DEF]	Page 57
Performance is degraded after a network outage	Performance	Page 62
Printers stop	Loop	Page 46
Program check message	Program Check	Page 25
PSWs point to a VTAM address	Loop	Page 46
Response time is slow	Performance	Page 62
Routing information is wrong	Incorrect Output	Page 57
SYSREC entries indicate a program check	Program Check	Page 25
SYSREC fills with repeated entries	Loop[H]	Page 46
System functions stop	Loop	Page 46
System light is on, Wait light is off	Loop	Page 46
Tapes stop	Loop	Page 46

Symptom	Problem Type	See
Terminal user cannot enter data	Incorrect Output	Page 57
Terminal user cannot log on or off	Incorrect Output	Page 57
Terminal user gets unexpected response	Incorrect Output ^{DEF}	Page 57
Terminal user reports incorrect or missing data	Incorrect Output ^{DEF}	Page 57
Termination problem (VSCS only)	Chapter 5	Page 93
Traffic ceases through a network component (BSC link, SDLC link, communication controller, control unit)	Wait	Page 29
VTAM does not work as described in a book	Documentation	Page 66
VTAM is not communicating with system console	Wait ^{L}	Page 29
VTAM process issues an error message	Message	Page 53
"5" error message	Message	Page 53
"5" message is wrong or formatted improperly	Message ^{IO}	Page 53
5C52I message appears	Program Check	Page 25

Chapter 3. Procedures for Specific Problem Types

Chapter 3. Procedures for Specific Problem Types

Now that you have classified your problem as a specific type, this chapter shows you how to collect the additional information you need when you talk with the IBM Support Center.

Turn to the section that describes your particular problem type:

- "MVS VM Abnormal End Procedure"
- "VSE Program Check Procedure" on page 25
- "Wait Procedure" on page 29
- "Loop Procedure" on page 46
- "Message Procedure" on page 53
- "Incorrect Output Procedure" on page 57
- "Performance Procedure" on page 62
- "Documentation Procedure" on page 66.

There are two other sections at the end of the chapter:

- "Failing Module Procedure" on page 67 tells what to do when you have isolated the problem to a specific VTAM module. You may be sent to this section from within the procedure for the problem type you have chosen.
- "Reporting Procedure" on page 68 describes briefly how to report the problem to the IBM Support Center.

MVS VM Abnormal End Procedure

If the problem is an **abend**, use the procedure in Figure 3 on page 20 to collect the following documentation:

- Dump output
- LOGREC
- Symptom string
- Abend or system completion code
- Contents of the general registers (at the time of the abend)
- Module ID and PTF eyecatcher
- PSW (at the time of the abend)
- VIT if available at the time of the abend.

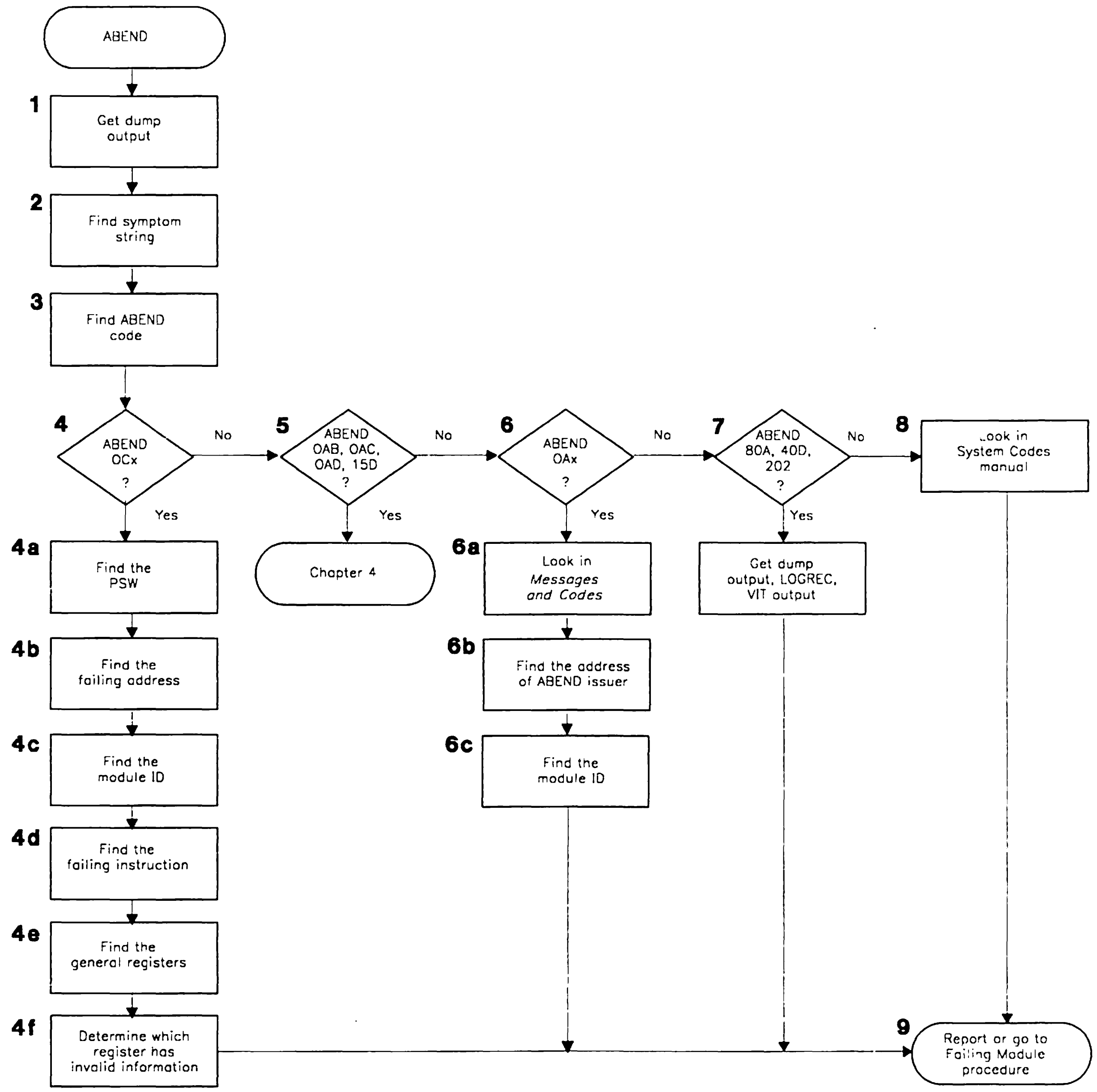

ABEND
1
Get dump output
2
Find symptom string
3
Find ABEND code
4
ABEND OCx ?
No
Yes
4a
Find the PSW
4b
Find the failing address
4c
Find the module ID
4d
Find the failing instruction
4e
Find the general registers
4f
Determine which register has invalid information
5
ABEND OAB, OAC, OAD, 15D ?
No
Yes
Chapter 4
6
ABEND OAx ?
No
Yes
6a
Look in Messages and Codes
6b
Find the address of ABEND issuer
6c
Find the module ID
7
ABEND 80A, 40D, 202 ?
No
Yes
Get dump output, LOGREC, VIT output
8
Look in System Codes manual
9
Report or go to Failing Module procedure

Figure 3. MVS VM Overview of the Abend Procedure

Detailed descriptions of the steps involved in the abend procedure begin below.

1. **Get dump output.**

   VTAM usually produces a dump for an abend. If no dump was taken, the dump files or spools may be full. Check for a message that an error occurred while VTAM was trying to produce the dump. If VTAM was not able to complete the dump, you will have to re-create the abend or wait for it to occur again.

2. **Find the symptom string.**

   A symptom string is a structured data base search argument. The symptom string gives information about what was happening at the time of the abend. A record is written to the LOGREC data set whenever VTAM takes a supervisor call (SVC) dump. For about 85% of all abends, recovery routines produce a symptom string subset, which you can get by printing LOGREC. The symptom string subset, if it occurs, is located in control block SDWA in LOGREC.

   MVS The SDWA address should be listed in the beginning of the dump, in the dump abstract information. It is also printed out in LOGREC, labelled "Hex Dump of Record" at the end of each software entry.

   The symptom string begins at X'194' in the SDWA. Field SDWAURAL gives the length of the symptom string, which can be up to 256 bytes.

   VM There are three ways to find the symptom string:

   a. If VIT is active, locate the ABND SNAP trace record. This shows the address of the SDWA.

   b. If VIT is not active, scan the registers shown at the beginning of a dump for the address of the SDWA.

      The SDWA can be difficult to recognize. However, the abend code is at + X'4', so knowing the abend code before you start looking for the SDWA can help.

   c. If neither the trace nor the dump is available, scan the software error portion of LOGREC for the symptom string.

   Figure 4 shows an example of a symptom string subset. This example is from the comments section, to the right of the SDWA.

```
AB/S00C4  PIDS/566528901  LVLS/113  RIDS/ISTCFCWM
RIDS/ISTAPCES#R  RIDS/ISTCFCR2#L  REGS/GR14  FLDS/PSW  ADRS/000006C4
VALU/HB0044770  FLDS/POWPSSQ  ADRS/027F9CB0  REGS/0C6A2  REGS/0A018
VALU/HE0044770  PIDF/566528901  PTFF/UY09850  PRCS/00000010
```

Figure 4. Example of a Symptom String Subset in Output

   The meanings of the fields are given below, in the order in which they occur in the example.

   AB Abend interrupt code, such as 0C4.

   PIDS VTAM component ID.

   LVLS VTAM version and release level.

   RIDS One of three kinds of modules:

   - Recovery module, if followed by .R
   - Load module, if followed by .L
   - CSECT name of the failing VTAM module, if not followed by anything.

   REGS The first byte shows the register number, and the next 1.5 bytes show the displacement. The displacement value is the difference between the value of the PSW I-Address and the content of the register. Each REGS field is shown only if the value is less than the PSW, and if the difference is less than 4K.

FLDS	Fields, control blocks, and DSECTs labels
ADRS	Offset into the failing VTAM module.
VALU	Field value or overlay length
PIDF	Software program component ID
PTFF	VTAM service level.
PRCS	Return code

3. **Find the abend or system completion code.**

   MVS You can find the system completion code in the output of several different service aids. The MVS system control block RTM2 work area (RTM2WA), the SYS1.LOGREC software record, and the Task Control Block (TCBCMPC) contain the completion code. The RTM2WA is pointed to by the TCB of the failing task (field TCBRTWA), and is listed after the abending TCB.

   VM You can find the abend code, ABNCODE, in the abend work area, ABNWA. Low core X'298' points to the ABNWA. You can also find the system completion code in the SDWA in a dump, or you can use the TACTIVE subcommand of the IPCS DUMPSCAN command to display the task completion code.

4. **Is the abend code 0Cx?**

   If the completion code is of the form 0C*x* (where *x* = the program interruption code from the PSW), continue with this step. If the abend code is not 0Cx, go to step 6 on page 23.

   a. Find the **program status word (PSW)** at the time of the abend.

      The PSW is found in:

      - MVS The LOGREC output, the SDWA, or the RTM2WA
      - VM The LOGREC output, the SDWA, or the ABNWA.

      The location of the PSW in the dump output varies depending on the type of dump taken. For assistance in locating the PSW in dump output, see the diagnostic books for your operating system.

   b. Find the **failing address**.

      The PSW contains either the address of the next instruction to be executed at the time of the abend or the instruction that failed at the time of the abend, depending on the interruption code. Scan the dump output to find the address given in the PSW

      If you cannot find the address, the dump may not contain the relevant portion of main storage.

   c. Find the **module ID** containing the failing address.

      VTAM identifies modules with the module name, Julian date, and PTF eyecatcher at or near the beginning of each module. This module identifier will be in the form:

      ```
      ISTxxxxx yy.ddd nnnnnnn
      ```

      where *xxxxx* is the last five characters of the module name, *yy.ddd* is the Julian date the module was assembled, and *nnnnnnn* is the latest PTF (if any) that has been applied to this module.

      Sometimes VTAM puts the module name of the failing module in LOGREC. If it is not there, you can find it in a dump.

To find the module ID in a dump, start at the failing address and scan in descending address order along the right side of the listing. The module ID is printed in EBCDIC.

MVS You can also scan the LPAMAP for the name of the module that contains the failing address.

d. Find the **failing instruction**.

   Use the address in the PSW to find the instruction that caused the abend.

e. Find the **general registers**.

   The general registers in use at the time of the abend are found in:

   - MVS The LOGREC output, the SDWA, the abend SVRB register save area, or the RTM2WA
   - VM The LOGREC output, the SDWA, the abend SVRB register save area, or the ABNWA.

   Use the diagnostic books for your operating system to help find the registers.

f. Determine which register has invalid information.

   The failing instruction often uses a register with an invalid address in one of the general registers, or will point to an invalid location (for example, low-address storage). Use the *Principles of Operation* book for your processor, the program interruption code from the SDWA (for MVS) or from the PSW (for VM), and the general registers used in the failing instruction, to determine (if possible) which register contains or points to incorrect data.

   If the interruption code is 10 or 11, then the PSW address points to the failing instruction. Otherwise, back up the PSW by the instruction length, and *that* is the failing instruction.

   Next go to step 9 on page 24.

5. MVS **Is the abend code 0AB, 0AC, 0AD, or 15D?**

   These abend codes indicate a TSO/VTAM abend. For diagnosis information, go to "MVS TSO/VTAM Abends" on page 77.

6. **Is the abend code 0Ax?**

   If the abend code is in the form 0A*x*, continue with this step. If not, go to step 7 on page 24.

   a. Find the **abend code explanation** in *VTAM Messages and Codes*.

      An abend code of 0A*x* indicates a problem within the VTAM network. The problem could have originated in VTAM, the NCP, an application program, or the hardware of some other network component. Look up the code in *VTAM Messages and Codes*. Most 0A*x* abends place a return or reason code in register 15 at the time of failure. You can find the return code in register 15 by using the set of general registers from:

      - MVS The LOGREC output, the SDWA, or the RTM2WA
      - VM The LOGREC output, the SDWA, or the ABNWA.

b. Find the **address** of the module that issued the abend, using the PSW, which points to SVC 13.

c. Find the **module ID**.

   From the address determined in the previous step, scan in descending address order through the dump to find the module ID (see step 4c on page 22).

   Go to step 9.

7. MVS **Is the abend code 80A, 40D, or 202?**

   If the **abend** code is one of these, continue with this step. Otherwise, continue with step 8.

   These abend codes indicate storage problems. Collect the following documentation:

   - A dump of the VTAM address space
   - A dump of the VTAM CSA
   - LOGREC output
   - VIT output at the time of the abend, if it is available.

   After obtaining this documentation, go to "Reporting Procedure" on page 68.

8. **Is the abend code none of the above?**

   MVS A complete list of abend codes is found in the *System Codes* book for your operating system.

   VM The abend codes are in *VM/SP System Messages and Codes*.

   Each code has an explanation of the documentation required and the problem determination steps to follow. For example, many abends occur during execution of SVC instructions. Parameter lists and register contents passed to SVC routines are in the diagnostic books for your operating system. These books may suggest that you obtain additional information such as a module name, a return code, a register containing invalid information, or the name of a system control block containing invalid parameters. After making a complete check of these sources, you are ready to report the problem; continue with step 9.

9. **Report or go to the failing module procedure.**

   If you determined the module ID, go to "Failing Module Procedure" on page 67. Otherwise, see "Reporting Procedure" on page 68.

VSE Program Check Procedure

If the problem is a **program check**, use the procedure in Figure 5 on page 26 to collect the following documentation:

- Dump of VTAM partition and supervisor
- Message information
- Contents of the general registers (at the time of the error)
- Module ID and PTF eyecatcher
- PSW (at the time of the error).

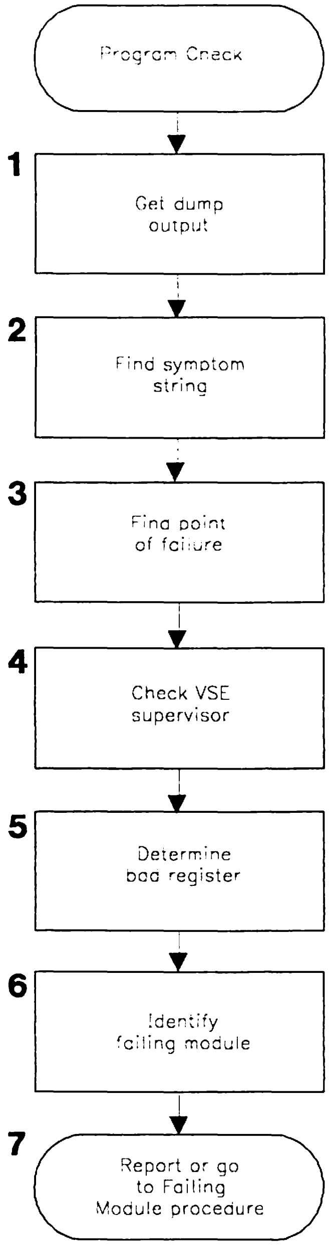


Figure 5. VSE Overview of the Program Check Procedure

Detailed descriptions of the steps involved in the program check procedure begin below.

1. **Get dump output**.

   In the hard wait cases, use the formatting option to print the VSE supervisor areas and the VTAM partition. Since VTAM programs are not executed in the shared virtual area (SVA), it is not usually necessary to print the SVA for VTAM problems.

   If the dump was produced by the VSE terminator upon cancelation of VTAM, it will produce a dump of both the VSE supervisor and the VTAM partition (if // OPTION DUMP,SYSDMP was specified in the VTAM procedure). A dump produced using // OPTION PARTDUMP will not contain all of the VSE supervisor areas used in performing VTAM functions.

See the *VSE Service Aids* book for more information on obtaining and formatting VSE dumps.

2. **Find the symptom string.**

   A symptom string is a structured data base search argument. The symptom string gives information about what was happening at the time of the program check. A record is written to the SYSREC file whenever VTAM takes an SVC dump. For about 85% of all program checks, recovery routines produce a symptom string subset, which you can get by printing SYSREC.

   Recovery routines produce the symptom string if storage is available for it. The symptom string subset, if it occurs, is located in the first records of the SVC dump output, usually on the first page. Figure 4 on page 21 shows an example of a symptom string subset in dump output.

3. **Locate the point of failure.**

   a. If the program check was reported by message 0S03I, the exception type and location are contained in the message text.

   b. For hard wait states (X'FFF' and X'FFB'), the failing instruction address can be determined from the program check old PSW (see the *VSE Service Aids* book for the location of the PSW in DOSVSDMP format).

   c. For task termination messages other than 0S03I, find the save area containing the error PSW and registers (in the beginning of the task termination dump).

   d. For message IST252I (or 5C52I, for VTAM V3R1 or V3R1.2) with return code 36, or VSE message 0V08I, the canceled task is VTAMRP, but the save area used is not the normal subtask save area. Instead, the VSE program check handler stores the PSW and registers at label SVPCSAVE.

   e. For message IST252I (or 5C52I, for VTAM V3R1 or V3R1.2) with return code 20, the VSE dump of VTAM will be taken under the main task. Locate the VTAMRP save area either by finding its TCB in the VSE supervisor dump or by using location 10 to find the ATCVT and field ATCFLDA to locate module ISTINCF0, which contains the save area.

      Identify the save area by the name VTAMRP in the first 8 bytes. This is the save area of the routine that generated the dump.

      The save area identified by the JOBNAME in the next 8 bytes contains the PSW and registers of the failing task.

   f. For wait state X'FED', use the address in register 5 as a failure address.

4. **For WAITFFF and WAITFFB, determine if the failure is in the VSE supervisor.**

   Though the cause of the problem may not lie in VTAM, there are several things to consider in making that determination.

   a. Check the VSE supervisor listing and include the **label** preceding the point of error in your information for the Support Center. If the logic pertains to a VTAM function, the problem may be in VTAM.

   b. If the instructions have been overlaid by data, try to identify that data from its contents, and note where the overlay begins and its length. If the overlay contains terminal data or a recognizable VTAM control block, the problem is probably in VTAM. Save any pertinent information about the overlay to discuss with the IBM Support Center. If the overlay is determined to be a VTAM control block, the name should be given to the Support Center.

c. If the last interruption was a VTAM **SVC 49 or SVC 53**, add that to the information in the form SVCxx.

   If the last interruption was some other VSE SVC issued by VTAM, then VTAM may be issuing the SVC incorrectly. Add that information in the form SVCxx.

   Proceed to step 5, using the address in the SVC old PSW as the failure address.

d. If the last interruption was an **I/O interruption** from a VTAM-owned channel-attached device, the problem may involve one of the VTAM appendage routines.

e. For the **FFB condition**, try to determine what was being referred to that caused the page fault. If the exception address at location 90 points to an area in the VTAM partition, the problem may be in VTAM. If the address is within a VTAM module (appears to be executable code), identify the module as in step 6. If not, try to identify it as a VTAM control block, using the list of control block IDs in Appendix E, "Storage and Control Block ID Codes" on page 657. If successful, add the control block name to your list of documentation. If not, the area should be either in a VTAM module (see step 6) or in a VTAM buffer pool (see Appendix A, "Finding VTAM Buffer Pools").

5. **Determine which register contains or points to incorrect data.**

   The failing instruction normally has an invalid address in one of the general registers, or points to an invalid location (for example, low storage). Use the *Principles of Operation* book for your processor, the program interruption code from the PSW, and general registers in the failing instruction to determine (if possible) which register contains or points to incorrect data. Add this register number string in the form REGnn to the information for the Support Center.

6. **If the failing instruction address found in step 3 on page 27 is in the VTAM partition, identify the failing VTAM module.**

   Most VTAM modules are identified by a field containing the name ISTxxxxx and the Julian date near the beginning of the module (for example, ISTACCQ4 86.175). To locate this identifier, scan the dump in descending address order from the point of failure. The identifier will be printed in EBCDIC along the right-hand side of the dump. Add the module name to your documentation list. Go to the "Failing Module Procedure" on page 67 to continue obtaining information for the Support Center.

7. **Report or go to the failing module procedure.**

   If you determined the module ID, go to the "Failing Module Procedure" on page 67. Otherwise, go to "Reporting Procedure" on page 68 for information on contacting IBM.

Wait Procedure

If the problem is a **wait**, use the procedure in Figure 6 on page 30 to collect the following documentation:

- I/O trace output
- Buffer contents trace output
- Session trace data (if using NetView or NLDM)
- Session awareness data (if using NetView or NLDM)
- Dump output:
  - MVS Dump of the VTAM address space and CSA
  - VM Dump of the VTAM virtual machine
  - VSE Dump of the VTAM partition and VSE supervisor
- List of:
  - Waiting Process Anchor Blocks (PABS)
  - Waiting Request Elements (WRES) and associated event IDS (EIDS)
  - Waiting Request Parameter Headers (RPHS)
- For problems associated with an application program:
  - Output from the VIT (all options except LOCK)
  - RPLS or FMCBS queued to the ACDEB
- For problems associated with the network:
  - Output from line trace, generalized PIU trace (GPT), transmission group (TG) trace, or (for 3720, 3725, and 3745 only) the scanner interface trace (SIT).
  - VSE Dynamic trace of communication adapter (if available)
  - NCP, MOSS, or CSP dump
  - Reports from NetView, NCCF, NPDA, IMR, or EREP (if available)

Note: Use the documentation you have available, unless you need to re-create the problem. In that case, make sure the above traces are active.

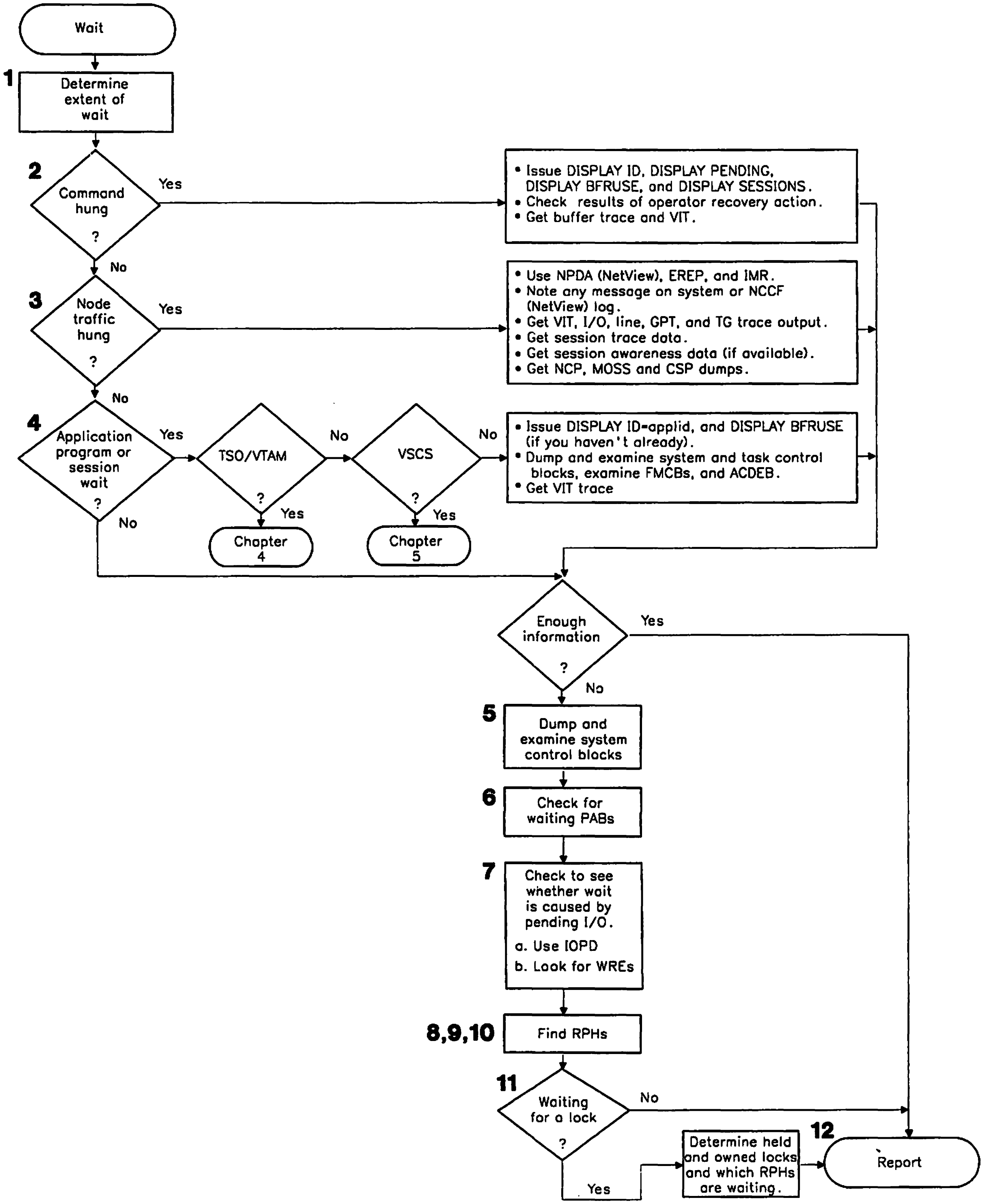

Wait
1
Determine extent of wait
2
Command hung ?
Yes
No
• Issue DISPLAY ID, DISPLAY PENDING, DISPLAY BFRUSE, and DISPLAY SESSIONS.
• Check results of operator recovery action.
• Get buffer trace and VIT.
3
Node traffic hung ?
Yes
No
• Use NPDA (NetView), EREP, and IMR.
• Note any message on system or NCCF (NetView) log.
• Get VIT, I/O, line, GPT, and TG trace output.
• Get session trace data.
• Get session awareness data (if available).
• Get NCP, MOSS and CSP dumps.
4
Application program or session wait ?
Yes
No
TSO/VTAM ?
No
Yes
Chapter 4
VSCS ?
No
Yes
Chapter 5
• Issue DISPLAY ID=applid, and DISPLAY BFRUSE (if you haven't already).
• Dump and examine system and task control blocks, examine FMCBs, and ACDEB.
• Get VIT trace
Enough information ?
Yes
No
5
Dump and examine system control blocks
6
Check for waiting PABs
7
Check to see whether wait is caused by pending I/O.
a. Use IOPD
b. Look for WREs
8,9,10
Find RPHs
11
Waiting for a lock ?
No
Yes
Determine held and owned locks and which RPHs are waiting.
12
Report

Figure 6. Overview of the Wait Procedure

Detailed descriptions of the steps involved in the wait procedure begin on page 31.

1. **Determine the extent of the wait condition.**

   Determine how extensive the wait state is in the operation of the VTAM network. Determine whether all VTAM processing stopped or only processing with respect to a single device, or something in between. Also determine what, if any, recovery action was taken at the time the wait was encountered by the operator or user. Some information about the activity that immediately preceded the wait may be available on the system log or in application program transaction logs.

2. **Did a logon, logoff, or command fail to complete?**

   If so, continue with this step; otherwise, go to step 3 on page 32.

   a. If the wait condition was actually the failure of a VTAM procedure to complete, use the **DISPLAY ID** command to identify the status of VTAM resources at the time of the problem. Note any status codes that are abnormal. (Resource status codes are explained in *VTAM Messages and Codes*.)

   b. Use the VTAM **DISPLAY PENDING, DISPLAY SESSIONS**, or **MODIFY IOPD** commands to identify I/O requests for which VTAM is awaiting a response from a network node. Sometimes a network node appears in a pending state awaiting the completion of activity at a higher- or lower-level node (for example, PSUB1, PTRM2). The pending status on the other node is needed in such a case.

   c. Use the **VTAM DISPLAY BFRUSE** command to get information about VTAM buffer pools. Save the output for use later in this procedure.

   d. A VTAM operator may have attempted a recovery action (such as issuing a **VARY INACT,FORCE** command; "Checking Whether the VARY INACT,FORCE Command Can Complete" on page 44 shows how to determine whether this command completed.) If the recovery action did not reset the state of the node for which the original command was issued, note any lack of change to the node status.

   e. If VTAM is waiting for an I/O response, look at the output of the VTAM **buffer contents trace** (assuming it was active when the problem occurred). If the trace shows that VTAM did send a request and is expecting a response, the problem is probably in another network node.

   f. You can get additional information on the status of a command from the **VTAM internal trace (VIT)**. With the SSCP and PIU options, you can match requests and responses and determine any requests that are outstanding (that is, for which responses have not been received). The SMS option supplies information about resource usage, and the PSS option provides information about VTAM'S scheduling of the dispatching process. (See Chapter 8, "Using the VTAM Internal Trace" for a description of the internal trace entries.)

   At this point you may have enough documentation to report the problem to the Support Center. If so, go to "Reporting Procedure" on page 68. Otherwise, go to step 5 on page 35.

3. **Is network traffic stopped through a specific node?**

   If so, continue with this step. Otherwise, go to step 4 on page 33.

   a. Add the **specific node type** to your problem documentation. For example, the node could be a 3705, a 3720, a 3725, a 3745, a 3790, or a 3274. NetView, NPDA, and EREP facilities show whether errors have been recorded for the node in question. Session trace data (collected by NetView or NLDM) shows whether the node is not responding to VTAM, or whether VTAM is discarding the responses. Consider using NCP intensive mode recording (IMR) for recurrent problems of this type.

   b. Note any **messages on the system** or **NetView command facility log** reporting ER-INOP outages or other failures. Use the VIT trace, or use the I/O trace with the EVERY operand, to trace the network flow up to the point of failure. NetView, NPDA, **MVS VM** LOGREC, and **VSE** SYSREC show the reason for the INOP.

   c. For NCP-related problems, use the **line trace** or **generalized PIU trace** if the affected node is in an adjacent subarea. Use the **transmission group trace** to record intermediate node flows up to the point where the problem occurred.

   d. If the problem may be in NCP software or communication controller hardware, obtain a **dump of NCP storage**. If the wait affects only part of the network, use the dynamic NCP dump facility. It allows the rest of the network to continue operating while the dump is taken. If the failure requires reactivating the NCP, use the MODIFY DUMP command. (See "Network Control Program (NCP) Dump" on page 274 and the *NCP, SSP, and EP Diagnosis Guide* for more information on NCP dumps.)

      If the NCP is hung, or the hung resource is attached downstream of an NCP, see the *NCP, SSP, and EP Diagnosis Guide* for information on hung resources.

   e. **VSE** If the problem is in a device attached to the communication adapter, use the **dynamic trace of the communication adapter** to obtain information about line-related errors. See *VSE Service Aids* for more information on this trace.

   f. If the problem is in a channel-attached device or a channel-to-channel attachment, examine one of the following traces, if available, to determine the sequence of events preceding the wait. (If no trace output is available, you will have to re-create the problem to get it.)

      - VIT trace with the CIO option
      - **MVS** CCWTRACE
      - **VSE** SDAID I/O and CCWD traces
      - **VM** CPTRAP SIO and I/O trace
      - **VM** CP virtual machine trace with I/O, SIO, CCW, and CSW.

The books listed below contain more information on operating system I/O control blocks:

- *MVS Service Aids* and *MVS Handbook*
- *VM/SP System Programmer's Guide* (for VM R4)
- *VM/SP Facilities for System Programming* (for VM R5)
- *VSE Service Aids* and *VSE Handbook.*

g. If enough information is available, go to "Reporting Procedure" on page 68. Otherwise, go to step 5 on page 35.

4. **Is it a session or application program wait?**

If the wait condition appears to be related to a particular VTAM application program, continue with this step. Otherwise, go to step 5 on page 35.

a. Enter the **DISPLAY ID** command for the application program, using the **EVERY** or **SCOPE=ALL** operand. If there are any nodes with status ACT/U, re-enter the DISPLAY command. If you are again informed that the status of a node is ACT/U, issue VARY INACT,FORCE for that node. If you still have a wait condition, continue with the next step.

b. Find out if the problem is in VTAM. If only one application program is waiting while others continue to communicate with VTAM, that application program probably contains an error. To determine what caused the problem, obtain a **dump** of the application program and the operating system supervisor at the time of the problem.

- Make sure that the error is not an operating system error. (Use the diagnostic books for your operating system.)

- If possible, use the dump to determine the reason the application program is waiting. If the application program is not waiting for VTAM, use the documentation for the application program to determine the reason for the wait. If the problem is in TSO/VTAM, go to Chapter 4, "**MVS** Procedures for TSO/VTAM Problems." If the problem is in VSCS, go to Chapter 5, "**VM** Procedures for VSCS Problems."

c. If VTAM still seems to be the cause of the problem, you will need output from the **VTAM internal trace (VIT)** to obtain a record of activity on the failing session. (If the VIT trace was not running at the time of the failure, you should re-create the problem with the VIT active.)

- If all application program sessions are hung, a small amount of trace data will be produced, and you can collect it all with MODE=INT.

- If only part of the network is affected, a large amount of trace data may be produced, and MODE=EXT is recommended.

See Chapter 8, "Using the VTAM Internal Trace" for more information on using the internal trace. You may also use the I/O or buffer contents traces with ID=*application program name* specified.

d. Find the address of the VTAM ACDEB for the application program, or the address of the VTAM APPCB for LU 6.2 applications.

You can find an ACDEB or APPCB associated with an application by using the SES(applname) option of VTAMMAP. If VTAMMAP can not be run then all ACDEBs are queued out of the ATCVT from the field ATCACDA. Sorry, there is no chain of APPCBs.

e. For LU 6.2 applications using the APPCCMD interface, go to step 4f on page 35.

   For non-LU 6.2 applications, continue with this step.

   1) Use the ACDEB's address to find it in the dump.

      On the FMCB RECEIVE ANY queue, ACDRAFQ points to the first FMCB.

      On the RPL RECEIVE ANY queue, ACDRARQ points to the first RPL.

      Notes:

      a) If there are FMCBS (ACDRAFQ ≠ 0), but no RPLS (ACDRARQ = 0), a problem has prevented the application program from issuing RECEIVES.

      b) If there are RPLS (ACDRARQ ≠ 0), but no FMCBS (ACDRAFQ = 0), there may be a problem involving the continue any/continue specific (CA/CS) state of the session. (See *VTAM Programming*.)

   2) Get the LUCB address (field ACDLUCBA in the ACDEB).

   3) Get the address of a chain of FMCB extensions (field LUCFMCBA in the LUCB). Each FMCB extension represents one LU-LU session.

   4) Each FMCB extension contains a pointer (field TSPFMCBA) to the address of an associated FMCB. Find the FMCBS associated with hung sessions.

   5) In those FMCBS, look for:

      - The CA/CS indicator (in TSPPSFL1 and TSPPSFL2)
      - The data queues (in TSPACCUM, TSPEWAIT, TSPEWORK, TSPNWAIT, TSPNWORK, TSPEDATA, TSPNDATA, TSPTSOP, and TSPTSIP)
      - Session state flags (in TSPSESSR, TSPDTSR, TSPCRVSR, and TSPRQRSR).

   6) Determine whether there are any indications of unusual conditions. (See *VTAM Data Areas* for more information on the fields in the FMCB.)

      Make a cross-reference listing of network addresses and node names to correlate the VIT PIU and I/O trace entries with VTAM session control blocks, such as the LUCB and FMCB.

   7) If enough information is available, go to "Reporting Procedure" on page 68. Otherwise, go to step 5 on page 35.

f. For the LU 6.2 applications using the APPCCMD interface:

   1) Use the APPCB's address to find it in the dump.

      On the queue of RABS in continue any (CA) mode, APPRBDAT points to the first RAB.

      On the RPL RECEIVE ANY queue, APPRANYQ points to the first RPL.

      Notes:

      a) If there are RABS (APPRBDAT ≠ 0), but no RPLS (APPRANYQ = 0), a problem has prevented the application program from issuing RECEIVES.

      b) If there are RPLS (APPRANYQ ≠ 0), but no RABS (APPRBDAT = 0), there may be a problem involving the continue any/continue specific (CA/CS) state of the conversation. (See *VTAM Programming*.)

   2) Find the chain of RABS in the APPCB (field APPRABC). Each RAB represents a conversation. The session associated with the conversation is represented by a SAB and is pointed to by RABSABPT. Find the RABS associated with hung sessions.

   3) In those RABS, look for:

      - The CA/CS indicator (in RABCMODE)
      - The data queues (in RABNDATQ and RABCONV)
      - The conversation state (RABPSFSM).

   4) Determine whether there are any indications of unusual conditions. (See *VTAM Data Areas* for more information on the fields in the RAB.)

      Make a cross-reference listing of network addresses and node names to correlate the VIT PIU and I/O trace entries with VTAM session control blocks, such as the RAB.

   5) If enough information is available, go to "Reporting Procedure" on page 68. Otherwise, continue with step 5.

   See the *NCP, SSP, and EP Diagnosis Guide* for more information on hung sessions.

5. **Examine system data areas.**

   MVS If you have not already done so, obtain a **dump** of the VTAM address space, CSA, and LSQA. Find and analyze the task control blocks. (See *Service Aids* and the *Debugging Handbook* for information on locating the task control blocks. See *Diagnostic Techniques* for information on analyzing them.)

   VM If you have not already done so, issue #CP VMDUMP, with the appropriate options, to obtain a **dump** of the VTAM virtual machine.

   VSE If you have not already done so, obtain a **dump** of the VTAM partition and VSE supervisor. Find and analyze the task control blocks. (See the *VSE Handbook* for information on locating the task control blocks. See *VSE Service Aids* for information on analyzing them.)

6. **Check for waiting PABS.**

   Look at the following PABS in the ATCVT:

ATCITPAB	Internal Trace PAB
ATCCSPAB	Configuration Services PAB
ATCVDPAB	VARY Definition PAB
ATCPXPAB	Buffer Pool Expansion PAB
ATCPUPAB	Physical Unit Services DYPAB
ATCPUIOP	Physical Unit Services I/O DYPAB
ATCLUSRT	Logical Unit Services Router DYPAB
ATCNSPAB	No Sessions PAB
ATCNOSPB	Network Operator Services PAB
ATCSSPAB	Session Serialization PAB
ATCTPMPB	TP Message PAB
ATCTRMPB	Termination Subtask DYPAB

   Check the contents of the PABWEQP and PABRPHA fields. The field PABWEQP in each PAB contains the address of a chain of work elements that have not been processed by VTAM. The field PABRPHA in each PAB contains the address of an RPH that is either running or waiting. Note the contents of these fields in each of the PABS, and have this information available when you contact IBM. Figure 7 on page 37 shows how to find each of these fields.

   Note: In some PABS, PABRPHA may contain the address of an RPH, even though the RPH is not running or waiting.

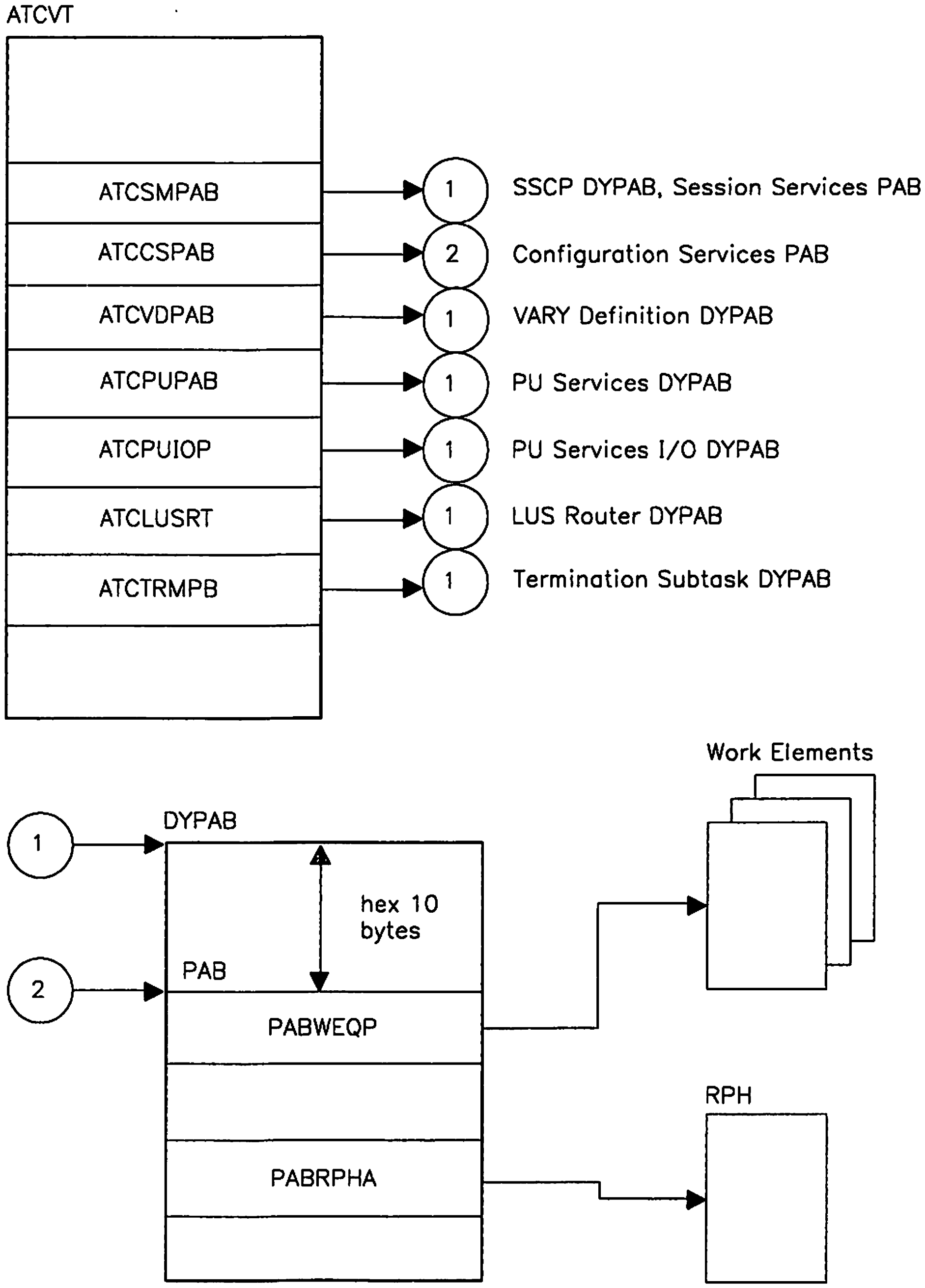


Figure 7. Finding PABs Using the ATCVT

7. **Is the wait caused by pending I/O?**

   a. Use the **Input/Output Problem Determination (IOPD) facility** to detect and report to the operator I/O operations that have been pending longer than a user-defined time limit. (For more information on using IOPD, see "Modify Input/Output Problem Determination" on page 270 and *VTAM Operation*.)

      When a VTAM process is waiting for a response, the process is represented by a waiting request element (WRE) queued to an I/O locked queue anchor block (LQAB). The WRE points to an event ID (EID), which indicates the reason for the wait. (Wait state event IDs are described in *VTAM Messages and Codes*.)

   b. **Look for the WREs and corresponding EIDs** in a dump by using Figure 8 on page 39 and the following steps:

      1) **Find the ATCVT** from low-storage address MVS X'408', VM X'200', or VSE X'10'.

         MVS If these low-address locations are not available in a dump, use the pointer in the CVT (CVTATCVT) to find the AVT. Location X'00' in the AVT points to the ATCVT.

      2) **Get the address of the SSCP I/O LQAB** in field ATCIOLQB. This is the address of the first LQAB. There is one LQAB for each subarea in the VTAM network. The LQABs are contiguous in storage. To find the LQAB for the next subarea, add X'10' to the address of the previous LQAB. To find an LQAB for a particular subarea, multiply the subarea number by X'14'. Add the result to the address in ATCIOLQB.

      3) **Scan each LQAB at offset X'00' (field LQABFRST) for a non-zero address.** This is the address of a WRE. List these addresses if any are found. If a WRE is present, look at WRE + X'04' (field WREQUE) for additional WREs queued off the same LQAB. Again list the addresses of any WREs found.

      4) **Find the waiting event.** WREs contain a code (WREIDCD) that identifies the waiting event. Use explanations of wait state event IDs in *VTAM Messages and Codes* to find the reason for the wait. The address and length of the waiting event are in the WRE in fields WREIDP and WREIDL respectively.

      5) **Find the LQAB.** For waiting events other than SSCP I/O, there is only one LQAB. Using the hex offsets in *VTAM Data Areas*, find each of these LQABs from pointers in the ATCVT:

         - ATCLUSMQ
         - ATCMCQAB
         - ATCPULQB
         - ATCNOSQ
         - ATCSSLQB

      6) **Look for WREs chained off the LQABs.** If a WRE is found, list its address and find the associated EID. Use explanations of wait state event IDs in *VTAM Messages and Codes* to find and list the reason for the wait.

7) **For each WRE found, examine the WREDATA field.** This field may contain a CPCB operation code. If so, look in Appendix D, "CPCB Operation Codes" on page 645 to determine the function represented by the operation code. Have the function name available when you report the problem. (You can also use the IOPD facility to determine the waiting event. This facility is described in "Modify Input/Output Problem Determination" on page 270.)

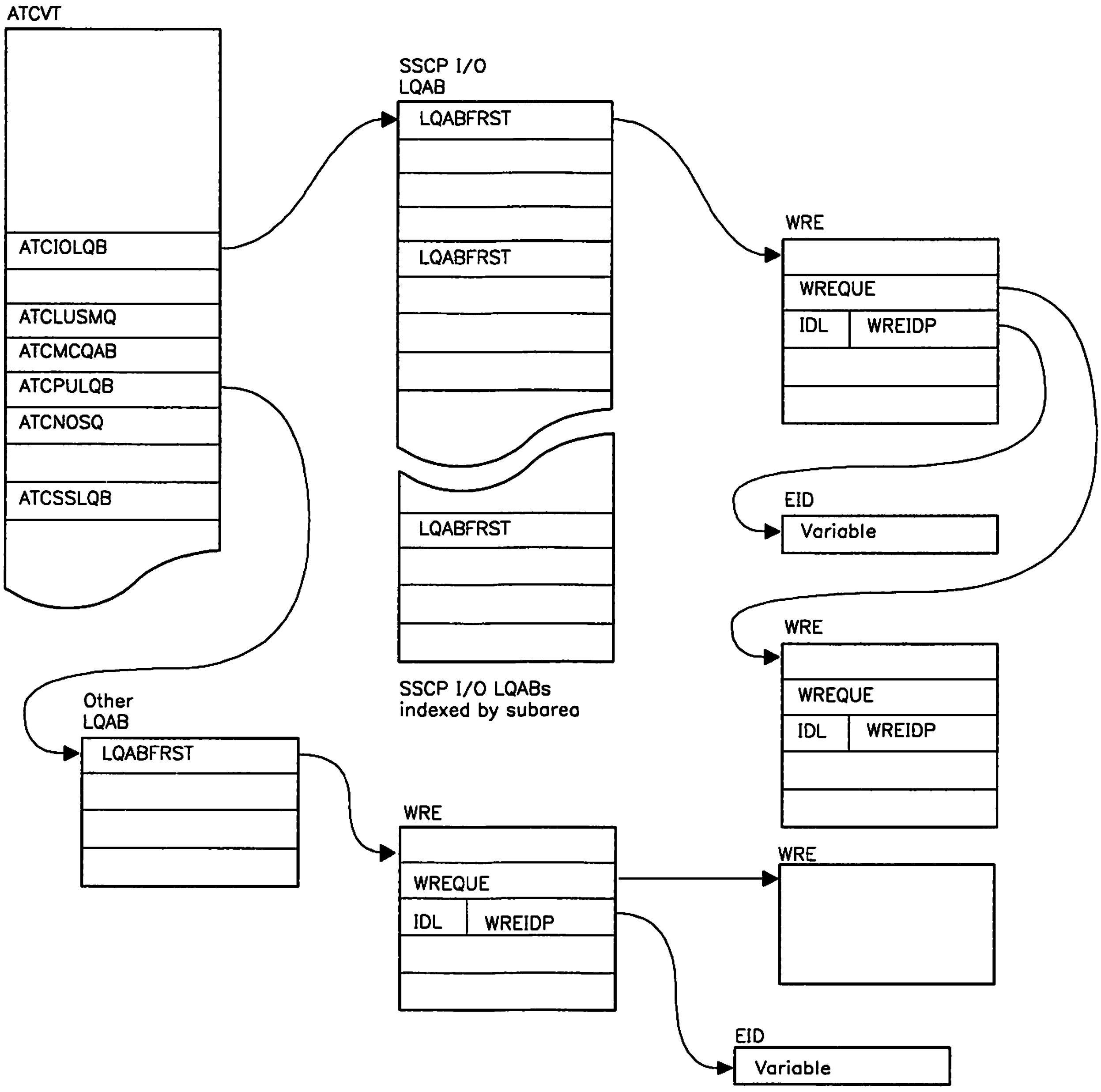


Figure 8. Waiting Request Elements

8. **Find waiting RPHS.**

   The following steps give instructions for examining two kinds of wait conditions: (1) a process is waiting for a buffer, and (2) a process is waiting for some other resource. Both kinds of waiting processes are represented by *request parameter header* (RPH) control blocks, but the RPH is found in different locations for each type of wait condition.

   - Step 9 explains how to find RPHS queued from a buffer pool control block. These RPHS show that the buffer pool cannot supply the required buffers, and as a result, the process is waiting. Note which buffer pool cannot supply the required buffers.

   - Step 10 explains how to find RPHS that indicate a waiting process. In this case, collect the documentation listed in step 10d on page 41.

9. **Find RPHS queued from buffer pool control blocks.**

   A buffer pool that has no available buffers can cause a wait state. There are many reasons for running out of buffers; for example, incorrect allocation in the VTAM start options, a VTAM programming problem, or an application programming problem. Use the DISPLAY BFRUSE output obtained in step 2 on page 31, if you were able to get it, to analyze buffer pool usage. Otherwise, use Appendix A, "Finding VTAM Buffer Pools" to find the buffer pools and determine whether any have used all their available buffers.

   Examine locations **MVS VM** BPCBRPHA and BPCBRPHB or location **VSE** PCBFRPHA in each buffer pool control block. If a non-zero address (indicating a waiting RPH) is found at these locations, list the name of the buffer pool associated with that buffer pool control block.

   Also, follow the chain at offset X'04' into the RPH to obtain the addresses of other RPHS waiting for the same pool.

10. **Find other waiting RPHS.**

    Waiting RPHS indicate a VTAM process that has not been completed. To locate the waiting RPHS, look at a formatted dump (in MVS and VM) or search the large pageable buffer pool (LPBUF) by hand.

    MVS Use the VTAMMAP dump formatter with the VTBASIC option. This formats the component recovery areas CRAS, which contain the RPHS.

    VM Use the PRTDUMP command of IPCS and the VTBASIC option. This formats the component recovery areas CRAS in a VM environment.

    If you cannot obtain a formatted dump, use Appendix A, "Finding VTAM Buffer Pools" to find LPBUF, and use steps 10a, 10b, and 10c to find the waiting RPHS.

    a. Find each allocated buffer in the LP00 buffer pool. The LP buffer is allocated if the high-order bit of the second word is on. Scan each allocated buffer for a waiting RPH. The buffer contains an RPH if the value **MVS VM** X'016C' or **VSE** X'0168' is shown at offset X'10' into the buffer (offset X'00' into the RPH). When you see this value, look at location X'10' into the RPH. A valid address at this location (not 1 or FFFFFFFF) indicates a waiting RPH. This valid address is the RPH *resume address* field (RPHRESUM), the location at which a VTAM routine will begin executing when posted. Write down the RPH address and the resume address.

b. Find the remaining buffers in the buffer pool. Look for a waiting RPH in each buffer as in the previous step. Continue with this process until the end of the buffer pool. If you find any more waiting RPHS, add the RPH address and the resume address to your documentation list.

c. Next look for any RPHS in the LPBUF expansion areas. (Appendix A, "Finding VTAM Buffer Pools" explains how to find the expansion areas.) If there are no expansion areas, go to step 10d. Otherwise, scan each expansion area for waiting RPHS as in steps 10a and 10b. Continue adding waiting RPH addresses and resume addresses to the list until you have examined all of the expansion areas.

d. For each waiting RPH, collect the following information:

- Address of the RPH
- Resume address associated with the RPH
- Name of the module that contains the resume address
- Starting address of the module in the dump
- PTF number (if it exists) or the Julian date of the module
- Any locks that this RPH is waiting for (step 11).

To find the name of the module that contains the resume address, start at the resume address and scan in descending address order (upwards) along the right side of the dump listing. The module ID (including the Julian date and the PTF eyecatcher) is printed in EBCDIC.

11. **Find RPHS waiting for locks.**

a. For each waiting RPH, look at the CRALXPTR fields. If any pointer (PTR) fields are non-zero, check the corresponding bit in CRALKACT. For example:

- If CRAL1PTR is non-zero, look at the last bit in CRALKACT.
- If CRAL2PTR is non-zero, look at the next-to-last bit in CRALKACT.
- If CRAL3PTR is non-zero, look at the third-from-last bit in CRALKACT.

If the corresponding bit in CRALKACT is off (zero), the RPH is waiting for this lock. If the bit is on (non-zero), the RPH is holding the lock and may be waiting for another lock. On your list of waiting RPHS, add the name of the lock being held or waited for. (See Figure 9 on page 42.)

b. If step 11a does not succeed, scan the LP00 buffer pool again, this time listing all allocated buffers that contain a non-zero value in field CRALKACT. This tells which RPHS own locks, if any, and which locks are held. It is possible for a CRA to hold several locks. For example, a value of 06 indicates two locks being held: the RDTLOCK (04) and the VOCLOCK (02). (See Figure 9 on page 42.)

For each allocated buffer with a non-zero CRALKACT field, look at the CRALXPTR fields. (The buffer may or may not contain a resume address.) A non-zero pointer field contains a lockword address. Find the lockword. The first word of the lockword shows a queue of RPHS waiting for that lock. Add these RPHS to your documentation list.

12. **Report the problem.**

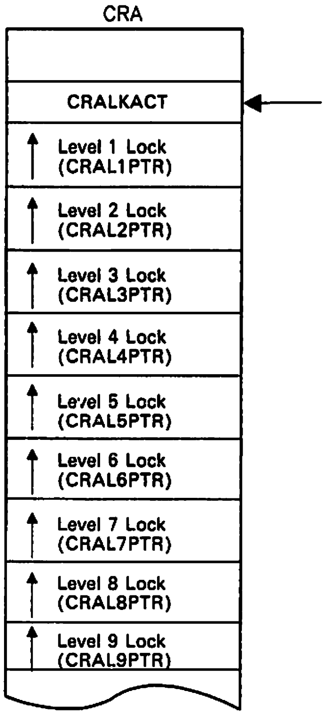
Go to "Reporting Procedure" on page 68.

Name	Lvl	Hex Value	Control Block	Field Name	Number	Function	Applies to
ADJLOCK	5	10	Adjacent Subarea Table (ISTADJSA)	ATCADJLK	One	Protects users of CIDCTL when adding or deleting an adjacent node.	MVS, VM, and VSE Pre-V3R2
DEBLOCK	5	10	ACDEB	ACDLOCK	One per OPEN application program	1. Protects FMCB queue off ACDEB. 2. Held by TSC and by OPEN or CLOSE.	Common
DWALOCK	8	80	DWA	DWALOCK	One	Used by certain disabled TSC modules to serialize use of the disabled work area (DWA).	MVS, VM, and VSE Pre-V3R2
HNTELOCK	7	40	Host Node Table Entry	HNTELOCK	One per minor node (per host element address)	Serializes updates and references to control blocks based off the HNTE (RDTE, NCB, LUCB, FMCB).	MVS, VM, and VSE Pre-V3R2
HNTLOCK	5	20	Host Node Table (ISTHNT)	ATCHNTLK	One	Protects updates and references to HNT during most CIDCTL functions.	MVS, VM, and VSE Pre-V3R2
HSQCHAIN	4	08	ATCVT	ATCHSQLK	One	Serializes usage of the HSQH queues. One lock is used to protect all of the queues.	Common
INNLOCK	9	100	ATCVT	ATCINNLK	One	Ensures that PIUs that are going to a node that is in slowdown mode are sent in FIFO order.	Common
PDBUFLK	9	100	ATCVT	ATCBUFLK	One	Allows the user to move in problem diagnosis trace data before the data is processed.	Common
PSTLOCK	8	80	ATCVT	ATCPSTLK	One	Serializes queueing and dequeueing of FMCB to PSTFMCB queue. Serializes release of PST storage.	Common

Figure 9 (Part 1 of 2). VTAM Locks

Name	Lvl	Hex Value	Control Block	Field Name	Number	Function	Applies to
QUEUE	9	100	PAB	PABLOCK	One per extended PAB	Serializes queueing and dequeueing of work elements to an extended PAB.	Common
RDTLOCK	3	04	ATCVT	ATCRDTLK	One	Protects users of CIDCTL (PAFIND). Obtained by PUNS when a network-addressable unit is to be added or deleted, or a use count decremented.	Common
VOCLOCK	2	02	ATCVT	ATCVOCLK	One	1. Serializes OPEN/CLOSE with VARY. 2. Serializes VARY Activate, VARY Deactivate, and VARY ERP.	Common
VRLOCK	3	04	VRBLK	VRBLOK	One per virtual route	Serializes usage of the VRBLK.	Common

Figure 9 (Part 2 of 2). VTAM Locks



Lock level held is relative to byte and bit position.
Byte 3, bit 6 corresponds to lock level 2;
byte 3, bit 5 corresponds to lock level 3;
and so on.
The value of this word corresponds to either
a hexadecimal value or to a combination
of hexadecimal values in the table above.

Figure 10. Pointers to VTAM Locks

Checking Whether the VARY INACT,FORCE Command Can Complete

If the operator attempted a VARY INACT,FORCE command, you should check whether the command can be completed or whether there is a VTAM problem. Determine this using the following steps according to the resource specified on the VARY INACT command.

1. **Channel-attached physical unit or logical unit, SNA or non-SNA:**

   a. Display the resource status. If it is PHLIN, PHLAC, PDLUC, or PSUB1, the channel is hung or a required interrupt is missing.
   b. If the status is PNFYX, go to step 11 on page 45.

2. **Link-attached SNA logical unit, switched logical unit:**

   a. Display the resource status. If it is PNFYX, go to step 11 on page 45.
   b. If it is anything else, there is a VTAM problem.

3. **Link-attached BSC 3270 logical unit:**

   a. Display the resource status. You should see PDACL or PFDLU.
   b. Issue VARY INACT,FORCE for the NCP or CA major node that NCP or CA defines the device.
   c. If the deactivation succeeded and all lower-level nodes are inactive, message IST105I (or 5B05I, for VTAM V3R1 or V3R1.2 running under VSE) will indicate this.
   d. Display the status of all the resources in the NCP or CA major node.
   e. If the status is PNFYX, go to step 11 on page 45.
   f. If it is anything else, there could be a VTAM or NCP problem. Go to "Reporting Procedure" on page 68.

4. **Link-attached SNA physical unit, switched physical unit:**

   a. Display the resource status. You should see PDISC or PFDSC.
   b. Issue VARY INACT,FORCE for either the physical unit to which the device is attached, or for the NCP or CA major node that defines the device.
   c. If the deactivation succeeded and all lower-level nodes are inactive, message IST105I (or 5B05I, for VTAM V3R1 or V3R1.2 running under VSE) will indicate this.
   d. Display the status of all the resources attached to the NCP.
   e. If the status is PSUBX, go to step 10.
   f. If the status is PNFYX, go to step 11 on page 45.
   g. If it is anything else, there is a VTAM problem.

5. **Link-attached BSC 3270 physical unit:**

   a. Display the resource status. You should see PDACP or PFDCP.
   b. Issue VARY INACT,FORCE for the NCP or CA major node that defines the device.
   c. If the deactivation succeeded and all lower-level nodes are inactive, message IST105I (or 5B05I, for VTAM V3R1 or V3R1.2 running under VSE) will indicate this.
   d. Display the status of all the resources in the NCP or CA major node.
   e. If the status is PSUBX, go to step 10.
   f. If the status is PNFYX, go to step 11 on page 45.
   g. If it is anything else, there is a VTAM problem.

6. **Local SNA or non-SNA major node, switched major node:**

   a. Display the resource status. You should see PSUBX.
   b. Issue VARY INACT,FORCE for any minor nodes that are not inactive. This should allow deactivation to be completed.

7. **Link:**

   a. Display the resource status. You should see PDLNK.
   b. Issue VARY INACT,FORCE for the NCP to which the link is attached. This should allow deactivation to be completed.

8. **Channel-attached NCP:**

   a. Display the resource status. You should see PDISC.
   b. Press the RESET LOAD button on the communication controller. This should allow deactivation to be completed.

9. **Link-attached NCP:**

   a. Display the resource status. You should see PSUBX.
   b. Display the status of the lower-level nodes.
   c. If the status is PNFYX, go to step 11.
   d. If the status is anything else, there is a VTAM problem.

10. **PSUBX status:**

    a. Display the status of the lower-level nodes to find any pending states.
    b. Deactivate any active or pending nodes. This should allow deactivation to be completed.

11. **PNFYX status:**

    a. **For application programs with an NS exit:**

       If the VARY INACT,FORCE command is unable to complete, there is a VTAM problem. Otherwise, deactivation should complete.

    b. **For application programs with only a LOSTERM exit:**

       1) If the application program has issued a CLSDST macroinstruction, deactivation should complete.

       2) If the application program has not issued a CLSDST macroinstruction for the logical unit, issue a second VARY INACT,FORCE for the logical unit in question. If that does not correct the problem, you may need to cancel the application program to allow the deactivation to complete. (Cancelling the application program terminates all of the LU-LU sessions with the application program.)

    c. **For application programs with neither exit:**

       Deactivation will not complete until the application program issues CLSDST, the application program closes its ACB, or the operator cancels the application program.

If you came to this procedure from step 2d on page 31, return now to step 2e on page 31.

Loop Procedure

If the problem is a **loop**, use the procedure in Figure 11 on page 47 to collect the following documentation:

- System console log
- Messages associated with the loop (if any)
- Failing module ID
- Dump output:
  - MVS Dump of the VTAM address space
  - VM Dump of the VTAM virtual machine
  - VSE Dump of the VTAM partition
- Error file output:
  - MVS VM LOGREC
  - VSE SYSREC
- For a problem associated with a device:
  - VIT trace output (all except LOCK)
  - I/O trace output (of NCP)
  - Session trace data (if using NetView or NLDM)
  - Session awareness data (if using NetView or NLDM)
  - NetView or NPDA report (if using NetView or NPDA)
  - MVS GTF CCWTRACE output (if available)
  - VM VM branch trace
  - VM VSCS internal trace
  - VM CP internal trace
  - VM CP virtual machine trace
  - VM GCS internal trace
  - VSE SDAID CCWD trace output

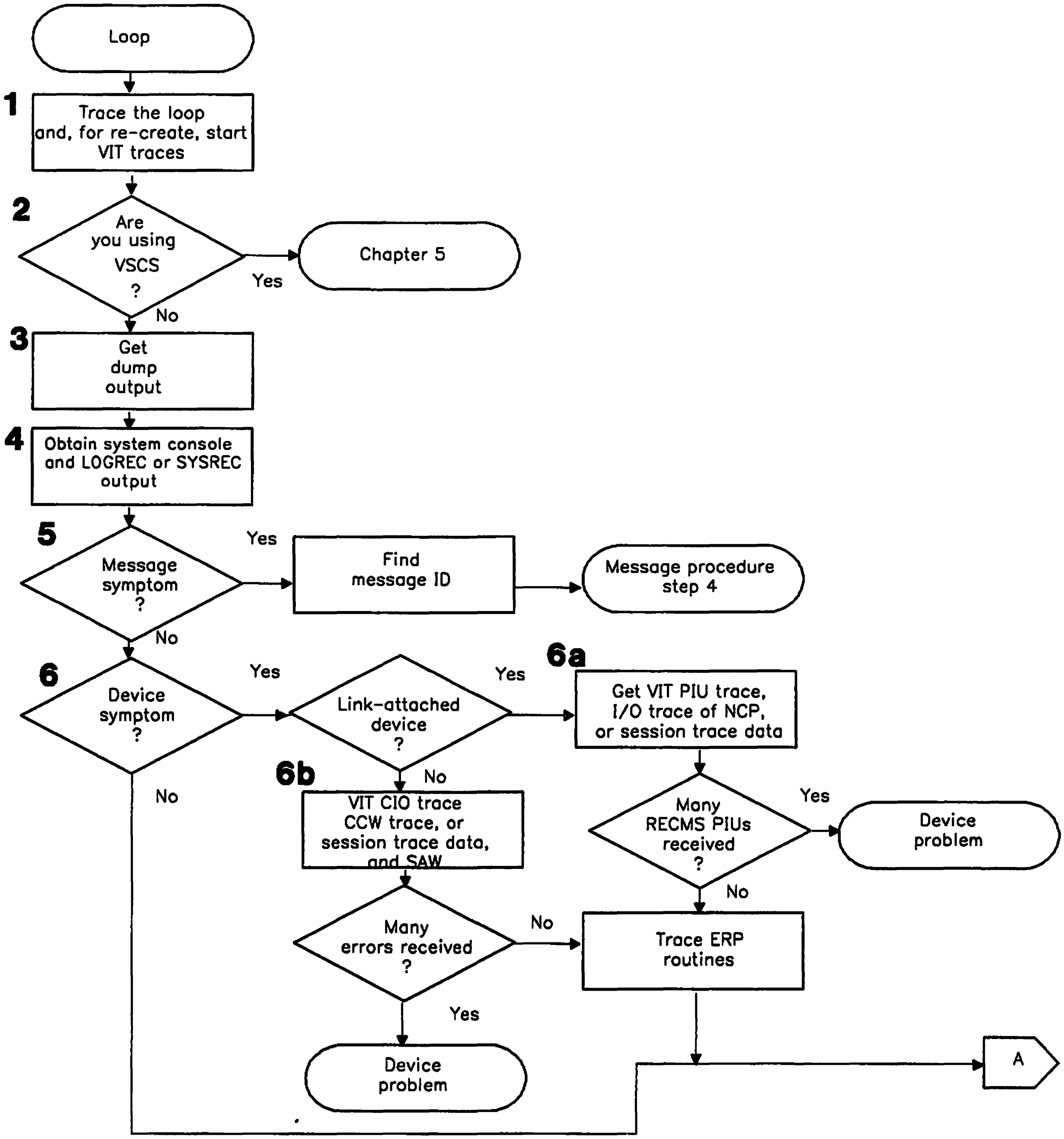


Figure 11 (Part 1 of 2). Overview of the Loop Procedure

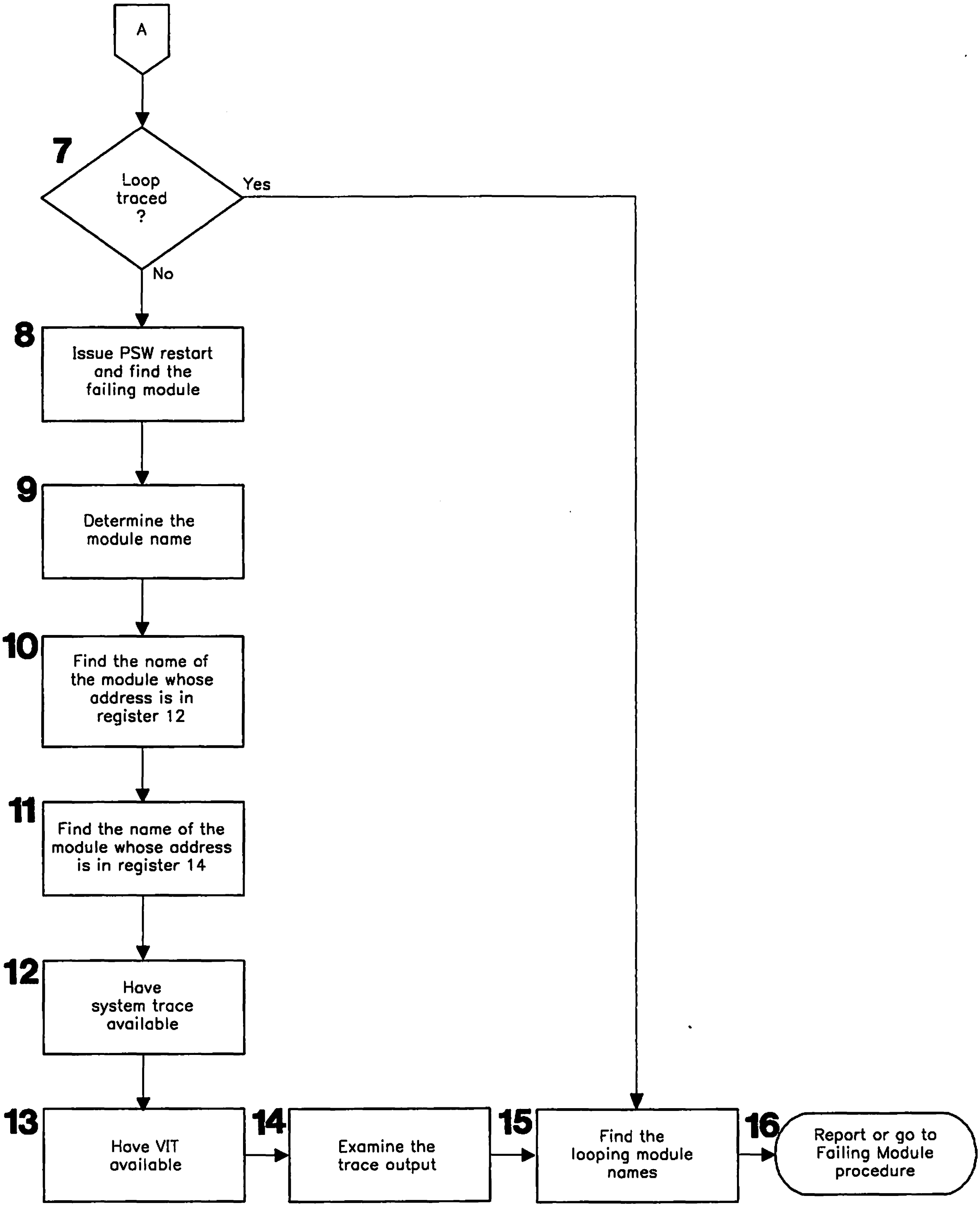

A
7
Loop traced ?
Yes
No
8
Issue PSW restart and find the failing module
9
Determine the module name
10
Find the name of the module whose address is in register 12
11
Find the name of the module whose address is in register 14
12
Have system trace available
13
Have VIT available
14
Examine the trace output
15
Find the looping module names
16
Report or go to Failing Module procedure

Figure 11 (Part 2 of 2). Overview of the Loop Procedure

Detailed descriptions of the steps involved in the loop procedure begin on page 49.

1. **Trace the loop**.

   Loop problems may involve many modules or a single module. If possible, trace the looping instructions. Using the *Operator's Reference* for your host processor, instruction-step through the looping addresses. Save these addresses for use in diagnosing the problem.

   In an enabled loop (PSW is enabled for interruptions), PSWS with low addresses may indicate a normal system process. Since most VTAM routines are enabled, look for PSW addresses that are in the range of the VTAM address space (MVS), virtual machine (VM), or partition (VSE).

   MVS You should also look for addresses within VTAM modules that are in the system range (LPA).

   If the VTAM internal trace was running when the loop started, look for any exception conditions that might have led to the loop. If the internal trace was not running, you will have to re-create the problem to get the trace at the time of the loop. Set the internal trace to MODE=EXT to record the trace entries in an external file.

   VM Issue DISPLAY PSW, then issue DISPLAY REGS several times to trace the loop. Then, obtain a VM branch trace of the loop, and start the VSCS internal trace, if you have VSCS. (The VSCS trace facility is explained in Chapter 6, "**VM** Using VSCS Service Aids.")

2. **VM Look for a VSCS loop**.

   If you are using VSCS, and the loop occurs during VSCS initialization or termination, go to Chapter 5, "**VM** Procedures for VSCS Problems" on page 93.

   If you are using VSCS, and the loop occurs at a time other than VSCS initialization or termination, go to "**VM** Presentation Services is in a Large Enabled Loop" on page 121. If you cannot identify the problem using those procedures, return to this procedure.

   MVS If you are using TSO/VTAM, use this procedure. You do not need to go to Chapter 4, "**MVS** Procedures for TSO/VTAM Problems" on page 71.

3. **Get dump output**.

   Get a dump of VTAM:

   - **MVS** Issue the DUMP command, or press the Program Restart key.
   - **VM** Issue #CP VMDUMP, with the appropriate options.
   - **VSE** Issue the DUMP command.

   If the loop is disabled, the system console will not be available for input, so take a stand-alone dump. (See "**MVS** Stand-Alone Dump" on page 284.) Print the dump output.

4. **Get the system console log and LOGREC or SYSREC output**.

   The system console log may contain information, such as error messages, that can help you diagnose the problem. Also, print the LOGREC or SYSREC file.

   MVS Use the LOGDATA option to print the in-core LOGREC buffers. See *MVS Service Aids* for more information.

5. **Is a message involved?**

   Determine whether there are any messages associated with the loop, such as a particular message always preceding the problem, or the same message being issued repeatedly. If so, add the message numbers to your problem documentation and go to the message procedure, step 4 on page 56.

6. **Is it a device error?**

   For any kind of device error, you should first check the NetView or NPDA report (if you have NetView or NPDA) and then the LOGREC or SYSREC output.

   Does the LOGREC or SYSREC output show repetitive entries for the same kind of error on a particular device? If so, VTAM is receiving several different errors from that device.

   a. If the LOGREC or SYSREC error records are for a link or link station attached to a communication controller, get **VIT PIU** records and an **I/O trace** of the NCP. If you have NetView or NLDM, get **session trace data** or **session awareness data** for the NCP. If the error records are for a link or device attached to a communication adapter, get **VIT PIU** records or a **dynamic trace of the communication adapter**.

      If the trace shows continual arrival of RECMS PIUs, then the repetitive entries in LOGREC or SYSREC are caused by a device error.

   b. For channel-attached devices, use one or more of the following traces for the device to determine whether VTAM is receiving many errors:

      - VTAM internal trace with CIO option
      - Session trace data (if using NetView or NLDM)
      - Session awareness data (if using NetView or NLDM).
      - MVS CCWTRACE (if available)
      - VM SIO and I/O traces for CP and GCS
      - VSE I/O interruption and CCWD SDAID traces

      If VTAM is receiving many errors, the problem is probably in the device. Try to trace the execution of VTAM ERP routines. Then continue with step 7.

7. **Is the loop traced?**

   If you were able to instruction-step through the loop, go to step 15 on page 52; otherwise, continue with step 8 on page 51.

8. **Find the failing module.**

   Use the PSW to find the failing module.

   The PSW also is found in:

   - MVS The LOGREC output, the SDWA, or the RTM2WA

     When you use PSW RESTART to terminate a looping task, a LOGREC entry is created with a completion code of X'071' for the task. An RTM2WA is also created for the task. Use the LOGREC record and the RTM work area to locate the failing module. See the diagnostic books for your operating system for help in locating the PSW in dump output.

     Depending on the PSW bit 32, the last 3 bytes (24-bit mode) or 4 bytes (31-bit mode) of the PSW contain the address being executed at the time of the dump. Scan the dump output to find the address given in the PSW.

     For more information on PSWS, see the *Principles of Operation* book for your processor.

     Note: Addresses may not always be in numeric order because the dump does not always generate output in sequential order.

     If you cannot find the address, the dump may not contain the relevant portion of main storage. For example, the address may be in LPA storage. Have this portion of storage dumped, or use output from LPAMAP to identify the module, and proceed as above.

   - VM You should have found the PSW when you issued DISPLAY PSW in step 1 on page 49.

   - VSE The location of the PSW in the dump will vary, depending on what type of dump was taken. See *VSE Service Aids* for help in finding the PSW in the dump output.

     The last 3 bytes of the PSW contain the address being executed at the time of the dump. Scan the dump output to find the address given in the PSW.

     For more information on PSWS, see the *Principles of Operation* book for your processor.

     Note: Addresses may not always be in numeric order because the dump does not always generate output in sequential order.

9. **Find the module name containing the failing address.**

   VTAM identifies modules with an EBCDIC module name and Julian date at or near the beginning of most modules. This module identifier will usually be in the form:

   IST*xxxxx yy.ddd [nnnnnnn]*

   where *xxxxx* is the last five characters of the module name, *yy.ddd* is the Julian date the module was assembled, and *nnnnnnn* is the latest PTF (if any) that has been applied to this module.

   To find the module ID, start at the failing address and scan upward (in descending address order) along the right side of the dump listing. The module ID is printed in EBCDIC. Add the module name to your documentation list.

10. **Find the module pointed to by register 12.**

    General register 12 (X'0C') is normally the base register for VTAM routines. In a VTAM loop, register 12 should point to the same module found in step 11. If not, add this module name to your documentation list.

11. **Find the module pointed to by register 14.**

    General register 14 (X'0E') may point to a module that called the routine that is looping. Add this module name to the documentation list.

    Normally, you should add the module names from steps 9, 10, and 11 to your documentation list. You can report the problem next, but you may need to continue with step 12.

12. **Get the system trace output.**

    The system trace may show many external and I/O interrupts. The PSW addresses in system trace entries will be part of the loop.

13. **Get the VIT output.**

    The VIT is useful in determining the reason for the loop, such as a process being continually re-dispatched for the same request. Get any VIT output available. If the VIT is not available, and VTAM will accept the command, start the VIT and specify MODE=EXT. If VTAM will not accept the command, it may be necessary to re-create the problem. For more information on the VIT, see Chapter 8, "Using the VTAM Internal Trace."

14. **Examine the trace entries.**

    By examining the trace entries, you should be able to determine whether there is a loop. The most obvious loops would be a module or modules getting continual control of the VTAM system, or a control block chaining to itself. Check the output of the PSS option to see which VTAM routines are getting control.

    If you see a pattern of repetition in the trace entries, it does not necessarily mean VTAM is looping. Some VTAM processes are timer-driven and repeat periodically. Also, the absence of any apparent loop does not necessarily mean VTAM is *not* looping. The loop may not contain a VTAM trace point.

    If a module or modules are looping, get their addresses from the trace entries. Step 15 explains how to find the module name.

    If you find a control block chained to itself, or if a queue of control blocks is in a cycle, try to identify the control block. Most control blocks have a 1-byte ID at offset X'00'. See the control block ID codes in Appendix E, "Storage and Control Block ID Codes" on page 657 to identify the control block name.

15. **Find the module names.**

    Use the addresses found in step 14 to find the module names involved in the loop.

    To find the module ID, start at the failing address and scan upward (in descending address order) along the right side of the dump listing. The module ID is printed in EBCDIC. Add this module ID to the documentation list. Continue with step 16.

16. **Report or go to the failing module procedure.**

    If you determined the module names, go to "Failing Module Procedure" on page 67. Otherwise, you are ready to contact IBM. Go to "Reporting Procedure" on page 68.

Message Procedure

If the problem is a **message**, use the procedure in Figure 12 on page 54 to collect the following documentation:

- Issuing module
- Message number
- System console log
- Dump output if required
- VIT output.

Note: If your installation changed the text of the message, the message ID may not be included, or may not match the ID of the message as it appears in *VTAM Messages and Codes*. Therefore, it is recommended that you re-create the problem using the VTAM-supplied message text. Otherwise, determine what VTAM-supplied message text corresponds to the message text your installation is using.

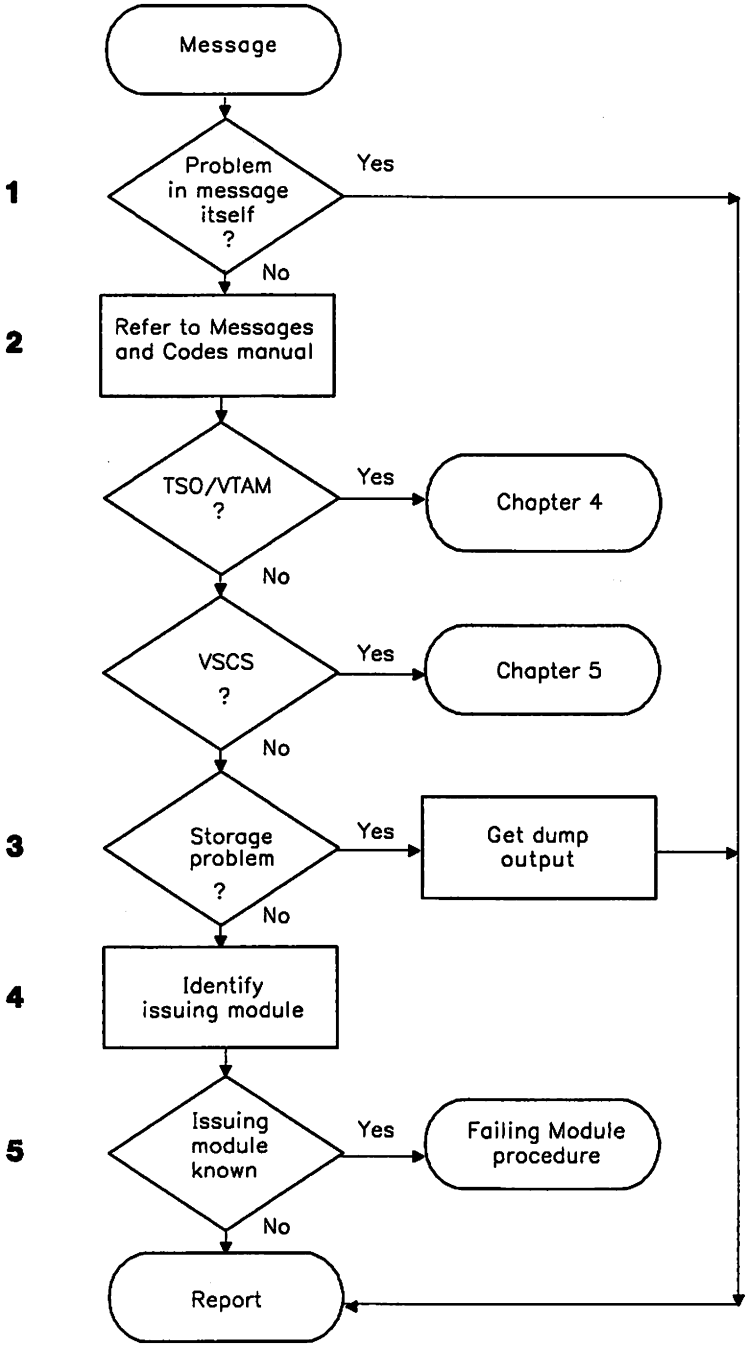


Figure 12. Overview of the Message Procedure

Detailed descriptions of the steps involved in the message procedure begin on page 55.

1. **Report if the problem is in the message itself.**

   If the content of the message is incorrect or the meaning of the message is not clear, go to "Reporting Procedure" on page 68.

2. **Follow the recommended action.**

   For all messages, see *VTAM Messages and Codes* for recommended operator and programmer actions. (See the list of VTAM books in the "Bibliography" on page 671 for the appropriate form number.) **In addition**:

   - If the message indicates a storage problem, go to step 3.
   - If the message indicates a TSO/VTAM problem, see Chapter 4, "MVS Procedures for TSO/VTAM Problems."
   - If message DTIS99I indicates a VSCS problem, see "VM Message DTIS99I Indicates a VSCS Internal Error" on page 135.

   The following list shows message prefixes and the components that issue those messages.

Prefix	**Issuing Component**
DTI	VSCS
IKT	TSO/VTAM
IST	VTAM (MVS, VM, and V3R2 VSE)
5	VTAM (VSE V3R1 and V3R1.2)

   If the message starts with any other characters, it comes from another network component.

3. **Get documentation for storage problems.**

   The following VTAM messages may indicate a storage problem:

MVS VM and V3R2 VSE	**V3R1 and V3R1.2 VSE**	**Description**
IST561I	5F61I	Indicates a buffer pool shortage
IST562I	5F62I	Indicates that CSA has been reached
IST563I	5F63I	Indicates that MAXPVT has been reached
IST564I	5F64I	Contains the subpool number
IST565I	5F65I	Contains the subpool number
IST566I	5F66I	Contains the subpool number

Figure 13. VTAM Messages Issued for a Storage Shortage

VSE Message IST561I (or 5F61I, for VTAM V3R1 or V3R1.2) may be issued when there is no problem. For example, a request may come in for a buffer pool before the pool expansion request can run. You must display buffer use to determine this.

If the message is issued frequently or continuously, get a dump:

MVS	Dump VTAM common and private storage areas.[2]
VM	Dump the VTAM virtual machine.
VSE	Dump the VTAM partition.

For messages that give a subpool number, dump the storage containing that subpool.

It is highly recommended that you use a dump formatter. Dump formatters are described in "MVS VM Formatting and Printing Dump Output" on page 279.

After getting the documentation for your storage problem, go to "Reporting Procedure" on page 68.

4. **Identify the issuing module.**

   Try to identify the module issuing the message. If the MSGMOD start parameter was active or the MODIFY MSGMOD command was issued before the problem occurred, the message text contains the last five characters of the issuing module name. Add the message prefix to the module name, and add this name to your problem documentation. (To modify the module identifier in messages, see "Modify Message Module Identification" on page 270.)

   The VTAM internal trace MSG entries contain the message number, the save area address, and the module ID (the 4th, 5th, 7th, and 8th characters of the module name). Use these to identify the issuing module. If the trace entry contains no module identifier, use the caller's address from the trace entry, or find the message issuer by using the message-to-module cross reference in *VTAM Messages and Codes*.

5. **Report or go to the failing module procedure.**

   If you know the name of the issuing module, go to "Failing Module Procedure" on page 67. If you are unable to determine the issuing module or resolve the problem, go to "Reporting Procedure" on page 68.

[2] Although VTAM detects storage shortages in the common storage area, VTAM may not be causing the shortage since this area is shared by all address spaces.

Incorrect Output Procedure

If the problem is **incorrect output**, use the procedure in Figure 14 on page 58 to collect the following documentation:

- Specific output that is incorrect
- Device type (if appropriate)
- Buffer contents trace output
- VIT trace output (all except LOCK)
- Session trace data (if using NetView or NLDM)
- Session awareness data (if using NetView or NLDM)
- Network controller line trace output (3710 only)
- **MVS**
  - GTF CCWTRACE output (if available)
  - For TSO problem — TGET/TPUT trace
- **VM**
  - CP internal trace
  - CP virtual machine trace
  - GCS internal trace
  - For VSCS problem — VSCS internal trace
- **VSE**
  - SDAID CCWD trace output
  - Communication adapter line trace output
- Network problem:
  - Line trace
  - Generalized PIU trace (GPT)
  - Transmission group (TG) trace
  - Scanner interface trace (SIT) for 3720, 3725, and 3745 only.

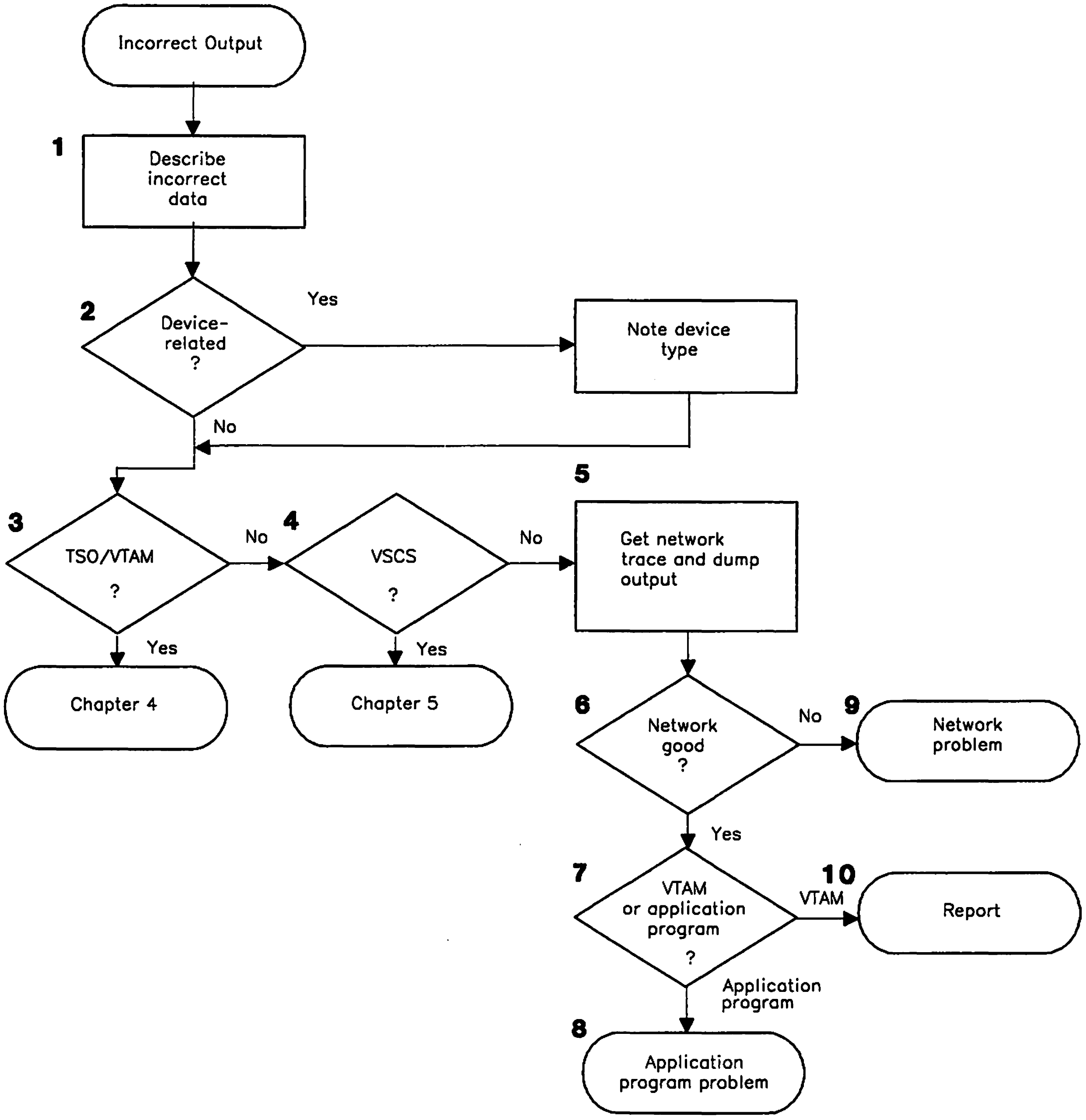


Figure 14. Overview of the Incorrect Output Procedure

Detailed descriptions of the steps involved in the incorrect output procedure begin on page 59.

1. **Describe the incorrect data.**

   Most incorrect output problems pertain to data contained in a PIU. This can be incorrectly formatted user data, invalid routing information, or other types of missing or incorrect data. These types of problems are generally difficult to diagnose, as they show up only at the end-user locations. From the following list, choose all the symptoms that apply to your problem and add them to your problem documentation:

   - Cursor is in the wrong place or missing.
   - Format of data is incorrect.
   - Data is truncated.
   - Data is incorrect.
   - Data is missing.
   - Problem is related to an application program macroinstruction.
   - Screen is improperly formatted.
   - Session is hung.
   - Terminal is hung.

   Note: The term "hung terminal" means the user is prevented by the system from entering data.

2. **Determine the device type.**

   If the problem is associated with a specific device type, add the device type (for example, 3270 terminal) to your problem documentation.

3. MVS **Is it a TSO/VTAM user problem?**

   If the incorrect output problem involves TSO/VTAM, go to "MVS Screen Management Problems" on page 81.

4. VM **Is it a VSCS user problem?**

   If the problem involves a VSCS user, follow the procedure in "VM Incorrect Output Problems" on page 138.

5. **Get network trace and dump output.**

   Re-create the problem with the following service aids active:

   Note: For more information on VTAM service aids, see Chapter 7, "Using VTAM Service Aids." Operating system service aids are documented in operating system books.

   - Start the VTAM **buffer contents trace** for the failing application program and terminal logical units.

   - MVS If you have an IBM 3710 Network Controller, start the **network controller line trace**. It traces information passing to and from a 3710.

   - Start the **VTAM internal trace** with MODE=EXT. (You do not need to use the LOCK option unless you suspect that a lock is not working.) This shows the type of requests being processed between the application program and the end user or terminal, and the control information for routing, pacing, and so on, in each PIU sent in the network.

   - You should start the **generalized PIU trace** if the problem involves lines or devices attached to a communication controller. This will show how far the PIU got within the NCP and what the PIU looked like (its control information) when it was sent to the line.

- VSE If the problem involves lines or devices attached through a communication adapter, start the **dynamic trace of the communication adapter**. This will show any problem related to the line.

- If you have NetView or NLDM, you can use the **session trace data** to determine the requests and responses received and sent by VTAM and the other network nodes.

- MVS VSE If it is available, you can now use the **GTF CCWTRACE** (in MVS) or the **SDAID CCWD trace** (in VSE). These trace the CCWs, I/O interruptions, and all CCW data for each Start I/O issued by the system. For a problem where data in the RU is invalid, this is the only service aid that will show the entire PIU. For more information on these traces, see the diagnostic books for your operating system.

- As soon after the problem occurs as possible, take a **dump** of the application program, VTAM, and (in MVS) TSO/TCAS. Stop all traces, and print the dump and trace output.

The dump is used to reference storage addresses, such as control blocks and module entry points. The trace data shows at what point the data was modified, and what PABs the data was on as it was processed by VTAM. Take the dump during the re-create, when the traces are running. A dump taken earlier may not be accurate because the terminal device may have been deactivated and reactivated. This would allocate a different set of control blocks.

6. **Examine the trace output.**

   Examine the individual trace entries to find the failure. If the problem concerns user data format, and the buffer contents trace or PIU trace does not show the invalid data, use the output from the VIT trace with the SSCP option.

   MVS The CCWTRACE (if available) can be used to see whether data is correct when it goes to the NCP or logical unit. The VTAM internal trace records CC2, CI2, and CO2 contain the first 24 bytes of this data.

   When output data is correct:

   If the traces show that the data or the control information in the RH/TH as it leaves VTAM is correct, the problem is not in VTAM or the application program; go to step 9 on page 61. If the data going to the network is invalid, continue with step 7.

   When input data is incorrect:

   If the traces show that invalid data is entering VTAM from an external network source, the problem is in the network; go to step 9 on page 61. If the data from the network is valid, the problem is in VTAM or an application program; continue with step 7.

7. **Is it VTAM or an application program?**

   The problem has been narrowed down to VTAM or the application program. Examine each trace entry to determine whether the information from the application program was incorrect. If VTAM seems to be responsible, go to "Reporting Procedure" on page 68; otherwise, continue with step 8 on page 61.

8. **Is more application program help needed?**

   For IBM application programs such as CICS or IMS, you may find additional diagnostic help in the IBM application program books. If you decide that the problem is with an IBM application program, contact the appropriate IBM representative for that product.

9. **Is the problem with an external network device?**

   The problem has been narrowed down to the VTAM network, but not to VTAM itself. Try to identify the device or program responsible. You can use service aids, such as the NCP line trace, generalized PIU trace, or transmission group trace, to trace data flow between the NCP and terminal logical units. For information on how to use these traces, see "Traces Provided by NCP" on page 245. For a 3720, 3725, or 3745, use the scanner interface trace (SIT) to distinguish between NCP problems and line or terminal problems. If you suspect the NCP, see the *NCP, SSP, and EP Diagnosis Guide*.

   Chapter 7, "Using VTAM Service Aids" explains how to use other service aids, including the NCP dump. Contact the appropriate IBM representative for the device or program identified as the cause of the problem.

   VSE The communication adapter line trace can be used to discriminate between a VTAM error and one that is related to a device or line attached through the communication adapter. For more information, see the diagnostic book for the host processor.

10. **Report the problem.**

    Go to "Reporting Procedure" on page 68.

Performance Procedure

If the problem is **performance**, use the procedure in Figure 15 on page 63 to collect the following documentation:

- System console log
- Error file output:
  - MVS VM LOGREC
  - VSE SYSREC
- Tuning statistics
- SMS (buffer use) trace output
- Network controller line trace output (3710 only).

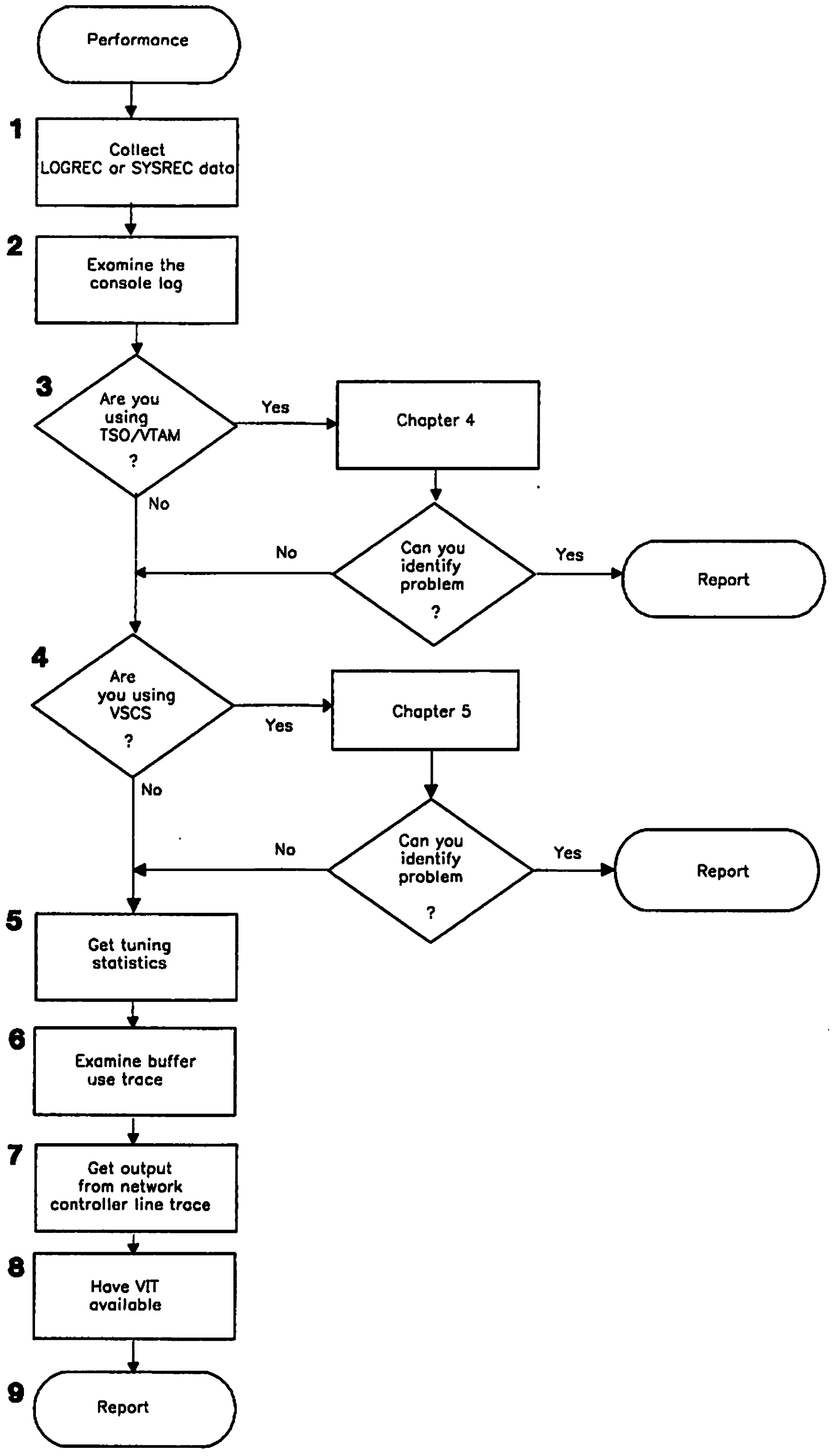


Figure 15. Overview of the Performance Procedure

Detailed descriptions of the steps involved in the performance procedure begin on page 64.

1. **Get LOGREC or SYSREC output.**

   Performance problems are often caused by hardware errors. These hardware errors cause software error recovery processing to occur, which degrades system performance. For this reason, you should get LOGREC or SYSREC output. LOGREC or SYSREC may show a large number of hardware errors for a particular device or group of devices. If the errors are limited to a single device, a hardware error is probably the cause. If the errors appear on many or all terminals of one type, software is more likely to be the problem, although hardware may still be at fault. If you suspect a particular device type, add it to your documentation list.

2. **Examine the system console log.**

   The system console log may contain messages to help diagnose a problem. Add the message ID to the documentation list:

   MVS — Document the message ID in the form *ISTxxxx* or *IKTxxxx*.

   VM — Document the message ID in the form *ISTxxxx* or *DTIxxxx*.

   VSE — Document the message ID in the form *5xxxx* (for VTAM V3R1 and V3R1.2) or *ISTxxxx* (for VTAM V3R2).

   The system console log may also contain information about command problems. For example, operator commands might be taking too long to complete. Add the command name (for example, VARY ACT) to the documentation list.

3. MVS **For TSO/VTAM, see Chapter 4, "MVS Procedures for TSO/VTAM Problems" on page 71.**

   If you are using TSO/VTAM, go to "MVS Performance Problems" on page 89. If you cannot resolve the problem with the procedure in Chapter 4, "MVS Procedures for TSO/VTAM Problems," return to this procedure.

4. VM **For VSCS, see Chapter 5, "VM Procedures for VSCS Problems" on page 93.**

   If you are using VSCS, and you have not identified the cause of the problem, go to "VM Performance Problems" on page 143.

5. **Get tuning statistics.**

   If the performance problem is associated with traffic through a channel-attached host, a channel-attached communication controller, or a channel-attached SNA physical unit, it may be helpful to get tuning statistics for VTAM. (For more information on tuning statistics, see "Modify Tuning Statistics" on page 272.)

6. **Get output from the SMS (buffer use) trace.**

   You may have enough information to identify the problem. If so, go to "Reporting Procedure" on page 68. If you do not, continue with this step.

   a. Buffer pool expansion can cause performance problems. During VTAM initialization, error recovery, and VARY and DISPLAY command processing, buffer usage is higher than normal. If buffer expansion is used, buffer pools should not expand except during such peak periods. Thus, what appears to be high buffer usage could be normal depending on the level of system activity.

      Run the buffer use trace (TYPE=SMS). For information on how to start the trace and examine the output, see "SMS (Buffer Use) Trace" on page 237. For diagnostic purposes, change the buffer request interval (RACBSNAP) in module ISTRACON to a value of 50 (X'32'). This change causes trace entries to be written more often to give a more accurate picture of buffer usage.

b. Using the time stamps in the system console and buffer use trace, correlate an excessive number of buffer pool expansions or large number of buffers used from a single pool with network activity recorded on the console. Constant high usage of a buffer pool may show that not enough buffers were allocated at VTAM initialization to properly support the level of network activity. Also look for a buffer pool that continually grows; buffers may not be released by some VTAM routine. Add the name of a particularly active buffer pool (for example, LPBUF or IOBUF) to your documentation list.

7. MVS **Get output from the network controller line trace**.

   If an IBM 3710 Network Controller is installed, start the network controller line trace. This traces information passing over the lines to and from a 3710. (For more information about this trace, see "**MVS** Network Controller Line Trace (3710 Only)" on page 250.) Print the trace output with TAP.

8. **Get additional documentation**.

   If no solid indication of a problem is apparent at this point, run the VTAM internal trace (VIT) with OPT=PSS,API,SSCP,PIU[*] and MODE=EXT. This creates a history of VTAM activity. At the time of performance degradation, stop VIT and take an SVC dump of VTAM. (For more information on the SVC dump, see "**MVS** SVC Dump" on page 284.) Print the dump and trace output for future reference.

9. **Report the problem**.

   Go to "Reporting Procedure" on page 68.

[*] If you are running an LU 6.2 application, include the APPC VIT option in this list.

Documentation Procedure

Note: Before using this procedure, be sure that documentation is the problem. A VTAM problem may cause the documentation to appear wrong.

If the problem is **documentation**, use the procedure in Figure 16 to collect the following documentation:

- Incorrect information
- Form number of book.

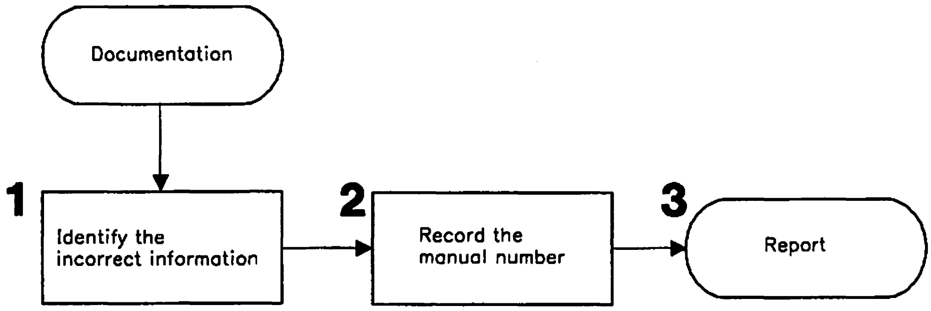


Figure 16. Overview of the Documentation Procedure

Detailed descriptions of the steps involved in the documentation procedure begin below.

1. **Identify the incorrect information.**

   Add to the documentation list the name of the macro, operand, or procedure that is incorrectly defined or explained in the documentation (for example, "line trace.")

2. **Record the form number.**

   Add the form number of the VTAM book to your documentation list in the form *ccnnnnnnrr* (omitting the dashes in the number; rr is the revision level). For example, report the form number of this book as LY30560100.

3. **Report the problem.**

   Go to "Reporting Procedure" on page 68.

   Note: Report a documentation problem only when it causes a VTAM problem. For suggestions, comments, or questions about VTAM books, use the Reader's Comment Form at the back of the book.

Failing Module Procedure

Use this procedure if you have identified a failing VTAM module in one of the other procedures (abnormal end, message, or loop). Figure 17 shows an overview of the failing module procedure.

Use this procedure to get the following documentation:

- Module ID and PTF eyecatcher
- Module descriptive name
- Caller of module
- VIT trace output.

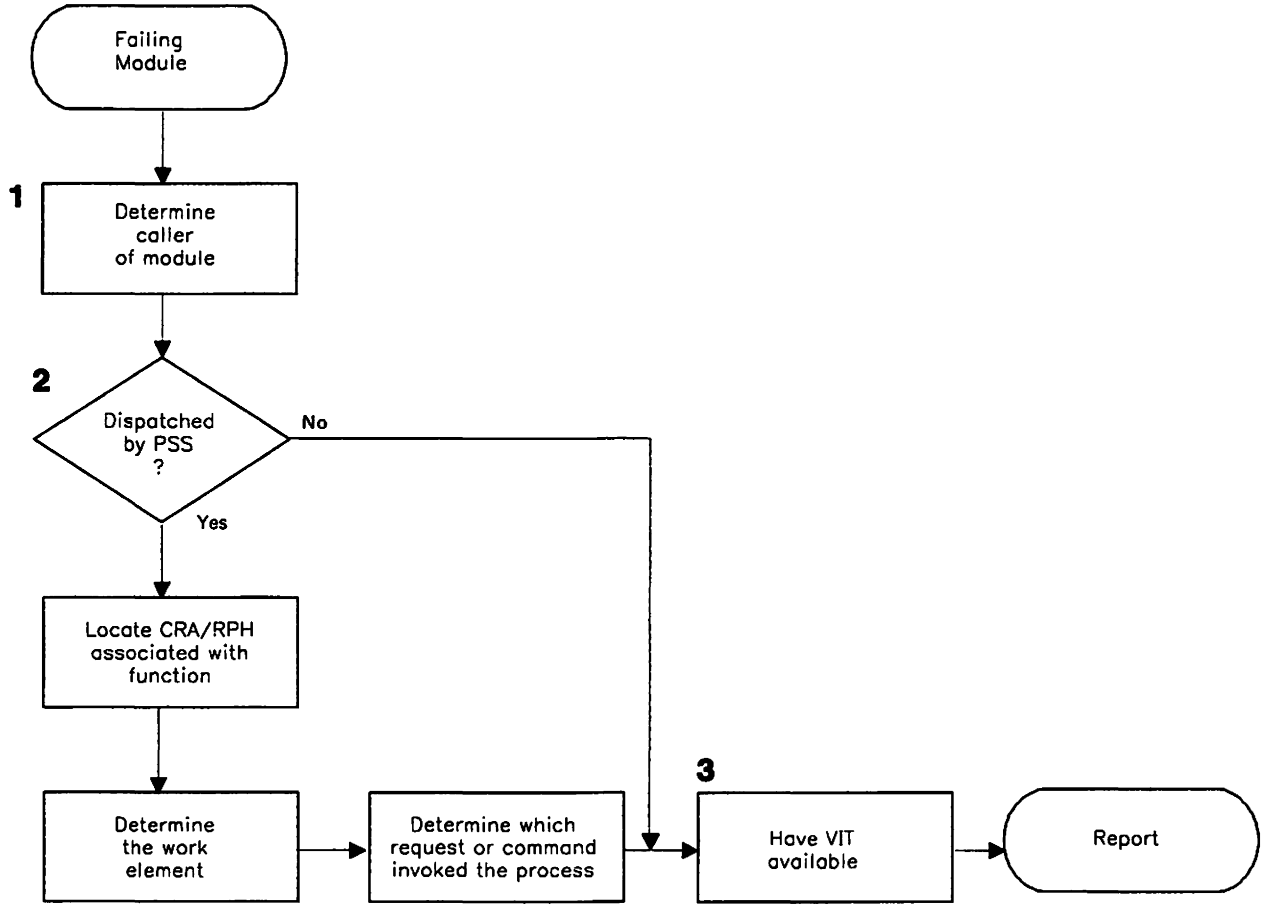

Failing Module
1
Determine caller of module
2
Dispatched by PSS ?
No
Yes
Locate CRA/RPH associated with function
Determine the work element
Determine which request or command invoked the process
3
Have VIT available
Report

Figure 17. Overview of the Failing Module Procedure

Detailed descriptions of the steps involved in the failing module procedure begin on page 68.

1. **Determine the caller of the service routine.**

   The failure may have occurred in a VTAM service routine used for many purposes. Determine the caller of the service routine. Use the save area conventions in "Using a Dump to Follow the Execution Sequence of VTAM Components" on page 302.

2. **Examine the work element structure.**

   If your problem type is an abnormal end or loop, and the module is dispatched under control of VTAM PSS, **find the CRA/RPH** associated with the process. (See step 10 on page 40 in the wait procedure.) The RPHWEA field (at offset X'1C') usually points to the work element associated with the process at the time it was dispatched. To identify the work element, see "Using a Dump to Follow the Execution Sequence of VTAM Components" on page 302.

   From the work element, it may be possible to **identify an SNA request/response type, an operator command, or an application program request** that ultimately caused the process to receive control. Add this request or command name to your documentation list.

   If the RPHWEA field does not point to the work element, continue with step 3.

3. **Get more trace output.**

   Use the **VIT options PSS and SMS** to get more information about how the failing module received control or where the relevant control blocks are found.

   Use the last DISP entry for the failing process to get the address of the work element and module name of the process entry point.

   Continue with "Reporting Procedure."

Reporting Procedure

Depending on the terms and conditions of your service agreement, you should report the problem to either the local branch office or the IBM Support Center.

After asking for your account name and other customer identification, the service representative will ask for a brief description of the problem. Your documentation list should contain the answers to all questions related to the problem.

You can find more information about the IBM Support Center and our problem solving process in the *Programming System General Information Manual.*

Chapter 4. MVS Procedures for TSO/VTAM Problems

Chapter 4. MVS Procedures for TSO/VTAM Problems

This chapter shows you what documentation to collect for each type of common TSO/VTAM problem. Use this chapter in conjunction with Chapter 3, "Procedures for Specific Problem Types" on page 19.

Note: Most traces discussed in this chapter are described in Chapter 7, "Using VTAM Service Aids." The exception is SVC 93 and SVC 94 entries, which are described in the *Debugging Handbook*, Volume 1. For VTAM and TSO/VTAM command syntax, see *VTAM Operation*.

To use this chapter, start below and follow the steps.

1. **Are you receiving one or more of the following messages?**

   - USS message 7 'LU-name UNABLE TO ESTABLISH SESSION – RU-name FAILED WITH SENSE sense', or similar USS message
   - USS message 10 (the "good morning" message)
   - Message IKT024I or IKT019I at the operator's console
   - Message IKJ608I at the operator's console.

   If so, go to "MVS Logon Problems" on page 73.

   Otherwise, continue with step 2.

2. **Have you encountered one of the following abends?**

   - ABEND0AB
   - ABEND0AC
   - ABEND0AD
   - ABEND15D.

   If so, go to "MVS TSO/VTAM Abends" on page 77.

   Otherwise, continue with step 3.

3. **Do you have a hung terminal?**

   - The terminal does not respond to any keys you press.
   - You must enter data from the terminal before processing will continue (in a situation where output is expected).

   If so, go to "MVS Hung Terminal Problems" on page 79.

   Otherwise, continue with step 4.

4. **Are you having one or more of the following screen management problems?**

   - Data is in the wrong place on the screen.
   - Data stream errors occur (such as operation checks, commands are rejected, PROGxxx).
   - Function errors occur (such as incorrect full-screen processing or incorrect line prompting in input mode of TSO EDIT).
   - Data length is incorrect.
   - Data content is incorrect.

If so, go to "MVS Screen Management Problems" on page 81.

Otherwise, continue with step 5.

5. **Are you having one or more of the following screen size problems?**

   - The terminal does not operate in the expected screen size after you log on.
   - The screen is not always the expected size during a TSO session.

   If so, go to "MVS Screen Size Problems" on page 86.

   Otherwise, continue with step 6.

6. **Are you having one or more of the following performance problems?**

   - Slow response time
   - An increase in the number of detected waits
   - An increase in the number of swap-outs.

   If so, go to "MVS Performance Problems" on page 89.

   Otherwise, if your problem is not listed in the steps above, it is probably not a TSO/VTAM problem. Go back to Chapter 3, "Procedures for Specific Problem Types" on page 19 and look for a more likely problem symptom. If you cannot find a more likely symptom, go to "Reporting Procedure" on page 68.

MVS Logon Problems

This section provides documentation requirements and diagnosis procedures for logon problems.

The recommended documentation is:

- VTAM buffer contents trace.

  To see the data in the buffer contents trace, set CONFTXT=NO in the TSOKEY00 member of SYS1.PARMLIB before starting TSO/VTAM.

- VTAM internal trace with MODE=EXT and OPTION=(API,PIU,SSCP,MSG).

  The VTAM internal trace may not be required. Review the diagnosis procedure for your problem to see if it is required.

1. **Did your first logon using USS commands fail?**

   If so, continue with the next step.

   Otherwise, go to step 5 on page 74.

2. In *VTAM Customization*, **review the process for setting up the USS table and using USS commands.** Check for the following errors:

   - Is your logon command syntax incorrect?

     If so, try to log on using the correct command syntax.

     Otherwise, continue with the next step.

   - Is the logmode name incorrect?

     If the logmode name is specified incorrectly, or if a default logmode entry that is inappropriate for the device type is used, you will get USSMSG7. Look up the sense code in *VTAM Messages and Codes* and correct the logmode name.

3. **Can you log on to TSO without using USS commands?**

   If you cannot log on at all, go to step 4.

   If you can log on, start the VTAM buffer contents trace and log on again. Look at the trace output to see what session parameters are contained in the BIND, and compare those parameters to the ones in your logmode table.

   If the session parameters in your logmode table are incorrect, make the necessary corrections. Also, make sure the DLOGMOD operand specifies the correct logmode table entry. (For more information on defining TSO/VTAM session parameters, see *VTAM Installation and Resource Definition*.)

   If you still cannot identify the problem, go to "Reporting Procedure" on page 68.

4. **If you cannot log on at all**, and if this is your first logon attempt from the device as well as your first logon attempt using USS commands, go to step 5 on page 74.

   If you cannot log on at all, and if this is **not** your first logon attempt from the device, go to "Reporting Procedure" on page 68.

5. **Did your first logon from a particular device fail?**

   If so, continue with the next step.

   Otherwise, go to step 7.

6. **Check for an error in the terminal definition statement, the logmode table, or the MODEENT macro.** (These are described in *VTAM Installation and Resource Definition.*)

   a. If you receive message IKT024I with return code X'210000' or X'220000', the BIND has been rejected. The following steps should help you find the invalid portion of the BIND:

      1) Locate the BINFM in the BIND. BINFM must be X'02' or X'03'. (For more information on coding the BIND, see *VTAM Programming.*)

      2) If a PSERVIC is coded, see *VTAM Installation and Resource Definition* to make sure all fields are coded correctly.

   b. Check to see if the DLOGMOD name is correct on the terminal definition statement.

      If it does not match an entry in the logmode table, the first entry in the logmode table is used as the default. The parameters on the default logmode table entry may not be appropriate for your device type, and as a result, the wrong BIND image may be passed to the logon exit and a CLSDST PASS failure may occur.

      Note: You can see this failure in the VTAM internal trace using the API option. For more information on the VTAM internal trace, see Chapter 8, "Using the VTAM Internal Trace."

   c. Check the logmode table entry to see if the parameters are specified correctly for your device type.

      If not, the wrong BIND image may be passed to the logon exit and a CLSDST PASS failure may occur.

   d. Check to see if the MODEENT macro is defined correctly.

      If it is not, the terminal may reject the BIND, or the terminal may indicate to the logon exit that the terminal is not supported by TSO/VTAM.

   e. If you have not identified the problem, and if users can log on to TSO from other terminals, start the VTAM buffer contents trace and check the BIND.

   f. If logons fail for all terminals, or if the BIND in the buffer contents trace is what you expected, run the VTAM internal trace with options API, PIU, SSCP, and MSG.

   g. If you still have not resolved the problem, go to "Reporting Procedure" on page 68.

7. **Did previous logons succeed, but now you cannot log on?**

   If so, continue with the next step.

8. **Did you get message IKJ608I?**

   If not, go to step 13 on page 75.

   If so, continue with the next step.

9. **Is message IKJ608I followed by message IKT104I?**

   If it was **not**, go to step 10.

   If if **was**, a storage shortage prevented the OPEN ACB from completing successfully. This is a VTAM storage shortage, not a TSO/VTAM storage shortage. See "Message Procedure" on page 53 for diagnosis information.

10. **Is message IKJ608I followed by messages IST804I, IST400I, and IST805I?**

    If so, the CLSDST PASS may have failed. Continue with the next step.

11. **Is this a cross-domain logon?**

    If not, go to step 12.

    If this is a cross-domain logon, make sure each domain has the same number of TSO/VTAM APPLIDs defined.

    If one domain has more APPLIDs defined than the other, the CLSDST PASS can fail, with an RPLRTNCD and RPLFDB2 combination of X'1453'. (The values of these RPL fields are shown in the VTAM internal trace using the API option. For more information on the VTAM internal trace, see Chapter 8, "Using the VTAM Internal Trace.")

12. **Is there an invalid TSO/VTAM APPLID?** (For more information on defining APPLIDS, see *VTAM Installation and Resource Definition.*)

    If so, correct the APPLID.

    If not, continue with step 13.

13. **Did you get message IKT111I?**

    If not, go to step 14 on page 76.

    If so, check the message text to determine the reason for the logon failure. See *VTAM Messages and Codes* for additional diagnostic information.

14. **Is an ABENDOAB with return code X'0201' or X'0202' associated with the logon attempt?**

    If not, go to step 15.

    If so, check the LOGREC entry for additional information.

Reason Code	Register	Contents
X'0105'	6	RPLRTNCD, RPLFDB2, and RPLDAF
	7	RPLFDBK2 (the word of sense)
X'0201'	8	ACBERFLG (for OPEN ACB failure)
	9, 10	TVWA ACB Name (TSOnnnn)
X'0202'	8	ACBERFLG (for OPEN ACB failure)
	9, 10	TVWA ACB Name (TSOnnnn)
X'0203'	5	ACBERFLG (for CLOSE ACB failure)
	6, 7	TVWA ACB Name (TSOnnnn)

Figure 18. MVS ABEND0AB Information in LOGREC

    Then go to "Reporting Procedure" on page 68.

15. **If none of the previous situations apply**, start the VTAM buffer contents trace and the VTAM internal trace, and trace the logon attempt. Then go to "Reporting Procedure" on page 68.

MVS TSO/VTAM Abends

TSO/VTAM issues several unique abends. This section briefly describes the causes and documentation requirements for each one. Use the information provided here and in *VTAM Messages and Codes* to try to resolve the problem.

If you are not able to do so, go to "Reporting Procedure" on page 68.

MVS ABEND0AB

ABEND0AB occurs when a VTIOC module issues a VTAM macroinstruction that fails. Depending on the values of the RPLRTNCD and RPLFDB2 fields, the macro may be retried. If the retry fails, ABEND0AB is issued.

Offset	**Length (bytes)**	**Description**
X'280'	8	Terminal name
X'289'	1	RPL request type: X'22' = SEND X'23' = RECEIVE
X'28A'	1	RPLRTNCD
X'28B'	1	RPLFDB2
X'28C'	4	RPLFDBK2 (Sense code)

Figure 19. MVS ABEND0AB Information in a Dump of SDWA

If this happens during the execution of a SEND or RECEIVE, the session is placed in reconnect status.

The recommended documentation is:

- Contents of register 15.

  This contains the reason code, which is explained in *VTAM Messages and Codes.*

- Message text for message IKT108I.

- The software LOGREC entry.

  If you have a LOGREC entry, look at an unformatted dump of the SDWA. Figure 19 describes the pertinent data you should look for in the dump.

- The dump that is created automatically for this abend.

  For abends associated with I/O errors, a dump is not generated automatically unless the RCFBDUMP parameter of the TSOKEY00 member of SYS1.PARMLIB is set for it.

- For errors that occur during session initialization or termination, run the VTAM internal trace with options API, PIU, MSG, SSCP, and PSS.

ABEND0AC

ABEND0AC occurs when an error halts TCAS processing.

The recommended documentation is:

- Contents of register 15.

  This contains the reason code, which is explained in *VTAM Messages and Codes*.

- TWARSON (IKTCASWA + X'02').

  This also contains a reason code, which is explained in *MVS System Codes* or *VTAM Messages and Codes*.

ABEND0AD

ABEND0AD occurs when the TSO/VTAM queue manager has a problem manipulating storage for the input and output queues.

The recommended documentation is:

- Contents of register 15.

  This contains the reason code, which is explained in *VTAM Messages and Codes*.

- The dump that is created automatically for this abend.

ABEND15D

ABEND15D occurs when the issuer of a TGET, TPUT, or TPG macro passes an invalid data area to the SVC 93 modules. A TPUT or TPG request requires read access to the area, and a TGET requires write access. ABEND0C4 occurs when IKTVTPUT or IKTVTGET tries to validate the data areas passed from the application program, and IKT93EST changes the ABEND0C4 to an ABEND15D.

The recommended documentation is:

- SLIP dump of the ABEND0C4.

  The SLIP dump is described in the *MVS Operator's Library*.

- GTF trace of SVC 93 entries.

Note: Either of these will show the invalid address.

MVS Hung Terminal Problems

This section provides documentation requirements and diagnosis procedures for problems with hung terminals. Use this information if this problem occurs while you are using TSO/VTAM. This procedure will help you determine when the hang occurred and what was happening at that time.

If the problem occurs during logon or logoff, get the recommended documentation and go to "Reporting Procedure" on page 68.

The recommended documentation is:

- VTAM buffer contents trace.

  To see the data in the buffer contents trace, set CONFTXT=NO in the TSOKEY00 member of SYS.1.PARMLIB before starting TSO/VTAM.

- GTF trace of SVC 93 and SVC 94 entries.

  See the *Debugging Handbook*, Volume 1, for information on interpreting SVC 93 and SVC 94 entries.

- VTAM internal trace with MODE=EXT and options API and PIU.

- TGET/TPUT trace.

  The TGET/TPUT trace records all TGET/TPUT/TPG data except address space ID TPUTS.

  You can get the TGET/TPUT trace by issuing the MODIFY TRACE command with TYPE=TSO. The MODIFY command is described in *VTAM Operation*.

- Dump of the nucleus, CSA, and user's address space.

  If you think you might not be able to re-create the problem, take a dump *before* you try to clear the hang.

  If input (such as ATTN, RESET, or ENTER) clears the hang, both a dump and traces may be necessary. You may want to start the traces, take a dump with the terminal hung, clear the hang, and then stop the traces.

- Collect additional general information:

  – Try to determine if the error is related to a certain type of hardware or a certain protocol (SNA or non-SNA).

  – Try to determine if the hang is related to a particular application program or type of application program (full-screen, graphics, and so on).

    If so, do other similar types of applications also hang?

1. **Was the last data that was sent from the application program to VTAM sent to the terminal before the hang occurred?**

   To determine this, look for a VTAM buffer contents trace entry that corresponds to the last TPUT trace entry.

   If you see these corresponding entries, the output was sent to the terminal.

2. **If the keyboard locked after data was sent to the terminal**, check the outbound buffer contents trace entry for:

   - Bracketing indicators in RH byte 2
   - Change direction indicator in RH byte 2
   - Write control character to unlock the keyboard in byte 2 of the output request unit.

   If an end bracket was sent, the keyboard should be available.

   If a change direction was sent and the keyboard has been unlocked, the keyboard should be available.

   If a TGET is issued after a full-screen TPUT, TSO/VTAM should unlock the keyboard.

3. **If the keyboard has not been unlocked**, see what kind of TPUT was issued last. (The option flag bytes in the TPUT entry show what kind of TPUT it is.)

   - For a NOEDIT or TPG TPUT, TSO/VTAM will not unlock the keyboard. The application program is supposed to send a write control character to unlock the keyboard.
   - For other TPUT options, is a TGET outstanding?

     If a TGET is outstanding, TSO/VTAM is supposed to unlock the keyboard.

     If no TGETs are outstanding, contact the group responsible for the application program.

4. **If the last activity prior to the hang was input from the terminal**, was the data passed to the application program? (If it was, the TGET trace entry will correspond to the inbound VTAM buffer contents trace entry.)

   - If so, did the application program ever issue another TPUT?

     If the application program never issued another TPUT, contact the group responsible for the application program.

   - If not, is there anything on the output queue? (TVWABOQ points to the first output queue element.)
   - If data has been received by TSO/VTAM, but it has not been sent to the terminal, go to "Reporting Procedure" on page 68.

5. **If you have not identified the problem**, go to "Reporting Procedure" on page 68.

MVS Incorrect Output Problems

Two main types of incorrect output problems are discussed in this section. **Screen management problems** involve mode errors, exception responses, and problems with the data on the screen. **Screen size problems** involve an incorrect or unexpected screen size, either in a particular mode or all the time. Choose the one that is most like your symptoms and follow the procedure for that problem.

MVS Screen Management Problems

This section provides the documentation requirements and diagnosis procedures for problems displaying data on the screen. This section deals with five types of screen management problems:

- Function error (incorrect screen management for mode)
- Exception responses
- Extra or missing data
- Data is not placed correctly on the screen
- Data appears to be translated incorrectly.

Choose the one that most closely matches your symptoms and follow the procedure for that problem.

Note: Problems with incorrect screen sizes are addressed in "MVS Screen Size Problems" on page 86.

The recommended documentation is:

- VTAM buffer contents trace.

  To see the data in the buffer contents trace, set CONFTXT=NO in the TSOKEY00 member of SYS.1.PARMLIB before starting TSO/VTAM.

- GTF trace of SVC 93 and SVC 94 entries.

  See the *Debugging Handbook*, Volume 1, for information on interpreting SVC 93 and SVC 94 entries.

- TGET/TPUT trace.

  The TGET/TPUT trace records all TGET/TPUT/TPG data except address space ID TPUTS.

  You can get the TGET/TPUT trace by issuing the MODIFY TRACE command with TYPE=TSO. The MODIFY command is described in *VTAM Operation*.

- Dump of CSA storage and the user's address space.

  This is only required for address space ID TPUT errors.

MVS The Screen Functions Improperly for the Mode

1. **Review the SVC 94 entries to determine the mode.** The STFSMODE and STLINENO macros set full-screen mode on and off.

   - Full-screen mode.

     In full-screen mode, the application handles screen management.

     If the problem relates to full-screen processing, review the information on full-screen mode in the *TSO Guide to Writing a Terminal Monitor Program*.

     If this does not describe the full-screen processing that you are experiencing, note the differences.

   - Line mode.

     In line mode, TSO/VTAM handles screen management.

     In line mode, the data generated by the application program is placed line by line down the screen. READY appears on the line below the data, and the cursor appears on the line below that. When the screen is full, TSO/VTAM sends a page prompt to the screen. When you press the ENTER key, TSO/VTAM clears the screen and sends any remaining data to the screen.

     If you enter data on the next to the last line of the screen, no page prompt is sent. Instead, TSO/VTAM clears the screen and reshows the data (or command) at the top of the screen.

     If this does not describe the line mode processing that you are experiencing, note the differences.

2. **Report to the group (TSO/VTAM or the application program) that appears to be responsible for the incorrect screen management.**

MVS Exception Responses

1. **Does the error indicator reflect an error in the RU portion of the PIU (sense = 1003 or 1005)?**

   If so, continue with this step.

   Otherwise, go to step 2.

   a. See what kind of TPUT was issued. (This is shown in the flag byte of the TPUT trace entry.)

   b. For a NOEDIT TPUT, TSO/VTAM should not change the data provided by the application program. Compare the data in the TPUT trace, which starts at X'2C' into the trace record, with the VTAM buffer contents trace.

      If TSO/VTAM has not changed the data, contact the group responsible for the application program that issued the TPUT.

   c. For a full-screen TPUT, was the data that is causing the error generated by the application program? (Data generated by the application program is present in the TPUT trace entry.)

      If so, contact the group responsible for the application program that issued the TPUT.

      If not, go to "Reporting Procedure" on page 68.

2. **Look at the error sense code.**

   If it is X'800A', the PIU is too long. This is probably a definition error. In this case:

   a. Look in the TH portion of the buffer contents trace entry to find out the length of the PIU that caused the error.

   b. See how MAXDATA is defined in your PCCU definition statement. The MAXDATA value that you code should be as large as the largest PIU that will be sent to the terminal by an application program.

      If you have more than one PCCU definition statement, check to see that the right one is being used.

3. **If there are other error indicators**, get the documentation shown in step 5 on page 59 and continue with that procedure.

4. **If you have not resolved the problem**, go to "Reporting Procedure" on page 68.

MVS Extra or Missing Data

1. **Is the problem with input (data received by the application)?**

   a. Is too much data being passed to the application?

      - If previous input is added to the end of the current input, the modified data tags may have been set improperly in previous TPUTS. Go to the "Reporting Procedure" on page 68.

      - If the backspace key, character delete key, or line delete key is not functioning properly, look for SVC 94 entries that may have changed the default in the STCC macro.

        If this does not account for the problem, get a dump of the nucleus, the CSA, and the user's address space. Then go to "Reporting Procedure" on page 68.

   b. Is insufficient data being sent to the application program?

      - Find out which TGET option was used. TSO/VTAM edits data sent from the terminal before it passes it to the application program. The type of editing that TSO/VTAM does depends on the TGET option. Certain characters may be deleted, such as control characters, aid characters, and set buffer address (SBA) sequences. Look at the flag bytes in the TGET trace entry to see which TGET option was specified. TGET options are described in the *TSO Guide to Writing a Terminal Monitor Program.*

      - Find out if any user edit exits are involved. User edit exits are listed in *VTAM Installation and Resource Definition.*

      - **If you have not resolved the problem**, go to "Reporting Procedure" on page 68.

2. **Is the problem with output (data sent by the application)?**

   a. Was extra data sent to the screen?

      TSO/VTAM should not generate any printable data. Compare the data portion of the TGET/TPUT trace with the data in the VTAM buffer contents trace.

      If you see printable data in the buffer contents trace that is not in the TGET/TPUT trace, go to "Reporting Procedure" on page 68.

      If you see the same data in both traces, contact the group responsible for the application program that issued the TPUT.

   b. Was data from the application lost?

      TSO/VTAM does not generally delete data sent by the application program unless it is doing reshow processing. In reshow processing, TSO/VTAM deletes the full-screen TPUT and sends a reshow character (X'6E') to the full-screen application program. This is shown in the TGET trace entry. Compare the data portion of the TPUT trace entry with the data in the VTAM buffer contents trace entry to see if any data has been deleted. Determine whether this is a TSO/VTAM problem or an application program error and contact the appropriate group.

      Note: For more information on reshow processing, see the *TSO Guide to Writing a Terminal Monitor Program*.

3. **If you have not resolved the problem**, go to "Reporting Procedure" on page 68.

MVS Data Is Misplaced on the Screen or Page

1. **Does data wrap around the screen?** (Wrapping means that data fills the line and splits inappropriately between lines.)

   If it does, continue with this step.

   Otherwise, go to step 2.

   a. Find out which TPUT option was issued. Bytes X'12' and X'13' of the TPUT trace entry contain the option flags.

   b. For a NOEDIT TPUT, TSO/VTAM does not edit data, and therefore does not change any SBA sequences that may be issued by the application program. Contact the group responsible for the application program that issued the TPUT.

   c. For a full-screen TPUT, TSO/VTAM does not generate SBA sequences to place the data on the screen. TSO/VTAM changes invalid buffer addresses to printable characters (X'7A'), but this would not cause the screen to wrap.

      If the symptom is incorrect screen size, go to "MVS Screen Size Problems" on page 86.

      Otherwise, contact the group responsible for the application program that issued the TPUT.

   d. If the data was sent without a full-screen or NOEDIT option, go to "Reporting Procedure" on page 68.

2. **If the problem is not on a display terminal**, go to "Reporting Procedure" on page 68.

3. **Are SBA sequences correct?** Check the type of TPUT that was issued. Bytes X'12' and X'13' of the TPUT trace entry contain the option flags. For full-screen and NOEDIT TPUTS, the application program usually generates the SBA sequences that determine where data is placed on the screen.

4. **Is the buffer address invalid?**

   - If the buffer address is invalid for the screen size, contact the group responsible for the application program.

   - If an invalid buffer address (shown in the TPUT trace entry) is translated to a printable character (X'7A'), the data will go to an incorrect location on the screen.

   - If the buffer address is valid for the terminal in its present screen size, but was translated, go to "Reporting Procedure" on page 68.

5. **If this is the first non-full-screen TPUT following a full-screen TPUT**, look for an SVC 94 trace entry for STLINENO. This macro may be issued by a full-screen application to indicate which line the next non-full-screen data should appear on.

   If the data was placed by the STLINENO macro, contact the support group for the application program that issued the macro.

6. **If you have not resolved the problem**, go to "Reporting Procedure" on page 68.

MVS

Data Appears To Be Translated Incorrectly

Incorrect output is the main symptom of this problem. Sometimes the incorrect output is colons. The problem is probably related to the TPUT option that was specified.

1. **Look at the option flag bytes** (X'12' and X'13') in the TPUT trace entry to determine what options were used.

2. **See the** *TSO Guide to Writing a Terminal Monitor Program* to see what editing occurs for each type of TPUT option.

3. **If the incorrect output consists of colons**, determine what data from the TPUT trace entry is being edited into the printable character X'7A', a colon.

4. **See if TSO/VTAM is editing correctly.**

   Note: Many applications use TPUT options that do extensive editing and translation. This allows many different hardware devices to communicate with the application program without causing I/O errors. You may need to write your own user edit exits to make sure that all characters that are valid for your terminals appear on the screen, especially if you are using type 1 logical unit devices.

5. **If the editing does not appear to conform to the options specified**, check for user edit exits or translation tables. (*VTAM Installation and Resource Definition* has a list of these exits.)

   If you have exits or translation tables, verify that they are not causing the problem.

6. **Did the application program send incorrect data to VTAM?** The data is shown in the data portion of the TPUT trace entry.

   If so, the problem is in the application program.

7. **If you have not resolved the problem**, go to "Reporting Procedure" on page 68.

MVS Screen Size Problems

This section provides documentation requirements and diagnosis procedures for screen size problems. Two major types of errors occur:

- The screen never operates in the expected size
- The screen is not always the expected size when you change modes.

Choose the one that most closely matches your symptoms and follow the procedure for that problem.

Recommended documentation includes:

All of the traces may not be required. Read the diagnosis procedure before you get them.

- VTAM buffer contents trace.

  To see the data in the buffer contents trace, set CONFTXT=NO in the TSOKEY00 member of SYS.1.PARMLIB before starting TSO/VTAM.

- GTF trace of SVC 93 and SVC 94 entries.

  See the *Debugging Handbook*, Volume 1, for information on interpreting SVC 93 and SVC 94 trace entries.

- TGET/TPUT trace.

  The TGET/TPUT trace records all TGET/TPUT/TPG data except address space ID TPUTS.

  You can get the TGET/TPUT trace by issuing the MODIFY TRACE command with TYPE=TSO. The MODIFY command is described in *VTAM Operation*.

MVS The Screen is Never the Expected Size

This is probably a definition problem.

1. **If you are using a USS command to log on**, try logging on without it.

   If this corrects the problem, review the use of the USS command in *VTAM Customization* and check the following:

   - Is the terminal a non-SNA 3270?

     If it is, does the USS command include a USSPARM macro for the logmode?

     If so, VTAM ignores the logmode name from the terminal definition statements and uses its own default BIND image instead.

   - Does the USS command establish a default logmode name?

     If so, the default name overrides the name in the terminal definition statement.

2. **If this is not a USS command problem**, check to see if the PSERVIC operand of the MODEENT macro is coded correctly. *VTAM Installation and Resource Definition* explains how to do this.

   - Are primary and alternate sizes coded correctly?
   - For screen switching, is BINPRESZ coded correctly as X'7F'?

3. **If you have not identified the problem**, look at the BIND that is sent. This is shown in a VTAM buffer contents trace of the logon, or in a dump (get the user's address space, nucleus, and CSA, and look at TSBXBIND).

   - If the BIND image is not what you expected, check the LU definition statement for an incorrect LOGMODE or DLOGMOD parameter.
   - If no logmode table or DLOGMOD operand is specified, no PSERVIC is passed to the TSO/VTAM logon exit routine. In this case, TSO/VTAM issues an INQUIRE DEVCHAR macro and VTAM indicates that the terminal is a logical unit. TSO/VTAM then uses the SCRSIZE operand found in TSOKEY00. The default value for SCRSIZE is 480 (12 rows and 40 columns).

4. **If you have not resolved the problem**, go to "Reporting Procedure" on page 68.

MVS The Screen Is Not the Expected Size for the Mode

Full-screen mode. The application program controls screen management in full-screen mode. The primary (small) screen size is considered by TSO/VTAM as the default size. The application program can control screen size by sending write commands in TPUTs that it issues. The write commands are X'F5', erase write, and X'7E', erase write alternate. The application program issues X'F5', erase write, to set the primary screen size, or X'7E', erase write alternate, to set the alternate (large) screen size.

If neither size is issued, the screen will remain whatever size was in effect when the application program entered full-screen mode.

Line mode. TSO/VTAM controls screen management in line mode. It generally uses the large (alternate) screen size when processing TPUTs in line mode. You can use the TSO TERMINAL command and STSIZE macro during a session to change the screen size for non-full-screen processing.

Using output from the VTAM buffer contents trace, the TGET/TPUT trace for TSO/VTAM, and the GTF trace of the SVC 93 and SVC 94 entries, try to locate the source of the problem.

1. **Check the SVC 94 trace entries** to see whether the processing is in full-screen mode or line mode. The STFSMODE and STLINENO macros set these modes on and off.

   If the processing is in line mode, go to step 6. For full-screen mode, continue with step 2.

2. **Note if the incorrect screen size is related to entering or exiting full-screen mode.**

3. **Locate the TPUT trace entry for the data that appears on the screen when the screen is the wrong size**. Determine the TPUT options for this TPUT and the one that precedes it by looking at the option flag bytes.

   If either is a full-screen TPUT, look at the first data byte.

   If the first data byte is an escape character (X'27'), the write command that follows has been specified by the application program. This write command should determine the screen size.

   If the write command is different in the buffer contents trace, go to "Reporting Procedure" on page 68.

4. **If there is not an escape character in the first data byte**, is a write command (X'F1', X'7E', or X'F5') provided?

   If one of these write commands exists, continue with this step.

   If not, go to the next step.

   If this write command is different in the VTAM buffer contents trace, go to "Reporting Procedure" on page 68.

   If the same write command appears in both traces, contact the group responsible for the application program.

5. **If a write command is not provided in the TPUT data**, and processing is in full-screen mode, does the write command generated by TSO/VTAM set the same screen size as the last write command provided by the full-screen application program? To determine this, compare the write command in the buffer contents trace entry with the last one provided in a TPUT trace entry. The write command is located in the data portion of the TPUT trace, at X'2C' into the entry.

   If the write commands are the same, go to "Reporting Procedure" on page 68.

   If the write commands are different, contact the group responsible for the application program.

6. **If processing is not in full-screen mode**, did the STSIZE macro set the screen size? To determine this, look in the SVC 94 trace entries. An entry code of X'0A' in the high order byte of register zero indicates that the STSIZE macro set the screen size.

   If the screen size was set by the application program, contact the group responsible for the application program.

   Otherwise, go to "Reporting Procedure" on page 68.

7. **If you have not resolved the problem**, go to "Reporting Procedure" on page 68.

MVS Performance Problems

This section provides documentation requirements and diagnosis procedures for performance problems. Use this section along with the information in "Performance Procedure" on page 62. In addition to the documentation required in "Performance Procedure," you will need:

GTF trace of SVC 93 and SVC 94 entries.

See the *Debugging Handbook*, Volume 1, for information on interpreting SVC 93 and SVC 94 trace entries.

1. **Are wait and hold options slowing response time?**

   The application program may be issuing TPUTs and TGETs with the wait or hold options. The wait option indicates that the application program should not regain control until output data has been placed on the output queue (TPUT) or input is available (TGET). A TPUT with a hold option indicates that control should not be returned to the application until the data has reached the terminal. These options may be necessary for screen management, but prolong response time and increase the number of times the address space is swapped.

2. **Is an external system resource slowing response time?**

   If all users must access the same resource, such as a system catalog, performance deteriorates. This problem is especially severe when exclusive ENQs are used to control access to the resource. To improve performance, redistribute resources.

3. **Are high and low buffer extents set at inappropriate values?**

   If the high and low buffer extents are too close together, output wait conditions occur. Buffer extents are specified in TSOKEY00, a member of SYS.1.PARMLIB. Tune the values of the high and low buffer extents to get optimum performance. Tuning is explained in *VTAM Customization.*

4. **Are APPL definition statements coded correctly?**

   Code the AUTH=NVPACE operand on all APPL definition statements for TSO/VTAM.

   If you do not set NVPACE, VTAM indicates that it has already received input data, instead of queuing the response until it receives the input data. Also, the swap count is incremented by two every time the ENTER key is pressed.

5. **Are pacing values set correctly for local SNA terminals?**

   VTAM ignores the NVPACE operand for sessions with logical units in a local major node. Therefore, you must set non-zero pacing values for these logical units.

6. **Is the MVS performance group specified correctly?**

   Set the application program's performance group five to ten percent lower than VTAM's performance group. To see the application program's performance group specification, look at the dispatching priority in the task's TCB.

7. **If you have not resolved the performance problem**, go to "Performance Procedure" on page 62.

Chapter 5. VM Procedures for VSCS Problems

Chapter 5. VM Procedures for VSCS Problems

VM Documentation You Should Collect for All VSCS Problems

Always run VSCS with one or more trace options active. The VSCS console communication services (CCS) trace option defaults to ON and shows what is happening between VSCS and CCS without significantly impacting system performance. (In VTAM V3R1.2 and V3R2, the VSCS VTAM trace option also defaults to ON.) When you need to re-create the problem, turn on all of the VSCS trace options for the failing logical unit (LU), or for all logical units if the problem is not related to a specific logical unit. To prevent the trace table from wrapping, run traces externally or increase the size of the VSCS internal trace table. For more information on setting the size of the trace table, see *VTAM Installation and Resource Definition.*

When getting a VSCS dump, use the dump formatting routine because control blocks are easier to locate in formatted output.

When you report a problem, be prepared to provide the IBM Support Center with the following information:

- VSCS link map
- VSCS maintenance list
- VSCS DTIUSER specifications (unless default parameters are used)
- VM directory entries for VTAM and VSCS (if VSCS is run in a separate virtual machine)
- CP user-specified information in DMKRIO, DMKSYS, and DMKSNT (particularly for installation and performance problems)
- GCS user-specific information (particularly for installation and performance problems)
- VTAM user-created logmode tables (particularly for logon and incorrect output problems).

Note: If VSCS dumps are enabled, and any VSCS dumps were created prior to the problem you are reporting, please have those dumps available.

"VM General Procedures to Diagnose Logical Unit Hang Problems" on page 94 and "VM Determining Device Characteristics" on page 97 explain how you can use the VSCS DISPLAY ID luname command to determine the status of a particular logical unit. In some cases you can also determine where the problem is most likely to be, or which products are involved. This can help you determine the type of documentation the IBM Support Center will require.

"VM VM/SP Logo Problems on an SNA Device" on page 113 is another new diagnostic procedure, introduced in V3R1.2.

VM General Procedures to Diagnose Logical Unit Hang Problems

If the problem is for one or more logical units, but not for all logical units, use the following commands to determine additional information as to where the problem is, from an external viewpoint:

1. Issue a VSCS DISPLAY ID=luname command. Use the display output to follow the steps below.

   a. Has VSCS issued a VTAM CLSDST on the logical unit? If so, then the logical unit was not found.

   b. Is message DTIS611 displayed? If so, is this a "DIALED" logical unit?

   c. Is message DTIS661 displayed? If so:

      - Note the current Presentation Services (PS) state of the logical unit.

        If the current state is one of the following—CIP, CLP, IIP, INVL, LGF, LIP, LUP, PRG, or SVR—then the logical unit is in an internal state and is waiting for another process to complete before continuing processing for this logical unit.

        If the current state is AWL, VSCS is waiting for an interrupt from the logical unit before displaying the VM/SP logo.

        If the current state is FSS (full-screen support mode), then the application is responsible for restoring the keyboard, not VSCS. If the application does not restore the keyboard, SNA logical units are left in a hang condition and cannot be reset.

      - Determine which response mode is active for this logical unit.

        – Exception response mode (ER; VTAM V3R1.2 and later)

        – Definite response mode (DR; VTAM V3R1.2 and later)

        Note: ER/DR refers to the mode that is allowed for this logical unit, not the mode requested for the current operation.

        See the explanations of DTIGEN options in *VTAM Installation and Resource Definition* and *VTAM Customization* for more information on exception response mode.

      - Determine whether recovery is in progress for this logical unit.

        SCIP indicates that a network failure occurred and that VSCS is attempting to reacquire the logical unit.

        RP indicates that recovery is pending. VSCS is waiting for a response before initiating recovery for the logical unit (exception response mode only).

        RIP indicates that recovery is in progress; VSCS is attempting to recover the logical unit (exception response mode only).

      - Determine the current VTAM Services (VS) state of the DEVICE.

        If the current state is TERM, then VSCS is attempting to terminate the logical unit.

        If the current state is NORM, then the logical unit has not been terminated.

d. Is message DTIS65I displayed? If so:

- Does this message contain any IUCV PATHIDS?

  If so, then a CP connection is currently active.

  If this message just contains blanks, then a CP connection is not currently active.

- What is the current RPL command?

- Is the SEND RPL currently busy? (VTAM V3R1.2 and later)

  If so, then VSCS is waiting for a VTAM command to complete before doing anything else on the user's behalf. Several conditions can cause this to occur, including:

  - Missing I/O interrupt
  - Blocked PAB in VTAM
  - Network problem with device or routes to the device
  - SIO never issued.

  Issue the VTAM D net,ROUTES TEST=YES to determine if the route is intact. If it is intact, you need VTAM BUFFER USE and LINE trace to determine whether the device responded to the last request.

e. Is message DTIS59I displayed? (VTAM V3R1.2 and later)

Note: This message will only be displayed if the RECEIVE SPECIFIC option is active.

If so, then is the RECEIVE SPECIFIC RPL busy?

If it is not busy, then VSCS cannot receive any input from the logical unit. Normally, this occurs only when the device is being returned to VTAM, is recovering from an error, or is processing input data for the logical unit.

If the current PS state is AWL, then the logical unit is hung, because VSCS is waiting for an interrupt from the logical unit that will never be received.

If a VSCS internal error did not occur for this condition, then get a dump of the VSCS virtual machine before recovering the logical unit by use of the VTAM VARY or VSCS FORCE command.

f. Is message DTIS68I displayed?

If so, then are the SEND or RECEIVE counts incrementing for the logical unit, or are both currently zero?

If both are zero, you need additional information to determine whether the problem is in VSCS or VTAM. For example:

1) If the problem occurred during logon, determine whether the device characteristics are correct for this logical unit. Check message DTIS65I. If the current RPL command is OPNDST and the RPL is busy, determine why VTAM is not not completing the request.

2) If SCIP is included with message DTIS66I, determine why recovery of the logical unit is not completing.

3) If the current RPL command in message DTIS65I is CLSDST, and RPL is busy, determine why VTAM is not completing the request.

2. Issue a CP QUERY NAMES to see if the logical unit name is still connected to the user ID.

3. If the user is still connected to CP, and the current PS state in message DTIS661 is **not** FSS, dump the VSCS virtual machine. Send a message to the hung user and see if that releases the logical unit.

4. Issue a VSCS QUERY to determine:

   - The current VSCS trace options that are active
   - The buffer status (LIFO or FIFO queuing).

   Note: If the logical unit has been hung for a while, the internal trace table has wrapped, unless the trace was running externally, or was running for only the logical unit in question.

VM Determining Device Characteristics

To determine if the right VTAM logmode entry was used, issue the VSCS DISPLAY ID=luname command. The resulting messages will indicate how the device characteristics appear to VSCS.

1. Is message DTIS61I displayed?

   If so, note the device type: 3277, 3278, 3767, 2741, TWX, or PRT.

   Specifying the wrong VTAM logmode entry, or using the VTAM default logmode entry, leads to incorrect operation of the logical unit. In some cases, message DTIS99I from DTIPIPUT will occur when input is received from the logical unit.

   Also note the device model. This determines the screen size that is used for display devices. (Screen size is displayed in message DTIS63I.)

2. Is message DTIS62I displayed?

   If so, what is the logical unit type?

   0 Non-SNA display or printer device

   1 Keyboard/printer device

   2 SNA display or printer device

   3 SNA printer device.

   What are the device characteristics?

   - EDS indicates that extended data stream commands are supported (for example, use of the Write Structured Field commands).
   - NEDS indicates that extended data stream commands are not supported.
   - APL indicates whether the device uses the NEW or OLD APL character set.
   - COLOR indicates whether the device supports color.
   - SHARED indicates that the device is a shared printer.

3. Is message DTIS63I displayed?

   Note: This message is issued for displays only.

   If so, note the screen size from the PSERVIC field of the VTAM logmode.

   Also note the alternate screen size from the PSERVIC field of the VTAM logmode (or, for some devices, from the Write Structured Field Query reply data).

4. Is message DTIS64I displayed?

   Note: This message is issued for keyboard/printers only.

VM Symptoms of VSCS Problems

This chapter divides VSCS problems into the following general problem types.

Note: Documentation requirements may differ if VSCS is running in a separate virtual machine from VTAM. It is also important to understand that VSCS errors do not always result in a message, user disconnection, or user logoff.

1. "VM Initialization Problems" on page 101
   - Initialization never completes
   - VSCS terminates immediately upon starting
   - A loop occurs during initialization
   - Other initialization problems.
2. "VM Termination Problems" on page 109
   - Termination never completes
   - VSCS terminates prematurely
   - User's session terminates prematurely
   - Operator commands terminate VSCS unexpectedly
   - Other termination problems.
3. "VM Logon Problems" on page 115
   - First logon attempt fails
   - Logon never completes (logical unit is hung).
4. "VM Logoff and Disconnect Problems" on page 117
5. "VM Wait and Loop Problems" on page 118
   - All VSCS logical units are hung
   - One or more VSCS logical units are hung.
6. "VM Message Problems" on page 135
   - Message DTIS991 indicates an internal error.
7. "VM Incorrect Output Problems" on page 138
8. "VM Performance Problems" on page 143
   - Paging and storage problems
   - Slow response time on terminals.
9. "VM Other VSCS Problems" on page 144
   - SNA dial problems
   - Printer sharing problems
   - Operator command is not processed properly.

VM Overview of Normal VSCS Initialization Process

Below is an overview of what takes place during normal VSCS initialization. This is the procedure that VSCS follows each time it is initialized. Figure 20 shows the components involved in initialization.

GCS processes IUCV requests from VSCS that affect path ID setup and breakdown. IUCV external interrupts are handled by GCS, which gives control to VSCS after some validation is done. All other IUCV requests and responses issued by VSCS are handled directly by CP.

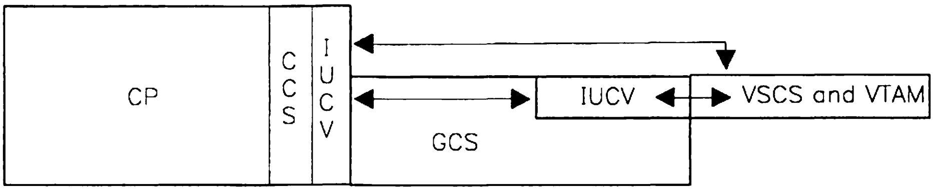


Figure 20. VM Components Involved in VSCS Initialization

1. VSCS storage is initialized with DTIUSERX, if specified.
2. VSCS issues message DTIC01I.
3. VSCS issues IUCVINI SET to GCS.
   - GCS sets return code after processing request.
4. VSCS issues IUCV QUERY to CP.
   - CP responds with MAXCONN and EIB size.
5. If VSCS trace is active, message DTIS74I is issued.
6. VSCS issues global IUCVCOM CONNECT and if successful:
   - CP provides path ID and a return code back to GCS
   - GCS passes path ID and a return code back to VSCS
   - **VSCS issues a system WAIT at this point**
   - CP generates an external interrupt for "CONNECT COMPLETE"
   - GCS passes "CONNECT COMPLETE" back to VSCS
   - CP generates an external interrupt for VM/SP logos
   - GCS notifies VSCS of a pending message.
7. VSCS issues IUCV RECEIVE
   - VSCS processes "CONNECT COMPLETE"
   - **IUCV should be functional**.
   - CP sends VM logo data to VSCS.
8. VSCS processes VM logo information.
9. VSCS issues message DTIC02I.
10. VSCS attaches VTAM Services and Utility Services subtasks to GCS.
11. VSCS issues VTAM OPEN ACB through GCS to VTAM.

12. VSCS schedules a work element containing the VTAM user ID to Presentation Services.

13. VSCS issues message DTIV04I.

    - **VTAM interface should be functional; VSCS is fully functional.**

14. VSCS issues an IUCV SEND to supply CP with the VTAM user ID.

VM Initialization Problems

First you should look for initialization problems in these areas:

1. A storage shortage could occur if:

   - VSCS is in the same virtual machine as VTAM in a large environment
   - VSCS is in a small virtual machine
   - VSCS DTIGEN parameters are incorrect
     - Trace table (too large for virtual machine size)
     - Printer table (too large for virtual machine size).
   - MAXCONN in the VM/SP directory entry for the VSCS user ID.

2. Misspelled or missing parameters: VSCS initialization uses most of the user parameters coded for:

   - CP
   - GCS
   - VTAM APPL definition
   - DTIUSERX (if specified)
   - VM directory entry for the VSCS virtual machine.

3. Linking to the wrong disk for the VSCS load module.

4. The disk containing the VSCS load module has changed since it was last accessed by this virtual machine.

5. The VTAM application program major node is inactive or invalid.

6. The wrong level of VTAM is specified.

7. The VM directory is incorrect (the user ID for the IUCV option card must be '*CCS').

If none of the previous items is causing the problem, follow the procedure for your symptom:

- Initialization never completes (a wait follows message DTIC01I or DTIS74I).
- VSCS terminates immediately upon starting.
- A loop occurs (using 90 to 100% of CPU time).
- Other initialization problems occur.

VM Initialization Never Completes (A Wait Follows Message DTIC01I)

1. Is SNA enabled?

   A delay during initialization is normal; VSCS must wait for external interrupts for "CONNECT COMPLETE" and the VM/SP logo. A long wait indicates a problem. ENABLE SNA is used to open a path from IUCV to CCS, and if SNA is disabled, then CCS will never process the connection request.

   Note: If SNA is disabled and VSCS started, it may appear that most operator commands do not complete.

   To make sure that SNA has been enabled, use one of these CP commands:

   ```
   #CP ENABLE SNA
   #CP ENABLE ALL
   #CP ENABLE SNA user ID
   ```

   For more information on the ENABLE command, see the *CP Command Reference.*

2. Are GCS and the VTAM recovery machine active?

   Issue the GCS command QUERY ETRACE to see if GCS is active. If it is not active, you will get the message "UNKNOWN CP/CMS COMMAND." To activate GCS, use the CP IPL GCS command. Use the CP QUERY NAMES command to see if the VTAM recovery machine is active. If it is not active, you need to initialize it.

3. Does GCS have control register 0 bit 30 on?

   To determine if control register 0 is set on, display the control registers from the VSCS operator console (DISPLAY X'0') or look in a dump of the VSCS virtual machine (all registers are shown at the beginning of the dump). GCS must set the control register during GCS initialization in order to receive IUCV external interrupts from CP. If control register 0 bit 30 is off, verify that the VSCS directory entry is correct (or the VTAM directory entry, if VSCS will run in the same virtual machine as VTAM). Continue problem resolution with GCS.

4. Did CP process the IUCV CONNECT request without error? ("Without error" means that no error messages were issued by CP, GCS, or VSCS.)

   If so, continue with step 5. If not, the problem is in CP or GCS.

   This is shown in the CP internal trace entries X'15' (IUCV) and X'16' (CCS) in the CP internal trace table. Before starting VSCS, limit the amount of the external trace generated by issuing the CP command CPTRAP 15 16. For more information on the CP trace, see the *VM/SP Operator's Guide.*

5. Did GCS reflect the external interrupts (for "CONNECT COMPLETE" and the VM/SP logo) to VSCS?

   To determine this, look at:

   - CP internal trace of IUCV and CCS requests
   - GCS external trace of external interrupts received
   - VSCS trace with CCS active.

   (See the *VM/SP System Programmer's Guide* (for VM/SP R4), or the *VM/SP Facility for System Programming* (for VM/SP R5) for CPTRAP information. See the *GCS Macro Reference* for ETRACE information.)

   If GCS did not reflect the external interrupts to VSCS, the problem is in GCS.

If all of these conditions are met, the IUCV will be properly initialized and you will see message DTIC02I.

6. Was VSCS previously active, and now cannot be restarted?

   If this is the case, the problem may arise from an abend or premature termination (message DTIC07I) of VSCS during the previous VSCS initialization (assuming CP has not been reinitialized since then). If CP and GCS did not clean up after the abend, VSCS cannot restart. Check the operator console listing to see if any problems were encountered on the previous termination. If there were problems, re-IPL GCS. If initialization still fails, re-IPL CP.

7. If you do not have enough information, or cannot solve the problem, re-create the problem with the GCS, CP, and VSCS traces active externally. Use your best judgment in trying to decide which product has the problem. If you suspect GCS or VSCS is causing the problem, get a VSCS dump. Otherwise, pursue the problem with CP. (See the *CP Command Reference*.)

8. Go to "Reporting Procedure" on page 68.

VM VSCS Terminates Immediately upon Starting

1. How far did initialization go?

   Get the operator console listing, and use "VM Overview of Normal VSCS Initialization Process" on page 99 to evaluate the console messages to see how far the initialization progressed. VSCS messages that occur during successful initialization are DTIC01I, DTIC02I, and DTIV04I in this order.

   Note: Messages issued for starting the VSCS internal trace are not included here.

   If you did not get DTIC01I, the problem may be:

   - GCS is not active
   - VSCS is not linked correctly
   - The VSCS virtual machine is not large enough to hold the DTIISTRT load module
   - Incorrect DTIGEN parameters.
   - GCS is dumping the VSCS virtual machine.

   If you got DTIC01I but not DTIC02I, the problem may be:

   - VM directory entry for the VSCS virtual machine is wrong, or
   - CP or GCS found a problem (other than an abend) while processing an IUCV request or response. VSCS should issue a message to identify the failure and reason. See step 2 on page 104.
   - GCS is dumping the VSCS virtual machine.

   If you got DTIC01I and DTIC02I, but not DTIV04I, the problem may be one of the following:

   - VTAM OPEN ACB error. (VTAM return codes show what caused this.)
   - VTAM APPL definition is incorrect.
   - VTAM APPL definition is not active.
   - VTAM storage shortage.

- VSCS storage shortage.
- VSCS DTIUSERX problem, such as incorrect parameters. (See "VM Other Initialization Problems" on page 107.)
- GCS is dumping the VSCS virtual machine.

2. Was an error message issued?

   If so, try to find the source of the message. If not, go to step 5 on page 105.

   - CP messages begin with DMK
   - GCS messages begin with CSI
   - CMS messages begin with DMS
   - VSCS messages begin with DTI
   - VTAM messages begin with IST.

   If GCS issued the message, and it is not an abend, get a VSCS dump. See "VM VSCS Dump" on page 155 for more information on the VSCS dump. Then go to "Reporting Procedure" on page 68.

3. Did you get an error message with a return code or IPTYPE?

   If so, find the explanation for it, and determine what component issued the message. If not, go to step 4 on page 105.

   Figure 21 shows the source of return codes and IPTYPES.

Message	Source of IPTYPE or Return Code
DTICxxx	• GCS • VM • VSCS (very rarely)
DTIIxxx	• GCS • VSCS
DTIPxxx	• VM • VSCS (very rarely)
DTISxxx	• GCS • VSCS
DTIVxxx	• GCS • VSCS • VTAM

Figure 21. VM Source of IPTYPE or Return Code During Initialization

- If you got message **DTIC05I with return code 878**, then GCS had a GETMAIN failure getting storage for an IUCV request. See "VM Storage-Related Problems" on page 140 for more information.
- If you got message **DTIC07I with IPTYPE=03**, the previous termination did not complete, or two IUCV CONNECTS with the same name have been issued from the same GCS group. IPTYPE=03 is an IUCV SEVER request from CP, indicating CP is rejecting the connection request.

If you can re-create the problem, follow these steps:

a. Use the DTIGEN options to activate VSCS internal trace options, including the external trace option.

b. Initiate CPTRAP.

c. Start VSCS using the newly created DTIUSERX.

d. After the failure occurs, stop CPTRAP.

e. Process the CPTRAP output, which should contain the failing sequence.

- If you got message DTIC09I, you need to determine which component generated the return code. See the *GCS Macro Reference*, the *VM/SP System Programmer's Guide* (for VM/SP R4), or the *VM/SP Facility for System Programming* (for VM/SP R5) for information on return codes and their meanings.

- If you got a different error message, look in the *GCS Macro Reference*, the *VM/SP System Programmer's Guide* (for VM/SP R4), or the *VM/SP Facility for System Programming* (for VM/SP R5) for an explanation of the return code.

  If the message has an IPTYPE, the IPTYPE is one of the following:

  1 Pending Connection
  2 Connection Complete
  3 Severed Connection
  4 Quiesced Condition
  5 Resumed Connection
  6 Incoming Priority Reply
  7 Incoming Reply
  8 Incoming Priority Message
  9 Incoming Message.

4. Did you get an error message without a return code or IPTYPE?

   If so, does the message text adequately explain the problem? If you can fix the problem by correcting user-specified parameters, do so. If not, get the VM directory entry for the VSCS virtual machine, VSCS installation information, and GCS specified parameters. Then go to "Reporting Procedure" on page 68.

   Note: Figure 22 on page 106 shows all the VSCS messages that can be issued for an initialization problem.

5. Is the problem an abend?

   If an abend occurs during initialization, the problem is most likely caused by an error in the installation process of VSCS. Find out how VSCS was installed and linked.

   An abend (0C2) may also occur if one of the following is true:

   - VSCS is not running in an authorized virtual machine.
   - The VSCS APPL definition does not specify AUTHEXIT=YES.

   If an abend (706) was received before any VSCS messages are displayed, then there was a failure linking the VSCS load module. That failure left the load module "non-executable."

   Otherwise, if the failure results from a link failure with a DTIUSERn load module, message DTII11I is issued.

If you cannot find and correct the error, get a VSCS dump (created automatically for an abend) and go to "Reporting Procedure" on page 68.

6. Is the problem something else?

   If you still cannot identify the problem, get a VSCS dump and go to "Reporting Procedure" on page 68. The VSCS dump is described in "VM VSCS Dump" on page 155.

DTIC03I VSCS INITIALIZATION FAILED--INSUFFICIENT STORAGE AVAILABLE

Note: The virtual machine is too small to initialize VSCS or the DTIGEN parameters are specified incorrectly.

DTIC04I VSCS INITIALIZATION FAILED--UNRECOVERABLE ERROR DETECTED

DTIC05I VSCS INITIALIZATION FAILED--IUCVINI SET MACRO FAILED WITH RETURN CODE nn

DTIC07I VSCS INITIALIZATION FAILED--GLOBAL CONNECT FAILURE, IPTYPE = nn

DTIC09I IUCV GLOBAL CONNECT FAILURE, RETURN CODE = nn

Note: This message is the result of an error from GCS or CP.

DTII03I DTIITASK COMMUNICATION SERVICES INITIALIZATION FAILED. VSCS TERMINATING.

DTII04I DTIITASK ATTACH FAILED FOR modname VSCS TERMINATING.

DTII06I DTIITASK IDENTIFY FAILED FOR modname RETURN CODE = nn

DTII14I modname ABNORMAL TERMINATION OF VSCS IS IN PROGRESS INITIATED BY modname

DTIP15I modname IUCV RECEIVE FAILURE, IUCV RETURN CODE = nn

Note: CP provides the IUCV return code.

DTIS38I modname VSCS ABEND FOR (PS|VS|US) TASK — SYSTEM COMPLETION CODE = nn

DTIS99I INTERNAL VSCS ERROR — modname n

Note: If this happens during initialization and you have supplied a DTIUSERX for initialization, it may be a DTIUSERX specification error.

DTIV01I modname FAILURE FOR reqtype RETURN CODE = nn nn

DTIV06I modname VSCS VTAM SERVICES INITIALIZATION FAILED, SHOWCB MACRO ERROR RTN CODES: R15 =nn R0 = nn

DTIV07I modname VSCS INITIALIZATION FAILED-OPEN ACB FAILURE, ACBERFLG = nn

DTIV09I modname VSCS VTAM SERVICES GENCB MACRO FAILURE FOR controlblkname RTN CODES: R15 = nn R0 = nn

Figure 22. VM Messages Indicating VSCS Initialization Problems

VM A Loop Occurs During Initialization

If 90 to 100% of CPU time is being used, and that is not a normal workload for the CPU, then the problem could be one of the following:

- GCS is dumping the virtual machine.
- A VTAM temporary storage shortage has occurred, and VSCS is trying the VTAM request again.
- VSCS, VTAM, or GCS may have a control block chained to itself, resulting in an infinite loop.
- There is a loop in CCS or IUCV, so that the VSCS virtual machine appears to be looping, but is not.

If VSCS is running in the same virtual machine as VTAM, it is very difficult to determine where VSCS is looping, because of the low priority at which VSCS is running. If you are using the PER trace to record the loop, the trace must run for at least one minute.

If a dump message is issued, see "VM VSCS Terminates Immediately upon Starting" on page 103 for information on how to proceed.

1. Determine amount of storage available.

   To determine the amount of VTAM storage available and which storage pools need to be enlarged, use either the SMS (buffer use) trace or the VTAM DISPLAY BFRUSE command. These service aids are described in Chapter 7, "Using VTAM Service Aids."

2. If you cannot find any errors from the display of the VTAM buffer pools, collect this documentation:

   - A few seconds of the CP virtual machine branch trace (described in the *CP Command Reference*)
   - VSCS dump
   - VSCS operator console listing
   - VSCS trace with CCS active, running before the loop started
   - VSCS trace with any other options active, running before the loop started
   - Series of loop addresses obtained with the CP DISPLAY PSW command or from the CP virtual machine trace output. This command is described in the *CP Command Reference.*

3. Go to "Reporting Procedure" on page 68.

VM Other Initialization Problems

1. Are parameters specified correctly in the DTIGEN macro?

   Figure 23 on page 108 shows messages issued for a problem involving incorrect parameters.

   The parameter value representing the DTIUSER number must be a number in the range 0 – 9, and one byte in length. If you do not get the correct DTIUSERX module in message DTI107I, re-access the VSCS LOADLIB and make sure you enter this command correctly:

   ```
   VSCS START PARM=x   (where x is the DTIUSER number)
   ```

If you are still experiencing problems, have an assembly listing of the DTIGEN macro available and go to "Reporting Procedure" on page 68.

DTII07I	USING INITIALIZATION PARAMETERS IN CSECT csectname
DTII08I	INITIALIZATION CSECT csectname NOT FOUND
DTII09I	INVALID INITIALIZATION START PARAMETER
DTII10I	DEFAULT INITIALIZATION PARAMETERS BEING USED
DTII11I	LOAD OF INITIALIZATION CSECT DTIUSERx FAILED, LOAD ABEND CODE = yyy, REASON CODE = zzz

Figure 23. VM VSCS Messages Useful in Analyzing a Parameter Problem

2. Is VSCS accessing the proper DTIUSERx?

   If you see message DTII10I, either you did not select the correct DTIUSERx, or the loadlib containing the proper DTIUSERx module is not accessible. Issue the CP QUERY LOADLIB command and see if the correct loadlib is displayed in the query output. If the correct loadlib is not displayed, reissue the CP GLOBAL LOADLIB command, specifying the proper loadlib and all of the other loadlibs displayed in the query output, then restart VSCS again. Make sure the VSCS LOADLIB contains the DTIUSER you are trying to use. If you have verified that the DTIUSERx is correct, and you still have problems, get the portion of the operator console listing showing VSCS initialization and a list of the VSCS LOADLIB.

   If you see message DTII11I, then the requested DTIUSERx was found, but an error occurred attempting to LOAD it. Check the LOADLIB containing DTIUSERx for errors.

3. Go to "Reporting Procedure" on page 68.

VM Termination Problems

VSCS issues one of the following messages to indicate termination is in progress:

DTI113I modname NORMAL TERMINATION OF VSCS HAS BEEN INITIATED BY mod2

DTI114I modname ABNORMAL TERMINATION OF VSCS IS IN PROGRESS INITIATED BY mod2

The DTI114I message is usually followed by a request for a dump.

If you receive message DTI113I, and you did not enter the VSCS HALT or QUIT operator command, and VTAM is not terminating, check for previous VSCS or VTAM messages. They may indicate why termination is in progress.

Problems during termination include:

- "VM Termination Never Completes"
- "VM VSCS Terminates Prematurely" on page 110
- "VM User's Session Terminates Prematurely" on page 111
- "VM Operator Commands Terminate VSCS Unexpectedly" on page 112
- "VM Other Termination Problems" on page 112.

Choose the symptom most like yours and follow the procedure for that problem.

VM Termination Never Completes

1. Was termination initiated by a VSCS operator command?

   If so, get a VSCS dump and go to "Reporting Procedure" on page 68. If not, issue the VSCS HALT command.

   Note: If the VSCS Utility Services subtask abended previously, normal termination cannot take place, and you must issue the VSCS CANCEL command to terminate VSCS.

2. Is VSCS still in a wait state?

   Display all active sessions with the VTAM DISPLAY LU and VSCS DISPLAY ID=luname commands. Issue a VTAM VARY INACT command for each active session. This should allow VSCS to continue the termination. If this does not complete termination, use the VSCS FORCE ID=luname command.

3. Is termination still not completing?

   Get the following documentation:

   - VSCS operator console listing
   - VSCS dump (see "VM VSCS Dump" on page 155).

   Use the VSCS CANCEL command to terminate VSCS.

4. Go to "Reporting Procedure" on page 68.

VM VSCS Terminates Prematurely

1. Did message DTIP15I appear?

   This indicates that Presentation Services caused the termination. Get the following documentation:

   - VSCS dumps, if available. (Generated by previous VSCS abends or VSCS internal errors)
   - VSCS operator console listing showing any messages that tell which module invoked VSCS termination.
   - Trace output from any VSCS traces active at the time of VSCS termination.

   Go to "Reporting Procedure" on page 68.

2. Did message DTIV01I appear?

   If you got this message, issue a VTAM BFRUSE command and see if the VTAM CRPL buffer pool has expanded. If the problem is not caused by buffer expansion, get the following documentation:

   - VSCS VTAM exit trace, if running at the time of the problem
   - A VSCS dump.

   Go to "Reporting Procedure" on page 68.

3. Did an ABEND0A9 occur?

   This abend code indicates that VTAM terminated while processing under control of the VSCS VTAM Services subtask. You must get this documentation:

   - VSCS VTAM exit trace if running at the time of the problem
   - The VSCS dump created for any previous VSCS abends.

   Go to "Reporting Procedure" on page 68.

4. Did VSCS terminate with abend code 878?

   If so, collect the following documentation:

   - The VSCS dump created for the abend.
   - VSCS global and storage control blocks. Process the VSCS dump with PRTDUMP, using VSCS dump formatter options 1 and 2. That formats the VSCS global and storage control blocks.
   - VTAM storage shortage documentation.

   Go to "Reporting Procedure" on page 68.

5. Did VSCS terminate with another abend code?

   If recovery is possible, a message indicates that recovery is in progress and processing will continue.

   If recovery is not possible, VSCS issues message DTII14I and begins termination. Recovery cannot occur for these abend codes:

   ```
   0A8 0A9 122 13E 222 322 422 522
   ```

   All but 13E and 222 automatically produce a dump.

In addition to this dump, get:

- VSCS trace with CCS active, at the time of the abend
- VSCS operator console listing showing messages around the time of the abend.

Go to "Reporting Procedure" on page 68.

6. Is there a storage shortage?

   A storage shortage can cause VSCS to terminate. See "**VM** Storage-Related Problems" on page 140 for more information.

7. Did any network failures just occur that involve VSCS logical units?

   If so, and if SCIP recovery is specified for VSCS, then both VSCS and VTAM require additional storage to perform recovery.

8. Is there a temporary VTAM storage shortage?

   Use the VTAM SMS (buffer use) trace or the VTAM DISPLAY BFRUSE to determine if any VTAM buffer pools are experiencing a shortage. These service aids are described in Chapter 7, "Using VTAM Service Aids." See *VTAM Customization* for information on VTAM buffer specifications. If you cannot resolve the problem by adjusting the depleted VTAM buffer pools, go to "Reporting Procedure" on page 68.

9. If you do not know the cause of the termination, collect the following documentation:

   - VSCS dump
   - VSCS operator console listing
   - Output from any traces active at the time of the problem.

   Go to "Reporting Procedure" on page 68.

VM User's Session Terminates Prematurely

1. Are there any messages on the VSCS operator console listing?

   Check the console for VSCS messages indicating that the logical unit is being terminated, or has just been terminated. These messages may indicate a reason for the termination.

2. Was a VTAM VARY INACT, VSCS FORCE, or CP FORCE command issued? If any of these operator commands were issued, VSCS responded by terminating the logical unit.

3. Did any errors occur during request processing for this device? If so, get the following documentation:

   - VSCS trace running externally with all options active (except FRE and GET, for VTAM V3R2)
   - VTAM internal trace, OPT=(API,MSG,PIU), MODE=EXT.

4. Did the termination occur right after a particular command was issued or a specific application program was started from a particular device type? If so, report the command or application program.

   If you can, re-create the problem with all VSCS trace options active, and report that information. Otherwise, provide as many details as you can about the device you were using, and about the problem.

5. If you have not resolved the problem, go to "Reporting Procedure" on page 68.

   Note: Use of exception response mode may increase the frequency of session terminations because of limited error recovery, which is considered normal operation for VTAM V3R1.2 and later.

VM Operator Commands Terminate VSCS Unexpectedly

Do commands other than HALT, CANCEL, or QUIT cause VSCS to terminate? If so, note what these commands are. Get the operator console listing and any dumps that were produced, and go to "Reporting Procedure" on page 68.

VM Other Termination Problems

If you have a different problem with termination, or you do not know what the problem is, get:

- VSCS dump
- VSCS operator console listing
- Output from any VSCS traces that are active.

Go to "Reporting Procedure" on page 68.

VM VM/SP Logo Problems on an SNA Device

Before looking into potential problems in VSCS logo processing, make sure that the logo appears correctly on a CP-controlled terminal.

If you have modified the VM/SP logo, see if the default VM/SP logo works. If it does, that indicates that your modified may not be supported on VSCS.

See *VTAM Customization* for any VSCS logo uniqueness.

1. Logo lines shifted all over the screen and logo is not readable.
   a. Verify that the appropriate VTAM logmode entry for the logical unit was used. (Use VSCS DISPLAY ID=luname command.)
   b. If VSCS data exit (DTIPDDSO) is coded, verify that the exit is not altering the start buffer addresses or the screen attributes in the outbound data stream.
2. Only the online message is displayed.

   This indicates that both the normal logo and the alternate logo exceeded the capacity of the device.
   a. Verify that the appropriate VTAM logmode entry for the logical unit was used. (Use VSCS DISPLAY ID=luname command.)
   b. Verify that either the normal or the alternate logo will fit on the screen size of the failing logical unit.
3. Only a single-line logo is displayed.

   For VTAM V3R1.2 and V3R2, this indicates that the normal logo, online message, and command lines would not fit on the screen because of the width of each line, or because the number of lines allowed was exceeded.

   For VTAM V3R1.1, this indicates that the online message and logo body would not fit on the screen.
   a. Verify that either the normal or the alternate logo will fit on the screen size of the failing logical unit.
   b. Verify that the appropriate VTAM logmode entry for the logical unit was used. (Use VSCS DISPLAY ID=luname command.)
4. Only the online message and logo body are displayed. (VTAM V3R1.1 only)

   Note: You must have VM/SP R5 to get the command line fields.
   a. Verify that the normal logo body plus two lines (online message followed by a blank line) and the command lines will fit on the failing logical unit.
   b. Verify that the appropriate VTAM logmode entry for the logical unit was used. (Use VSCS DISPLAY ID=luname command.)
5. Screen colors or highlighting are incorrect or missing.
   a. Verify that the appropriate VTAM logmode entry for the logical unit was used. (Use VSCS DISPLAY ID=luname command.)
   b. Check the modified logo to confirm that screen color and highlighting are inserted correctly.
   c. If the VSCS data exit (DTIPDDSO) is coded, verify that it is not altering the start buffer addresses or the screen attributes in the outbound data stream.

6. The logo fields are unprotected.

   a. Check the modified logo to confirm that screen color and highlighting are inserted correctly.

   b. Ensure that the embedded SET ATTRIBUTE commands are paired (balanced) in the normal logo.

   c. If the VSCS data exit (DTIPDDSO) is coded, verify that it is not altering the start buffer addresses or the screen attributes in the outbound data stream.

7. Command line problems.

   Note: Alteration of the command lines is not recommended.

   a. Cursor positioned wrong, or missing.

      The command lines provided by VM contain the Insert Cursor command. If this command is missing, then the cursor can be positioned anywhere.

   b. Command lines protected

      1) Ensure that the embedded SET ATTRIBUTE commands are paired (balanced) in the normal logo.

      2) If the VSCS data exit (DTIPDDSO) is coded, verify that it is not altering the start buffer addresses or the screen attributes in the outbound data stream.

   c. Input from command lines results in incorrect input.

      1) Verify that the appropriate VTAM logmode entry for the logical unit was used. (Use VSCS DISPLAY ID=luname command.)

      2) If the VSCS data exit (DTIPDDSI) is coded, verify that it is not altering the start buffer addresses in the data stream.

Go to "Reporting Procedure" on page 68.

If you decide to write your own logo, keep the following points in mind:

- Changes to the VM/SP logo are considered to be user modifications. For VSCS restrictions on altering the VM/SP logo, see *VTAM Customization.*

- For VM/SP R5, user modification of the command lines is not recommended.

- You cannot use the start buffer address (SBA) or insert cursor (IC) commands when you modify the logo body. Other screen commands may not be accepted by all types of display devices, and VSCS does not offer recovery procedures for any I/O errors that may result.

- The default colors are white for the online message and blue for the logo body and input area. If you change the colors, you must start with the online message. If you use start field commands to control the colors, you must protect the online message and logo body.

VM Logon Problems

1. Issue the VTAM DISPLAY and VSCS DISPLAY ID=luname commands and answer the following questions:

   - Does the session exist?
   - If the session exists, do the device characteristics match the logical unit characteristics?
   - If the RPL is busy, what is the current RPL command?

   See "VM General Procedures to Diagnose Logical Unit Hang Problems" on page 94 and "VM Determining Device Characteristics" on page 97 for more information.

2. Are you logging on to VSCS for the first time?

   If so, and if you are not using the VTAM-supplied logmode entries, the logmode entries are probably incorrect. If you are logging on from an MVS or VSE host, that host must provide customized logmode entries. Review the process for building your own MODTAB and logmode entries in *VTAM Customization* (VSCS requirements in the PSERVIC field).

3. Are you trying to log on in logo mode?

   When you first turn on the power for your terminal, the VM/SP logo appears on the screen. To log on, press ENTER, the PA1 or PA2 key, and start your input.

   Note: With VM/SP R5, command lines are included as part of the logo, which allows you to log on directly from the logo screen.

   If you cannot log on in logo mode, try logging on in no-logo mode. If the problem still exists, you are probably using the wrong logmode entry. (See "VM Determining Device Characteristics" on page 97.)

   If the problem goes away, a user-designed logo may be causing the problem. (See "VM VM/SP Logo Problems on an SNA Device" on page 113.)

   To pursue the problem further, collect the following documentation:

   - VSCS trace for the failing logical unit with all options active (except FRE and GET, for VTAM V3R2)
   - VTAM internal trace, OPT=API
   - VTAM buffer contents trace for the failing logical unit
   - Output from the VSCS DISPLAY ID=luname command, if possible
   - Description of what is on the screen
   - The VSCS console log.

4. Did you get message DTIP141I with a return code indicating that the IUCV global path is severed?

   If so, go to page 120.

   If not, issue VMEXIT. That should return the logical unit to VTAM. Then issue the VSCS DISPLAY command. If the logical unit is not found, the problem is with CP.

5. Is the terminal hung?

   - If so, was the VM/SP logo removed from the screen?

     – If so, the documentation you need is:

       — VSCS trace for the failing logical unit with all options active (except FRE and GET, for VTAM V3R2)

       — VTAM internal trace, OPT=API

       — VTAM buffer contents trace for the failing logical unit

       — The VSCS console log.

     – If not, go to step 3.

   - If not, is the VM/SP logo or VTAM USSMSG10 on the screen?

     – If so, the documentation you need is:

       — VSCS trace for the failing logical unit with all options active (except FRE and GET, for VTAM V3R2)

       — VTAM internal trace, OPT=API

       — VTAM buffer contents trace for the failing logical unit

       — The VSCS console log.

     – If not, go to step 6.

6. Get other documentation.

   In addition to any documentation you have collected so far, get the VSCS CCS and VTAM exit trace for the failing logical unit.

7. Are all terminals hung?

   If so, go to " **VM** All VSCS Logical Units Are Hung (Affects Logons and Current Sessions)" on page 119.

8. Go to "Reporting Procedure" on page 68.

VM Logoff and Disconnect Problems

If the terminal is hung, go to page 129.

1. Record the status of the terminal. For example, on a display screen, note what is in the status area, the last input or the last output to the logical unit.
2. Issue the CP QUERY NAMES command and note whether the logical unit name is known to CP.
3. Issue the VTAM DISPLAY and VSCS DISPLAY ID=luname commands and note the following:
   - Whether the session exists
   - The current RPL command
   - Whether the RPL is busy (for VTAM V3R1.2 and later).

   See " VM General Procedures to Diagnose Logical Unit Hang Problems" on page 94 for more information.
4. Collect more documentation:
   - Device type, model, and current state (use VSCS DISPLAY ID=luname)
   - CCS and VTAM trace output
   - Any other VSCS trace output available
   - VSCS dump.

   Note: The VSCS dump formatter will print a single logical unit on request.
5. Find the logical unit control blocks in the VSCS dump and look at the following:
   a. The current state of each logical unit. It will be PLB LBCSTATE or VLB LBCSTATE. *VTAM Data Areas* describes these states in more detail.
   b. What is the status of the SEND RPL? Note whether it is busy, and note whether it contains any errors.
   c. Note whether the logical unit has any work to process. This is indicated by the existence of DTIWEBs in the dump.
6. After collecting all documentation, go to "Reporting Procedure" on page 68.

VM Wait and Loop Problems

Complete the preliminary procedure below, and if the problem persists, choose the symptom that most closely matches yours. Follow the procedure given for your symptom.

- "VM All VSCS Logical Units Are Hung (Affects Logons and Current Sessions)" on page 119
- "VM One or More Logical Units Hung" on page 123
- "VM VSCS Operator Commands Do Not Complete" on page 134.

Note: The term "hung terminal" means the user is prevented by the system from entering data.

VM Preliminary Procedure for Problems Involving All Logical Units

This is a fairly short procedure that can help you isolate some common reasons that cause all logical units to hang.

1. Is VSCS dispatchable?

   Use the following commands to see if VSCS can process anything:

   - Does VSCS respond to the VSCS QUERY command?
   - Does 'VSM' show up when the CP QUERY NAMES command is issued?
   - Does VTAM respond to the VTAM DISPLAY commands?

   If the answer to all three questions is yes, VSCS is dispatchable and the global IUCV connection is intact. Continue with step 2.

   If one or more answers is no, the problem could be one of the following:

   - A higher-priority task is looping
   - VSCS is looping
   - VSCS (or GCS) is waiting for an IUCV instruction to complete.

   If the loop or wait occurs during initialization, go to "VM Initialization Problems" on page 101.

   If the problem occurs during termination, go to "VM Termination Problems" on page 109.

   If the problem occurs during logon, go to "VM Logon Problems" on page 115.

   If the problem occurs during logoff, go to "VM Logoff and Disconnect Problems" on page 117. Get a VSCS dump (see step 3), and go to "Reporting Procedure" on page 68.

   Note: If a VSCS dump is in progress, it will appear that all users are hung and no console commands will complete. When the dump has completed, processing will resume.

2. Look at operator console output.

   The VSCS, VTAM, and CP console listings may contain messages that explain your present condition. Check for VSCS or VTAM messages that indicate that a network failure occurred and error recovery is in progress. For example, message DTIS311 might tell you that VSCS is taking a dump.

3. Get a VSCS dump and a VSCS internal trace.

   If a dump is not in progress and the VSCS console does not respond to commands issued in "running" status, use the CP DUMP command in CP mode. For more information on the VSCS dump, see VM VSCS Dump" on page 155. For more information on CP operator commands, see the *CP Command Reference*. Get the VSCS internal trace before recovering the logical unit if possible.

4. Examine the logical unit control blocks and trace output.

   In the dump, find the logical unit control blocks TAB, PLB, and VLB. The VSCS internal trace, used along with information from the control blocks, may show the reason for the hang.

   Note: The VTAM VARY INACT, VSCS FORCE, and CP FORCE commands turn bits on and off in the PLB and VLB. Those bits are used to determine the reason for a hung logical unit. Therefore, attempt to gather documentation *before* using those commands, unless you suspect one of them is the cause of the problem.

5. Is it a network problem?

   A network problem, such as VTAM failing to terminate a session successfully, or outstanding I/O, may cause logical units to hang. To diagnose VTAM problems, see Chapter 3, "Procedures for Specific Problem Types."

   Note: Before determining that the problem is a VSCS problem, issue the VSCS DISPLAY ID=luname to determine whether I/O is active (RPL is busy; VTAM V3R1.2 and later).

VM All VSCS Logical Units Are Hung (Affects Logons and Current Sessions)

If all VSCS logical units are hung, the problem is likely to be in one of the following four areas:

Problem	Symptoms
Global IUCV path severed	• VSCS issues great quantities of DTI messages • Users cannot log on (DTIV12I) • Message DTIP14I.
VTAM Services task problem	• Wait • VTAM Services save areas show that a VTAM request is being processed.
Presentation Services in a large enabled loop	• Loop symptoms – Response time is slower than normal – 90 to 100% of CPU time is used – VSCS console does not respond • All VIT entries are the same.
No VTAM RECEIVE ANY RPLs active	• Message DTIV05I • VSCS dump shows inactive RPLs.

It can be difficult to determine whether VSCS is the cause of a loop. The loop can be caused by one of the following:

- VTAM is recovering from a network failure (if VSCS is in the same virtual machine as VTAM).
- VSCS is in SCIP recovery after a network failure.

- GCS is dumping the virtual machine.
- A VTAM temporary storage shortage has occurred, and VSCS is trying the VTAM request again.
- VSCS, VTAM, or GCS may have a control block chained to itself, resulting in an infinite loop.
- There is a loop in CCS or IUCV, so that the VSCS virtual machine appears to be looping, but is not.

Note: Check the VSCS console log to see if error recovery is in progress.

Choose the problem most like yours and follow the procedure for it.

VM Global IUCV Path is Severed

1. Determine whether the IUCV path is severed.
   a. Issue CP QUERY NAMES. If you *do not* see a "VSM" followed by names, then the global path is probably severed. Go to step 2.
   b. VSCS issues many DTIxxxx messages because all attempts to communicate with CP are rejected. Look at the IPTYPE or return codes in these messages (a list of IPTYPEs is shown on page 105). If the IPTYPE or return code indicates that the IUCV path is not active, or that the path ID is invalid, go to step 2. VSCS issues message DTIP14I with a reason code indicating why the logical unit cannot connect with CP. If the reason code indicates that the IUCV path is not active, or that the path ID is invalid, go to step 2.
2. If you have determined that the VSCS global path is severed, get data from the VSCS CCS trace and any VSCS internal error dumps taken. Get the documentation if available, and then go to "Reporting Procedure" on page 68.

   Note: VSCS dumps must be enabled at the time of the IUCV SEVER request to get the required documentation.

VM VTAM Services Task Problems

1. Is VSCS in a wait?

   Because VSCS uses both synchronous and asynchronous processing, timing problems can cause a wait. A wait condition could indicate a problem in the VSCS VTAM Services subtask.
2. Is a request presently being processed?

   Find the VTAM Services save areas and determine if a VTAM request is presently being processed. If VTAM does not complete a request, all VSCS terminals will be hung.
3. If the problem is in VTAM Services, collect the following documentation and go to "Reporting Procedure" on page 68:
   - VSCS dump
   - VSCS operator console listing
   - VSCS CCS and VTAM exit traces active externally
   - VTAM internal trace, OPT=(API,MSG,PIU), MODE=EXT
   - VTAM dump (use GDUMP command), unless VSCS is in the same virtual machine.

VM Presentation Services is in a Large Enabled Loop

1. Is VSCS showing symptoms of a loop?

   These symptoms are typical of a loop problem:

   - Extremely slow response time
   - 90 to 100% of CPU time used
   - VSCS console does not respond.

2. Do any VSCS trace entries indicate a loop?

   Look at the VSCS internal trace table on the screen or in a dump and see if all of the trace entries are the same. (See "VM VSCS Trace Facility" on page 159 for a description of the VSCS trace records). If so, the VSCS CCS trace entries could indicate if VSCS or CP is continuously sending IUCV requests or replies. The VSCS dispatcher trace entries show whether VSCS is in an internal loop processing a single work element.

   On the screen, use the VSCS QUERY command (see *VTAM Operation*) to find the starting address of the VSCS internal trace table. Then display the address of the current entry with the command

   ```
   D xxxxxx.20
   ```

   where xxxxxx is the current trace entry (the VSCS trace table address +8 is the address of the current entry). Record what you see. Wait a minute or two and repeat the procedure above. If the VSCS trace entries are the same, and only the VSCS CCS trace is active, then VSCS is in a loop, VSCS is waiting for an event to complete, or there is currently no work to process.

   To find the internal trace table in a dump, use the pointer at SABTRHDR in the DTISAB. If VSCS is in a loop, the current and preceding trace entries may look the same, depending on the trace options that are active.

3. Find the current logical unit.

   You can find the logical unit being processed by Presentation Services in the current (last used) entry in the VSCS internal trace table.

4. Use the dispatcher trace to verify the loop.

   If VSCS is looping on one work element block (DTIWEB), you will see it re-dispatched for the same logical unit over and over. For information on the dispatcher trace, see "VM VSCS Trace Facility" on page 159.

   Note: If you limit the number of users on the system at the time of the loop, the extent of the loop may be more apparent in the save areas.

5. Examine the dispatcher work element queue in a VSCS dump.

   The Presentation Services dispatcher work element queue is located in module DTIISDA1 in a dump. The address of this module in a dump is shown in the VSCS trace table header and in the link-edit listing. If the queue shows a long chain of work to do, VSCS may be in a loop or a permanent wait. If so, the problem will be associated with the current user identified by the dispatcher.

6. Get the following documentation:

7. If so, the documentation you need is:

   - VSCS trace for the failing logical unit with all options active (except FRE and GET, for VTAM V3R2)
   - VTAM internal trace, OPT=(API,MSG,PIU), MODE=EXT.

- A few seconds of virtual machine branch trace on the VSM, to trace only the VSCS address range.

  The beginning of the VSCS load module can be found by use of the VSCS QUERY command (if VSCS TRACE is active). Message DTIS74I contains the starting address of the VSCS trace table. Use CP DISPLAY TXXXXX.50 and *VTAM Data Areas* to find the addresses in the THDRSTRT and THDRSDA2 fields. Those addresses represent the extent of the VSCS load module.

  Note: Use the PER instruction to trace only address ranges of VSCS that you want to trace. (For more information on the branch trace, see the *CP Command Reference.*)

- VSCS dump
- VSCS operator console listing.

8. Go to "Reporting Procedure" on page 68.

VM No VTAM RECEIVE ANY RPLs Active

1. Did you get message DTIV05I?

   VSCS maintains a pool of RECEIVE ANY RPLS, which are used by the receive RPL exit to receive data from VTAM for all VSCS logical units. VSCS monitors the number of active RECEIVE ANY RPLS. If none of the RPLS are active, VSCS cannot receive data from VTAM. VSCS issues message DTIV05I for this error condition. If you received this message, go to step 3.

   Under certain error conditions, however, the RECEIVE ANY is not issued, and VSCS runs out of RECEIVE ANY RPLS without issuing DTIV05I. Then VSCS cannot communicate with VTAM.

2. Look for the RPL pool in a dump.

   To check for this condition, look for the RPLS in a dump using this method:

   a. Find DTIISDA1.

   b. Locate the DTIVGB contained in DTIISDA1.

   c. Locate the RECEIVE ANY RPLS.

      DTIVGB + X'18' is the address of the 1st RECEIVE ANY RPL.
      DTIVGB + X'1C' is the address of the 2nd RECEIVE ANY RPL.
      DTIVGB + X'20' is the address of the 3rd RECEIVE ANY RPL.

   Keep adding X'4' to the previous address to find the next RECEIVE ANY RPL.

   The total number of RECEIVE ANY RPLS is controlled by the DTIGEN parameter RPLNUM. The default is 8, but RPLNUM can range from 3 to 16. DTIVGB contains the address of each RECEIVE ANY RPL.

   Under normal conditions, all RPLS are active. Check the RPLACTIV field of the RPL. If RPLACTIV is X'FF', then the RPL is active.

   If RPLACTIV is *not* X'FF', then the RPL is temporarily inactive, most likely because one of the VTAM Services exits is currently working with it, or VTAM is momentarily using it. Check the VSCS VTAM Services save areas to see what is being processed. The RPL command code is usually X'23', but it can be X'22' or X'24' when used by the RPL exits. The command code is in the RPLREQ field of the RPL.

3. If RECEIVE ANY RPLS are inactive, get the following documentation:

   - VTAM internal trace, OPT=API,MSG,PIU; active externally
   - VSCS VTAM exit trace
   - VSCS dump
   - VSCS operator console listing.

4. Go to "Reporting Procedure" on page 68.

VM One or More Logical Units Hung

If a single terminal is hung, follow these steps:

1. Use the "VM General Procedures to Diagnose Logical Unit Hang Problems" on page 94.
2. Get the required documentation from "VM Determine When the Hang Occurred."
3. Try to clear the hang.

- For local non-SNA VSCS terminals, press the RESET key.
- For remote VSCS terminals, do one of the following:
  1. Turn the power off, then on.
  2. Use the TEST ON/OFF switch.
  3. For SNA logical units, press the ALT and SYS REQ keys, then enter LOGOFF to VTAM.
  4. Press the ATTN key.
  5. Have the operator issue one or more of the following commands from the VM or VTAM console:
     - VTAM VARY INACT,F,ID=luname
     - VTAM VARY INACT,I,ID=luname
     - VSCS FORCE ID=luname
     - CP FORCE userid

VM Determine When the Hang Occurred

Choose from the following list the description that most closely matches your situation, and go to that procedure.

1 Logical unit hangs while using a full-screen application program, such as SPF or XEDIT (see page 124).

2 Logical unit hangs while using console mode or CMS mode (see page 126).

3 Logical unit hangs while moving between full-screen mode and console or CMS mode (see page 128).

4 Logical unit hangs after error condition during purge, logoff, or cleanup. This may be indicated by a message (see page 129).

5 Logical unit hangs after message DTIC101 because the path is already severed (see page 131).

6 Logical unit remains hung after using VTAM VARY INACT, VSCS FORCE, or CP FORCE commands (see page 132).

7 Other (see page 133).

1 Logical Unit Hangs While You Are Using Full-Screen Mode or the Logical Unit Is Dialed

1. Is a full-screen application program (such as XEDIT or SPF) responsible for the failure?

   In full-screen mode, the application program is responsible for formatting the screen, unlocking the keyboard, and reading input data. If the application program does not handle these functions properly, the device can be hung.

   Indications of this are:

   - PROGxxx message
   - Message limit is reached
   - Message 'NOT ACCEPTED' appears when you press the enter key.
     This indicates that the application issued a WRITE with keyboard restore, but did not retry the previous data.
   - The keyboard is locked.

   If you have indications of a full-screen application program failure, get a dump of the user's virtual machine and go to "Reporting Procedure" on page 68.

2. Is the logmode defined incorrectly?

   Signs of this are:

   - Invalid characters appear on the screen.
   - The colors are wrong.
   - The screen size is wrong.
   - The logical unit is disconnected.

   Use the VSCS command DISPLAY ID=luname to see if the device is defined correctly. (See "VM Determining Device Characteristics" on page 97 for more information.) If the logmode is defined incorrectly, see *VTAM Customization* and correct the logmode definition.

   If logmode is correct, continue with this procedure.

3. Did an OPCHECK occur?

   Signs of this are:

   - The application program ends.
   - The output for a color display shows the wrong colors.
   - The logical unit is disconnected.

   If OPCHECK conditions exist, get the VSCS CCS trace and VTAM buffer contents trace on the failing logical unit for documentation.

   Go to "Reporting Procedure" on page 68.

4. Is the logical unit hung or data lost after a CP message was displayed or input was entered?

   While a logical unit is in full-screen state, VSCS should display **no** CP messages (except warnings and priority messages) until input is received (at which time the full-screen application is notified of input available).

If DTIGEN option FSREAD=N has been coded, then you can expect loss of data:

- If the screen is interrupted before input is received from the logical unit
- If modified data input is available, CP messages were displayed, and the buffer data is requested by the full screen application.

Note: The full-screen application program is responsible for restoring the keyboard after input is received. If the keyboard is not restored, the logical unit remains hung.

Get the VSCS trace with CCS, DISP, and VTAM active, and a VSCS dump. (See "VM VSCS Trace Facility" on page 159 and "VM VSCS Dump" on page 155 for more information on getting this output.)

Go to "Reporting Procedure" on page 68.

5. Is the logical unit still hung? If so, get this documentation:

   - VSCS CCS, DISP, and VTAM exit traces for the failing logical unit
   - VSCS dump
   - VTAM SMS (buffer use) trace for the failing logical unit.

   Go to "Reporting Procedure" on page 68.

2 Logical Unit Hangs While You Are Using Console Mode or CMS Mode

Note: The CMS full-screen command puts the logical unit into full-screen state.

1. Is the logical unit waiting for the previous command to be displayed again?

   If so, then one of the following occurred:

   - The command has not completed and CP did not notify VSCS (via command completion or timer expiration) to display the input again.
   - The command was executed by the full-screen application program and the application did not display the information again.
   - The input has not been accepted.

   Get a VSCS CCS trace for the failing logical unit and a VSCS dump. Then go to "Reporting Procedure" on page 68.

2. Is the logical unit hung after a 'MORE', 'HOLDING', or 'NOT ACCEPTED' condition?

   If so, get a VSCS dump and the VSCS CCS and dispatcher traces. Then go to "Reporting Procedure" on page 68.

3. Are the VSCS data translation exits coded?

   The following VSCS user exits affect the logical units' reaction to any outbound data streams and CP's reaction to inbound data streams:

DTIPDNTI	LU1 inbound data
DTIPDNTO	LU1 outbound data
DTIPDDSI	display input data
DTIPDDSO	display outbound data.

   If the problem does not appear to be a problem in the user-coded exit, get the following documentation:

   - VSCS CCS trace
   - VSCS DATA trace (V3R2)
   - VTAM buffer contents trace for the failing LU.

   Go to "Reporting Procedure" on page 68.

4. Is the problem with a 3279 color terminal (type 0 logical unit)?

   Make sure FEATUR2=EDATS is specified in one or more of these NCP definitions:

   LINE
   GROUP
   CLUSTER
   TERMINAL.

   Also, check the coding of the user exits (DTIPDDSI and DTIPDDSO).

   If color or highlighting still does not work correctly, get the following documentation:

   - VSCS CCS and VTAM exit traces, and data for the failing logical unit
   - VTAM buffer contents trace for the failing logical unit.

   Go to "Reporting Procedure" on page 68.

5. Is the device dialed to another system?

   If so, go to the full-screen procedure on page 124.

   Note: Dialing to PVM and passing through to another CMS system gives the appearance of normal console and CMS screens. However, VSCS receives the entire screen already formatted, including the status area, and processes the data as full-screen data.

6. Documentation for problems not described above:

   If you can re-create the problem, get:

   - VSCS trace for the failing logical unit with all options active (except FRE and GET, for VTAM V3R2)
   - VSCS dump
   - VTAM internal trace with OPT=(API,MSG,PIU,PSS,SSCP), MODE=EXT
   - VTAM buffer contents trace for failing logical unit.

   If you cannot re-create the problem, get:

   - VSCS trace with CCS active, for just the failing logical unit, if possible
   - VSCS dump
   - VTAM logmode being used.

   Go to "Reporting Procedure" on page 68.

3 Logical Unit Hangs When You Switch Modes

1. Does your terminal hang when you switch between modes? For example:

   - Console mode to CMS mode
   - CMS mode to console mode
   - CMS or console mode to full-screen mode
   - Full-screen mode to CMS or console mode.

2. If so, get the following documentation:

   If you can re-create the problem, get:

   - VSCS dump
   - VTAM buffer contents trace
   - VSCS CCS and dispatcher traces with just the failing logical unit traced
   - VTAM internal trace with OPT=(API,MSG,PIU,PSS,SSCP), MODE=EXT.

   If you cannot re-create the problem, get at least:

   - VSCS dump
   - VSCS trace with CCS and DISP active.

3. Go to "Reporting Procedure" on page 68.

4 Logical Unit Hangs During Logoff or Disconnect Processing

1. Have any messages been issued?

   Get the operator console log and look for messages that pertain to the hang.

2. Determine the component that sent the message.

   If the message has a return code, use Figure 24 to determine the component that may have sent the message. Then look in *VTAM Messages and Codes* to see what the return code means.

   If the message has no return code, use the message prefix to determine where the message is coming from. Page 104 has a list of the message prefixes for each component.

Message	Source of Return Codes
DTIPxxx	GCS VM VSCS
DTISxxx	VM VSCS
DTIVxxx	VTAM

Figure 24. VM Source of Return Codes

3. Are other components connected to the logical unit?

   Answering these questions may indicate which components are still communicating with the logical unit.

   - Does the VSCS command DISPLAY ID=luname show the status of the device? (See "VM Determining Device Characteristics" on page 97 for more information.)

   - Is the user still logged on to VM? Try the CP QUERY NAMES command. If the logical unit name and user ID appear, the logical unit is connected to VM and the CP IUCV connection is intact. If the response is gibberish, the logical unit is not connected to VM and the CP IUCV connection has been severed.

   - Does VTAM show the logical unit still in session with VSCS? (Use the VTAM DISPLAY command.)

4. Note the status of the logical unit at this time. For example, on a display screen note what is in the status area, and what was the last input or the last output to the logical unit.

5. Collect documentation before recovery.

   If you can re-create the problem, get:

   - VSCS dump
   - VTAM buffer contents trace
   - VSCS trace for the failing logical unit with all options active (except FRE and GET, for VTAM V3R2).

   If you cannot re-create the problem, get:

   - VSCS dump
   - VSCS trace with CCS active.

6. Recover the logical unit with one or more of these commands:

   - CP FORCE
   - VTAM VARY INACT
   - VSCS FORCE.

7. Go to "Reporting Procedure" on page 68.

5 Logical Unit Hangs After Receiving Message DTIC10I

1. Did these symptoms occur?

   Message DTIC10I indicates that an IUCV request or response was received on the global path after an IUCV SEVER was issued by VSCS. Most of the time this does not cause a problem, because CP logs off or disconnects the logical unit when the IUCV SEVER is processed. If it is a problem, it could produce these symptoms:

   - Abend or program check
   - VSCS message DTII14I
   - Loss of operator communications.
   - Message CSIIXT413T indicates that the signal services path was severed.

2. Collect documentation.

   - VSCS dump

   - VSCS trace with CCS active, running for all logical units.

3. Go to "Reporting Procedure" on page 68.

6 Logical Unit Stays Hung After You Use VARY INACT or FORCE Command

1. Try both the VTAM VARY INACT and CP FORCE commands.

   If the logical unit is being terminated by VM, the VMBLOK status prevents the CP FORCE command from being effective. The VMKILL bit in the VMBLOK indicates whether CP is in the process of terminating the logical unit. If you issue the CP FORCE command and the resource is unknown, CP may be waiting for an IUCV reply before completing the termination process for the logical unit. Try using the VTAM VARY INACT command.

   If the logical unit is still hung, continue with step 2.

2. Determine which components are still connected to the logical unit.

   - Use the VSCS command DISPLAY ID=luname to determine the status of the logical unit and of the SEND RPL.
   - Is the user still logged on to VM?
   - Does the logical unit name appear when you use the CP QUERY NAMES command? If so, the VM connection has not been severed.
   - Does VTAM show the logical unit still in session with VSCS? Use the VTAM DISPLAY command to determine this.

   From this information, try to determine which components are connected to the logical unit.

3. Note the status of the logical unit at this time. For example, on a display screen, what is in the status area, and what was the last input or the last output to the logical unit?

   Note: If the user logged off and the screen says "HOLDING," you must clear the screen before the logical unit can be disconnected from VM. (This is normal for VSCS.)

4. Collect documentation.

   If you can re-create the problem, get:

   - VSCS dump
   - VSCS operator console listing
   - VTAM buffer contents trace
   - VSCS trace for the failing logical unit, running externally, with all options active (except FRE and GET, for VTAM V3R2)
   - VTAM internal trace with OPT=(API,MSG,PIU,PSS,SSCP), MODE=EXT.

   If you cannot re-create the problem, get:

   - VSCS dump
   - VSCS operator console listing
   - VSCS CCS and dispatcher trace.

5. Go to "Reporting Procedure" on page 68.

7 Logical Unit is Hung for Other Reasons

1. Answer these questions.
   a. Issue the CP QUERY NAMES command on the CP console. Does the logical unit name appear?
   b. Is VSCS still controlling the logical unit? (Use the VTAM DISPLAY command to discover this.)
   c. Have any messages appeared on the VTAM or VSCS operator console for this logical unit?
   d. What is the last sequence of messages issued? (See the VSCS operator console listing.)
   e. Issue VSCS DISPLAY ID=luname. Is the RPL busy?
2. Collect documentation.

   If you can re-create the problem, get:

   - VSCS dump
   - VSCS operator console listing
   - VTAM buffer contents trace
   - VSCS traces for the failing logical unit, running externally, with all options active (except FRE and GET, for VTAM V3R2).
   - VTAM internal trace with OPT=(API,MSG,PIU,PSS,SSCP), MODE=EXT.

   If you cannot re-create the problem, get:

   - VSCS dump
   - VSCS operator console listing
   - VSCS CCS and dispatcher trace.
3. Go to "Reporting Procedure" on page 68.

VM VSCS Operator Commands Do Not Complete

1. Has the VSCS Utility Services subtask abended?

   Check the operator console listing to see if the Utility Services subtask abnormally terminated. If this occurs VSCS can do no further processing of operator commands. Use the VSCS CANCEL command to terminate VSCS. Then get the dump generated by the abend and go to "VM Termination Problems" on page 109.

2. Does a GCS message indicate a problem with the GCS console task? If that task abended or could not get storage, operator commands cannot reach VSCS.

3. If the problem is not an abend or GCS console task, try these commands:

   - If VSCS is running in the same virtual machine as VTAM, enter a VTAM command such as DISPLAY LU. Does it complete?
     - If yes, VSCS may be in a loop or a permanent wait.
     - If no, the problem is in the VTAM virtual machine. Continue down this list to determine where the problem is.
   - Enter the CP QUERY TIME command. This shows whether GCS is processing and passing CP commands to CP.
   - Enter #CP QUERY NAMES. This shows whether CP commands are being processed.

4. Does GCS process some commands, but not others? Does the PA1 key put the terminal in CP READ mode?

   If the answer to both questions is yes, GCS is looping or waiting for an event to complete. In CP READ mode, issue SET RUN OFF. Then enter the CP command D PSW to display the current PSW. Display the storage around the address shown in the second half of the PSW. If the last instruction issued was X'B2F0xxxx', CP has not completed an IUCV request, and VSCS and the virtual machine are permanently waiting for a response. If the problem does not involve an IUCV instruction, check to see if the WAIT bit (bit 14) is on in the PSW. If so, the problem is in the VSCS virtual machine. Get a VSCS dump and go to "Reporting Procedure" on page 68. The VSCS dump is described in "VM VSCS Dump" on page 155.

VM Message Problems

For messages other than DTIS99I, go to "Message Procedure" on page 53.

VM Message DTIS99I Indicates a VSCS Internal Error

Message DTIS99I tells you that a VSCS internal error occurred for module DTIXXXXX and a number (n). The number identifies the reason the message was issued. Use the lists below to determine which component of VSCS issued the message. See the following example:

```
DTIS99I INTERNAL VSCS ERROR - DTIPRSVR 1
```

This indicates that VSCS attempted to issue an IUCV SEVER for the VSCS global path. For this error, a dump is taken (if dumping is enabled) and the SEVER request is ignored.

1. Presentation Services issues messages with this information:

Module	n	Error Description
DTIPDADV	1	SEND-in-progress bit error (VTAM V3R1.2 and later)
DTIPICPY	1	SEND-in-progress bit error (VTAM V3R1.2 and later)
DTIPIPUT	1	Error in logical aid
	2	Invalid WEBLAID (logical aid)
DTIPISYN	1	Invalid SEND/REPLY WEB type
DTIPOCWR	1	SEND-in-progress bit error (VTAM V3R1.2 and later)
DTIPOFWR	1	SEND-in-progress bit error (VTAM V3R1.2 and later)
DTIPOVSR	1	SEND-in-progress bit error (VTAM V3R1.2 and later)
DTIPOWIC	1	SEND-in-progress bit error (VTAM V3R1.2 and later)
DTIPOWRT	1	SEND-in-progress bit error (VTAM V3R1.2 and later)
DTIPRACT	1	Received an invalid WEB
DTIPRSVR	1	Attempting to issue a SEVER for global path ID
DTIPRWEB	1	Invalid WEB
DTIPSCFS	1	Invalid FS response WEB
DTIPSCPY	1	SEND-in-progress bit error (VTAM V3R1.2 and later)
DTIPSFCR	1	SEND-in-progress bit error (VTAM V3R1.2 and later)
DTIPSPIN	1	SEND-in-progress bit error (VTAM V3R1.2 and later)
DTIPSTAM	1	Queue tag error for work element queue
	2	Invalid WEBFUN or WEBMODE
	3	Invalid WEBFUN or LBCSTATE

Note: DTIPIPUT will request a dump if you use the wrong logmode entry when logging on to VSCS with a type 1 logical unit (TWX, 2741, or 3767). The PSERVIC field in the logmode entry you use must contain X'01' in byte one, or VSCS assumes the terminal is a type 0 logical unit (display terminal). VSCS requires a valid aid character to process input from a type 0 logical unit, and the type 1 logical unit does not provide this.

2. VTAM Services issues messages with this information:

Module	n	Error Description
DTIVCLSX	1	Schedule request to DTIVTRME failed
	2	No storage available
DTIVLOFF	1	IUCV connection exists
DTIVPURG	1	Schedule failed
DTIVRECA	1	Error in building WEB
DTIVRECX	1	Invalid input chain
DTIVREJR	1	Logical unit not in termination
DTIVSEND	1	Invalid request (VTAM V3R1.2 and later)
	2	Invalid request (VTAM V3R1.2 and later)
	3	Recovery queue empty during recovery (VTAM V3R1.2 and later)
	4	Schedule request failed (VTAM V3R1.2 and later)
DTIVSNDX	1	Invalid RPL response
	2	Dequeue error in recovery (VTAM V3R1.2 and later)
	3	Invalid data response in recovery (VTAM V3R1.2 and later)
	4	Invalid RPL response (VTAM V3R1.2 and later)
	5	Recovery queue empty during recovery (VTAM V3R1.2 and later)
	6	Recovery queue empty during recovery (VTAM V3R1.2 and later)
	7	Attempting to schedule DTICMT (VTAM V3R1.2 and later)
DTIVSTAM	1	Queue tag error for work element queue
	2	Invalid WEBFUN or WEBMODE
	3	Invalid WEBFUN or LBCSTATE
DTIVTPNX	1	Cannot schedule request

3. System Services issues messages with this information:

Module	n	Error Description
DTISCHED	1	Invalid process identifier
DTISDEQU	1	Queue tag does not match tag in queue header
DTISENQU	1	Invalid chain pointer
	2	Queue tag does not match tag in queue header
DTISFREB	1	Invalid SMP tag
	2	Invalid request (releasing storage that is already free)
DTISFREM	1	Invalid SMP tag
DTISGETB	1	Control block error or request is too big to obtain
	2	Invalid free block count
DTISGMGR	1	PDBUSCNT (chain pointer) is invalid
	2	Empty PDBSDB queue found

4. Documentation to collect for VSCS internal error problems:

   - Issuer of the message: Presentation Services, VTAM Services, or System Services
   - Reason the message was issued
   - VSCS dump (produced automatically as a result of message DTIS991 if dumps were enabled prior to the problem occurring)
   - VSCS operator console listing
   - Any VSCS traces that were active at the time of the error.

VM Incorrect Output Problems

1. Describe the incorrect output. The following list shows some of the possible symptoms:

   - The size of display screen is incorrect.
   - The status area of the display screen does not work as expected.
   - Highlighting or color do not work as expected.
   - Invalid characters appear in the output, such as EBCDIC, APL, TEXT, or double-byte Kanji characters. (Invalid characters are any characters that are obviously wrong or that you don't expect.)
   - Input data is lost.
   - PROGxxxx occurs on SNA 3270s. If this happens during full-screen mode, the contents of the data stream may be invalid. If the terminal is not in full-screen mode, this message is more likely to indicate a VSCS problem or a logmode problem.
   - The device does not function like a VM-supported terminal and the problem is not due to a known restriction.
   - The incorrect output does not involve full-screen data or the VM/SP logo, and one or more of the following symptoms occur:
     - Data is misplaced on the screen.
     - Data is missing.
     - Content of data is incorrect.
     - Device errors occur or the logical unit is disconnected when you execute a command or a program.

2. Check to see that product restrictions are not causing the incorrect output.

3. Does the incorrect output invoive the logo? See "VM VM/SP Logo Problems on an SNA Device" on page 113 for more information.

   Note: If you enter both the user ID and the password on the user ID line, you may receive a different DMK message than you would if you entered only the user ID.

4. Is the device defined correctly to VSCS?

   Use the VSCS command DISPLAY ID=luname to determine if the device is defined correctly to VSCS. If the DISPLAY command is not successful, get a dump of VSCS. The logical unit control blocks in a dump also show the LU definition.

   If the LU definitions are not correct for the device, correct them and retest the logical unit. See *VTAM Installation and Resource Definition* for more information on defining logical units.

5. Check to see if the appropriate VTAM logmode for the device is being used. See *VTAM Customization* for more information on logmodes.

6. Are user translation exits involved?

   If your system has user-written translation exits, try to re-create the problem without them. Use the DTIGEN parameters and see if the problem still occurs.

7. Is Kanji support involved?

Double-byte character support for Kanji does limited editing. Check the data stream sent to see if it is within the limitations of double-byte support.

Note: VSCS does not check whether the device supports Kanji. The application has this responsibility.

8. Are TWX 3101 configuration switches set up as documented by VM? See the *VM/SP Terminal Reference* to find out how these switches should be set up.
9. Other possible causes of incorrect output are listed here. You may choose to investigate these further.
   - Functional limitation due to the release of VM/SP used
   - Hardware problem
   - VSCS problem
   - VM problem
   - VM application problem.
10. If the problem has not been solved, collect the following documentation:
    - For status area problems, lost input, device errors, and disconnections:
      - VSCS dispatcher trace of the failing logical unit.
      - VSCS trace with CCS active, of the failing logical unit.
      - VSCS VTAM exit trace of the failing logical unit.
      - VTAM buffer contents trace.
    - For all other incorrect output problems:
      - VTAM buffer contents trace of the failing logical unit.
      - VSCS trace with CCS active, of the failing logical unit.
      - If a VSCS dump is necessary, limit the number of users on the system and use the VSCS BFRFIFO command before re-creating the problem.

Storage-Related Problems

Storage problems in VSCS can occur for several reasons; they may indicate a problem in VSCS. For VTAM V3R2, issue the VSCS STORAGE command whenever you suspect that storage is causing problems.

1. Determine whether the virtual machine in which VSCS resides in is large enough. Estimate this using *NPP Storage Estimates*.

   Note: If VSCS resides in the same virtual machine as VTAM, add the storage requirements for VTAM.

   If the virtual machine is large enough for your environment, continue. Otherwise, correct the VSCS virtual machine size.

2. If you are using new VSCS start parameters, evaluate changes made with DTIGEN parameters:

   - Trace table size
   - Increase in the number of printers
   - Receive buffer size
   - BLKMULT greater than one
   - VSAMLM number increased
   - Transmit buffer sizes for both logical units and displays
   - RECEIVE SPECIFIC option (VTAM V3R1.2 and later)
   - Exception response option (VTAM V3R1.2 and later).

   If VSCS ran properly before these changes, review the size of the VSCS virtual machine before reporting a problem.

   You can use the VSCS STORAGE command (VTAM V3R2) to see:

   - Current number of VTAM sessions and CP connections
   - Indicator of storage shortage in message DTIS15I
   - Some of the DTIGEN options that are currently active
   - Storage management parameters for VTAM V3R2 that have been specified
   - Number of VEIBs allocated (controlled by VSAMLM).

   If VSCS is running in the same virtual machine as VTAM, use the VTAM DISPLAY BFRUSE command to determine VTAM's storage usage.

3. Is the network in slow-down or recovery mode?

   If the network is in slow-down mode, it will take additional storage to maintain data destined for the logical unit. This symptom could also be caused by one of the following:

   - VTAM is recovering from a network failure.
   - VSCS is in SCIP recovery after a network failure.
   - VSCS is in a different virtual machine than VTAM, and a dump is being taken for VSCS or VTAM
   - A VTAM temporary storage shortage has occurred.
   - The I/O buffer has expanded for a VTAM-controlled device.

4. Are several logical units hung?

   The storage being used by hung logical units remains allocated until the logical unit is released or is terminated. If you are using the exception response mode and RECEIVE SPECIFIC options, the amount of storage used could be enough to cause a storage shortage. See "VM General Procedures to Diagnose Logical Unit Hang Problems" on page 94 before continuing.

   If you can re-create the problem, or if it is occurring frequently, gather the following documentation:

   - VSCS internal trace with options CCS, DISP, FRE, and GET active
   - Dump of the VSCS virtual machine.

5. Did you receive message DTIS991 for DTISFREB or DTISGETB?

   This indicates that VSCS is having a problem managing storage blocks.

   Note: As with other DTIS991 problems, dumps must be enabled prior to the error or you cannot get proper documentation.

   If you can re-create the problem, or if it is occurring frequently, gather the following documentation:

   - VSCS internal trace with the options CCS, DISP, FRE, and GET active
   - Dump created when message DTIS991 is issued.

   Get a VSCS dump and go to "Reporting Procedure" on page 68.

6. Did you receive message DTIS101, followed by DTIV121?

   This sequence of messages indicates that either the virtual machine is too small, or all available storage has been depleted. If the VSCS virtual machine is large enough, the available storage may be allocated to a different VSCS storage pool than the current storage requester.

   For example, if storage is available in the Presentation Services pool, but VTAM Services needs it, then GCS cannot provide additional storage. Logons are rejected until Presentation Services releases the storage.

   In VTAM V3R2, you can control the distribution of dynamic storage by coding the DTIGEN parameters that allow swapping of storage segments in one pool to a different pool, or returning them to GCS. The default for the DTIGEN storage options is to work as previous release of VSCS.

   Get the following documentation:

   - VSCS trace with CCS and DISP active (and FRE and GET, for VTAM V3R2), for all logical units, before the problem occurred (externally if possible).
   - VSCS dump.

   Go to "Reporting Procedure" on page 68.

7. Is it an IUCV pacing problem?

   IUCV pacing is controlled by the value specified in the DTIGEN parameters (DPACE for displays and KPACE for keyboard/printers).

   For displays, the default value is the screen depth, which is also the default if the pace is set to 0. The screen depth varies by display type.

   For keyboard/printers, the default is 10. If the pace is set to 0 for keyboard/printers, there is no IUCV pacing.

   For TWX logical units, the default is 1, and cannot be changed in VSCS. That can lead to increased IUCV traffic, which can lead to a VSCS storage shortage in a large network, or if VSCS is running in a small virtual machine.

   If IUCV pacing is not set properly, change the DTIGEN parameter and reinitialize VSCS with the new DTIGEN. If the problem persists, get the following documentation:

   - VSCS trace with CCS active, started before the problem occurs
   - VSCS dump.

   Go to "Reporting Procedure" on page 68.

8. Other

   If the problem cannot be isolated from previous steps, then a dump of the VSCS virtual machine is required. After obtaining the dump, use the VSCS dump formatter to print the VSCS storage control blocks. If VSCS is in the same virtual machine as VTAM, then also evaluate the VTAM storage.

   Get the output from the VSCS formatter and the console log, and go to "Reporting Procedure" on page 68.

VM Performance Problems

1. Do not use a large value for the DTIGEN parameter VSAMLM. This can lead to excessive paging in a medium-to-large environment. See *VTAM Customization* for more information on VSAMLM, and *NPP Storage Estimates* to see if the virtual machine size is correct.

   Note: If VSCS is running in the same virtual machine as VTAM, include the VTAM storage size required when you determine the virtual machine size.

   If you think your performance problem is related to paging and storage, (for example, you are getting slow response time for operator commands or logical units,) collect this documentation:

   - VSCS dump
   - DTIGEN parameters
   - GCS internal trace for GETMAIN and FREEMAIN.

2. If you think your performance problem is related to logical units, (for example, you are getting slow response time for I/O operations,) collect this documentation:

   - VTAM I/O trace
   - VTAM buffer contents trace
   - VSCS trace for the failing logical unit, running externally, with all options active (except FRE and GET, for VTAM V3R2)
   - GCS internal trace for SIO and I/O.

3. If you do not know the cause of the performance problem, collect all of the documentation in the two lists above, plus supply the values for MAXBFRU, UNITSZ, and IOBUF.

4. When you have collected the necessary documentation, go to "Reporting Procedure" on page 68.

VM Other VSCS Problems

Miscellaneous topics are covered in this section:

- "VM SNA Dial Problems" on this page
- "VM Printer Sharing Problems" on page 146
- "VM Operator Commands Are Rejected" on page 147.

VM SNA Dial Problems

SNA dial keeps the logical unit in full-screen state from the time the dial is complete until the dial is dropped or reset. All of the information on the screen is supplied by the resource that was dialed.

1. If the screen size changed after one of the following occurred, take the indicated action:

   a. A VTAM logical unit dialed another virtual machine.

      Make sure the virtual machine you are dialing supports the features provided by your logical unit.

   b. A VTAM logical unit, dialing PVM, dialed another GCS group or host containing VTAM.

      When dialing PVM, the logical unit is defined as a logical address, which may prevent the real device characteristics from being known. If you are dialing another VTAM machine, supply a logmode name that describes the type of terminal you are using.

   If the screen size is still incorrect, collect the following documentation:

   - VSCS trace with CCS active
   - VSCS dump
   - VTAM buffer contents trace (in both hosts if connecting to another VTAM machine).

2. Is the logical unit hung?

   If you are using PVM, find out whether the resource you are attempting to dial is active. An inactive resource could result in a hung logical unit. If the resource is active and the logical unit is hung, the resource, PVM, VSCS, CP, or GCS may be experiencing problems. Note whether other PVM users' terminals are also hung.

   If you cannot determine the reason, get the following documentation:

   - VSCS dump
   - VSCS trace with CCS active (must be active before the hang occurs).

   In the VSCS dump, find the logical unit control blocks and the last few trace entries for that specific logical unit. From this, answer the following questions:

   a. Is the PLB LBCSTATE of the logical unit 20, 40 or 60?
   b. Is the SEND RPL busy, indicating VSCS is waiting for a response?
   c. Did VSCS respond to all outstanding requests?

If everything is specified correctly, then get the following documentation:

- VSCS dispatcher and VTAM exit traces
- VTAM internal trace, OPT=API
- VSCS dump.

Go to "Reporting Procedure" on page 68.

3. Is the I/O interrupt missing?

   If the situation is caused by missing I/O, run these traces externally on just the failing logical unit:

   - VSCS CCS and VTAM exit traces
   - VTAM internal trace, OPT=CIO, MODE=EXT
   - CP SIO SIOF, and I/O traces (See the *CP Command Reference*)
   - GCS SIO and I/O traces (See the *VM/SP Planning Guide and Reference*).

   Go to "Reporting Procedure" on page 68.

VM Printer Sharing Problems

1. Is VSCS printer sharing failing?

   If you coded a DTIGEN macro with PRTSHR=N, then change to PRTSHR=Y, reassemble DTIUSER, link DTIUSER into the VSCS LOADLIB, and reinitialize VSCS with the correct DTIUSER number.

   If you are using default parameters or you are using a DTIUSER with PRTSHR=Y coded, then get the following documentation;

   - VSCS dispatcher and VTAM exit traces
   - VSCS dump.

   Go to "Reporting Procedure" on page 68.

2. Does VTAM SIMLOGON fail after you use the VSCS PRINTER command?

   If the VTAM SIMLOGON fails, then make sure:

   - The printer name was spelled correctly
   - The resource is known and available to VTAM (use VTAM DISPLAY command)
   - The VTAM logmode for the printer logical unit is correct.

3. Does 'NOT AVAILABLE' appear in the status area of the display?

   In order for VSCS to access a VTAM printer, either the VSCS PRINTER command or a VTAM VARY ACT command must be issued first. See *VTAM Operation* for more information on commands.

   If the proper command was not issued or the printer name was misspelled, issue the command again with the correct name.

   If the proper command was issued to identify the printer to VSCS, get the following documentation:

   - VSCS dispatcher and VTAM exit traces
   - VTAM internal trace, OPT=API
   - VTAM buffer contents trace for the printer logical unit
   - VSCS dump.

   Go to "Reporting Procedure" on page 68.

4. Is VSCS not releasing a printer when requested to?

   If the printer was allocated to VSCS by using the VTAM VARY ACT,LOGON=VSCS command, then a VTAM VARY INACT command is the only way to release it.

   If the VTAM SIMLOGON was used (check the VSCS operator console listing), then VSCS will use the time value specified in the DTIGEN parameters. TIMEREL is used to determine when the printer will be released. (See *VTAM Installation and Resource Definition* for more information on TIMEREL.)

   If the correct value for TIMEREL is specified and VSCS was initialized with the proper DTIUSER module, then get the following documentation:

   - VSCS dispatcher and VTAM exit traces
   - VSCS dump
   - VTAM internal trace, OPT=API

- VTAM buffer contents trace for the printer logical unit before the Release was requested.

Go to "Reporting Procedure" on page 68.

5. Other printer problems:

For problems not described above, get the following documentation:

- VSCS dispatcher and VTAM exit traces
- VTAM internal trace, OPT=API
- VSCS dump.

Go to "Reporting Procedure" on page 68.

VM Operator Commands Are Rejected

Does VSCS reject a command that was entered correctly and issue an error message? You may have entered the command at an inappropriate time. For example, a VSCS START command is rejected when VSCS is already active. Try using some other commands to see if they work.

Note: Avoid using VSCS HALT, CANCEL, or QUIT; they could cause further problems.

If the problem is unique to one command, or to a certain sequence of commands, get the operator console listing and go to "Reporting Procedure" on page 68.

Chapter 6. VM Using VSCS Service Aids

Chapter 6. VM Using VSCS Service Aids

The following tools can help you diagnose VSCS problems:

- Messages
- Operator console listing
- Operator commands: BFRFIFO, QUERY, DISPLAY, FORCE, and STORAGE
- VSCS dump
- VSCS trace facility.

These service aids, along with service aids for VTAM, can help you isolate problems to a single component and identify problems within VSCS.

VM Message and Module Conventions

For most messages, the module name appears as the first word in the message text. For example, in message DTIC01I below:

```
DTICINIT VSCS IUCV INITIALIZATION IS IN PROGRESS
```

DTICINIT is the module name. If the module name is not given in the message text, see the message-to-module cross-reference in *VTAM Messages and Codes* for the module name.

All VSCS message and module identifiers begin with DTI. Module names are in the form DTI*caaaa*, where *aaaa* is the unique module identifier and *c* represents the issuing subcomponent, as follows:

C Communication Services for Inter-User Communication Vehicle (IUCV)
F System Services
I Initialization and Termination Services
P Presentation Services
  In this subcomponent (DTIPaaaa), the fifth letter is one of the following:
  D Device manager
  I Input manager
  O Output manager
  R Request services
  S State manager
S System Services
V VTAM Services
  In this subcomponent, a module with the last character of X (DTIVaaaX) indicates a VTAM exit routine.

Message identifiers are in the form DTI*cnni*, where *c* indicates the issuing component (as above), *nn* is the two-digit serial number, and *i* is the action indicator:

A Action
E Eventual action
I Information.

For internal VSCS errors, module DTIMSGE issues the following message:

```
DTIS99I INTERNAL VSCS ERROR modname n
```

where *modname* indicates the name of the module that issued the message, and *n* identifies the *n*th occurrence of DTIMSGE within the module. This number *n* can be used to identify the reason the message was issued. See "**VM** Message DTIS99I Indicates a VSCS Internal Error" on page 135 and *VTAM Messages and Codes* for more information.

VM Operator Console Listing

The operator console listing is a printout of the messages that appear at the VSCS operator console. The listing contains everything that appears on the operator's console during the entire session. It can help you find out the sequence of events when a problem occurs. If VSCS is running in the same virtual machine as VTAM, the console listing includes VTAM messages as well as VSCS messages.

To obtain a console listing, issue the CP command SPOOL CONSOLE START. If you are using a hardcopy terminal, such as a 3767, this is not necessary because all input and output already comes out on paper. For more information on this command, see the *VM/SP CP Command Reference*.

Note: If the operator console is in conmode 3270, the CP SPOOL CONSOLE command produces no output.

VM Operator Commands

VM BFRFIFO Command

BFRFIFO affects the dequeueing and dispatching structure for storage. Generally, storage is used from the top in a last-in-first-out fashion (BFRLIFO). This is the default. For diagnostic purposes, however, you can turn on BFRFIFO. This first-in-first-out method of storage preserves the queue long enough for you to see the elements that have just been processed.

To start BFRFIFO, use the VSCS BFRFIFO command or the DTIGEN macro. For more information, see *VTAM Operation*.

Note: Using BFRFIFO can seriously degrade system performance, especially in a large network.

VM DISPLAY Command

The VSCS command DISPLAY ID=luname shows the status of a logical unit. Figure 25 shows an example of the VSCS DISPLAY command.

```
vscs display id=13e0
R;
DTIS60I DISPLAY COMMAND ACCEPTED
DTIS61I LU NAME = L3E0 DEVICE TYPE = 3277 MODEL = 2
DTIS62I LU TYPE = 0 DEVICE FEATURES = NEDS APL( OLD )
DTIS63I SCREEN SIZE = 24 BY 80 ALTERNATE SIZE =    BY      PACE = 24
DTIS65I IUCV PATHIDS = 2 3 CURRENT RPL COMMAND = SEND
DTIS59I RECEIVE SPECIFIC RPL BUSY
DTIS66I CURRENT STATES: PS = NMH VS = NORM ER
DTIS67I LOGON TIME = 16 : 05 : 17
DTIS68I SEND COUNT = 00000001 RECEIVE COUNT = 00000000
DTIS77I VSCS COMMAND PROCESSING COMPLETE
```

Figure 25. VM Example of the VSCS DISPLAY Command

VM FORCE Command

The VSCS command FORCE ID=luname terminates the CP session and VTAM session for a logical unit. This command may be used when the CP FORCE and VTAM VARY INACT commands do not terminate a logical unit.

VM QUERY Command

The VSCS QUERY command displays the following information:

- The location of the VSCS trace table (if it has been allocated)
- The status of VSCS traces
- The status of a VSCS dump
- The status of buffer queues.

Figure 26 shows an example of the VSCS QUERY command.

```
vscs query
R;
DTIS74I TRACE TABLE STARTS AT 001C5020 , ENDS AT 002BF05F
DTIS84I SELECTIVE TRACE ACTIVE FOR LOGICAL UNIT ELU3101
DTIS84I SELECTIVE TRACE ACTIVE FOR LOGICAL UNIT LU1920
DTIS87I SELECTIVE VTAM, DISPATCHER AND CCS TRACE ACTIVE FOR 2 LU(S), 30
DTIS93I VSCS EXTERNAL TRACE IS ENABLED
DTIS76I VSCS DUMP PROCESSING IS ENABLED
DTIS78I BUFFER DEQUEUE IS IN NORMAL MODE(BFRLIFO)
DTIS77I VSCS COMMAND PROCESSING COMPLETE
```

Figure 26. VM Example of the VSCS QUERY Command

VM STORAGE Command (V3R2)

The VSCS STORAGE command shows the status of VSCS dynamic storage, and some DTIGEN options that are in effect.

Figure 27 shows an example of the VSCS STORAGE command when both STCHKTM and STRELTM are zero.

```
vscs storage
R;
DTIS15I DYNAMIC STORAGE UTILIZATION
DTIS16I RCVBFRL - 00010C DPXMTL   - 0006BC  KPXMTL  - 00010C
DTIS17I RPLNUM  - 08     VSAMLM   - 000A    VEIBS   - 000003
DTIS18I BLKMULT - 01     STCHKTM  - 000000  STRELTM - 000000
DTIS19I GETMAIN - 00000B FREEMAIN - 000000  SWAP    - 000000
DTIS20I VTAMSES - 000000 CPCONNS  - 000000  CPMAX   - 000000
DTIS77I VSCS COMMAND PROCESSING COMPLETE
```

Figure 27. VM Example of the VSCS STORAGE Command, STCHKTM = 0 and STRELTM = 0

Figure 28 shows an example of the VSCS STORAGE command when either STCHKTM or STRELTM is greater than zero.

```
vscs storage
Ready;
DTIS15I DYNAMIC STORAGE UTILIZATION
DTIS16I RCVBFRL - 00011C DPXMTL   - 00079C  KPXMTL  - 00011C
DTIS17I RPLNUM  - 08     VSAMLM   - 000A    VEIBS   - 000003
DTIS18I BLKMULT - 08     STCHKTM  - 000001  STRELTM - 00000A
DTIS19I GETMAIN - 00000A FREEMAIN - 000000  SWAP    - 000000
DTIS20I VTAMSES - 000000 CPCONNS  - 000000  CPMAX   - 000000
DTIS21I POOL   HIGHWATER/VTAMSES    CURRENT    AVAILABLE     LIFO
DTIS22I  PS      000004 / 000000     000004     000002
DTIS22I  VS      000001 / 000000     000001     000001
DTIS22I  VX      000001 / 000000     000001     000000
DTIS77I VSCS COMMAND PROCESSING COMPLETE
```

Figure 28. VM Example of the VSCS STORAGE Command, STCHKTM > 0 or STRELTM > 0

VM VSCS Dump

The VSCS dump contains all of the storage for VSCS. It also contains common, group control system (GCS), and VTAM storage if they are in the same virtual machine as VSCS.

To enable the dump, use the VSCS DMPENABL command or the DTIGEN macro. For more information, see *VTAM Operation*. With dumping enabled, VSCS can create a dump as soon as an internal error occurs. This ensures that the proper data is recorded before recovery processing begins. If an abend occurs, VSCS creates a dump automatically, whether dumping is enabled or not. Use the GCS GDUMP command to obtain a dump of the VSCS virtual machine for problems that do not cause VSCS to dump automatically.

While the dump is being taken, all network activity temporarily stops in order to keep the contents of the dump intact. When the dump is requested, VSCS issues a message containing the name of the module requesting the dump and a dump identifier number. VSCS dumps are numbered to correlate to the console messages.

Use the link-edit listing to locate key modules or work areas in the dump. The addresses of CSECTS DTIISDA1 and DTIISDA2 also appear in the trace table header record, as shown in Figure 31 on page 162. DTIISDA1 and DTIISDA2 contain the global control blocks, which can be identified in the dump by their unique identifiers. For more information on control block identifiers, see " VM VSCS Storage and Control Block Identifiers" on page 659.

VM Dump Formatting Routine

VSCS provides a dump formatter to format VSCS control blocks and the VSCS internal trace table in a dump. The routine runs as part of PRTDUMP.

1. Use the GCS GDUMP command to dump VSCS storage.
2. Use IPCS to process the reader file produced by the dump.
3. Use PRTDUMP to format the VSCS control blocks and internal trace table.

Note: Use the VSCS QUERY command to find the location of the VSCS internal trace table, and record its address. That way, if you cannot find the proper control blocks, you can still use the table's address to find them via the dump formatter.

Five options allow you to choose what portion of the dump you want to format:

1	2	3	4	5
Trace table	DTIPDBs	DTITAB	Trace table	Module list
DTIPRM	DTISDBs	DTIPIBs		
DTISAB	SCB summary	DTIPLB		
DTICGB		DTICLB		
DTIPGB		DTIVLB		
DTIVGB		Send RPL		
DTIGTD		DTIWEB		
DTISCB		DTIVEIB		
DTIUGB		DTICMT		
DTICIA		DTIXMT		
DTICMDs		Logical unit trace entries		
DTISWBs				
DTIPACT				
DTIDSBs				
VTAM work area				
Path table				
DTITABs				
DTIPIBs				
DTIPLBs				
DTICLBs				
DTIVLBs				
Send RPL				
SCB summary				

Formatting may terminate if the following conditions occur:

- If VSCS cannot find the control blocks in the dump. Early in initialization and late in termination, the appropriate pointers and exits may not exist. If VSCS cannot find the control blocks, it issues a message asking you for the VSCS trace table address. From this, it can locate the control blocks.
- If validity checking shows that a control block has been overlaid with data.
- If a page of storage that VSCS needs is unavailable.

If formatting fails, you may print the dump unformatted.

Figure 29 on page 157 shows how to use the dump formatting routine. It is taken from an operator console listing.

```
prtdump prb00001

PROCESSING FILE 'PRB00001 DUMP'
CSIIPR550R DO YOU WISH TO FORMAT VTAM CONTROL BLOCKS?  REPLY YES OR NO

y

FORMAT VSCS CONTROL BLOCKS? (YES OR NO)

y

DTIFT04 - VSCS DUMP FORMATTING
DTIFT05 - VSCS CONTROL BLOCKS LOCATED
DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
P. PRINT ENTIRE DUMP
Q. EXIT

ENTER OPTION(1-5) P-PRINT OR Q-EXIT:

3

ENTER LUNAME(1-8 CHARACTERS):

vtam

DTIFT03 - FORMATTING VTAM LU CONTROL BLOCKS
DTIF204 - SEARCHING FOR LU....
DTIF205 - EXTRACTING LU TRACE ENTRIES...
DTIF201 - LU NAME VTAM PROCESSED
DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
P. PRINT ENTIRE DUMP
Q. EXIT

ENTER OPTION(1-5) P-PRINT OR Q-EXIT:

5

DTIFT09 - SEARCHING FOR MODULE NAMES
DTIFT10 - SORTING MODULE NAMES...
DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
P. PRINT ENTIRE DUMP
Q. EXIT
```

Figure 29 (Part 1 of 2). **VM** Using the VSCS Dump Formatting Routine

```
ENTER OPTION(1-5) P-PRINT OR Q-EXIT:

1

DTIF101 - SHOULD THE VSCS INTERNAL TRACE TABLE BE FORMATTED? Y]N

n

DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
P. PRINT ENTIRE DUMP
Q. EXIT

ENTER OPTION(1-5) P-PRINT OR Q-EXIT:

2

DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
P. PRINT ENTIRE DUMP
Q. EXIT

ENTER OPTION(1-5) P-PRINT OR Q-EXIT:

3

ENTER LUNAME(1-8 CHARACTERS):

vtam

DTIFT03 - FORMATTING VTAM LU CONTROL BLOCKS
DTIF204 - SEARCHING FOR LU....
DTIF205 - EXTRACTING LU TRACE ENTRIES...
DTIF201 - LU NAME VTAM PROCESSED
DTIFT01 - VSCS DUMP FORMATTING OPTIONS
1. TRACE, GLOBAL AND LU RELATED CONTROL BLOCKS
2. STORAGE CONTROL BLOCKS
3. SPECIFIC LU CONTROL BLOCKS
4. VSCS TRACE TABLE ONLY
5. VSCS MODULE CROSS REFERENCE LISTING
P. PRINT ENTIRE DUMP
Q. EXIT

ENTER OPTION(1-5) P-PRINT OR Q-EXIT:

q

FORMAT VTAM CONTROL BLOCKS? (YES OR NO)

n
```

Figure 29 (Part 2 of 2). VM Using the VSCS Dump Formatting Routine

The general format of control blocks in a formatted dump is:

```
Control block name and address
Control block information
Offset .........data in dump format......... *..printable..*
```

VM VSCS Trace Facility

The VSCS trace facility offers the following trace options:

- CCS (Console Communications Services)
- Cleanup
- Data (V3R2)
- Dispatcher
- FRE (V3R2)
- GET (V3R2)
- VTAM exit.

All of the traces except cleanup are started with operator commands. (For the CCS and VTAM exit trace options, the default is active.) The cleanup trace starts automatically with any of the other traces.

The following list shows when each kind of trace entry is created:

CCS trace — Creates one entry for each IUCV request or response sent or received by VSCS.

Cleanup trace — Creates one or two entries for each logical unit at logoff or disconnect. This trace is activated automatically when one or more of the other traces is activated. It cannot run by itself.

Data trace (V3R2) — Creates one entry for logical unit-bound data (output), and one entry for CCS-bound logical unit data (input).

Dispatcher trace — Creates one entry each time DTIPSTAM or DTIVSTAM gives control to another module to process a work element block. This can occur more than once for a single work element block.

FRE trace (V3R2) — Creates one entry for each request to free a block of storage.

GET trace (V3R2) — Creates one entry for each request for a block of storage.

VTAM exit trace — Creates one entry each time VTAM gives control to a VSCS VTAM exit.

Figure 30 on page 160 shows where each trace occurs.

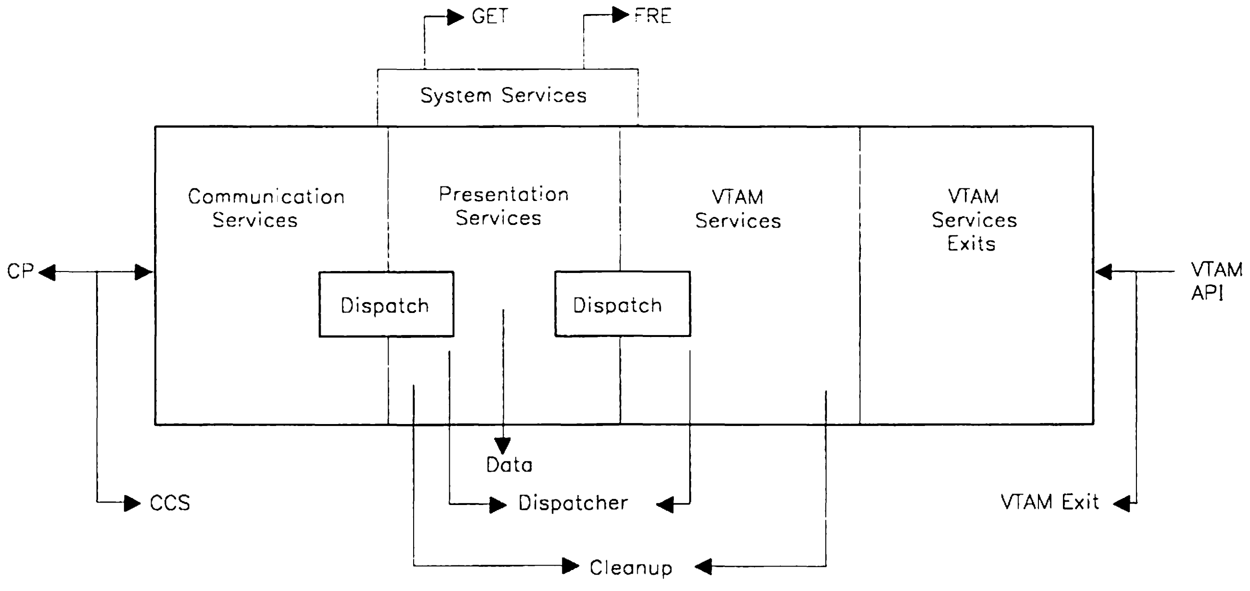


Figure 30. VM The VSCS Traces

VSCS records the data in an internal table that wraps around when full. The size of the trace table is variable and is specified during VSCS initialization. The default size is 1000 records of 32 bytes each.

The external trace option prevents trace information from being overlaid when the internal table wraps. If MODE=EXT, VSCS maintains both the internal trace table and the external trace table. This ensures that a dump taken for an unrecoverable error will contain the relevant data from the internal trace table. The external VSCS trace entries are sent to CPTRAP. See "VM How to Run External VSCS Traces" on page 161 for information on obtaining the external trace output.

Storage for the trace table is not allocated until the trace is activated and tracing begins. When the trace facility is turned off, storage for the trace table is preserved and can be dumped by the operator using the CP DUMP command. The VSCS QUERY command gives the starting and ending address of the trace table. In addition, the trace table can be found through a pointer at SABTRHDR in the DTISAB. See "VM How to Run External VSCS Traces" on page 161 and *VTAM Operation* for information on starting the VSCS traces.

VM Trace Degradation

Each trace option causes some system degradation, but it can be controlled to some extent. The cleanup and CCS traces cause the least system degradation. The VTAM exit trace causes a little more; and the data, dispatcher, FRE, and GET traces cause the most. That is why CCS is the default.

If one or very few logical units are experiencing a problem, and you are re-creating the problem to get documentation, run the traces on only one logical unit. This reduces the impact of the trace on system performance and reduces the frequency of the trace table wrapping.

VM How to Run External VSCS Traces

To run the VSCS traces externally, issue the following commands from the VSCS console:

`CPTRAP 3D`	
`CPTRAP GROUPID gcsgroup`	
`CPTRAP START TO userid`	*userid* is the receiver of the spool file
`VSCS TRACEON (EXT ALL`	EXT means run the trace externally. ALL means turn on the CCS, data, dispatcher, FRE, GET, and VTAM exit trace options.
`VSCS TRACEON luname`	This is required only for selective tracing.

VSCS starts tracing to a spool file. When you have collected enough data and you want to stop tracing, issue the following commands from the VSCS console:

`VSCS TRACEOFF (EXT ALL`	Turn off external tracing.
`VSCS TRACEON *`	Turn off selective tracing. This command is needed only if selective tracing was used above.
`VSCS TRACEON (CCS`	Activate the CCS trace option and use the internal trace table. It is recommended that you run with this minimum trace at all times.
`CPTRAP STOP`	

CP spools a file to the reader of the user ID specified in the CPTRAP START command. Use TRAPRED to look at the output as follows:

`ACCESS 193 x`	The GCS formatting routines are on this disk. "x" is a user-specified file mode.
`TRAPRED fileno`	"fileno" is the reader file number (Q RDR). (Make sure the reader file is not in hold status.)
`3D`	Read VTAM and VSCS records only.
`FORMAT`	Format the records.
`PRINTER 999999`	Print the records. (There are other commands which allow you to view the CPTRAP file online. See the *VM/SP System Programmer's Guide* for VM R4, or the *VM/SP Facility for System Programming* for VM R5.)
`QUIT`	

VM VSCS Trace Table

A trace may be started for one logical unit, up to 32 specific logical units, or all logical units at any time. If a trace is running for all logical units and the TRACEON command is specified for a single logical unit, then the trace only records data for the logical unit specified. In the same manner, if you are tracing a specific logical unit and issue TRACEON ALL, the trace now records data for all logical units.

The VSCS QUERY command shows:

- What traces are active
- Which logical units are being traced (if selective tracing is used)
- Which trace options are in effect
- The location of the internal trace table (if it is allocated).

VM Trace Table Header

The VSCS trace table contains a header record (DTITHDR) followed by the trace records (DTITREC). Figure 31 shows the trace table header.

00	T	R	Number of entries
04	Zero		
08	Current trace entry address		
0C	First trace entry in table		
10	Last trace entry in table		
14	Storage address used to free table		
18	Zero		
1C	Zero		
20	Address of DTIISDA1		
24	Address of DTIISDA2		
28	Address of first CSECT in load module		
2C	Address of DTIPATCH		
30	Address of DTIIPARM		
34	Zero		
38	Zero		
3C	Zero		

Figure 31. VM VSCS Trace Header Format

The header shows this information:

Byte (hex)	Contents
00–01	DTITHDR element header (TR)
02–03	Number of entries in table
04–08	Zero
08–0B	Pointer to current entry in table
0C–0F	Pointer to first entry in table
10–13	Pointer to last entry in table
14–17	Address to be used in freeing the trace table storage
18–1F	Zero
20–23	Address of DTIISDA1
24–27	Address of DTIISDA2

28–2B	Address of DTIISTRT
2C–2F	Address of DTIIPATCH
30–33	Address of DTIIPARM
34–3F	Zero

VM Examples of VSCS Trace Output

There are three ways to look at the VSCS trace table:

- Online at the operator console
- In dump output
- Formatted with TRAPRED.

1. To view the trace table **online**, issue the VM DISPLAY command from the VSCS operator console. (For more information on this command, see the *VM/SP CP Command Reference*.) Figure 32 on page 164 shows an example of the output. (The trace record titles were added to identify the different trace entry types.)

   Note: You can also use TRAPRED to view the output online. The output looks the same whether you use the VM DISPLAY command or TRAPRED. See the *VM/SP System Programmer's Guide* (for VM R4) or the *VM/SP Facility for System Programming* (for VM R5) for more information about TRAPRED.

2. Figure 33 on page 165 shows what the trace table looks like **in dump output**. (The trace record titles were added to identify the different trace entry types.)

3. Figure 34 on page 166 shows an example of a VSCS trace **running externally** and formatted with TRAPRED. (The procedure for obtaining this output is described in "**VM** How to Run External VSCS Traces" on page 161.) For more information on using TRAPRED, see the *VM/SP System Programmer's Guide*. (for VM R4) or the *VM/SP Facility for System Programming* (for VM R5).

```
Storage
Address
         Trace header
15E020   E3D97F58  00000000  0015E780  0015E060     *TR".......X....-*
15E030   0025CB40  0015E008  00000000  00000000     *... ............*
15E040   0013A948  001588F8  0012E000  0015C0D8     *.......8.......Q*
15E050   0013A568  00000000  00000000  00000000     *................*

         VSCS CCS data trace entries
15E060   C3D5C3E3  00012000  000A0000  5CC3C3E2     *CNCT........*CCS*
15E070   40404040  01010000  0000000A  E5D44040     *    ........VM  *

         VSCS CCS global trace entries
15E080   00125028  00000001  C7D3C2D3  D9010310     *..&.....GLBLR...*
15E090   0C000000  00110000  00000000  E2E240C5     *............SS E*

         VSCS VTAM exit trace entries
15E0A0   E5D3D6C7  00124028  001241B8  00001910     *VLOG.. .........*
15E0B0   00008000  00000000  10000000  00000000     *................*
15E0C0   E5D6D7D5  00124028  171241B8  00000000     *VOPN.. .........*
15E0D0   10758000  00000000  10000000  00070002     *................*

         VSCS dispatcher trace entries
15E0E0   D7020704  00124028  00127028  80000000     *P..... .........*
15E0F0   00000000  00000010  00000000  E2C9D5E3     *P...........SINT*
15E100   D7020000  00124028  401254E0  81000080     *P..... . .......*
15E110   00001D00  00000010  00000000  D6E6D9E3     *P...........OWRT*
15E120   E5031004  00124028  4025D028  81000080     *V..... . .......*
15E130   00000000  00000010  00000000  E2C5D5C4     *V...........SEND*

         VSCS CCS LU trace entries
15E220   0011E2B0  00020000  10751054  F2830009     *..S.........2...*
15E230   01000001  00020000  00000000  40040050     *............ ..&&
15E240   0011E310  00020000  10751054  C3830002     *..T.........C...*
15E250   0E000000  00040000  00000000  00040000     *................*

         VSCS cleanup trace entries
15E6C0   D7D9C2D2  02124028  00840000  80000000     *PRBK.. .........*
15E6D0   00000000  00000000  10751054  0025F188     *..............1.*
```

Figure 32. **VM** VSCS Trace Records Displayed at the Operator Console

```
STORAGE
 ADDR
         Trace header

15E020   E3D97F58 00000000 0015E780 0015E060 0025CB40 0015E008 00000000 00000000  *TR".......X....-... .............*
15E040   0013A948 001588F8 0012E000 0015C0D8 0013A568 00000000 00000000 00000000  *.......8.......O.................*


         VSCS CCS data entry

15E060   C3D5C3E3 00012000 000A0000 5CC3C3E2 40404040 01010000 0000000A E5D44040  *CNCT........*CCS    ........VM  *


         VSCS CCS global trace entry

15E080   00125028 00000001 C7D3C2D3 D9010310 0C000000 00110000 00000000 E2E240C5  *..&.....GLBLR...............SS E*


         VSCS VTAM trace entries

15E0A0   E5D3D6C7 00124028 001241B8 00001910 00008000 00000000 10000000 00000000  *VLOG.. .........................*
15E0C0   E5D6D7D5 00124028 171241B8 00000000 10758000 00000000 10000000 00070002  *VOPN.. .........................*


         VSCS Dispatcher trace entries

15E0E0   D7020704 00124028 00127028 80000000 00000000 00000010 00000000 E2C9D5E3  *P..... ....................SINT*
15E100   D7020000 00124028 401254E0 81000080 00001D00 00000010 00000000 D6E6D9E3  *P..... . ..................OWRT*
15E120   E5031004 00124028 4025D028 81000080 00000000 00000010 00000000 E2C5D5C4  *V..... . ..................SEND*


         VSCS CCS trace entries

15E220   0011E2B0 00020000 10751054 F2830009 01000001 00020000 00000000 40040050  *..S.........2............... ..&*
15E240   0011E310 00020000 10751054 C3830002 0E000000 00040000 00000000 00040000   *..T.........C..................*


         VSCS Cleanup trace entry

15E6C0   D7D9C2D2 02124028 00840000 80000000 00000000 00000000 10751054 0025F188  *PRBK.. .......................1.*
```

Figure 33. VM VSCS Trace Records in Dump Output

```
3D OE VTAM       VM/GCS USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 9A1222FFE3ABC000
       LENGTH OF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = F5
       EVENT IDENTIFICATION = EFAA
  VREC   RTN CODE 00  TAB   179B40  RPLREQ 23  RPL 17627C RPLFDBK 0000 RPLSENS 0000  NAU 0100
         PS STATE 00  PLBSM 00 00 00 00  VS STATE 00 VLBSWAP 30 80 00 00   VLBLRCD 00 EXIT DATA 04 00 00
3D OE VTAM       VM/GCS USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 9A1222FFE727C000
       LENGTH OF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = F5
       EVENT IDENTIFICATION = EFAA
  PDISP  LBCSTTUS 02 WEBFUN 04 WEBMODE 04  WEBVSRPS 7D TAB 179B40 WEB 17A9C8 PS STATE 00 PLBSM 00 00 98 00 PLBDFLG 02
         PLB- OFLG1 00 NOM 00 00 EDIT 00  PLBIA 00  VS STATE 00 VLBSWAP 30 80 00 00  VLBLRCD 00 MODULE IKIN
3D OE VTAM       VM/GCS USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 9A1222FFEBEF8000
       LENGTH OF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = F5
       EVENT IDENTIFICATION = EFAA
  CCS    WEB   17A9C8 PATHID 0003 0006   CID  0100 000A TYPE Y  PS STATE 00 WEBCOUNT 0003  WEBFUN 04 WEBMODE 00
         WEB- CHAR 00 EDIT 00 FLAGS 00 04 LINE 000A  CURSR 0000    TABCH 00 WEBRSVSA  00   WEBDATA 172FB100 *....*
```

Figure 34. vm VSCS Trace Records Formatted with TRAPRED

vm VSCS CCS Trace Record Format

When the CCS trace option is active, VSCS traces all traffic to and from CCS. You can limit the CCS trace to a selected set of logical units when you start it. The CCS trace produces three types of record formats:

Logical unit-related for traffic associated with logical units
Global for traffic not associated with a logical unit
Data for traffic associated with logical units or global traffic.

vm Format for a CCS Trace Entry for a Logical Unit

00	B'0'	DTIWEB ADDRESS		
04	PATH IDs			
08	VTAM CID			
0C	Flow ind.	LBCSTATE	WEBCOUNT	
10	WEBFUN	WEBMODE	WEBCHAR	WEBEDIT
14	WEBFLAGS	WEBCPFLG	WEBLINE	
18	WEBCURSR		WEBTABCH	WEBRSVSA
1C	First four characters of data			

Displacement:

00 Zero
01 – 03 Address of DTIWEB associated with this request or response
04 – 07 Path ID from Communication Services local block (CLB)

04 – 05 VSCS path ID
06 – 07 CCS path ID

08 – 0B VTAM communication identifier (CID), either OAF or DAF

0C

Flow indicator	Flow direction
R (Receive)	CCS = > VSCS (outbound)
Y (Reply)	VSCS = > CCS (inbound)
1 (1-way send)	VSCS = > CCS (inbound)
2 (2-way send)	VSCS = > CCS (inbound)
T (Reject)	VSCS = > CCS (inbound)
C (Message complete)	CCS = > VSCS (outbound)

Note: In this book, whenever a flow is referred to as *inbound* or *outbound*, you should take the Console Communications Services (CCS) as the reference point. That is, flows *toward* the CCS are *inbound*; flows *away from* the CCS are *outbound*.

0D LBCSTATE from PLB (See " VM LBCSTATE Values for PLBINxxx" on page 214.)

1...	LBCINT	Internal mode if on (on=1). If LBCINT is on, see " VM LBCSTATE Values for PLBINxxx" on page 214. If LBCINT is off, the following values apply:
.11.	LBCFSSMF	Full-screen support mode flags
.1..	LBCFSAV	Screen available
..1.	LBCFSII	Full-screen input inhibited:

B'00'	CP (Console) mode
B'01'	Virtual machine mode state between ATTN and READ, QUEUE CP messages.
B'10'	Virtual machine mode queue CP messages.
B'11'	Virtual machine mode available for switch.

...1	LBCCOPY	Copy mode: on=1
.... 1...	LBCNAL	NAL flag: on=1
.... .11.	LBCHOLD	Holding flag: on=B'11'
.... .1..	LBCM10	'More' 10 flag: on=1
.... ..1.	LBCM50	'More' 50 flag: on=1
....1	LBCCMSD	Screen written with a CMS DIAGNOSIS: on=1
....		No more holding state

0E – 0F WEBCOUNT Length of actual data in WEBDATA

10 WEBFUN WEB function code. This is explained in "VM WEB Function Codes" on page 215.

11 WEBMODE Mode:

00	Console
01	CMS
02	Full-screen
04	Internal

12 WEBCHAR Character set (set by CCS for Console and CMS modes)

00	EBCDIC character set
01	APL character set
02	TEXT character set

13 WEBEDIT Editing characteristics

1111	Zero

	.... 1...	WEBHIEKO	Highlight input display
	.... .1..	WEBEDPTI	Do not print or do not display
	.... ..1.	WEBNOEKO	Do not redisplay input on next output line
	1	WEBEDCUP	Convert to upper case
14	WEBFLAGS		
	11..	WEBRSV1	Zero
	.1..	WEBCONMD	Request for display in conmode 3270
	..1.	WEBLED	Limited edit diagnose
	...1	WEBANFSW	Allow any full-screen write; first full-screen write after reset.
	.... 1...	WEBDIAL	SNA dialed logical unit
	.... .1..	WEBNOMOR	CMS ERASE WRITE NO 'MORE' requested
	.... ..1.	WEBPRMPT	WEBDATA contains pre-logon prompt
	1	WEBPASPA	Pass PA1 (FSSM) as data to application
15	WEBCPFLG	CCS status flags for VSCS	
15	WEBSAFLG	VSCS status flags	
	1...	WEBCPNCR	No carriage return (hardcopy terminal)
	.1..	WEBCPPRI	Priority flag
	..1.	WEBCPALM	Ring the alarm
	..1.	WEBSAHLD	Holding
	...1	WEBCPMDE	CP-generated bit = 1
	...1	WEBSAMOR	'More' VM-generated bit = 0
	.... 1...	WEBCPHLW	Highlight write requested
	.... .1..	WEBCPRSP	Indicates CCS response
	.... .1..	WEBSARSP	Indicates VSCS response
	.... ..1.	WEBCPRRR	CCS request, requires response
	.... ..1.	WEBSARRR	VSCS request, requires response
	1	WEBCPRNR	CCS request, requires no response
	1	WEBSARNR	VSCS request, requires no response
16–17	WEBLINE	Line number for CMS WEBCMWRT	
16	WEBLAID	Logical attention identifier	
17	WEBNLLOS	Number of lines left on screen	
18–19	WEBCURSR	Cursor position may be a relative address or a buffer address	
1A	WEBTABCH	Tab (not DTITAB) character	
1B	WEBRSVSA	Reserved	
	1...	WEBCPRDS	DTIWEB is a redisplay DTIWEB
	.1..	WEBNOVCK	Do not validity check data
	..1.	WEBATTNW	Attention write (VSCS only) yes = 1
	...1 1...		Zero
	.... .1..	WEBCHDIR	Change direction required
	.... ..1.	WEBMASKW	Write inhibit mask
	1		Zero
1C–1F	WEBDATA	First four bytes of data field	

VM Format for a Global CCS Trace Entry

A global entry occurs if CCS receives a request for a path ID that has been severed.

<table>
<tr><td>00</td><td>B'0'</td><td colspan="3">DTIWEB address</td></tr>
<tr><td>04</td><td colspan="4">PATH IDs</td></tr>
<tr><td>08</td><td>G</td><td>L</td><td>B</td><td>L</td></tr>
<tr><td>0C</td><td>Flow ind.</td><td>LBCSTATE</td><td colspan="2">WEBCOUNT</td></tr>
<tr><td>10</td><td>WEBCAN</td><td>WEBMODE</td><td>WEBCHAR</td><td>WEBEDIT</td></tr>
<tr><td>14</td><td>WEBFLAGS</td><td>WEBCPFLG</td><td colspan="2">WEBLINE</td></tr>
<tr><td>18</td><td colspan="2">WEBCURSR</td><td>WEBTABCH</td><td>WEBRSVSA</td></tr>
<tr><td>1C</td><td colspan="4">First four characters of data</td></tr>
</table>

Displacement:

00 Zero
01 – 03 Address of DTIWEB associated with this request or response
04 – 07 Path ID (from CLB)

04 – 05 VSCS path ID
06 – 07 CCS path ID

08 – 0B Characters 'GLBL'

0C

Flow indicator	Flow direction
R (Receive)	CCS = > VSCS
Y (Reply)	VSCS = > CCS
1 (1-way send)	VSCS = > CCS
2 (2-way send)	VSCS = > CCS
T (Reject)	VSCS = > CCS
C (Message complete)	CCS = > VSCS

0D WEBCAN Number of control areas
0E – 0F WEBCOUNT Length of actual data in WEBDATA
10 WEBFUN WEB function code. This is explained in "VM WEB Function Codes" on page 215.
11 WEBMODE Mode:

00 Console
01 CMS
02 Full-screen
04 Internal

12 WEBCHAR Character set (set by CCS for Console and CMS modes)

00 EBCDIC character set
01 APL character set
02 TEXT character set

13 WEBEDIT Editing characteristics

1111		Zero
.... 1...	WEBHIEKO	Highlight input display
.... .1..	WEBEDPTI	Do not print or do not display
.... ..1.	WEBNOEKO	Do not redisplay input on next output line

	1	WEBEDCUP	Convert to upper case
14	WEBFLAGS		
	11..	WEBRSV1	Zero
	.1..	WEBCONMD	Request for display in conmode 3270
	..1.	WEBLED	Limited edit diagnose
	...1	WEBANFSW	Allow any full-screen write; first full-screen write after reset.
	.... 1...	WEBDIAL	SNA dialed logical unit
	.... .1..	WEBNOMOR	CMS ERASE WRITE NO 'MORE' requested
	.... ..1.	WEBPRMPT	WEBDATA contains pre-logon prompt
	1	WEBPASPA	PA1 (FSSM) pass as data to application
15	WEBCPFLG	CCS status flags for VSCS	
	1...	WEBCPNCR	No carriage return (hardcopy terminal)
	.1..	WEBCPPRI	Priority flag
	..1.	WEBCPALM	Ring the alarm
	..1.	WEBSAHLD	Holding
	...1	WEBCPMDE	CP-generated bit = 1
	...1	WEBSAMOR	'More' VM-generated bit = 0
	.... 1...	WEBCPHLW	Highlight write requested
	.... .1..	WEBCPRSP	Indicates CCS response
	.... .1..	WEBSARSP	Indicates VSCS response
	.... ..1.	WEBCPRRR	CCS request, requires response
	.... ..1.	WEBSARRR	VSCS request, requires response
	1	WEBCPRNR	CCS request, requires no response
	1	WEBSARNR	VSCS request, requires no response
16−17	WEBLINE	Line number for CMS WEBCMWRT	
16	WEBLAID	Logical attention identifier	
17	WEBNLLOS	Number of lines left on screen	
18−19	WEBCURSR	Cursor position may be a relative address or a buffer address	
1A	WEBTABCH	Tab (not DTITAB) character	
1B	WEBRSVSA	Reserved	
	1...	WEBCPRDS	DTIWEB is a redisplay DTIWEB
	.1..	WEBNOVCK	Do not validity check data
	..1.	WEBATTNW	Attention write (VSCS only) yes = 1
	...1 1...		Zero
	.... .1..	WEBCHDIR	Change direction required
	.... ..1.	WEBMASKW	Write inhibit mask
	1		Zero
1C−1F	WEBDATA	First four bytes of data field	

VM Format for CCS Data Trace Entries

```
0      4                                              1F
┌──────┬──────────────────//─────────────────────────────┐
│      │                                                 │
│Header│          Data                                   │
│      │                                                 │
└──────┴──────────────────//─────────────────────────────┘
```

Displacement:

00 – 03	Header		
	'ACNT'	Accept connection	CCS = > VSCS
	'CMSG'	Completion of a 1-way send	CCS = > VSCS
	'CNCT'	Connect	VSCS = > CCS
	'CXIT'	Invalid external interrupt	CCS = > VSCS
	'GMSG'	Ignored traffic on global path	CCS = > VSCS
	'LIER'	Logical interface error	CCS = > VSCS
	'SCNT'	Sever connection	CCS = > VSCS

	'SEVR'	Sever	VSCS = > CCS
			or
			CCS = > VSCS

04 – 1F	First 28 bytes of data from area passed		

ACNT Entry: Accept connection, CCS = > VSCS

```
0      4       8        C                     1F
┌──────┬───────┬────────┬─────────────────────┐
│      │       │        │                     │
│'ACNT'│Module │  Zero  │ IUCV parameter list │
│      │  ID   │        │                     │
└──────┴───────┴────────┴─────────────────────┘
```

Displacement:

00 – 03	Characters 'ACNT'		
04 – 07	Last four characters of module making trace entry. First four characters are always DTIP.		
08 – 0B	Zero		
0C – 0D	Path ID		
0E	IPFLAGS1		
	1...	IPALL	Quiesce, resume, and sever all
	.1..	IPQUSCE	Connect in quiesce mode
	..1.	IPPRTY	Priority message, reply required
	...1	IPNORPY	One-way protocol
	.... 1...	IPCPENTY	Entry from CP
	.... .1..	IPFGMID	Message ID specified
	.... ..1.	IPFGPID	Path ID specified
	1	IPFGMCL	Message class specified
0F	IPTYPE	External interrupt code	
10 – 11	IPSMGLIM	Message limit	
12 – 13	Zero		
14 – 15	IPAUDIT	Audit trail as follows:	
14	IPAUDIT1	Audit trail byte 1	
	1...	IPADRPLE	Reply too long for buffer

	.1..	IPADSNPX	Protection exception on send buffer
	..1.	IPADSNAX	Address exception on send buffer
	...1	IPADANPX	Protection exception on answer buffer
	.... 1...	IPADANAX	Address exception on answer buffer
	.... .1..	IPADRJCT	Message was rejected
	11		Zero
15	IPAUDIT2	Audit trail byte 2	
	1...	IPADRCPX	Protection exception on receive buffer
	.1..	IPADRCAX	Address exception on receive buffer
	..1.	IPADRPPX	Protection exception on reply buffer
	...1	IPADRPAX	Address exception on reply buffer
	.... 1...	IPADSVRD	Path was severed
	.... .111		Zero
16–17	Zero		
18–1B	User data		
1C–1F	First four bytes of logical unit name		

CMSG Entry: Completion of a one-way send, CCS = > VSCS

```
0         4           8           C                          1F
'CMSG'    Module ID   Zero        IUCV parameter list
```

Displacement:

00–03	Characters 'CMSG'		
04–07	Last four characters of module making trace entry. First four characters are always DTIP.		
08–0B	Zero		
0C–0D	Path ID		
0E	IPFLAGS1		
	1...	IPALL	Quiesce, resume, and sever all
	.1..	IPQUSCE	Connect in quiesce mode
	..1.	IPPRTY	Priority message, reply required
	...1	IPNORPY	One-way protocol
	.... 1...	IPCPENTY	Entry from CP
	.... .1..	IPFGMID	Message ID specified
	.... ..1.	IPFGPID	Path ID specified
	1	IPFGMCL	Message class specified
0F	IPTYPE	External interrupt code	
10–13	IPMSGID	Message ID	
14–17	IPTRGCLS	Target class	
14–15	IPAUDIT	Audit trail as follows:	
14	IPAUDIT1	Audit trail byte 1	
	1...	IPADRPLE	Reply too long for buffer
	.1..	IPADSNPX	Protection exception on send buffer
	..1.	IPADSNAX	Address exception on send buffer
	...1	IPADANPX	Protection exception on answer buffer

	.... 1...	IPADANAX	Address exception on answer buffer
	.... .1..	IPADRJCT	Message was rejected
	11		Zero
15	IPAUDIT2	Audit trail byte 2	
	1...	IPADRCPX	Protection exception on receive buffer
	.1..	IPADRCAX	Address exception on receive buffer
	..1.	IPADRPPX	Protection exception on reply buffer
	...1	IPADRPAX	Address exception on reply buffer
	.... 1...	IPADSVRD	Path was severed
	.... .111		Zero
16—17		Zero	
18—1B	IPBFADR1	Address of buffer 1	
1C—1D		Zero	
1E—1F	IPBFLN1	Length of buffer 1	

CNCT Entry: Request for connection, VSCS = > CCS

```
0       4                                    1F
'CNCT'  IUCV parameter list
```

Displacement:

00—03	Characters 'CNCT'		
04—05	Path ID		
06	IPFLAGS1		
	1...	IPALL	Quiesce, resume, and sever all
	.1..	IPQUSCE	Connect in quiesce mode
	..1.	IPPRTY	Priority message, reply required
	...1	IPNORPY	One-way protocol
	.... 1...	IPCPENTY	Entry from CP
	.... .1..	IPFGMID	Message ID specified
	.... ..1.	IPFGPID	Path ID specified
	1	IPFGMCL	Message class specified
07	IPRCODE	Return code	
08—09	IPSMGLIM	Message limit	
0A	IPFCNCD	Function code	
0B	Zero		
0C—13	IPVMID	Target virtual machine ID	
14—1B	'*CCS'		
1C—1F	First four bytes of logical unit name		

CXIT Entry: Invalid external interrupt, ccs = > vscs

```
0        4                                      1C       1D       1F
┌────────┬──────────────────────────────────────┬────────┬────────┐
│        │                                      │ Error  │        │
│ 'CXIT' │Bytes 0-23 of external parameter list │ Code   │  VEIB  │
│        │                                      │        │        │
└────────┴──────────────────────────────────────┴────────┴────────┘
```

Displacement:

00–03	Characters 'CXIT'		
04–05	Path ID		
06	IPFLAGS1		
	1...	IPALL	Quiesce, resume, and sever all
	.1..	IPQUSCE	Connect in quiesce mode
	..1.	IPPRTY	Priority message, reply required
	...1	IPNORPY	One-way protocol
	.... 1...	IPCPENTY	Entry from CP
	.... .1..	IPFGMID	Message ID specified
	.... ..1.	IPFGPID	Path ID specified
	1	IPFGMCL	Message class specified
07	IPTYPE	External interrupt code	
08–0B	IPMSGID	Message ID	
0C–0F	IPTRGCLS	Target class	
0C–0D	IPAUDIT	Audit trail as follows:	
0C	IPAUDIT1	Audit trail byte 1	
	1...	IPADRPLE	Reply too long for buffer
	.1..	IPADSNPX	Protection exception on send buffer
	..1.	IPADSNAX	Address exception on send buffer
	...1	IPADANPX	Protection exception on answer buffer
	.... 1...	IPADANAX	Address exception on answer buffer
	.... .1..	IPADRJCT	Message was rejected
	11		Zero
0D	IPAUDIT2	Audit trail byte 2	
	1...	IPADRCPX	Protection exception on receive buffer
	.1..	IPADRCAX	Address exception on receive buffer
	..1.	IPADRPPX	Protection exception on reply buffer
	...1	IPADRPAX	Address exception on reply buffer
	.... 1...	IPADSVRD	Path was severed
	.... .111		Zero
0E–0F		Zero	
10–13	IPBFADR1	Address of buffer 1	
14–15		Zero	
16–17	IPBFLN1	Length of buffer 1	
18–1B	IPSRCCLS	Source class	

1C	DTICEXIT	Error code point
	01	VSCS abnormally terminating
	02	Pending connect received - not supported
	03	Error dequeueing VEIB or no available VEIBs
	04	Error enqueueing VEIB on CGBCGMQ
	05	Error found - not specific
	06	Error enqueueing or dequeueing
	07	Scheduling with priority failed
	08	Scheduling failed
	09	No match found for path ID in VSCS path table
	0A	Control block address passed does not point to a valid VSCS control block (SAB or TAB)
	0B	Storage shortage obtaining additional VEIBs
	0C	Storage shortage obtaining additional VEIBs
1D–1F	VEIB Address	

GMSG Entry: Traffic on global path is ignored, ccs => vscs

0	4	1C 1F
'GMSG'	Bytes 0-23 of external parameter list	VEIB

Displacement:

00–03	Characters 'GMSG'		
04–05	Path ID		
06	IPFLAGS1		
	1...	IPALL	Quiesce, resume, and sever all
	.1..	IPQUSCE	Connect in quiesce mode
	..1.	IPPRTY	Priority message, reply required
	...1	IPNORPY	One-way protocol
	.... 1...	IPCPENTY	Entry from CP
	.... .1..	IPFGMID	Message ID specified
	.... ..1.	IPFGPID	Path ID specified
	1	IPFGMCL	Message class specified
07	IPTYPE	External interrupt code	
08–0B	IPMSGID	Message ID	
0C–0F	IPTRGCLS	Target class	
0C–0D	IPAUDIT	Audit trail as follows:	
0C	IPAUDIT1	Audit trail byte 1	
	1...	IPADRPLE	Reply too long for buffer
	.1..	IPADSNPX	Protection exception on send buffer
	..1.	IPADSNAX	Address exception on send buffer
	...1	IPADANPX	Protection exception on answer buffer
	.... 1...	IPADANAX	Address exception on answer buffer
	.... .1..	IPADRJCT	Message was rejected
	11		Zero
0D	IPAUDIT2	Audit trail byte 2	
	1...	IPADRCPX	Protection exception on receive buffer

.1..	IPADRCAX	Address exception on receive buffer
..1.	IPADRPPX	Protection exception on reply buffer
...1	IPADRPAX	Address exception on reply buffer
.... 1...	IPADSVRD	Path was severed
.... .111		Zero

0E—0F		Zero
10—13	IPBFADR1	Address of buffer 1
14—15		Zero
16—17	IPBFLN1	Length of buffer 1
18—1B	IPSRCCLS	Source class
1C—1F	VEIB Address	

LIER Entry: Logical interface error, ccs => vscs

0	4	8	10	14	18	1F
'LIER'	Last four char of module	LU name with logic err	DTITAB ADDR	INPUT DTIWEB ADDR	'0'	

Displacement:

00—03	Character 'LIER'
04—07	Last four characters of module making trace entry. First four characters are always DTIP.
08—0F	Logical unit name of device with logical interface error.
10—13	Address of DTITAB
14—17	Address of DTIWEB that was input to module making trace entry.
18—1F	Zero

SCNT Entry: Request to sever connection, ccs => vscs

0	4	6	14	18	1C	1D	1F
'SCNT'	Path ID	'0'	Address of TAB or SAB	'0'	Reason Code	'0'	

Displacement:

00—03	Characters 'SCNT'
04—05	ID of path that was severed
06—13	Zero
14—17	Address of Terminal Anchor Block (TAB) or Service Application Block (SAB)
18—1B	Zero

1C Reason code

1	VSCS not enabled by CP
2	Maximum logical unit limit exceeded
3	No VSCS global CONNECT accepted
4	VTAM service machine (VSM) not enabled
5	Duplicate logical unit name found on VSM RDEVBLOK chain
6	This path ID is currently in use
7	VTAM service machine was previously connected
8	Message limit exceeded
9	VTAM service machine global path severed

1D – 1F Zero

SEVR Entry: Connection severed, VSCS = > CCS or CCS to = > VSCS

0	4	6	14	18 1F
'SEVR'	Path ID	'0'	Address of TAB or SAB	'0'

Displacement:

00 – 03	Characters 'SEVR'
04 – 05	ID of path that was severed
06 – 13	Zero
14 – 17	Address of TAB or SAB
18 – 1F	Zero

VM VSCS Dispatcher Trace Record Format

When the dispatcher trace option is active, VSCS traces all traffic through the VSCS state managers (DTIPSTAM and DTIVSTAM). This trace can be limited to one or several logical units by starting the VSCS dispatcher trace for selected logical units. Use the VSCS TRACEON (DISP command to start the VSCS dispatcher trace. See *VTAM Operation* for additional information. The dispatcher trace produces two types of record formats:

Logical unit-related for traffic associated with logical units
Global for traffic not associated with a logical unit.

The following examples show the format of records created for each event.

VM Format for a Dispatcher Trace Entry for a Logical Unit

00	Header	LBCSTTUS	WEBFUN	WEBMOD
04	WEBVSRPS	DTITAB address		
08	PLBDFLG1	DTIWEB address		
0C	LBCSTATE	PLBFSSMF	PLBLFLG1	PLBFLG1
10	PLBPRLBC	PLBOFLG1	PLBNOM	
14	PLBEDIT	PLBIAFLG	LBCSTATE	VLBFLG
18	VLBFLG1	VLBFLG2	VLBSENDF	VLBLRCD
1C	Last four characters of module name			

Displacement:

00 Header ID

D	No module ID found or address of module is zero
P	Presentation Services dispatcher
V	VTAM Services dispatcher

01 LBCSTTUS and current task running

1...	LBCSIP	1 = busy, 0 = not busy
.111 1...		Zero
.... ..1.		Presentation Services task running
.... ..11		VTAM Services task running
.... .1..		Unknown task running

02 WEBFUN WEB function code from input WEB. This is explained in "VM WEB Function Codes" on page 215.

03 WEBMODE WEB mode from input WEB

00	Console
01	CMS
02	Full-screen
04	Internal

04 WEBVSRPS VTAM Services response to SEND request from Presentation Services

01	WEBVSOPC	Operation check
02	WEBVSLFU	Unconditional logoff
03	WEBVSERR	Unrecoverable I/O error
04	WEBVSBSY	Printer busy
05	WEBVSIRQ	Intervention required
06	WEBVSNAV	Printer not available
07	WEBTWXER	TWX read error, sense code 0827
08	WEBVSAOK	Successful completion
0C	WEBVSVER	Unrecoverable internal error
0D	WEBVSPSL	Presentation space integrity lost
0E	WEBVSCNL	SEND canceled by cancel key
0F	WEBVSCC1	Reflect attention, CC1 on SIO
10	WEBVSECU	Equipment check and unit specify

	11	WEBVSDCU	Data check and unit specify
	12	WEBVSEC	Equipment check
	13	WEBVSCRJ	Command reject
	14	WEBVSDC	Data check
	15	WEBVSCC	Command check
05—07	DTITAB	Address of TAB passed	
08	PLBDFLG1	Device manager flags	
	1...	PLBDASS	Display has alternate size
	.1..	PLBDNEW	Bit is on for a 3278, bit is off for a 3277
	..1.	PLBNOVCK	Do not validity check data
	...1 11..	PLBDCH	Display has new color or highlight functions, no B'000'
	...1	PLBDXTDS	Extended data stream available
	.... 1...	PLBDXCLR	Extended color available
	.... .1..	PLBDXHLI	Extended highlight available
	.... ..1.	PLBDATTN	Printer attention
	1	PLBDMASK	Password mask
09—0B	DTIWEB	Address of current WEB	
0C	LBCSTATE from PLB (See "VM LBCSTATE Values for PLBINxxx" on page 214.)		
	1...	LBCINT	Internal mode if on (on=1). If LBCINT is on, see "VM LBCSTATE Values for PLBINxxx" on page 214. If LBCINT is off, the following values apply:
	.11.	LBCFSSMF	Full-screen support mode flags
	.1..	LBCFSAV	Screen available
	..1.	LBCFSII	Full-screen input inhibited: B'00' CP (Console) mode B'01' Virtual machine mode state between ATTN and READ, QUEUE CP messages. B'10' Virtual machine mode QUEUE CP messages. B'11' Virtual machine mode available for switch.
	...1	LBCCOPY	Copy mode: on=1
	.... 1...	LBCNAL	NAL flag: on=1
	.... .11.	LBCHOLD	Holding flag: on=B'11'
	.... .1..	LBCM10	'More' 10 flag: on=1
	.... ..1.	LBCM50	'More' 50 flag: on=1
	1	LBCCMSD	Screen written with a CMS DIAGNOSE: on=1
	....		No more holding state
0D	PLBFSSMF	Full-screen support mode flags	
	1...	PLBFPA1	PA1 key received switch mode
	.1..	PLBFSRFC	Mode switch required (FSSM to Console)
	..1.	PLBFSAFC	Switch available (FSSM to Console)
	...1	PLBFCNMP	Console message pending
	.... 1...	PLBFVSIP	VTAM Services SEND in progress

	.... .1..	PLBFCLGF	Conditional logoff received
	.... ..1.	PLBFCMRR	CP message pending, response required
	1	PLBCMSEW	Screen has just been erased
0E	PLBLFLG1	VTAM Services logoff flags for Presentation Services	
	1...	PLBLVS	VTAM Services logoff received, bypass logo
	.1..	PLBLGFC	WEBINLGF sent to CCS conditional logoff
	..1.	PLBLGFU	WEBINLGF sent to CCS unconditional logoff
	...1	PLBLERR	WEBINERR sent to CCS for hard error
	.... 1...	PLBLLERR	WEBINLER sent to CCS for interface error
	.... .1..	PLBLVERR	WEBINVER sent to CCS for internal logic error
	.... ..1.	PLBLSEVR	CP CCS SEVER received, must sever
	1	PLBLNSTR	No storage indicator in logoff
0F	PLBFLG1	PLB flags	
	1...	PLBLGO	Logo mode logical unit
	.1..	PLBINAR	APL/TEXT starting in input area
	..1.	PLBLFLNP	Logon from logo not permitted
	...1	PLBCONCT	Current count incremented
	.... 1...	PLBCNCTI	IUCV CONNECT issued
	.... .1..	PLBINREC	Input received during purge processing
	.... ..1.	PLBPRMPT	Usability prompt message flag
	1	PLBSDIAL	SNA-dialed logical unit
10	PLBPRLBC	Previous logical unit LBCSTATE	
11	PLBOFLG1	Output manager flags	
	1...	PLBSDIAG	CMS DIAGNOSE on screen
	.1..	PLBPSMSG	Priority message on screen
	..1.	PLBOSRIP	SEND retry in progress
	...1	PLBOKLOK	Keyboard locked
	.... 1...	PLBOKPMI	Keyboard printer mask input area sent
	.... .1..	PLBOSWRT	CMS WRITE active (no input received)
	.... ..1.	PLBOCRRQ	Carriage return required before SEND on hardcopy terminal
	1	PLBOEXRQ	'!' required before SEND on hardcopy terminal
12–13	PLBNOM	Notify output manager	
12	PLBNFLG1	NOM flag 1	
	1...	PLBCEND	Command end (read)
	.1..	PLBRDTX	The redisplay timer has expired
	..1.	PLBSAC	Status area change required
	...1	PLBCLROA	Clear output area
	.... 1...	PLBCLRIA	Clear input area
	.... .1..	PLBRING	Ring the alarm
	.... ..1.		Zero
	1	PLBUNLK	Unlock keyboard
13	PLBNFLG2	NOM flag 2	

	1...	PLBRDPND	Redisplay pending
	.111 1111		Zero
14	PLBEDIT	Editing flags	
	1...	PLBNCONV	User-translated output data
	.111		Zero
	.... 1...	PLBHIEKO	Highlight input redisplay
	.... .1..	PLBEDPTI	Do not print or do not display
	.... ..1.	PLBNOEKO	Do not redisplay input on next output line
	1	PLBEDCUP	Convert to upper case
15	PLBIATTN	Input manager attention handling	
15	PLBIAFLG	Input manager attention handling flags	
	1...	PLBMATTN	Multiple attention received
	.1..	PLBSATTN	Single attention received
	..1.	PLBIPA1	Attention received in 'MORE'
	...1 1111		Zero
16	LBCSTATE	VLB	
	1	LBCVSST	VTAM Services state: 0 = Normal, 1 = Termination
17	VLBFLG	VLB flags	
	1...	VLBBSPBB	Bracket-state pending begin bracket
	.1..	VLBBSPEB	Bracket-state pending end bracket
	..1.	VLBBSINB	Bracket-state in bracket
	...1	VLBSR	Application SEND or RECEIVE state: 1 = SEND state, 0 = RECEIVE state
	 1111		Zero
18	VLBFLG1	VLB flag 1	
	1...	VLBIOTR	Inbound data received since last SEND
	.1..	VLBCLSD	CLSDST not required
	..1.	VLBPCS	Purging chain state
	...1	VLBSIGNL	Signal sent
	.... 1...	VLBCHGD	Change direction required
	.... .1..	VLBRETR	Retry RECEIVE request
	.... ..1.	VLBRETS	Retry SEND request
	1		Zero
19	VLBFLG2	VLB flag 2	
	1...	VLBRPLB	RPL in use
	.1..	VLBCLPND	VTAM CLSDST pending
	..11		Zero
	.... 1...	VLBBUSY	Printer busy
	.... .1..	VLBNAV	Printer not available
	.... ..1.	VLBOPCK	Operation check occurred
	1	VLBWAIT	Waiting for data or LUSTAT
1A	VLBSENDF	Special SEND indicator	
	1...	VLBCLR	Clear required
	.1..	VLBSDT	Start data traffic (SDT) required
	..1.	VLBCANC	Cancel required
	...1	VLBSIGRQ	Signal required

	 1111		Zero
1B	VLBLRCD	Logoff reason codes	
	1...	VLBVTLO	VTAM LOGOFF
	.1..	VLBVRYI	VARY INACTIVE
	..1.	VLBHLTQ	HALT QUICK
	...1	VLBHLTC	HALT CANCEL
	.... 1...	VLBIOE	Unrecoverable I/O error
	.... .1..	VLBINERR	Internal error
	11		Zero
1C – 1F	Last four characters of module ID		

VM Format for a Dispatcher Trace Entry for a Global Connection

00	Header ID	Task/Busy	SABIQECB	
04	Address passed			
08	DTIWEB address			
0C	SABFLG1	SABABIPF	SABLFLG1	SABRFLG1
10	WEBFUN	WEBMODE	WEBCHAR	WEBEDIT
14	WEBFLAGS	WEBCPFLG	WEBLINE	
18	WEBCURSR		WEBTABCH	WEBRSVSA
1C	Last four characters of module name			

Displacement:

0	Trace record header ID		
	D	No module ID found or module address is zero	
	P	Presentation Services dispatcher	
	V	VTAM Services dispatcher	
1	LBCSTTUS and current task running		
	...1	Address of SAB included	
	.111 1...	Zero	
	.... ..1.	Presentation Services task running	
	11	VTAM Services task running	
	.... .1..	Unknown task running	
02 – 03	SABIQECB	First two bytes of SABIQECB event control block (ECB)	
04 – 07		Address passed	
08 – 0B	DTIWEB	Address of WEB	
0C	SABFLGS1		
	1...	SABDMPNO	Do not take a dump
	.1..	SABVSCSV	VSCS running in VTAM machine
	..1.	SABDUIP	Dump in progress
	...1	SABCSOPN	VSIUCV open

	.... 1...	SABBUFDQ	DTISGETB dequeue type: FIFO=1, LIFO=0
	.... .1..	SABGETFL	GETMAIN failed
	11		Zero
	....1.	SABNEWLG	New logo format allowed
	1	SABVEIBS	VEIB shortage has occurred
	11		Zero
0D	SABABIPF	Abend-in-progress flags	
	1...	SABVABIP	VTAM Services abend in progress
	.1..	SABPABIP	Presentation Services abend in progress
	..1.	SABUABIP	Utility Services abend in progress
	...1 1111		Zero
0E	SABTFLG1	Global termination flags	
	1...	SABTETPE	TPEND in progress
	.1..	SABTESIQ	Termination scheduled by DTIIQUIT
	..1.	SABTETIP	Termination in progress
	...1	SABTEATP	Abnormal termination in progress
	.... 1...	SABINVQT	Invalid termination request detected
	 1111		Zero
0F	SABRFLG1	Global recovery flags	
	1...	SABVSCTL	VTAM Services task in control
	.1..	SABPSCTL	Presentation Services task in control
	..1.	SABTICTL	Timer exit in control
	...1	SABCXCTL	Communications Services exit in control
	 1111		Zero
10	WEBFUN	WEB function code. This is explained in "VM WEB Function Codes" on page 215.	
11	WEBMODE	Mode:	
		00 Console	
		01 CMS	
		02 Full-screen	
		04 Internal	
12	WEBCHAR	Character set (set by CCS for Console and CMS modes)	
		00 EBCDIC character set	
		01 APL character set	
		02 TEXT character set	
13	WEBEDIT	Editing characteristics	
	1111		Zero
	.... 1...	WEBHIEKO	Highlight input display
	.... .1..	WEBEDPTI	Do not print or do not display
	.... ..1.	WEBNOEKO	Do not redisplay input on next output line
	1	WEBEDCUP	Convert to upper case
14	WEBFLAGS		
	11..	WEBRSV1	Zero
	.1..	WEBCONMD	Request for display in conmode 3270
	..1.	WEBLED	Limited edit diagnose
	...1	WEBANFSW	Allow any full-screen write; first full-screen write after reset.

	.... 1...	WEBDIAL	SNA-dialed logical unit
	.... .1..	WEBNOMOR	CMS ERASE or WRITE NO 'MORE' requested
	.... ..1.	WEBPRMPT	WEBDATA contains pre-logon message
	1	WEBPASPA	Pass PA1 (FSSM) as data to application
15	WEBCPFLG	CCS status flags for VSCS	
	1...	WEBCPNCR	Do not imbed carriage return in data stream (on hardcopy terminal)
	.1..	WEBCPPRI	Priority flag
	..1.	WEBCPALM	Ring the alarm
	..1.	WEBSAHLD	Holding
	...1	WEBCPMDE	CP-generated bit = 1
	...1	WEBSAMOR	'More' VM-generated bit = 0
	.... 1...	WEBCPHLW	Highlight write requested
	.... .1..	WEBCPRSP	Indicates CCS response
	.... .1..	WEBSARSP	Indicates VSCS response
	.... ..1.	WEBCPRRR	CCS request, requires response
	.... ..1.	WEBSARRR	VSCS request, requires response
	1	WEBCPRNR	CCS request, requires no response
	1	WEBSARNR	VSCS request, requires no response
16–17	WEBLINE	Line number for CMS WEBCMWRT	
16	WEBLAID	Logical attention identifier	
17	WEBNLLOS	Number of lines left on screen	
18–19	WEBCURSR	Cursor position may be a relative address or a buffer address	
1A	WEBTABCH	Tab (not DTITAB) character	
1B	WEBRSVSA	Reserved for VSCS use	
	1...	WEBCPRDS	DTIWEB is a redisplay DTIWEB
	.1..	WEBNOVCK	Do not validity check data
	..1.	WEBATTNW	Attention write (VSCS only) yes = 1
	...1 1...		Zero
	.... .1..	WEBCHDIR	Change direction required
	.... ..1.	WEBMASKW	Write inhibit mask
	1		Zero
1C–1F	Last four characters of module ID (or entry address)		

VM VSCS VTAM Exit Trace Record Formats

If you are running the default VSCS trace (VSCS V3R1.2 and later) and you have activated the trace with an operator command, or by starting VSCS with the VTAM exit trace active, VSCS traces all inbound traffic through the VSCS VTAM exits. You can limit the VTAM exit trace to a selected set of logical units by starting VSCS VTAM trace for those logical units.

You can produce the VTAM exit trace in two formats:

Normal trace record if the request is a valid VSCS logical unit
Data trace record if the request or response was invalid, or if the data traced is different from the normal VTAM exit trace record.

Below is the format for the records created for each event.

VM VSCS VTAM Exit Trace Entries

00	V	Module ID		
04	RTN CODE	DTITAB address		
08	RPL REQ	RPL address		
0C	RPLFDBK		RPLSSNSI or RPLUSNSI	
10	SOURCE NAU		LBCSTATE	PLBFSSMF
14	PLBLFLG1	PLBNFLG1	PLBOFLG1	LBCSTATE
18	VLBSWAP			
1C	VLBLRCD	VXIT data		

DTIVCLSX
DTIVDFAX
DTIVLOGX
DTIVNSEX
DTIVOPNX
DTIVRECX
DTIVRESX
DTIVRLQX
DTIVSCIX
DTIVSNDX

Displacement:

00	V=VTAM Services
01–03	Three-character module ID DTIVxxxx
04	Return code from module
05–07	TAB address for current request
08	RPL request code
09–0B	RPL address
0C–0D	RPLFDB2
0E–0F	RPLUSNSI (user sense) or RPLSSNSI (system sense)
10–11	Source network address
12	LBCSTATE from PLB (See "VM LBCSTATE Values for PLBINxxx" on page 214.)

1...	LBCINT	Internal mode if on (on=1). If LBCINT is on, see "VM LBCSTATE Values for PLBINxxx" on page 214. If LBCINT is off, the following values apply:
.11.	LBCFSSMF	Full-screen support mode flags
.1..	LBCFSAV	Screen available
..1.	LBCFSII	Full-screen input inhibited: B'00' CP (Console) MODE B'01' Virtual machine mode state between ATTN and READ, QUEUE CP messages. B'10' Virtual machine mode QUEUE CP messages. B'11' Virtual machine mode available for switch.
...1	LBCCOPY	Copy mode: on=1
.... 1...	LBCNAL	NAL flag: on=1
.... .11.	LBCHOLD	Holding flag: on=B'11'
.... .1..	LBCM10	'More' 10 flag: on=1
.... ..1.	LBCM50	'More' 50 flag: on=1

	1	LBCCMSD	Screen written with a CMS DIAGNOSE: ON=1
	....		No more holding state
13	PLBFSSMF	Full-screen support mode flags	
	1...	PLBFPA1	PA1 key received switch mode
	.1..	PLBFSRFC	Mode switch required (FSSM to Console)
	..1.	PLBFSAFC	Switch available (FSSM to Console)
	...1	PLBFCNMP	Console message pending
	.... 1...	PLBFVSIP	VTAM Services SEND in progress
	.... .1..	PLBFCLGF	Conditional logoff received
	.... ..1.	PLBFCMRR	CP message pending, response required
	1	PLBCMSEW	Screen just cleared
14	PLBLFLG1	VTAM Services logoff flags for Presentation Services	
	1...	PLBLVS	VTAM Services logoff received, bypass logo
	.1..	PLBLGFC	WEBINLGF sent to CCS conditional logoff
	..1.	PLBLGFU	WEBINLGF sent to CCS unconditional logoff
	...1	PLBLERR	WEBINERR sent to CCS for hard error
	.... 1...	PLBLLERR	WEBINLER sent to CCS for interface error
	.... .1..	PLBLVERR	WEBINVER sent to CCS for internal logic error
	.... ..1.	PLBLSEVR	CP CCS SEVER received, must sever
	1	PLBLNSTR	No storage indicator in logoff
15	PLBNFLG1	NOM flag 1	
	1...	PLBCEND	Command end (read)
	.1..	PLBRDTX	The redisplay timer has expired
	..1.	PLBSAC	Status area change required
	...1	PLBCLROA	Clear output area
	.... 1...	PLBCLRIA	Clear input area
	.... .1..	PLBRING	Ring the alarm
	.... ..1.		Zero
	1	PLBUNLK	Unlock keyboard
16	PLBOFLG1	Output manager flags	
	1...	PLBSDIAG	CMS DIAGNOSE on screen
	.1..	PLBPSMSG	Priority message on screen
	..1.	PLBOSRIP	SEND retry in progress
	...1	PLBOKLOK	Keyboard locked
	.... 1...	PLBOKPMI	Keyboard printer mask input area sent
	.... .1..	PLBOSWRT	CMS WRITE active (no input received)
	.... ..1.	PLBOCRRQ	Carriage return required before SEND (on hardcopy terminal)
	1	PLBOEXRQ	'!' required before SEND (on hardcopy terminal)
17	LBCSTATE	From VLB	
	1	LBCVSST	VTAM Services state: 0 = Normal, 1 = Termination

18	VLBFLG	VLB flags	
	1...	VLBBSPBB	Bracket state-pending begin bracket
	.1..	VLBBSPEB	Bracket state-pending end bracket
	..1.	VLBBSINB	Bracket state-in bracket
	...1	VLBSR	Application SEND or RECEIVE state: 1 = SEND state, 0 = RECEIVE state
	 1111		Zero
19	VLBFLG1	VLB flag 1	
	1...	VLBIOTR	Inbound data received since last SEND indicator
	.1..	VLBCLSD	CLSDST: 0 = required, 1 = not required
	..1.	VLBPCS	Purging chain state
	...1	VLBSIGNL	Signal sent
	.... 1...	VLBCHGD	Change direction required
	.... .1..	VLBRETR	Retry RECEIVE request
	.... ..1.	VLBRETS	Retry SEND request
	1		Zero
1A	VLBFLG2	VLB flag 2	
	1...	VLBRPLB	RPL in use
	.1..	VLBCLPND	VTAM CLSDST pending
	..11		Zero
	.... 1...	VLBBUSY	Printer busy
	.... .1..	VLBNAV	Printer not available
	.... ..1.	VLBOPCK	Operation check occurred
	1	VLBWAIT	Waiting for data or LUSTAT
1B	VLBSENDF	Special function flags	
	1...	VLBCLR	Clear required
	.1..	VLBSDT	SDT required
	..1.	VLBCANC	Cancel required
	...1	VLBSIGRQ	Signal required
	 1111		Zero
1C	VLBLRCD	Logoff reason codes	
	1...	VLBVTLO	VTAM LOGOFF
	.1..	VLBVRYI	VARY INACTIVE
	..1.	VLBHLTQ	HALT QUICK
	...1	VLBHLTC	HALT CANCEL
	.... 1...	VLBIOE	Unrecoverable I/O error
	.... .1..	VLBINERR	Internal error
	11		Zero
1D–1F	VTAM exit-module-dependent data		

VM VTAM Exit-Module-Dependent Data

DTIVCLSX

1-3	WEB address or zero	

DTIVDFAX

1	WEBFUN value. This is explained in "VM WEB Function Codes" on page 215.	
2	Zero	
3	Module flow indicator	
	1...	DTIVPURG called
	..xx xxxx	Space reserved for module flow indicator
	1	SEND issued support command
	2	SEND issued unsupported command
	3	DTIWBLD issued
	4	DTISCHED issued

DTIVLOGX

1	WEBFUN value. This is explained in "VM WEB Function Codes" on page 215.	
2	SABTFLG1 field	
3	Module flow indicator	
	1...	Asynchronous CLSDST issued
	.1..	Synchronous CLSDST issued
	..xx xxxx	Space reserved for module flow indicator
	1	Start NIB build process
	2	INQUIRE TERMS issued
	3	Started building session parameters
	4	INQUIRE SESSPARM issued
	5	DTIVTRMI called
	6	Attempting OPNDST
	7	Invalid device or invalid bind format
	8	Unsupported auxiliary device or unsupported screen size

DTIVNSEX

1-3	Bytes 0-2 of input data field	

DTIVOPNX

1	WEBFUN value. This is explained in "VM WEB Function Codes" on page 215.	
2	Zero	
3	Module flow indicator	
	1...	Check failed
	..xx xxxx	Space reserved for module flow indicator
	1	DTIWBLD issued
	2	DTISKED issued

DTIVRECX

1 WEBFUN value. This is explained in 'VM WEB Function Codes" on page 215.

2 Return code from CHECK macro

3 Module flow indicator

11..	Zero
..xx xxxx	Space reserved for module flow indicator
1	Major (non-recoverable) error detected
2	Asynchronous device end received
3	Failure sending exception response (to exception request)
4	SEND failure trying to issue a normal response
5	DTIVRECM failure
6	DTINQ failure
7	DTIWBLD failure
8	DTIDQ failure
9	Chaining accumulator error
A	Error attempting to send chain format error response
B	Error attempting to send RU length error response
C	In purging chain state
D	First in chain received
E	Middle of chain received
F	End of chain received
10	VTAM chain indicator error
11	LUSTAT = No input mechanism
12	TWX read error
13	Null RU with CHANGE DIRECTION processed
14	Entry invalid — TAB invalid or CID mismatch
15	Entry ignored — logical unit undergoing CLSDST
16	DTIVLOFF called for logical unit
17	Null RU without CHANGE DIRECTION processed

DTIVRESX

1 WEBVSRPS

X'15'	WEBVSCC	Command check
X'14'	WEBVSDC	Data check
X'13'	WEBVSCRJ	Command reject
X'12'	WEBVSEC	Equipment check
X'11'	WEBVSDCU	Data check and unit specify
X'10'	WEBVSECU	Equipment check and unit specify
X'0F'	WEBVSCC1	Reflect attention, CC1 on SIO
X'0E'	WEBVSCNL	SEND canceled - CANCEL key
X'0D'	WEBVSPSL	Presentation space integrity lost
X'0C'	WEBVSVER	Unrecoverable internal error
X'08'	WEBVSAOK	Successful completion
X'07'	WEBTWXER	TWX read error sense X'0827'
X'06'	WEBVSNAV	Printer not available
X'05'	WEBVSIRQ	Intervention required

	X'04'	WEBVSBSY	Printer busy
	X'03'	WEBVSERR	Unrecoverable I/O error
	X'02'	WEBVSLFU	Unconditional logoff
	X'01'	WEBVSOPC	Operation check
2	VLBERCVY		
	1...	VLBERCLR	Clear requested
	.1..	VLBERBRK	Bracket error occurred
	..1.	VLBERSDT	Start Data Traffic required
	...1	VLBERSIG	Signal required
	.... 1...	VLBERCNL	Change received
	.... .1..	VLBERCDR	Change direction required
	.... ..1.	VLBERPSL	Presentation space integrity lost
	1	VLBERHRD	Hard error occurred
3	Module flow indicator		
	11..	Reserved	
	..xx xxxx	Module flow indicator	
	01	Purge logical unit	
	02	SEND requested	
	03	Error recovery pending	
	04	Error recovery in progress, request ignored	
	05	Error element not found	
	06	Unrecoverable error occurred	
	07	Dequeue failure	
	08	Logical unit being purged	
	14	Invalid entry — TAB invalid or CID mismatch	
	15	Entry ignored — logical unit undergoing CLSDST	

DTIVRLQX

1-3	Zero

DTIVSCIX

1	Request code
2	UNBIND type
3	Zero

DTIVSNDX

1	WEBVSRPS		
	X'15'	WEBVSCC	Command check
	X'14'	WEBVSDC	Data check
	X'13'	WEBVSCRJ	Command reject
	X'12'	WEBVSEC	Equipment check
	X'11'	WEBVSDCU	Data check and unit specify
	X'10'	WEBVSECU	Equipment check and unit specify
	X'0F'	WEBVSCC1	Reflect attention, CC1 on SIO
	X'0E'	WEBVSCNL	SEND canceled - CANCEL key
	X'0D'	WEBVSPSL	Presentation space integrity lost

	X'0C'	WEBVSVER	Unrecoverable internal error
	X'08'	WEBVSAOK	Successful completion
	X'07'	WEBTWXER	TWX read error sense X'0827'
	X'06'	WEBVSNAV	Printer not available
	X'05'	WEBVSIRQ	Intervention required
	X'04'	WEBVSBSY	Printer busy
	X'03'	WEBVSERR	Unrecoverable I/O error
	X'02'	WEBVSLFU	Unconditional logoff
	X'01'	WEBVSOPC	Operation check
2	VLBERCVY		
	1...	VLBERCLR	Clear requested
	.1..	VLBERBRK	Bracket error occurred
	..1.	VLBERSDT	Start Data Traffic required
	...1	VLBERSIG	Signal required
	.... 1...	VLBERCNL	Change received
	.... .1..	VLBERCDR	Change direction required
	.... ..1.	VLBERPSL	Presentation space integrity lost
	1	VLBERHRD	Hard error occurred
3	Module flow indicator		
	1...	DTIVPURG called	
	.1..	DTIVLOFF called	
	..1.	Entry ignored — logical unit undergoing CLSDST	
	...1	Entry invalid — TAB invalid or CID mismatch	

VM Format for VTAM Exit Data Trace Entries

These trace records are created when:

- Minimum data is available
- VTAM issues an invalid request to VSCS
- VTAM calls fail validity checks.

DTIVCLSX

00	V	C	L	S
04	RPL REQ	RPL address		
08	DTITAB address			
0C	RPL		CID	
10	Zero			
1C	FF	DTIWEB address or zero		

Displacement:

00–03	Characters 'VCLS'
04	RPL request code
05–07	RPL address
08–0B	Invalid TAB address
0C–0F	RPL CID
10–1B	Zero
1C	X'FF' to indicate data trace entry
1D–1F	WEB address or zero

DTIVDFAX

00	V	D	F	A
04	ACB address			
08	RPL CID			
0C	DTITAB address			
10	Zero			
14	RPL address			
18	Zero			
1C	FF	WEBFUN	Zero	Mod. flow

Displacement:

00–03	Characters 'VDFA'
04–07	Input ACB address
08–0B	Input CID
0C–0F	Input TAB address
10–13	Zero
14–17	Input RPL address

18–1B Zero
1C X'FF' to indicate data trace entry
1D WEBFUN value. This is explained in "VM WEB Function Codes" on page 215.
1E Zero
1F Module flow indicator

1... .... DTIVPURG called
..xx xxxx Module flow indicator
1 SEND issued supported command
2 SEND issued unsupported command
3 DTIWBLD issued
4 DTISCHED issued

DTIVLOGX

00	V	L	O	G
04	ACB address			
08	LU name address			
0C	Reserved			
10	Length of logon message			
14	LU name			
1C	FF	WEBFUN	SABTFLG1	Mod. flow

Displacement:

00–03 Characters 'VLOG'
04–07 Input ACB address
08–0B Address of logical unit name
0C–0F Zero
10–13 Length of logon message
14–1B Logical unit name
1C X'FF' to indicate data trace entry
1D WEBFUN value. This is explained in "VM WEB Function Codes" on page 215.
1E SABTFLG1 field
1F Module flow indicator

1... .... Asynchronous CLSDST issued
.1.. .... Synchronous CLSDST issued
..xx xxxx Module flow indicators
1 Start NIB build process
2 INQUIRE TERMS issued
3 Started building session parameters
4 INQUIRE SESSPARM issued
5 DTIVTRMI called
6 Attempting OPNDST
7 Invalid device or invalid BIND format
8 Unsupported auxiliary device or unsupported screen size

DTIVLOSX

00	V	L	O	S
04	ACB address			
08	RPL CID			
0C	NIB user field			
10	Reason code			
14	DTITAB address			
18	DTIWEB address			
1C	FF	Zero		Mod. flow

Displacement:

00–03 Characters 'VLOS'
04–07 Input ACB address
08–0B Input CID
0C–0F Input NIB user field
10–13 Reason code
  0C Contact lost
  14 Unconditional logoff
  20 Conditional logoff
  24 Buffer limit
14–17 TAB address or zeros
18–1B WEB address or zeros
1C X'FF' to indicate data trace entry
1D–1E Zero
1F Module flow indicator
  1... .... DTIVPURG called
  ..xx xxxx Module flow indicator
  1 DTIWBLD issued
  2 DTISKED issued

DTIVNSEX

00	V	N	S	E
04	ACB address			
08	RPL CID			
0C	DTITAB address			
10	Zero			
14	RPL address			
18	RPL REQ	RPL address		
1C	FF	Bytes 0 - 2 of user data field		

Displacement:

00–03 Characters 'VNSE'
04–07 Input ACB address
08–0B Input CID
0C–0F Input TAB address
10–13 Zero
14–17 Input RPL address
18 RPL request code
19–1B Input RPL address
1C X'FF' to indicate data trace entry
1D–1F First three bytes of input data field

DTIVOPNX

00	V	O	P	N
04	RPL REQ	RPL address		
08	DTITAB address			
0C	Zero			
1C	FF	WEBFUN	Zero	Mod. flow

Displacement:

00–03 Characters 'VOPN'
04 RPL request code
05–07 Input RPL address
08–0B Invalid TAB address
0C–1B Zero
1C X'FF' to indicate data trace entry
1D WEBFUN. This is explained in "VM WEB Function Codes" on page 215.
1E Zero
1F Module flow indicator

1...	Check failed
..xx xxxx	Module flow indicator
1	DTIWBLD issued
2	DTISKED issued

DTIVRECX

00	V	R	E	C
04	RPL REQ	RPL address		
08	DTITAB address			
0C	DTIVLB address			
10	DTIWEB address			
14	Zero			
1C	FF	WEBFUN	CHK RC	Mod. Flow

Displacement:

00 – 03	Characters 'VREC'		
04	RPL request code		
05 – 07	Input RPL address		
08 – 0B	Invalid TAB address		
0C – 0F	VLB address		
10 – 13	Input WEB address		
14 – 1B	Zero		
1C	X'FF' to indicate data trace entry		
1D	WEBFUN. This is explained in "VM WEB Function Codes" on page 215.		
1E	Return code from CHECK macro		
1F	Module flow indicator		
	11..	Zero	
	..xx xxxx	Module flow indicator	
		1	Major (non-recoverable) error detected
		2	Asynchronous device end received
		3	Failure sending exception response (to exception request)
		4	SEND failure trying to issue a normal response
		5	DTIVRECM failure
		6	DTINQ failure
		7	DTIWBLD failure
		8	DTIDQ failure
		9	Chaining accumulator error
		A	Error attempting to send chain format error response
		B	Error attempting to send RU length error response
		C	In purging chain state
		D	First in chain received
		E	Middle of chain received
		F	End of chain received
		10	VTAM chain indicator error
		11	LUSTAT = No input mechanism
		12	TWX read error
		13	Null RU with CHANGE DIRECTION processed
		14	Invalid entry — TAB invalid or CID mismatch
		15	Entry ignored — logical unit undergoing CLSDST
		16	DTIVLOFF called for logical unit
		17	Null RU without CHANGE DIRECTION processed

DTIVRESX

Note: This trace record is created only when exception response is used.

00	V	R	E	S
04	REQCODE	RPL ADDRESS		
08	DTITAB ADDRESS			
0C	DTIVLB ADDRESS			
10	DTIWEB ADDRESS			
14	RESERVED			
1C	X'FF'	RESERVED	RESERVED	MOD FLOW

Displacement:

00–03	Characters 'VRES'	
04	RPL request code	
05–07	RPL address	
08–0B	TAB address	
0C–0F	VLB address	
10–13	WEB address	
14–17	Zero	
1C	X'FF' to indicate data trace entry	
1D–1E	Zero	
1F	Module flow indicator	
	11..	Reserved
	..xx xxxx	Module flow indicator
	01	Purge logical unit
	02	SEND requested
	03	Error recovery pending
	04	Error recovery in progress, request ignored
	05	Error element not found
	06	Unrecoverable error occurred
	07	Dequeue failure
	08	Logical unit being purged
	14	Invalid entry — TAB invalid or CID mismatch
	15	Entry ignored — logical unit undergoing CLSDST

DTIVRLQX

00	V	R	L	Q
04	ACB name			
0C	LU name			
14	DTITAB address			
18	Zero			
1C	FF	Ret. code	ZERO	

Displacement:

00–03	Characters 'VRLQ'
04–0B	ACB name
0C–13	Logical unit name
14–17	TAB address
18–1B	Zero
1C	X'FF' to indicate data trace entry
1D	Return code
1E–1F	Zero

DTIVSCIX

00	V	S	C	I
04	ACB address			
08	CID (not used)			
0C	DTITAB address			
10	Zero			
14	RPL address			
18	RPLREQ	RPL address		
1C	FF	Req code	Unbind	Zero

Displacement:

00–03	Characters 'VSCI'
04–07	ACB address
08–0B	CID (not used)
0C–1F	DTITAB address
10–13	Zero
14–17	RPL address
18	RPL request
19–1B	RPL address
1C	X'FF' to indicate data trace entry
1D	Request code
1E	Unbind type
1F	Zero

DTIVSIMX

00	V	S	I	M
04	RPL REQ	RPL address		
08	RPLFDBWD			
0C	RPLFDBK2			
10	LU name			
18	DTITAB address			
1C	FF	Return code from CHECK macro		

Displacement:

00–03	Characters 'VSIM'
04	RPL request code
05–07	Input RPL address
08–0B	RPL feedback word
0C–0F	RPLFDBK2
10–17	Logical unit name
18–1B	DTITAB address
1C	X'FF' to indicate data trace entry
1D–1F	Return code from CHECK macro

DTIVSEND

Note: This trace record is created only when exception response is used.

00	V	S	E	N
04	DTITAB ADDRESS			
08	DTIWEB ADDRESS			
0C	XMTFLG1	XMTFLG2	MODE/TYPE	LBCSTATE
10	PLBFSSMF	PLBLFLG1	PLBNFLG1	PLBOFLG1
14	VLBLRCD	VLBERFLG	VLBERSQ1	
18	VLBSWAP			
1C	X'FF'	CALLER'S RETURN ADDRESS		

Displacement:

00–03	Characters 'VSEN'		
04–07	DTITAB address		
08–0B	DTIWEB address (DTICMT, DTIWEB, DTIXMT)		
0C	XMTFLG1 — Transmit flag 1		
	1...	XMTCD	Change direction required (for start/stop)
	.1..	XMTRSHOW	Redisplay in DTIXMT buffer
	..1.	XMTPMSG	Priority message on screen

	...1	XMTDIAG	CMS DIAGNOSE on screen
	.... 1...	XMTLRO	Logo RU
	.... .1..	XMTATTN	Attention write (VSCS only) yes = 1
	.... ..1.	XMTINPT	Input expected from logical unit
	1	XMTPWMSK	Password mask
0D	XMTFLG2 — Transmit flag 2		
	1...	XMTVDATR	DTIVDATR called
	.1..	XMTERCMT	DTIXMT is really a DTICMT
	..1.	XMTERERR	Error retry necessary
	...1	XMTERPSL	Presentation space integrity lost
	.... 1...	XMTLMPEO	LMPEO SEND initiated
	.... .111		Zero
0E	MODE or TYPE of request		
	B	Busy	
	C	Control command	
	D	Definite response	
	E	Exception response	
	P	Logical unit purged	
	X	Recovery request	
	W	Logical unit waiting	
0F	LBCSTATE from PLB (See "VM LBCSTATE Values for PLBINxxx" on page 214.)		
	1...	LBCINT	Internal mode if on (on=1). If LBCINT is on, see "VM LBCSTATE Values for PLBINxxx" on page 214. If LBCINT is off, the following values apply:
	.11.	LBCFSSMF	Full-screen support mode flags
	.1..	LBCFSAV	Screen available
	..1.	LBCFSII	Full-screen input inhibited: B'00' CP (Console) mode B'01' Virtual machine mode state between ATTN and READ, QUEUE CP messages. B'10' Virtual machine mode queue CP messages. B'11' Virtual machine mode available for switch.
	...1	LBCCOPY	Copy mode: on=1
	.... 1...	LBCNAL	NAL flag: on=1
	.... .11.	LBCHOLD	Holding flag: on=B'11'
	.... .1..	LBCM10	'More' 10 flag: on=1
	.... ..1.	LBCM50	'More' 50 flag: on=1
	1	LBCCMSD	Screen written with a CMS DIAGNOSIS: on=1
	....		No more holding state
10	PLBFSSMF — Full-screen support mode flags		
	1...	PLBFPA1	PA1 key received switch mode
	.1..	PLBFSRFC	Mode switch required (FSSM to Console)
	..1.	PLBFSAFC	Switch available (FSSM to Console)

	...1	PLBFCNMP	Console message pending
	.... 1...	PLBFVSIP	VTAM Services SEND in progress
	.... .1..	PLBFCLGF	Conditional logoff received
	.... ..1.	PLBFCMRR	CP message pending, response required
	1	PLBCMSEW	Screen has just been erased
11	PLBLFLG1 — VTAM Services logoff flags for Presentation Services		
	1...	PLBLVS	VTAM Services logoff received, bypass logo
	.1..	PLBLGFC	WEBINLGF sent to CCS conditional logoff
	..1.	PLBLGFU	WEBINLGF sent to CCS unconditional logoff
	...1	PLBLERR	WEBINERR sent to CCS for hard error
	.... 1...	PLBLLERR	WEBINLER sent to CCS for interface error
	.... .1..	PLBLVERR	WEBINVER sent to CCS for internal logic error
	.... ..1.	PLBLSEVR	CP CCS SEVER received, must sever
	1	PLBLNSTR	No storage indicator in logoff
12	PLBNFLG1 — NOM flag 1		
	1...	PLBCEND	Command end (read)
	.1..	PLBRDTX	The redisplay timer has expired
	..1.	PLBSAC	Status area change required
	...1	PLBCLROA	Clear output area
	.... 1...	PLBCLRIA	Clear input area
	.... .1..	PLBRING	Ring the alarm
	.... ..1.		Zero
	1	PLBUNLK	Unlock keyboard
13	PLBOFLG1 — Output manager flags		
	1...	PLBSDIAG	CMS DIAGNOSE on screen
	.1..	PLBPSMSG	Priority message on screen
	..1.	PLBOSRIP	SEND retry in progress
	...1	PLBOKLOK	Keyboard locked
	.... 1...	PLBOKPMI	Keyboard printer mask input area sent
	.... .1..	PLBOSWRT	CMS WRITE active (no input received)
	.... ..1.	PLBOCRRQ	Carriage return required before SEND on hardcopy terminal
	1	PLBOEXRQ	'!' required before SEND on hardcopy terminal
14	VLBLRCD — Logoff reason codes		
	1...	VLBVTLO	VTAM LOGOFF
	.1..	VLBVRYI	VARY INACTIVE
	..1.	VLBHLTQ	HALT QUICK
	...1	VLBHLTC	HALT CANCEL
	.... 1...	VLBIOE	Unrecoverable I/O error
	.... .1..	VLBINERR	Internal error
	11		Zero
15	VLBERFLG — Exception response flags		
	1...	VLBPSHLD	DTIXMT pending (PS held)

	.1..	VLBERPND	Error recovery pending
	..1.	VLBERRIP	Error recovery in progress
	...1	VLBERPRG	Permanent error: purge logical unit
	.... 1...	VLBPRGDN	Purge complete
	.... .1..	VLBERPRM	Permanent error
	.... ..1.	VLBEREXT	Exit receive control before return
	1	VLBRCVNO	Do not reissue RECEIVE SPECIFIC for logical unit
16–17	VLBERSQ1 — Sequence number of first error		
18	VLBFLG — VLB flag		
	1...	VLBBSPBB	Bracket-state pending begin bracket
	.1..	VLBBSPEB	Bracket-state pending end bracket
	..1.	VLBBSINB	Bracket-state in bracket
	...1	VLBSR	Application SEND or RECEIVE state: 1 = SEND state, 0 = RECEIVE state
	 1111		Zero
19	VLBFLG1 — VLB flag 1		
	1...	VLBIOTR	Inbound data received since last SEND
	.1..	VLBCLSD	CLSDST not required
	..1.	VLBPCS	Purging chain state
	...1	VLBSIGNL	Signal sent
	.... 1...	VLBCHGD	Change direction required
	.... .1..	VLBRETR	Retry RECEIVE request
	.... ..1.	VLBRETS	Retry SEND request
	1		Zero
1A	VLBFLG2 — VLB flag 2		
	1...	VLBRPLB	RPL in use
	.1..	VLBCLPND	VTAM CLSDST pending
	..11		Zero
	.... 1...	VLBBUSY	Printer busy
	.... .1..	VLBNAV	Printer not available
	.... ..1.	VLBOPCK	Operation check occurred
	1	VLBWAIT	Waiting for data or LUSTAT
1B	VLBSENDF — Special function flags		
	1...	VLBCLR	Clear required
	.1..	VLBSDT	Start data traffic (SDT) required
	..1.	VLBCANC	Cancel required
	...1	VLBSIGRQ	Signal required
	 1111		Zero
1C	X'FF' to indicate data trace entry		
1D–1F	Caller's return address		

DTIVSNDX

00	V	S	N	D
04	RPL REQ	RPL address		
08	DTITAB address			
0C	DTIVLB address			
10	DTIWEB address			
14	Zero			
18	Zero			
1C	FF	WEBVSRPS	Zero	Mod. flow

Displacement:

00–03	Characters 'VSND'		
04	RPL request code		
05–07	Input RPL address		
08–0B	Invalid TAB address		
0C–0F	VTAM Services local block (VLB) address		
10–13	Input WEB address		
14–1B	Zero		
1C	X'FF' to indicate data trace entry		
1D	WEBVSRPS		
	X'15'	WEBVSCC	Command check
	X'14'	WEBVSDC	Data check
	X'13'	WEBVSCRJ	Command reject
	X'12'	WEBVSEC	Equipment check
	X'11'	WEBVSDCU	Data check and unit specify
	X'10'	WEBVSECU	Equipment check and unit specify
	X'0F'	WEBVSCC1	Reflect attention, CC1 on SIO
	X'0E'	WEBVSCNL	SEND canceled - CANCEL key
	X'0D'	WEBVSPSL	Presentation space integrity lost
	X'0C'	WEBVSVER	Unrecoverable internal error
	X'08'	WEBVSAOK	Successful completion
	X'07'	WEBTWXER	TWX read error sense X'0827'
	X'06'	WEBVSNAV	Printer not available
	X'05'	WEBVSIRQ	Intervention required
	X'04'	WEBVSBSY	Printer busy
	X'03'	WEBVSERR	Unrecoverable I/O error
	X'02'	WEBVSLFU	Unconditional logoff
	X'01'	WEBVSOPC	Operation check
1E	Zero		

1F Module flow indicator

1...	DTIVPURG called
.1..	DTIVLOFF called
..1.	Entry ignored — logical unit undergoing CLSDST
...1	Entry invalid — TAB invalid or CID mismatch

VM Cleanup Error Trace Record Format

The VSCS cleanup trace is activated automatically if any of the other VSCS traces are active. The cleanup trace applies only to the set of logical units specified for the other VSCS traces. Normal session terminations and error situations are traced. Two kinds of cleanup trace records are produced:

- Logical unit-related
- Data-related (associated with VSCS global path).

VM Format for a Cleanup Error Trace Entry

<table>
<tr><td>00</td><td colspan="4">Module ID</td></tr>
<tr><td>04</td><td>WEBFUN</td><td colspan="3">DTITAB address</td></tr>
<tr><td>08</td><td>RPL REQ</td><td>LBCSTATE</td><td>PLBFSSMF</td><td>PLBLFLG1</td></tr>
<tr><td>0C</td><td>PLBFLG1</td><td colspan="2">PLBNOM</td><td>PLBOFLG1</td></tr>
<tr><td>10</td><td>LBCSTATE</td><td>VLBLRCD</td><td colspan="2">Zero</td></tr>
<tr><td>14</td><td colspan="4">VLBSWAP</td></tr>
<tr><td>18</td><td colspan="4">VLBCID</td></tr>
<tr><td>1C</td><td>TRCFLOW</td><td colspan="3">DTIWEB or RPL address</td></tr>
</table>

Displacement:

00–03 Four-character module ID (characters 'PBRK' or 'VPUR')

04 WEBFUN code or zero (See "VM WEB Function Codes" on page 215.)

05–07 TAB address for current request

08 RPL request code

09 LBCSTATE from PLB (See "VM LBCSTATE Values for PLBINxxx" on page 214.)

1...	LBCINT	Internal mode if on (on=1). If LBCINT is on, see "VM LBCSTATE Values for PLBINxxx" on page 214. If LBCINT is off, the following values apply:
.11.	LBCFSSMF	Full-screen support mode flags
.1..	LBCFSAV	Screen available

	..1.	LBCFSII	Full-screen input inhibited: B'00' CP (Console) MODE B'01' Virtual machine mode state between ATTN and READ, QUEUE CP messages B'10' Virtual machine mode queue CP messages B'11' Virtual machine mode available for switch.
	...1	LBCCOPY	Copy mode: on=1
	.... 1...	LBCNAL	NAL flag: on=1
	.... .11.	LBCHOLD	Holding flag: on=B'11'
	.... .1..	LBCM10	'More' 10 flag: on=1
	.... ..1.	LBCM50	'More' 50 flag: on=1
	1	LBCCMSD	Screen written with a CMS DIAGNOSE: on=1
	....		No more holding state
0A	PLBFSSMF	Full-screen support mode flags	
	1...	PLBFPA1	PA1 key received switch mode
	.1..	PLBFSRFC	Mode switch required (FSSM to Console)
	..1.	PLBFSAFC	Switch available (FSSM to Console)
	...1	PLBFCNMP	Console message pending
	.... 1...	PLBFVSIP	VTAM Services SEND in progress
	.... .1..	PLBFCLGF	Conditional logoff received
	.... ..1.	PLBFCMRR	CP message pending, response required
	1	PLBCMSEW	Screen just cleared
0B	PLBLFLG1		
	1...	PLBVS	VTAM Services logoff received bypass logo
	.1..	PLBLGFC	WEBINLGF sent to CCS conditional logoff
	..1.	PLBLGFU	WEBINLGF sent to CCS unconditional logoff
	...1	PLBLERR	WEBINERR sent to CCS for hard error
	.... 1...	PLBLLERR	WEBINLER sent to CCS for interface error
	.... .1..	PLBLVERR	WEBINVER sent to CCS for internal logic error
	.... ..1.	PLBLSEVR	CP CCS SEVER received, VSCS must sever
	1	PLBNSTR	No storage indicator in logoff processing
0C	PLBFLG1		
	1...	PLBLGO	Logo or no logo bit (logo = 1)
	.1..	PLBINAR	APL or TEXT starting in input area
	..1.	PLBPASPA	Pass PA1 as data (FSSM; VTAM V3R1 only)
	..1.	PLBLFLNP	Logon from logo not permitted
	...1	PLBCONCT	Current connect count incremented for logical unit
	.... 1...	PLBCNCTI	IUCV CONNECT has been issued

	.... .1..	PLBINREC	Input received during purge
	.... ..1.	PLBPRMPT	Usability prompt message flag. For display terminals: CP READ received but prompt not sent. For hardcopy terminals: VSCS sent prompt to logical unit.
	1		Zero (VTAM V3R1 only)
	1	PLBSDIAL	SNA-dialed logical unit
0D–0E	PLBNOM	Notify output manager	
0D	PLBNFLG1	NOM flag 1	
	1...	PLBCEND	Command end (read)
	.1..	PLBRDTX	The redisplay timer has expired
	..1.	PLBSAC	Status area change required
	...1	PLBCLROA	Clear output area
	.... 1...	PLBCLRIA	Clear input area
	.... .1..	PLBRING	Ring the alarm
	.... ..1.		Zero
	1	PLBUNLK	Unlock keyboard
0E	PLBNFLG2	NOM flag 2	
	1...	PLBRDPND	Redisplay pending
	.111 1111		Zero
0F	PLBOFLG1	Output manager flags	
	1...	PLBSDIAG	CMS DIAGNOSE on screen
	.1..	PLBPSMSG	Priority message on screen
	..1.	PLBOSRIP	SEND retry in progress
	...1	PLBOKLOK	Keyboard locked
	.... 1...	PLBOKPMI	Hardcopy terminal mask input area sent
	.... .1..	PLBOSWRT	CMS WRITE active (no input received)
	.... ..1.	PLBOCRRQ	Carriage return required before SEND (on hardcopy terminal)
	1	PLBOEXRQ	'!' required before SEND (on hardcopy terminal)
10	LBCSTATE	From VLB	
	1	LBCVSST	VTAM Services state: 0 = Normal, 1 = Termination
11	VLBLRCD	Logoff reason codes	
	1...	VLBVTLO	VTAM LOGOFF
	.1..	VLBVRYI	VARY INACTIVE
	..1.	VLBHLTQ	HALT QUICK
	...1	VLBHLTC	HALT CANCEL
	.... 1...	VLBIOE	Unrecoverable I/O error
	.... .1..	VLBINERR	Internal error
	11		Zero
12–13	Zero		
14	VLBFLG	VLB flags	
	1...	VLBBSPBB	Bracket state - pending begin bracket

	.1..	VLBBSPEB	Bracket state - pending end bracket
	..1.	VLBBSINB	Bracket state - in bracket
	...1	VLBSR	Application SEND or RECEIVE state: 1 = SEND state, 0 = RECEIVE state
	 1111		Zero
15	VLBFLG1	VLB flag 1	
	1...	VLBIOTR	Inbound data received since last SEND indicator
	.1..	VLBCLSD	CLSDST: 0 = required, 1 = not required
	..1.	VLBPCS	Purging chain state
	...1	VLBSIGNL	Signal sent
	.... 1...	VLBCHGD	Change direction required
	.... .1..	VLBRETR	Retry RECEIVE request
	.... ..1.	VLBRETS	Retry SEND request
	1		Zero
16	VLBFLG2	VLB flag 2	
	1...	VLBRPLB	RPL in use
	.1..	VLBCLPND	VTAM CLSDST pending
	..11		Zero
	.... 1...	VLBBUSY	Printer busy
	.... .1..	VLBNAV	Printer not available
	.... ..1.	VLBOPCK	Operation check occurred
	1	VLBWAIT	Waiting for data or LUSTAT
17	VLBSENDF	Special send indicator	
	1...	VLBCLR	CLEAR required
	.1..	VLBSDT	SDT required
	..1.	VLBCANC	CANCEL required
	...1	VLBSIGRQ	SIGNAL required
	 1111		Zero
18 – 1B	VLBCID	Network address pair (CID)	
19	TRCFLOW	Cleanup-module-dependent data	
1D – 1F	Address	DTIWEB if WEBFUN not zero, RPL if RPLREQ not zero	

Cleanup-Module-Dependent Data

DTIPRBRK (ID='PBRK'): Trace flow flags are set as follows:

01	TRXPURG	Logical unit in PLBINPRG (88) state
02	TRXNWEB	No input WEB
03	TRXLOGO	Logo error
04	TRXCOND	Conditional logoff

DTIVPURG (ID='VPUR'): Trace flow flags are set as follows:

1...	Schedule failed	
.1..	Storage shortage	
..xx xxxx	Space reserved for module flow indicator	
01	TRXNTERM	NOTERM entered
02	TRXLTERM	Logical unit in termination
03	TRXWORK	VSCS ignoring work to process request
04	TRXNWORK	No work to process DTIVLOFF called
05	TRXCONN	Logical unit in termination state with CCS connection
06	TRXACTIV	Logical unit state is not termination with no connection
07	TRXIGNOR	CCS has been notified or DTIPRBRK has set a timer to force purge. VSCS is waiting.

VM VSCS FORCE Trace Entry (V3R2)

00	P	F	O	R
04	DTITAB ADDRESS			
08	LBCSTATE	PLBLFLG1	PLBNFLG1	PLBOFLG1
0C	TABDTYPE	TABDSUB	LBCSTATE	VLBLRCD
10	VLBSWAP			
14	LU NAME			
1C	'FF'	MOD FLOW 1	MOD FLOW 2	MOD IND

Displacement:

00–03	Four-character module ID		
04–07	Tab address for current request		
08	LBCSTATE from PLB (See "VM LBCSTATE Values for PLBINxxx" on page 214.)		
	1...	LBCINT	Internal mode if on (on=1). If LBCINT is on, see "VM LBCSTATE Values for PLBINxxx" on page 214. If LBCINT is off, the following values apply:
	.11.	LBCFSSMF	Full-screen support mode flags
	.1..	LBCFSAV	Screen available
	..1.	LBCFSII	Full-screen input inhibited: B'00' CP (Console) mode B'01' Virtual machine mode state between ATTN and READ, QUEUE CP messages. B'10' Virtual machine mode queue CP messages. B'11' Virtual machine mode available for switch.
	...1	LBCCOPY	Copy mode: on=1
	.... 1...	LBCNAL	NAL flag: on=1
	.... .11.	LBCHOLD	Holding flag: on=B'11'
	.... .1..	LBCM10	'More' 10 flag: on=1
	.... ..i.	LBCM50	'More' 50 flag: on=1
	1	LBCCMSD	Screen written with a CMS DIAGNOSIS: on=1
	....		No more holding state
09	PLBLFLG1		
	1...	PLBVS	VTAM Services logoff received bypass logo
	.1..	PLBLGFC	WEBINLGF sent to CCS conditional logoff
	..1.	PLBLGFU	WEBINLGF sent to CCS unconditional logoff
	...1	PLBLERR	WEBINERR sent to CCS for hard error

	.... 1...	PLBLLERR	WEBINLER sent to CCS for interface error
	.... .1..	PLBLVERR	WEBINVER sent to CCS for internal logic error
	.... ..1.	PLBLSEVR	CP CCS SEVER received, VSCS must sever
	1	PLBNSTR	No storage indicator in logoff processing
0A – 0B	PLBNOM	Notify output manager	
0A	PLBNFLG1	NOM flag 1	
	1...	PLBCEND	Command end (read)
	.1..	PLBRDTX	The redisplay timer has expired
	..1.	PLBSAC	Status area change required
	...1	PLBCLROA	Clear output area
	.... 1...	PLBCLRIA	Clear input area
	.... .1..	PLBRING	Ring the alarm
	.... ..1.		Zero
	1	PLBUNLK	Unlock keyboard
0B	PLBOFLG1	Output manager flags	
	1...	PLBSDIAG	CMS DIAGNOSE on screen
	.1..	PLBPSMSG	Priority message on screen
	..1.	PLBOSRIP	SEND retry in progress
	...1	PLBOKLOK	Keyboard locked
	.... 1...	PLBOKPMI	Hardcopy terminal mask input area sent
	.... .1..	PLBOSWRT	CMS WRITE active (no input received)
	.... ..1.	PLBOCRRQ	Carriage return required before SEND (on hardcopy terminal)
	1	PLBOEXRQ	'!' required before SEND (on hardcopy terminal)
0C	TABDTYPE		
	01	DTYPEDSP for a display	
	02	DTYPEKP for a keyboard/printer	
	02	DTYPEPRT for a printer	
0C	TABDSUB		
	01	VDEV3767 for a 3767	
	02	VDEV2741 for a 2741	
	02	VDEVTWX for a TXW/TTY/3101	
0E	LBCSTATE	VLB	
	1	LBCVSST	VTAM Services state: 0 = Normal, 1 = Termination
0F	VLBLRCD	Logoff reason codes	
	1...	VLBVTLO	VTAM LOGOFF
	.1..	VLBVRYI	VARY INACTIVE
	..1.	VLBHLTQ	HALT QUICK
	...1	VLBHLTC	HALT CANCEL
	.... 1...	VLBIOE	Unrecoverable I/O error
	.... .1..	VLBINERR	Internal error
	11		Zero
10	VLBFLG	VLB flags	

	1...	VLBBSPBB	Bracket state-pending begin bracket
	.1..	VLBBSPEB	Bracket state-pending end bracket
	..1.	VLBBSINB	Bracket state-in bracket
	...1	VLBSR	Application SEND or RECEIVE state: 1 = SEND state, 0 = RECEIVE state
	 1111		Zero
11	VLBFLG1	VLB flag 1	
	1...	VLBIOTR	Inbound data received since last SEND indicator
	.1..	VLBCLSD	CLSDST: 0 = required, 1 = not required
	..1.	VLBPCS	Purging chain state
	...1	VLBSIGNL	Signal sent
	.... 1...	VLBCHGD	Change direction required
	.... .1..	VLBRETR	Retry RECEIVE request
	.... ..1.	VLBRETS	Retry SEND request
	1		Zero
12	VLBFLG2	VLB flag 2	
	1...	VLBRPLB	RPL in use
	.1..	VLBCLPND	VTAM CLSDST pending
	..11		Zero
	.... 1...	VLBBUSY	Printer busy
	.... .1..	VLBNAV	Printer not available
	.... ..1.	VLBOPCK	Operation check occurred
	1	VLBWAIT	Waiting for data or LUSTAT
1B	VLBSENDF	Special function flags	
	1...	VLBCLR	Clear required
	.1..	VLBSDT	SDT required
	..1.	VLBCANC	Cancel required
	...1	VLBSIGRQ	Signal required
	 1111		Zero
14–1B	TABLUNM	Logical unit name being forced	
1C		X'FF' to indicate data trace entry	
1D		Module flow indicator 1 done by SPECIALX	
	1...	SPECIALX called	
	.1..	CONECTYS called	
	..1.	CONECTNO called	
	...1	COPYSIT called	
	.... 1...	DTIPRCLP called	
	.... .1..	DTIPRBRK called	
	.... ..1.	DTIPRSVR called	
	1	DTIPRLGF called	
1E		Module flow indicator 2 done by SPECIALX	
	1...	SPECIALX called	
	.1..	CONECTYS called	
	..1.	CONECTNO called	
	...1	COPYSIT called	

	.... 1...	DTIPRCLP called	
	.... .1..	DTIPRBRK called	
	.... ..1.	DTIPRSVR called	
	1	DTIPRLGF called	
1F		Queue status indicators	
	1...	LUNORMAL	Logical unit control blocks on normal queue
	.1..	LUREC	Logical unit control blocks on recovery queue
	..1.	LUPQUEUE	PS PIB on PS dispatcher queue
	...1	LUVQUEUE	VS PIB on VS dispatcher queue
	.... 1...	LUBFKPIB	WEBCPBRK on PS PIB work queue
	.... .1..	LUBRKLBC	WEBCPBRK on LBC pending queue
	.... ..1.	LUVSBRK	WEBVSBRK found on one of the work queues
	1	LUNTF60	WEBNTF60 found on PS PIB

VM VSCS Storage Trace Entries

VM FREB Trace Entry (V3R2)

00	F R E B	
04	DTIPDB ADDRESS	
08	BLOCK ADDRESS	
0C	BLOCK ID	HEADER ID
10	POOL NUMBER	
14	RETURN1 ADDRESS	
18	RETURN2 ADDRESS	
1C	RESERVED	

Displacement:

00–03	Characters 'FREB'
04–07	DTIPDB address
08–0B	Block being freed
0C–0D	SMP header
0E–0F	Block ID
10	Storage pool number 1 Presentation Services 2 VTAM Services 3 VTAM Exit Services
11–13	Zero
14–17	Caller's return address
18–1B	Caller's caller's return address
1C–1F	Zero

GETB Trace Entry (V3R2)

00	GETB	
04	DTIPDB ADDRESS	
08	BLOCK ADDRESS	
0C	REQ SIZE	CLR SIZE
10	POOL NUMBER	
14	RETURN1 ADDRESS	
18	RETURN2 ADDRESS	
1C	CALLERS STATE	

Displacement:

00 – 03	Characters 'GETB'
04 – 07	DTIPDB address
08 – 0B	Block being obtained
0C – 0D	Size requested
0E – 0F	Size to be cleared
10	Storage pool number
	1 Presentation Services
	2 VTAM Services
	3 VTAM Exit Services
	4 Special request
11 – 13	Zero
14 – 17	Caller's return address
18 – 1B	Caller's caller's return address
1C – 1F	State
	ENAB Caller running enabled
	DISB Caller running disabled

VSCS Data Trace Entries

Input Trace Entry (V3R2)

00	'I'	DTITAB ADDRESS	
04	DTIWEB ADDRESS		
08	WEBFUN	WEBMOD	WEBCOUNT
10	WEBDATA		

Displacement:

00	Character 'I' for Input
01 – 03	DTITAB address
04 – 07	Work element address
08	WEB function code

09	WEB mode
	00 Console
	01 CMS
	02 Full-screen
	04 Internal
0A – 0B	Length of actual data in WEBDATA
10-xx	WEBDATA (20 bytes for internal, or 223 bytes for external)

VM Output Trace Entry (V3R2)

00	'O'	DTITAB ADDRESS		
04	DTIWEB ADDRESS			
08	WEBFUN	WEBMOD	WEBCOUNT	
10	WEBDATA			

Displacement:

00	Character 'O' for Output
01 – 03	DTITAB address
04 – 07	Work element address
08	WEB function code
09	WEB mode
	00 Console
	01 CMS
	02 Full-screen
	04 Internal
0A – 0B	Length of actual data in WEBDATA
10-xx	WEBDATA (20 bytes for internal, or 223 bytes for external)

VM LBCSTATE Values for PLBINxxx

VM Console Mode

Name	Hex Value	Description
PLBINNMH	00	No 'More' or 'Holding' condition
PLBINM50	02 – 03	More condition, 50 second timer set
PLBINM10	04 – 05	More condition, 10 second timer set
PLBINHLD	06 – 07	Holding
PLBINNAL	08 – 0F	Not Accepted logic applies
PLBINCPY	10 – 1F	PF key Copy in progress

VM CMS Edit Mode

Name	Hex Value	Description
PLBINCMS	01	No 'More' or 'Holding' condition
PLBINM50	02 – 03	More condition, 50 second timer set
PLBINM10	04 – 05	More condition, 10 second timer set
PLBINHLD	06 – 07	Holding
PLBINNAL	08 - 0F	Not Accepted logic applies

VM Full-Screen Support Mode (FSSM)

Name	Hex Value	Description
PLBINFSS	20, 40, 60	Full-screen support mode

VM Internal Modes

Name	Hex Value	Description
PLBINLIP	80	Logon in progress
PLBINLUP	81	Logo up
PLBINCIP	82	Connect in progress
PLBINIIP	83	Inlog in progress
PLBINAWL	84	Attention wait before logo
PLBINLGF	85	Logoff
PLBINCLF	86	Conditional logoff
PLBINSVR	87	Sever required
PLBINPRG	88	Purge in progress

VM WEB Function Codes

The tables in this section show the bits defined in field WEBFUN, the function code for VSCS. The function code depends on the value in WEBMODE:

- If WEBMODE is 04, use "VM Function Codes for Internal States" on page 216.

  Internal function codes represent work functions within VSCS. These are not considered to be inbound or outbound.

- For all other WEBMODES:

  – For inbound, use "VM Function Codes for Inbound Processing in All Modes" on page 216.

    Inbound function codes represent work elements that are received by CCS.

  – For outbound, use the WEBMODE:

    — If the WEBMODE is 00, use "VM Function Codes for Outbound Processing in Console Mode" on page 217.

    — If the WEBMODE is 01, use "VM Function Codes for Outbound Processing in CMS Mode" on page 217.

    — If the WEBMODE is 02, use "VM Function Codes for Outbound Processing in Full-Screen Mode" on page 218.

    Outbound function codes represent work functions sent from CCS to VSCS, or generated internally in VSCS. The mode of the request (CMS, Console, or FSSM) determines which function is performed.

Note: In this book, whenever a flow is referred to as *inbound* or *outbound*, you should take the Console Communications Services (CCS) as the reference point. That is, flows *toward* the CCS are *inbound*; flows *away from* the CCS are *outbound*.

VM Function Codes for Internal States

Name	Hex Value	Description
WEBNTM10	00	Timer expired - 'More' 10 second, source = System Services
WEBNVLFC	01	Logoff conditional, source = VTAM Services
WEBNVLFU	02	Logoff unconditional, source = VTAM Services
WEBINERR	03	Unrecoverable I/O error, logoff condition
WEBINPUT	04	Attention identifier and optional data input
WEBNVFLW	05	Logoff warning, source = VTAM Services
WEBNSRSP	06	Input/output response (could be either normal or error), source = VTAM Services
WEBNVLGO	07	Logon response, source = VTAM Services
WEBNACON	08	Accept connect, source = CCS
WEBNSEVR	09	Sever connection, source = CCS
WEBNMSGC	0A	IUCV one-way message complete, source = CCS
WEBNTM50	0B	Timer expired - 'More' 50 second, source = System Services
WEBNTNAL	0D	Timer expired - not accepted 3 second
WEBNTF60	0E	Timer expired = FSSM 60 second, source = System Services
WEBVSBRK	0F	Logoff request, source = Presentation Services
WEBNVASY	10	Asynchronous error - WEBVSRPS indicates type
WEBPSSRU	10	Send request, source = Presentation Services
WEBNVCLU	11	Internal VTAM request cleanup, source = VTAM Services
WEBNSIML	12	Request to VTAM Services to issue SIMLOGON
WEBNTCPY	13	PF key Copy wait for printer SIMLOGON timer expired
WEBNTPRT	14	Timer expired for request to release printer after RELREQ exit driven
WEBINOFC	14	Operator issued VSCS FORCE command — processing continues in DTIPFORC

VM Function Codes for Inbound Processing in All Modes

Name	Hex Value	Description
WEBINLOG	01	Logon request
WEBINATT	02	Attention interrupt without data
WEBINCB1	05	Move cursor back one
WEBINLGF	06	VTAM-generated logoff condition, CCS logs user off
WEBINFNU	07	FSSM request rejected, screen not in user mode
WEBINAOK	08	Normal response
WEBINACT	09	Accounting data for CCS
WEBINOPC	0A	Operation check for FSSM request
WEBINLER	0B	Logical interface error (DTIWEB)
WEBINVER	0C	VSCS internal processing error
WEBRDERR	0D	Error in input data found by the NCP
WEBINVTM	0E	Identify VTAM user ID to CCS

VM Function Codes for Outbound Processing in Console Mode

Name	Hex Value	Description
WEBCPWRT	00	Write to next available line in output area
WEBCPRED	01	Read request
WEBCPBRK	02	Break connection (CP logoff or disconnect)
WEBCPWIC	03	Write to input area and then position cursor as indicated in WEBCURSR
WEBCPCPY	04	Copy request
WEBCPIED	05	Invalid environment definition
WEBCPFKR	06	PFK reply - treat as terminal input
WEBCPIDA	07	Data accepted, proceed with input processing
WEBCPINA	08	Input not accepted
WEBCPREX	09	IREPLY expected by way of synchronous
WEBCPTAB	0A	PF key tab operation
WEBCPLER	0B	Logical interface error (WEB) (ISEND or reply)
WEBCPLOG	0C	Logo
WEBCPTMR	0D	Command end (no output for command entered)
WEBCPPTH	0E	IUCV path ID
WEBCPCAH	0F	Color and highlight map for advantage II
WEBCPLSA	10	Device information (terminal command)

VM Function Codes for Outbound Processing in CMS Mode

Name	Hex Value	Description
WEBCMWRT	00	Write to line number specified in WEBLINE
WEBCMEWT	01	Erase output area and write to line number specified in WEBLINE
WEBCMCLR	02	Erase entire screen and refresh the format of the output, input, and status areas

VM Function Codes for Outbound Processing in Full-Screen Mode

Name	Hex Value	Description
WEBFSWRT	00	FSSM write
WEBFSEWT	01	FSSM erase/write
WEBFSEWA	02	FSSM erase/write alternate
WEBFSRDM	03	FSSM read modified
WEBFSRDB	04	FSSM read buffer
WEBFSWSF	05	FSSM write structured field
WEBFSEAU	06	FSSM erase all unprotected
WEBFSRBP	07	FSSM read buffer by position
WEBFSRMP	08	FSSM read modified by position
WEBINCC1	0F	FSSM reflect attention, CC1 on SIO
WEBINECU	10	FSSM equipment check and unit specify
WEBINDCU	11	FSSM data check and unit specify
WEBINEC	12	FSSM equipment check
WEBINCRJ	13	FSSM command reject
WEBINDC	14	FSSM data check
WEBINCC	15	FSSM control check

Chapter 7. Using VTAM Service Aids

Chapter 7. Using VTAM Service Aids

Many different service aids are available to help you collect information about SNA network problems. This chapter describes when to use some of these aids and provides examples to help you interpret their output.

Note: In addition to the service aids discussed in this chapter, you can use:

- Chapter 6, "VM Using VSCS Service Aids"
- Chapter 8, "Using the VTAM Internal Trace."

This chapter discusses a number of service aids. Figure 35 on page 222 shows where in the network you can use these aids.

Traces Provided by VTAM

VTAM provides several kinds of traces to record the flow of network events. Each trace occurs at a different point in the network (see Figure 35 on page 222). This difference allows you to narrow down the problem by following an RU through the network and determining where in the network the RU is incorrect. (The RU could be out of sequence or lost, the data in the RU could have been changed, and so forth.)

The VTAM traces and their results are:

- The buffer contents trace shows the contents of inbound and outbound message buffers.
- The I/O trace shows (in order) all I/O sent between VTAM and a particular network node.
- The SMS (buffer use) trace shows information about the use of buffers, including how often a buffer pool has expanded, how many buffers are currently being used, and what was the maximum number of buffers used since the last trace record was written.
- MVS The TGET/TPUT trace shows each message as it passes between a TSO command processor and TSO/VTAM.

Notes:

1. The VTAM internal trace (VIT) is discussed in Chapter 8, "Using the VTAM Internal Trace."
2. The VSCS trace facility (VM only) is discussed in Chapter 6, "VM Using VSCS Service Aids."

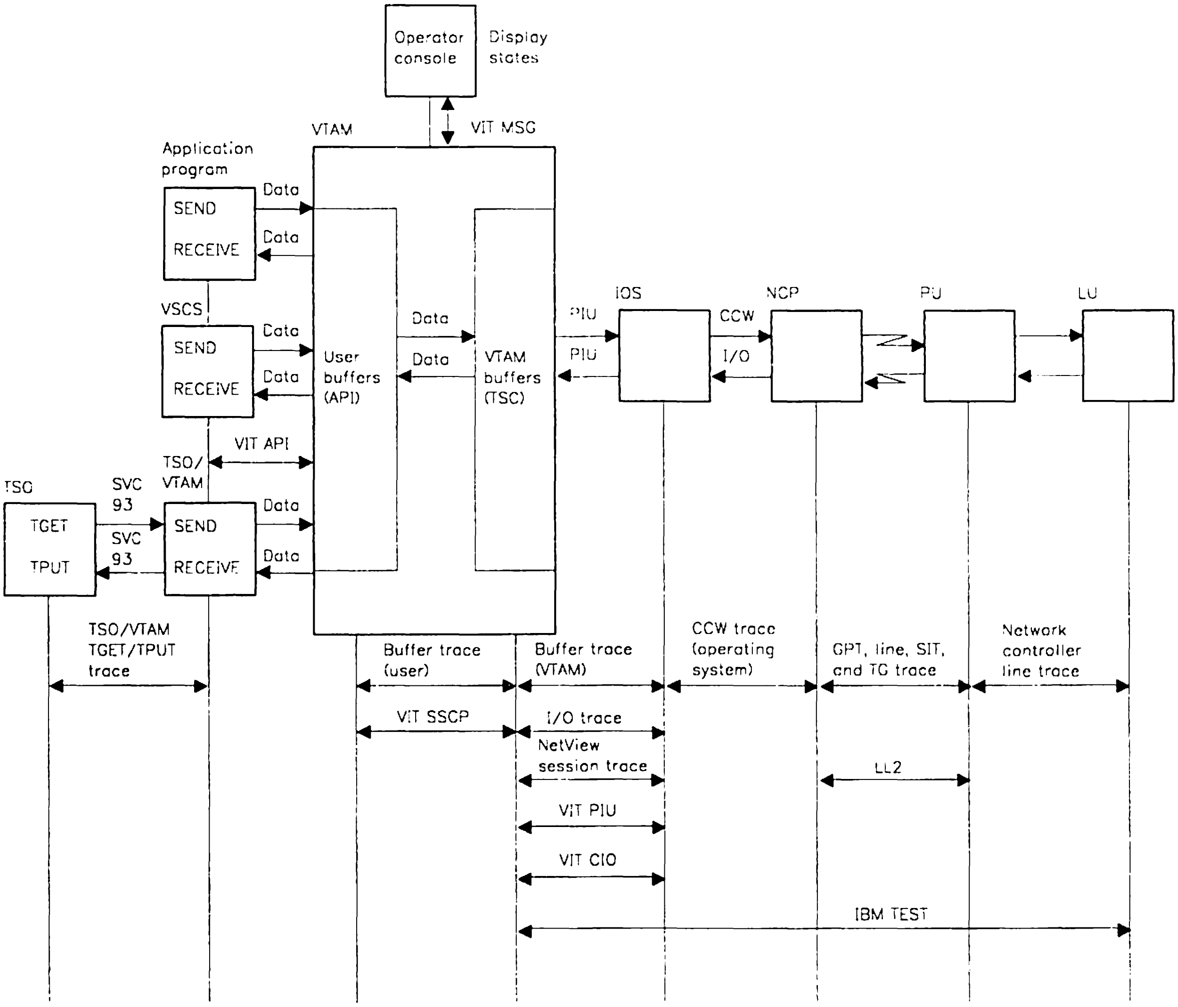


Figure 35. VTAM Traces

Activating VTAM Traces

You can activate VTAM traces when VTAM is started, or you can activate them only when they are needed. Use the TRACE option on the START command to begin the trace when VTAM is started. Use the MODIFY TRACE command to begin the trace when VTAM is already running. The TRACE start option and the MODIFY TRACE command have additional operands that are unique to the type of trace being started. For more information about using the TRACE start option, see *VTAM Installation and Resource Definition*. For more information about using the MODIFY TRACE command, see *VTAM Operation*.

MVS VTAM passes all trace data to the Generalized Trace Facility (GTF), which records all system trace data. Because of this, GTF *must* be active to use VTAM traces. Specify TRACE=RNIO for the VTAM I/O trace. Specify TRACE=USR for all other VTAM traces.

VM VTAM and VSCS pass all trace data to CPTRAP, using the GTRACE macroinstruction. CPTRAP *must* be active with format type 3D active. In addition, an ETRACE command (with the GTRACE and GROUP operands) must be entered for the VTAM and VSCS virtual machine. For details, see "VM Using CPTRAP and TRAPRED" on page 225.

VSE VTAM passes all trace data to the VTAM trace file.

Printing Trace Records

Once the trace data is recorded, you can use one of the following facilities to print the traces:

- Trace analysis program (TAP)
- MVS PRDMP
- VM TRAPRED
- VSE TPRINT.

The following table indicates which traces can be formatted and printed by each of the formatting programs. In this table, 'FP' indicates that the trace is both formatted and printed; 'UP' indicates that the trace is printed but not formatted. A blank entry indicates that the trace is neither formatted nor printed.

Trace	TAP	PRDMP	TRAPRED	TPRINT
Buffer contents trace	FP	FP	FP	FP
Generalized PIU trace	FP			
I/O trace	FP	FP	FP	FP
Line trace	FP	FP	FP	FP
Network controller line trace	FP			
Scanner interface trace	FP			
SMS (buffer use) trace	UP	FP	FP	FP
TGET/TPUT trace		UP		
Transmission group trace	FP	FP	FP	FP
VSCS internal trace			FP	
VTAM internal trace	UP	FP	FP	FP

Using TAP

You can format and print the following VTAM traces (and *only* these traces) with the trace analysis program (TAP) by setting the indicated option on the INPUT operand:

Specify:	To format and print:
LINE	Line trace
LINE	Scanner interface trace (3720, 3725, and 3745 only)
LINE	Transmission group trace
GPT	Generalized PIU trace
RNIO	I/O trace
BUFFER	Buffer contents trace
NETCTLR	Network controller line trace

TAP will print the SMS (buffer use) trace, but not format it. TAP also prints the VTAM and VSCS internal traces, unformatted. TAP can print all internal trace entries, or you can select up to ten entry types. For more information on TAP, see the *NCP, SSP, and EP Diagnosis Guide.*

MVS Using PRDMP

For the VTAM I/O trace, set RNIO on the EDIT option. For all other VTAM traces, set USR=(*symnum1[,symnum2]...[,symnum6]* | *ALL*) on the EDIT option.

symnum can be either a symbolic name or a number representing the trace that you want formatted and printed. If you specify USR=ALL, PRDMP formats and prints all user and subsystem traces recorded by GTF (except for TGET/TPUT trace, which is printed unformatted). Following are the valid symbols and numbers for the VTAM traces:

Symbol	Number	Trace
INT1	FE1	VTAM internal trace
TPIO	FEF	VTAM buffer contents trace — VTAM (traced in TSC)
CL01	FF1	VTAM buffer contents trace — user (traced in API)
CL02	FF0	SMS (buffer use) trace
LINE	FF2	NCP 3705 line or TG trace
APTH	FE2	TSO/VTAM TGET/TPUT trace

Note: The symbol and the number can be used interchangeably; for example, USR=(LINE) and USR=(FF2) are the same.

See the *Service Aids* manuals for your operating system for more information on PRDMP and GTF.

MVS Fields in VTAM Trace Output

In addition to the fields produced by VTAM, the following MVS-only trace fields appear in VTAM traces:

Field Header	Meaning
ASCB *nnnnnnnn*	The address of the ASCB for the address space that created the record. This field appears in a buffer contents trace, I/O trace, line trace, SMS trace, or transmission group trace.
CPU *nnnn*	The ID of the host processor in which the trace was run (applies only in a multiprocessor configuration). This field appears in an I/O trace.
JOBN *cccccccc*	The name of the job associated with the I/O operation (for an I/O trace) or the address space (for a buffer contents, SMS, line, or transmission group trace).
RNIO	The trace record was created by GTF (with TRACE=RNIO).
USR*xx* *nnn*	The formatting routine (AMDUSR*xx*). VTAM's trace formatter is AMDUSRFD. *nnn* is the event ID that represents this trace. This field appears in a buffer contents trace, line trace, SMS trace, or transmission group trace.

VM Using CPTRAP and TRAPRED

To collect VTAM trace output in VM, issue the following commands from the VTAM console.

Note: To collect VSCS trace output, see "VM How to Run External VSCS Traces" on page 161.

In CP

```
CPTRAP 3D

CPTRAP GROUPID gcsgroup

CPTRAP START TO userid
```

userid is the receiver of the spool file

In GCS

```
ETRACE GTRACE GROUP
```

In VTAM

```
VTAM F TRACE,TYPE=type,ID=term_name
```

VTAM will now start tracing to a spool file. When you have collected enough data and you want to stop tracing, issue the following commands:

In CP

```
CPTRAP STOP
```

In GCS

```
ETRACE GTRACE OFF
```

In VTAM

```
VTAM F NOTRACE,TYPE=type,ID=term_name
```

At this point you will get a CP spool file in the reader of the user ID specified in the CPTRAP START command. There are two ways to format the data, using TRAPRED and using TAP. TAP is available only if you have installed SSP. To use TRAPRED, issue the following commands:

`ACCESS 193 x` The GCS formatting routines are on this disk. "x" is a user-specified file mode.

`TRAPRED fileno` fileno is the spool reader file number. (Make sure the reader file is not in hold status.)

`3D` Read VTAM records only.

`FORMAT` Format the records.

`PRINTER 999999` Print the records. (There are other commands that allow you to view the CPTRAP file online. See the *VM/SP System Programmer's Guide* for VM R4, or the *VM/SP Facility for System Programming* for VM R5.)

`QUIT`

For more information on CPTRAP, see the *VM/SP Operator's Guide*. For more information on TRAPRED, see the *VM/SP System Programmer's Guide*. (for VM R4) or the *VM/SP Facility for System Programming* (for VM R5).

VM VM-Only Fields in VTAM Trace Output

In addition to the fields produced by VTAM, the following VM-only trace fields appear in VTAM traces:

Field Header	Meaning
Time-of-day clock	Time when trace record was created.
Length of GTF header and trace data	Indicates the length of this trace record.
Format routine ID	Identifies formatting module used for this record.

Field Header	Meaning
Event identification	Information from GTRACE macroinstruction's event ID parameter.

VSE Using TPRINT

The trace records generated by VTAM traces are directed to a VTAM trace file. Trace records in the trace file can then be printed by using the TPRINT utility program.

The trace print utility runs as a subtask under VTAM or as a job step under VSE. To run the trace print utility as a subtask, the operator initiates printing with the MODIFY SUBTASK command while VTAM is active. When the utility is run as a separate job step, printing can be done whether VTAM is active or inactive.

Job Step Procedure: The trace print utility can be executed through a job step procedure that is defined when VTAM is installed. SYSLST is assigned to a tape, printer, or disk, and the trace file is defined in DLBL and EXTENT statements, for example:

```
// DLBL     TRFILE,'VTRACE',1,SD
// EXTENT   SYS004
// ASSGN    SYS004,132
// ASSGN    SYSLST,02E
   .
   .
   .
// EXEC     TPRINT
```

SYSLST must be assigned to a printer, tape, or disk with the name IJSYSLS. If it is not assigned, TPRINT is canceled.

MODIFY SUBTASK Command: To start the trace print facility with the MODIFY SUBTASK command, use the ID=TPRINT option. The operator is prompted to request a print of the TRFILE, using any of the print options available or a snapshot print. A snapshot prints all the I/O, BUF, LINE, and TNSTAT records in VTAM's main storage I/O buffers, without accessing records previously written to TRFILE. (A snapshot print option is not available for the VTAM internal trace.) When the contents of TRFILE are being printed, all tracing is suspended until the printout is completed.

Note: When the trace print facility is run as a subtask, the trace output is not printed by VSE until VTAM is halted.

For more detailed information about initiating the trace print facility and the options available, see *VTAM Operation*.

All trace records contain either a time-stamp or a formatted time and date field from which the order of recording can be determined. This information can be helpful in synchronizing buffer contents and I/O traces and, to a lesser extent, VTAM and non-VTAM traces.

The TPRINT User Exit Facility: An interface is provided in the trace print facility to allow the user to write an exit routine for the editing of VTAM trace records. The dummy user exit module, ISTRAEUE, can be replaced by an installation-supplied module. The dummy module is called by the trace writing module, ISTRACTO, with the following register contents:

Register	Contents
0	0 = first entry to module 1 = return for more output
1	Address of parameter list
13	Save area address
14	Caller's return address
15	ISTRAEUE entry point address

The parameter list pointed to by register 1 contains the following information:

- Word 1 address of trace record (ISTTRAB)
- Word 2 address of output buffer
- Word 3 address of date and time stamp for this record

Consider the following points when programming a user exit routine for the TPRINT facility:

- TPRINT provides support for reading TRFILE with VSE access methods DTFSD or DTFMT. The file format is standard variable blocked records with BLKSIZE=2048. Each physical record contains a dummy 16-byte logical record that provides control information for TPRINT and is not passed to the user exit routine. The block header and the dummy record are mapped by the TRB.

- The remaining logical records contain trace data. Each has a 32-byte header mapped by the TRAB. Each is passed to the user exit routine for processing. Contents of the TRAB header and data format and length vary according to the trace record type.

- TPRINT provides access method support for all SYSLST output. Line counting and page headers are provided automatically. DTFDI is used with ASA control characters. The output buffer is cleared to blanks before the user exit routine is called. Use of control characters is limited to single space ( ), double space (0), triple space (-), and skip to channel 1 (1). When a skip is specified, TPRINT inserts its own page header and double space before printing the specified line.

- The TPRINT user exit routine may provide its own access method interface to SYSLOG (console) or any other device. For example: A trace record showing only errors is desired; or data reduction processing of trace records, such as line or internal trace, may require operator specification of additional record selection options.

- Buffer, I/O, and TNSTAT records are single entry data records. Line and internal trace records contain one or more entries following the TRAB header. The user exit routine must perform all de-blocking logic for multiple-entry records, using return code 8 to request re-entry from TPRINT for additional processing of the same trace record. See Figure 36 on page 229 for an example of user exit routine logic.

Figure 37 on page 230 shows the format of trace records as recorded on the VTAM trace file (TRFILE).

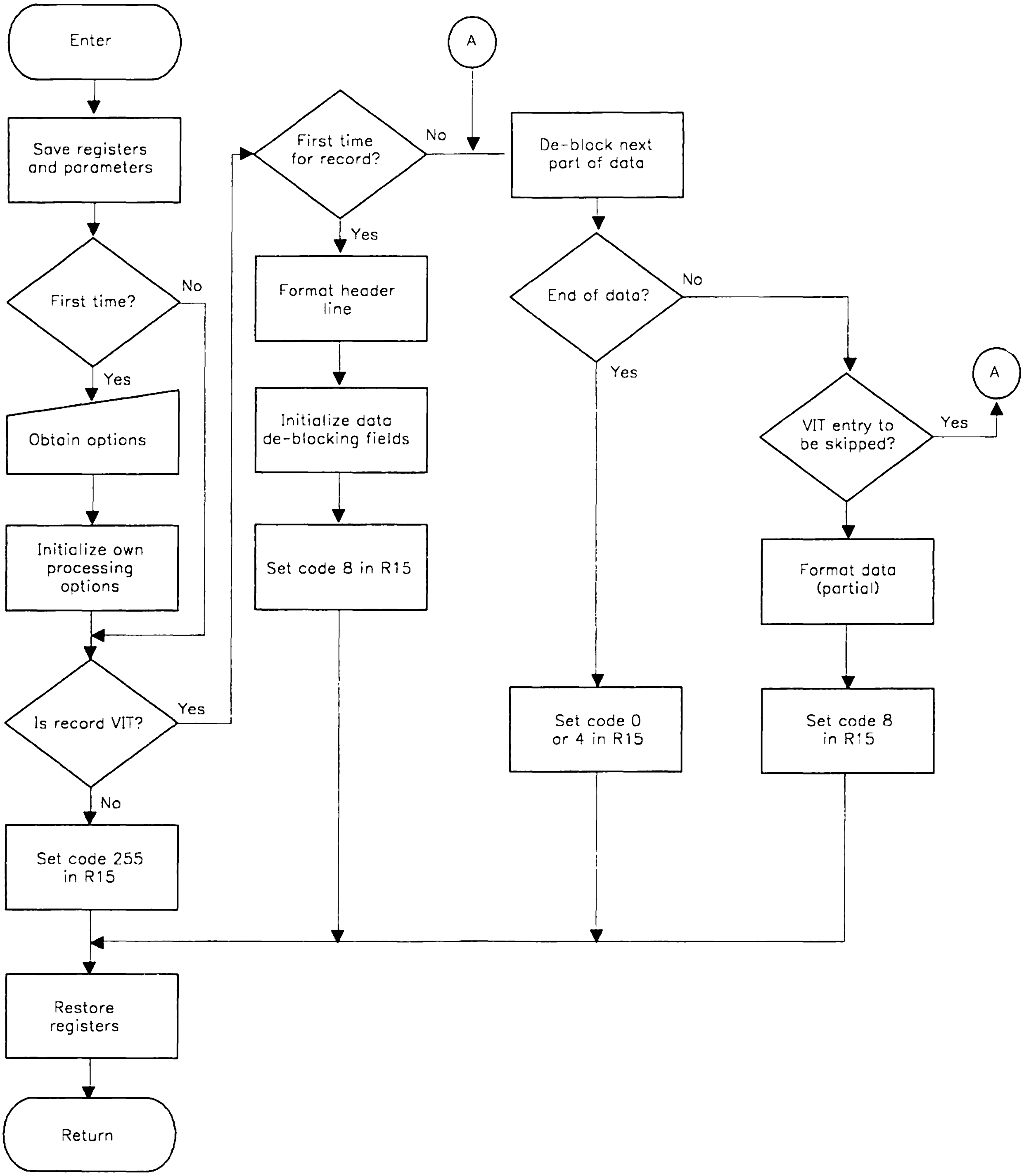


Figure 36. VSE TPRINT User Exit Routine Logic

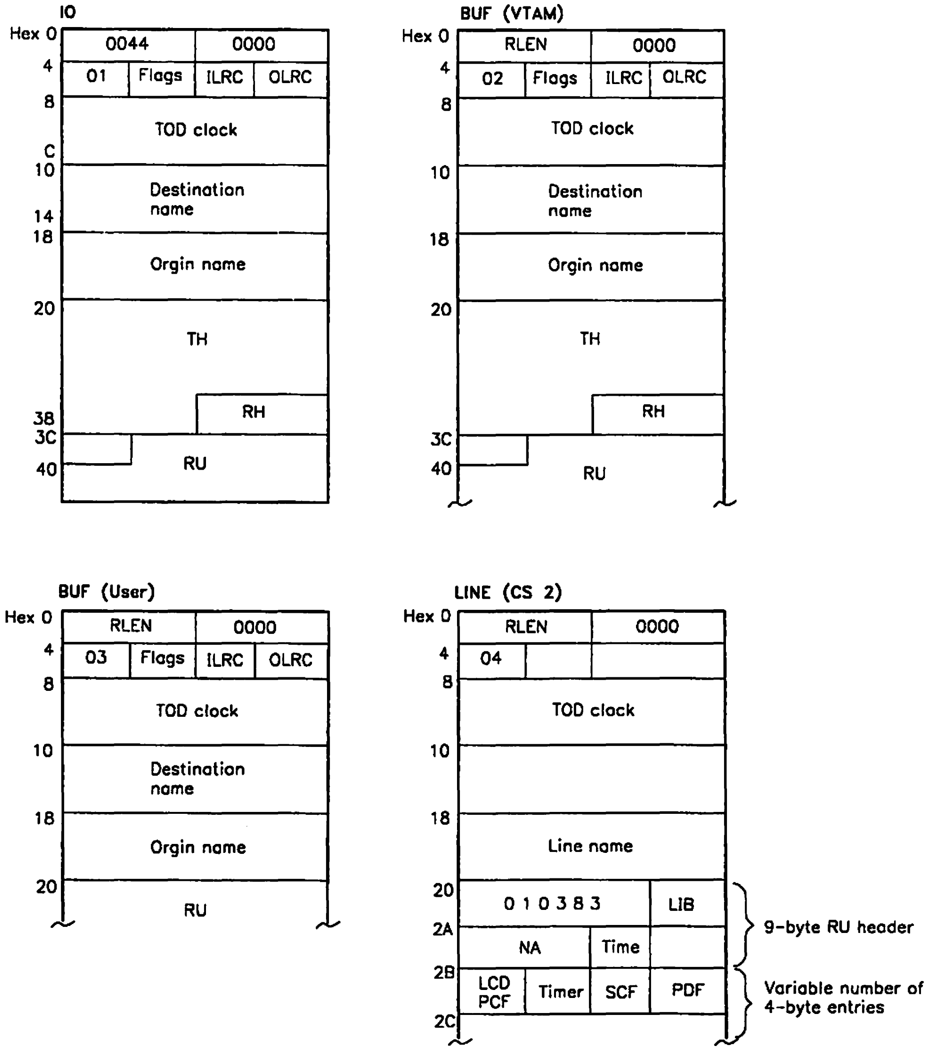
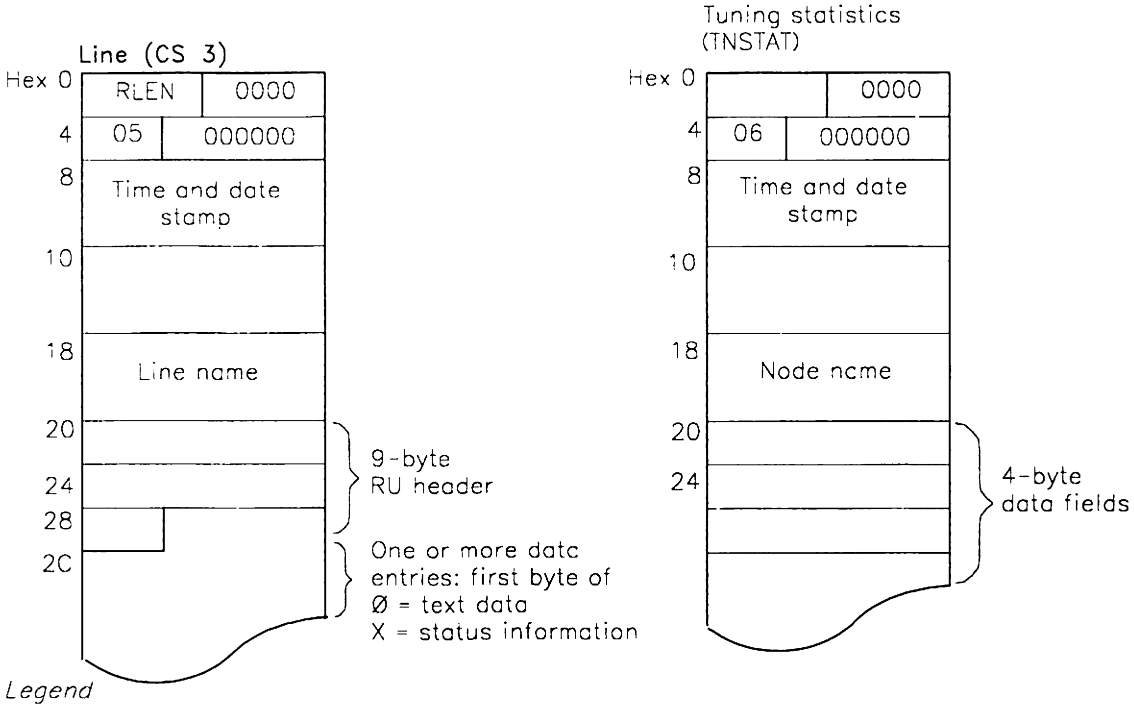


Figure 37 (Part 1 of 2). VSE Trace Records on TRFILE



Legend
Flag bit 0 (direction): 1 = Inbound
Flag bit 1: 1 = Confidential text
ILRC = Inbound lost record count
OLRC = Outbound lost record count
RLEN = Total record length

Figure 37 (Part 2 of 2). VSE Trace Records on TRFILE

Buffer Contents Trace

The buffer contents trace shows the contents of message buffers in two places: the application program interface (API) and the transmission subsystem component (TSC). When data is sent by an application program (outbound), API is the first component of VTAM to process it, and TSC is the last component of VTAM to process it. When data is received from the network (inbound), TSC is the first component of VTAM to process it, and API is the last. To enable you to distinguish where in VTAM the trace data was recorded, the output specifies either USER (for data recorded in API) or VTAM (for data recorded in TSC).

The API writes user buffer contents trace records while user data is still in the application program's buffers, before it is copied into VTAM buffers. Only user data is recorded.

TSC writes VTAM buffer contents trace records while the data is in VTAM fixed I/O buffers. The entire PIU — that is, the transmission header (TH), request/response header (RH), and user data — is recorded.

You can trace buffer contents data for messages to and from any of the following nodes:

- Application program
- Logical unit
- Channel-attached 3270
- SNA cluster controller
- NCP
- SSCP
- Host physical unit
- Cross-domain resource
- Cross-domain resource manager.

The trace record has a maximum length of 256 bytes, including trace field headers, transmission headers, request/response headers, and data.

Note: VTAM can start a buffer contents trace only for the resources that it owns. A data host, which does not own any NCPs, cannot start a buffer contents trace for an NCP or any of the NCP's subordinate resources.

When to Use the Buffer Contents Trace

The buffer contents trace can help you determine whether a problem is in the host (VTAM or an application program) or in the network. For example, if an application program sends a message to a terminal, and the message is correct in VTAM buffer contents trace output, but the message does not appear correctly at the terminal, then the problem is probably in a system resource other than VTAM or the application program.

The buffer contents trace cannot always be used to distinguish an application program problem from a VTAM problem. However, it can confirm the order in which data is passed between an application program and a logical unit. It can also record all the data passing to and from an application program.

If you do not need to trace user data, use the I/O trace.

Buffer Contents Trace Operation

Start the buffer contents trace with the MODIFY TRACE command or the TRACE start option.

Specify:	To Trace:
ID=*nodename*	Requests and responses flowing between VTAM and *nodename*.
ID=VTAM	Request and response units (RUS) for SSCP sessions.
ID=ISTPUS	Request and response units (RUS) for sessions between the host physical unit and another physical unit type 5 or a physical unit type 4 (these RUS include ER-ACT, ER-OP, and ER-TEST RUS).

If you use the SCOPE=ALL or EVERY operand when you start the trace, VTAM records messages to and from the specified node and all its valid subnodes.

Note: You cannot use the SCOPE=ALL or EVERY operands with ID=ISTPUS.

For more information on the MODIFY TRACE command, see *VTAM Operation*. For more information on the TRACE start option, see *VTAM Installation and Resource Definition*.

MVS Make sure that GTF (with TRACE=USR) is active before starting this trace. To format and print the data recorded by GTF, use PRDMP and set USR=(TPIO,CL01) or USR=(FEF,FF1) on the EDIT option.

Specify:	To Cause PRDMP to:
FF1	Format and print the buffer data recorded in the API component of VTAM. The trace output says "USER".
FEF	Format and print the buffer data recorded in the TSC component of VTAM. The trace output says "VTAM".

VM Use CPTRAP (with a format ID of 3D) to collect buffer contents trace data. Enter the ETRACE command, using the GTRACE and GROUP operands. Use TRAPRED or TAP to format and print the output.

VSE Use TPRINT to print the buffer contents trace records.

For more information on printing trace output, see "Printing Trace Records" on page 223.

Buffer Contents Trace Output

Figure 38 on page 234 shows an example of buffer contents trace output in MVS. (The output for VM and VSE are similar.)

The trace fields are explained after the figure. In addition to the fields described here, additional operating-system–dependent fields may appear (in MVS and VM). For a description of these fields, see "MVS Fields in VTAM Trace Output" on page 225 and "VM VM-Only Fields in VTAM Trace Output" on page 226.

```
BUFF  APPL12  /TERM200 LRC(000,000)   INBOUND
VTAM  TH=40000000 200000C2 0000 000C 00000004 IC000005 0028004 0040 RH=030080
          7DD8E811 D7F1F3C8 C1E3C140 C9E240C3       *'QY.P1THIS DATA IS C*
          D6D4D4C9 D5C740C9 D540D6D5 40E4E2C5       *OMMING IN ON THE USE*

BUFF  APPL12  /TERM200 LRC(000,000)  OUTBOUND
USER      C5D5E3D9 40C4C1E3 C140E3D6 40C5C8D6       *ENTER DATA TO ECHO B*
          C5D3D6E6 4B5CF04B D7D9C5E2 E240C5D5       *ELOW.*0.PRESS ENTER *
```

Figure 38. Example of Buffer Contents Trace Output

The following fields appear in the buffer contents trace:

Field Header	Meaning
BUF *destname/origname*	Is the destination (*destname*) and origin (*origname*) node name. In Figure 38 the destination is APPL12 and the origin is TERM200.
LRC(*xxx*,*yyy*)	The number of records lost since the last trace record was written because the trace facility could not get a VTAM buffer. *xxx* is the destination's lost record count, and *yyy* is the source's lost record count.
INBOUND or OUTBOUND	The direction of the traced data (inbound or outbound) with respect to this host subarea. By use of this field and the PIU sequence number in the TH, requests and corresponding responses can be matched.
VTAM or USER	Indicates where the message buffers were traced. VTAM means the buffers were traced in TSC (in which case the TH and the RH are included in the trace record). USER means the buffers were traced in API (in which case the TH and the RH are not included).
TH	The 26-byte transmission header portion of the path information unit (PIU).
RH	The 3-byte request/response header portion of the PIU.

The rest of the trace record shows the contents of the buffer. Each line contains 20 bytes of user data in five groups of eight hexadecimal digits followed by the equivalent 20 EBCDIC characters.

Confidential data will *not* be recorded in trace records. When the trace facility detects confidential data (CONFTXT=YES is specified on the applications program's NIB macroinstruction), the user data is replaced with the marker in the trace output.

> MVS VM The marker is 'CONFIDENTIAL AND SUPPRESSED'.
> VSE The marker is 'CONFIDENTIAL DATA'.

MVS If you are using the VTAM encryption facility, data on a cryptographic session is handled in the same way as confidential data.

I/O Trace

The I/O trace shows requests and responses that flow between VTAM and network nodes. You can trace I/O activity for any of the following types of nodes:

- Application program
- Physical unit
- Logical unit
- SNA cluster controller
- NCP
- SSCP
- Host physical unit
- Host as an intermediate routing node
- Channel attachment major node
- Cross-domain resource
- Cross-domain resource manager.

The maximum I/O trace record length is 36 bytes.

When to Use the I/O Trace

Use the I/O trace to record the order that PIUs flow between network nodes and VTAM. For example, you might use this trace to determine whether an application program receives all the responses that it should and whether VTAM forwards all the requests issued by the application program.

The content of the I/O trace is similar to that of the buffer trace, except that the I/O trace records less user data. If you do not need this user data, you can save trace file space by using the I/O trace.

I/O Trace Operation

Start the I/O trace with the MODIFY TRACE command or the TRACE start option.

Specify:	To Trace:
ID=*nodename*	Requests and responses flowing between VTAM and *nodename*.
ID=VTAM	Request and response units (RUs) for SSCP sessions.
ID=ISTPUS	Request and response units (RUs) for sessions between the host physical unit and another physical unit type 5 or a physical unit type 4 (these RUs include ER-ACT, ER-OP, and ER-TEST RUs).
ID=ISTIRN	Request and response units (RUs) that flow through this host while this host is acting as an intermediate routing node.

If you use the SCOPE=ALL or EVERY operand when you start the trace, the trace records I/O activity for the specified node and all its valid subnodes.

Notes:

1. You **cannot** use the SCOPE=ALL or EVERY operands with ID=ISTPUS or ID=ISTIRN.
2. You **must** specify SCOPE=ALL when tracing a channel-attachment major node.
3. You may trace a link in a channel-attachment major node, but not a link station.

See *VTAM Operation* for information on the MODIFY command. See *VTAM Installation and Resource Definition* for more information on the TRACE start option.

MVS Make sure that GTF (with TRACE=RNIO) is active before starting this trace. To format and print the data recorded by GTF, use either PRDMP or TAP. For PRDMP, set RNIO on the EDIT option.

VM Use CPTRAP (with a format ID of 3D) to collect I/O trace data. Enter the ETRACE command, using the GTRACE and GROUP operands. Use TRAPRED or TAP to format and print the output.

VSE Use TPRINT or TAP to print the trace records collected by the I/O trace.

For more information on printing trace output, see "Printing Trace Records" on page 223.

I/O Trace Output

MVS VM Figure 39 shows an example of I/O trace output.

VSE Figure 40 shows an example of I/O trace output.

The trace fields are explained after each figure. In addition to the fields described here, additional operating system dependent fields may appear (in MVS and VM). For a description of these fields, see 'MVS Fields in VTAM Trace Output" on page 225 and "VM VM-Only Fields in VTAM Trace Output" on page 226.

```
 .
 .
 IN 40000000 200080B1 0000000C 00000004 0E000005 000B808D 000A9B80 00000841 00006000
 .
OUT 42000000 2000009A 00000004 0000000C 0E00000B 0005808D 000ACB80 00000823 00000000
 .
 .
```

Figure 39. MVS VM Example of I/O Trace Output

MVS VM The following fields appear in the I/O trace:

Field Header	Meaning
IN or OUT	The direction of the trace record: IN for inbound to VTAM, and OUT for outbound to the specified node. When ISTIRN is specified, each RU is traced twice — once when it enters VTAM (IN) and once when it leaves (OUT).

The rest of the trace record is some or all the path information unit (PIU). It consists of a 26-byte transmission header (TH), a request/response header (RH), and as much of the request/response unit (RU) as will fit. The data length field in the TH indicates the length of the RH and RU in bytes. The RU may contain sense data.

```
IO  TERM12  /        87.153/10:31:02.13          OUTBOUND
TH=40000000000000000000000C0000000C1D0000390005000D0028  RH=6B8000  RU=31010202714020

IO          /TERM12  87.153/10:31:02.15           INBOUND
TH=40000000000000000000000C0000000C1D0000050039000D0004  RH=EB8000  RU=31
```

Figure 40. VSE Example of I/O Trace Output

VSE The following fields appear in the I/O trace:

Field Header	Meaning
IO *destname/origname*	Is the destination (*destname*) or origin (*origname*) node name. Usually, only one node name will be resolved by the I/O trace. In Figure 40 on page 236 the node name is TERM12 in both records.
yy.jjj/hh:mm:ss.hh	Is the year (yy), the Julian date (jjj), and the time to hundredths of a second (hh:mm:ss.hh) that trace record was written. To find this field in Figure 40 on page 236, look for "87.153/10:31:02.13" in the first trace record.
LRC(xxx,yyy)	Is the number of records lost since the last trace record was written because of the inability of the trace facility to get VTAM buffers. *xxx* is the destination's lost record count, and *yyy* is the source's lost record count. This field is not present in the trace record unless records are lost.
INBOUND or OUTBOUND	The direction of the traced data (inbound or outbound) with respect to this host subarea. With this field and the PIU sequence number in the TH, requests and corresponding responses can be matched.
TH	Is the 26-byte transmission header portion of the path information unit (PIU). The data length field in the TH indicates the length of the RH and RU in bytes.
RH	Is the 3-byte request/response header portion of the PIU.
RU	Ten bytes of the request/response unit, which may contain sense data.

SMS (Buffer Use) Trace

The storage management services (SMS) trace records contain information on the use and availability of VTAM buffer pools. SMS trace records are written after a predetermined number of requests occur for VTAM buffers. An IBM-supplied threshold causes a trace record to be written after every 1000 (hex 3E8) requests.

To change the IBM-supplied number (the halfword at label RACBSNAP in VTAM module ISTRACON), use the service aid SPZAP (in MVS and VM) or the utility aid MSHP (in VSE).

MVS See *MVS Service Aids* for a complete description of SPZAP.

VM See the *VM/SP Group Control System Guide* for a complete description of SPZAP.

VSE See the *VSE/AF MSHP Reference* for a complete description of MSHP.

Notes:

1. If the DISPLAY BFRUSE command is issued while this trace is running, the fields MAX TOTAL, MAX USED. and TIMES EXP reflects buffer usage only since the last trace record was written, because the SMS trace resets these fields.
2. The SMS trace is **not** the same thing as the VTAM internal trace with the SMS option specified. The SMS trace **is** similar to the DISPLAY BFRUSE command. The SMS trace displays in trace output the same information that the DISPLAY BFRUSE command displays on the screen.

When to Use the SMS Trace

Use the SMS trace during VTAM installation to evaluate VTAM use of buffer pools, to help estimate how many buffers VTAM needs for normal operation, and, with dynamic buffering, to limit buffer pool expansions to peak use periods. You can use the SMS trace in conjunction with tuning statistics.

SMS Trace Operation

Start the SMS trace with the MODIFY TRACE command, or use the TRACE start option with TYPE=SMS and ID=VTAMBUF specified. For more information about the MODIFY TRACE command, see *VTAM Operation.* For more information about the TRACE start option, see *VTAM Installation and Resource Definition.*

MVS Make sure that GTF (with TRACE=USR) is active before starting this trace. To format and print the data recorded by GTF, use PRDMP and set USR=(FF0) or USR=(CL02) on the EDIT option.

VM Use CPTRAP (with a format ID of 3D) to collect SMS (buffer use) trace data. Enter the ETRACE command, using the GTRACE and GROUP operands. Use TRAPRED or TAP to format and print the output.

VSE Make sure that SDAID is active, and TRACE VTAMBU is specified. The trace output can be printed with the DOSVSDMP utility program. For more information on SDAID and DOSVSDMP, see *VSE Service Aids.*

For more information on printing trace output, see "Printing Trace Records" on page 223.

MVS VM SMS Trace Record Output

The SMS trace record format is the same for MVS and VM. Figure 41 and Figure 42 on page 239 show examples of SMS trace output.

The trace fields are explained after the figures. In addition to the fields described here, other operating-system-dependent fields may appear. These fields are described in "MVS Fields in VTAM Trace Output" on page 225 and "VM VM-Only Fields in VTAM Trace Output" on page 226.

```
VTAM BUFFERS     MAXU MAXQ AVNO TEXP MBUF TOTL     MAXU MAXQ AVNO TEXP MBUF TOTL
              IO 0021 0000 0044 0000 0064 0064 PP 0000 0000 0000 0000 0000 0000
              LP 0004 0000 003E 0000 0040 0040 WP 0001 0000 001B 0000 0010 0010
              CR 0001 0000 0000 0000 0021 0021 UE 0000 0000 0022 0000 0022 0022
              SF 0002 0000 0021 0000 0033 0033 SP 0000 0000 0000 0000 0000 0000
```

Figure 41. MVS VM Example of SMS Trace Output for VTAM Pre-V3R2

```
VTAM BUFFERS        MAXU     MAXQ     AVNO     TEXP     MBUF     TOTL
                 IO 00000021 00000000 00000044 00000000 00000064 00000064
                 PP 00000000 00000000 00000000 00000000 00000000 00000000
                 LP 00000004 00000000 0000003E 00000000 00000040 00000040
                 WP 00000001 00000000 0000001B 00000000 00000010 00000010
                 NP 00000000 00000000 00000000 00000000 00000000 00000000
                 LF 00000000 00000000 00000000 00000000 00000000 00000000

VTAM BUFFERS        MAXU     MAXQ     AVNO     TEXP     MBUF     TOTL
                 CR 00000001 00000000 00000000 00000000 00000021 00000021
                 UE 00000000 00000000 00000022 00000000 00000022 00000022
                 SF 00000002 00000000 00000021 00000000 00000033 00000033
                 SP 00000000 00000000 00000000 00000000 00000000 00000000
                 AP 00000000 00000000 00000000 00000000 00000000 00000000
```

Figure 42. MVS VM Example of SMS Trace Output for VTAM V3R2. Two separate records will be printed, and they may be separated by another trace entry.

The following fields appear in the SMS trace for MVS and VM:

Field Header	Meaning
Pool ID	Is the first field in each record and identifies the buffer pool. Pool IDs and their corresponding buffer pool names are: AP Application program pageable pool (APBUF) CR Copied RPL pool (CRPLBUF) IO Fixed I/O pool (IOBUF) LF Large fixed pool (LFBUF) LP Large pageable pool (LPBUF) SF Small fixed pool (SFBUF) SP Small pageable pool (SPBUF) WP Working set pageable pool (WPBUF). **Note:** NPBUF, PPBUF, and UPBUF are no longer used by VTAM.
MAXU	The maximum number of buffers in the pool that were in use at any time since the last trace record was written.
MAXQ	The maximum number of requests for buffers that were queued waiting for storage at any time since the last trace record was written.
AVNO	The number of available buffers (those not in use at the time the trace record was written).
TEXP	The number of times the buffer pool was expanded since the last trace record was written.
MBUF	The maximum number of buffers that were in the pool at any time since the last trace record was written. This includes both used and unused buffers.
TOTL	The total number of buffers that were in the pool at the time this record was written. This includes both used and unused buffers.

VSE SMS Trace Record Output

Figure 43 shows an example of SMS trace output for VSE. The trace fields are explained after the figure.

```
VTAM BUFFER POOL USE  SEQ. NO = 00000001  DAY 223  TIME 08:21:43
VF IN USE=00011 MAX ALLOC=00011 MAX WAIT=00000 EXPAND=00000 MAX AVAIL=00000 CUR AVAIL=00000
VP IN USE=00397 MAX ALLOC=00424 MAX WAIT=00000 EXPAND=00000 MAX AVAIL=00000 CUR AVAIL=00000
SF IN USE=00009 MAX ALLOC=00011 MAX WAIT=00000 EXPAND=00000 MAX AVAIL=00022 CUR AVAIL=00002
LF IN USE=00045 MAX ALLOC=00109 MAX WAIT=00000 EXPAND=00027 MAX AVAIL=00116 CUR AVAIL=00063
SP IN USE=00000 MAX ALLOC=00001 MAX WAIT=00000 EXPAND=00000 MAX AVAIL=00022 CUR AVAIL=00003
LP IN USE=00005 MAX ALLOC=00011 MAX WAIT=00000 EXPAND=00003 MAX AVAIL=00015 CUR AVAIL=00009
WP IN USE=00079 MAX ALLOC=00087 MAX WAIT=00000 EXPAND=00003 MAX AVAIL=00099 CUR AVAIL=00027
```

Figure 43. **VSE** Example of SMS Trace Output

The following fields appear in the SMS trace for VSE:

Field Header	Meaning
Pool ID	Is the first field in each record. It identifies the buffer pool. Pool IDs and their corresponding buffer pool names are: LF Large fixed pool (LFBUF) LP Large pageable pool (LPBUF) SF Small fixed pool (SFBUF) SP Small pageable pool (SPBUF) VF Variable fixed pool (VFBUF) VP Variable pageable pool (VPBUF) WP Working set pageable pool (WPBUF). **Notes:** 1. Sometimes pool names are not listed. The pool names are not available when VTAM is running disabled. 2. APBUF, NPBUF, PPBUF, and UPBUF are no longer used by VTAM.
IN USE	Indicates the number of buffers in the pool that were in use at any time during the time covered by the snapshot dump. For the VP and VF pools it indicates the number of pages that were in use at the time the snapshot dump was taken.
MAX ALLOC	Indicates the maximum number of buffers in the pool that were in use at any time during the time covered by the snapshot dump. For the VP and VF pools it indicates the maximum number of pages in use.
MAX WAIT	Indicates the maximum number of requests for buffers that were queued waiting for storage at any time during the time covered by the snapshot dump. For VF and VP pools this field has no meaning.
EXPAND	Indicates the number of times the buffer pool was expanded during the time covered by the snapshot dump. For VF and VP pools this field has no meaning.

Field Header	Meaning
MAX AVAIL	Indicates the maximum number of buffers that were in the pool at any time during the time covered by the snapshot dump. For VF and VP pools this field has no meaning.
CUR AVAIL	Indicates the number of buffers that were in the pool at the time the snapshot dump was taken. For VF and VP pools this field has no meaning.

MVS TGET/TPUT Trace for TSO/VTAM

The TGET/TPUT trace for TSO/VTAM writes a GTF trace record for each inbound and outbound message that uses the TGET/TPUT/TPG interface (SVC 93) between a TSO command processor and the VTIOC component of TSO/VTAM. Outbound messages are traced before being placed in the VTIOC queue manager output buffer. Inbound messages are traced before the data is sent to the TSO command processor.

Note: The TGET/TPUT trace does not trace address space ID TPUTS.

MVS When to Use the TGET/TPUT Trace

Use this trace if the failure is restricted to TSO sessions. This trace can help you determine whether TSO/VTAM or your TSO command processor is causing the problem. For example, if outbound data is correct in the TGET/TPUT trace output, but incorrect in the buffer trace output, the problem is probably in TSO/VTAM or VTAM. Use the following tables as a guideline to determine where the error is occurring:

Direction of Data	If TPUT Trace Data Is:	And Buffer or I/O Trace Data Is:	Then Possible Error Is In:
Outbound	Correct	Incorrect	• VTAM • TSO/VTAM • TPUT option • User edit exits
Outbound	Incorrect	Trace not required	TSO or the command processor
Outbound	Correct	Correct	Network

Figure 44. MVS Location of TPUT (Outbound) Error

Direction of Data	If TGET Trace Data Is:	And Buffer or I/O Trace Data Is:	Then Possible Error Is In:
Inbound	Incorrect	Correct	• VTAM • TSO/VTAM • TGET option • User edit exits
Inbound	Correct	Trace not required	TSO or the command processor
Inbound	Incorrect	Incorrect	Network

Figure 45. MVS Location of TGET (Inbound) Error

MVS TGET/TPUT Trace Operation

Start the TGET/TPUT trace with the MODIFY TRACE command and specify TYPE=TSO.

The trace output is a record of inbound and outbound messages for the specified TSO user ID.

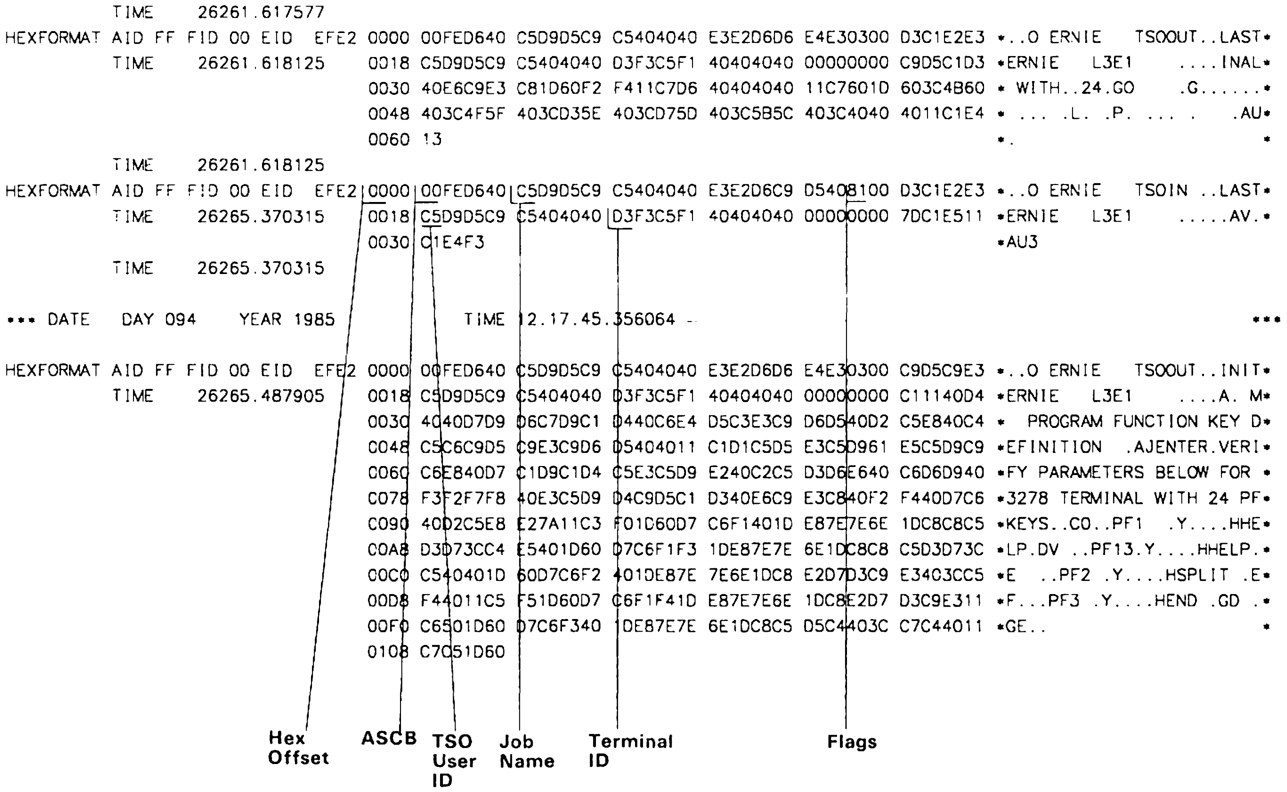
Make sure that GTF (with TRACE=USR) is active before starting this trace. To print these trace records, use PRDMP with USER=(FE2) specified.

For more information on printing trace output, see "Printing Trace Records" on page 223.

MVS TGET/TPUT Trace Output

The trace records created by the TGET/TPUT trace have a 12-byte GTF header and a 32-byte trace header followed by the data portion of the RU in unformatted hexadecimal. The entire RU is traced, but will span several trace entries if it is longer than 228 bytes.

Figure 46 on page 243 shows an example of TGET/TPUT trace output. The trace fields are explained after the figure.



```
          TIME      26261.617577
HEXFORMAT AID FF FID 00 EID  EFE2 0000 00FED640 C5D9D5C9 C5404040 E3E2D6D6 E4E30300 D3C1E2E3 *..O ERNIE   TSOOUT..LAST*
          TIME      26261.618125    0018 C5D9D5C9 C5404040 D3F3C5F1 40404040 00000000 C9D5C1D3 *ERNIE   L3E1     ....INAL*
                                    0030 40E6C9E3 C81D60F2 F411C7D6 40404040 11C7601D 603C4B60 * WITH..24.GO     .G......*
                                    0048 403C4F5F 403CD35E 403CD75D 403C5B5C 403C4040 4011C1E4 * ... .L. .P. ... .     .AU*
                                    0060 13                                                     *.                       *
          TIME      26261.618125
HEXFORMAT AID FF FID 00 EID  EFE2 0000 00FED640 C5D9D5C9 C5404040 E3E2D6C9 D5408100 D3C1E2E3 *..O ERNIE   TSOIN ..LAST*
          TIME      26265.370315    0018 C5D9D5C9 C5404040 D3F3C5F1 40404040 00000000 7DC1E511 *ERNIE   L3E1     .....AV.*
                                    0030 C1E4F3                                                 *AU3
          TIME      26265.370315

*** DATE   DAY 094    YEAR 1985              TIME 12.17.45.356064 -                                                ***

HEXFORMAT AID FF FID 00 EID  EFE2 0000 00FED640 C5D9D5C9 C5404040 E3E2D6D6 E4E30300 C9D5C9E3 *..O ERNIE   TSOOUT..INIT*
          TIME      26265.487905    0018 C5D9D5C9 C5404040 D3F3C5F1 40404040 00000000 C11140D4 *ERNIE   L3E1     ....A. M*
                                    0030 4040D7D9 D6C7D9C1 D440C6E4 D5C3E3C9 D6D540D2 C5E840C4 *  PROGRAM FUNCTION KEY D*
                                    C048 C5C6C9D5 C9E3C9D6 D5404011 C1D1C5D5 E3C5D961 E5C5D9C9 *EFINITION  .AJENTER.VERI*
                                    C060 C6E840D7 C1D9C1D4 C5E3C5D9 E240C2C5 D3D6E640 C6D6D940 *FY PARAMETERS BELOW FOR *
                                    C078 F3F2F7F8 40E3C5D9 D4C9D5C1 D340E6C9 E3C840F2 F440D7C6 *3278 TERMINAL WITH 24 PF*
                                    C090 40D2C5E8 E27A11C3 F01D60D7 C6F1401D E87E7E6E 1DC8C8C5 *KEYS..CO..PF1  .Y....HHE*
                                    C0A8 D3D73CC4 E5401D60 D7C6F1F3 1DE87E7E 6E1DC8C8 C5D3D73C *LP.DV ..PF13.Y....HHELP.*
                                    00C0 C540401D 60D7C6F2 401DE87E 7E6E1DC8 E2D7D3C9 E3403CC5 *E  ..PF2 .Y....HSPLIT .E*
                                    00D8 F44011C5 F51D60D7 C6F1F41D E87E7E6E 1DC8E2D7 D3C9E311 *F...PF3 .Y....HEND .GD .*
                                    00F0 C6501D60 D7C6F340 1DE87E7E 6E1DC8C5 D5C4403C C7C44011 *GE..                    *
                                    0108 C7C51D60
```

Figure 46. MVS Example of TGET/TPUT Trace Output

The following fields appear in the TGET/TPUT trace.

The first two bytes in each row show the hex offset in storage. The data follows after that.

Byte (hex)	**Meaning**
00–03	ASCB address
04–0B	Job name
0C–0E	C'TSO'
0F–11	C'IN' for inbound data (TGET); C'OUT' for outbound data (TPUT)
12	TGET/TPUT option flags (see the SVC 93 description in the *Debugging Handbook* for bit definitions)
13	TGET: return code (see the *TSO Guide to Writing a Terminal Monitor Program* for descriptions) TPUT: X'00' EDIT, ASID, FULLSCREEN, or CONTROL options. X'01' NOEDIT option X'02' TPG macro issued
14–17	C'INIT' for the first 228-byte section of a PIU; C'MID' for the middle sections of a PIU; C'LAST' for the last section of a PIU
18–1F	TSO user ID
20–27	Symbolic terminal ID
28–2B	Zero
2C	Start of user data.

Traces Provided by NCP

NCP provides several kinds of traces to record the flow of network events. Each trace occurs at a different point in the network (see Figure 35 on page 222). This allows you to follow an RU through the network and determine where in the network the RU is incorrect. (The RU could be out of sequence or lost, the data in the RU could have been changed, and so forth.)

The NCP traces are:

- "Generalized PIU Trace"
- "Line Trace" on page 246
- "MVS Network Controller Line Trace (3710 Only)" on page 250
- "Scanner Interface Trace (3720, 3725, and 3745 Only)" on page 251
- "Transmission Group Trace" on page 252.

For more information on all service aids for NCP, see the *NCP, SSP, and EP Diagnosis Guide*.

Generalized PIU Trace

The generalized PIU trace collects PIU trace data collected by the NCP. The resulting trace output shows the flow of PIUs exchanged between the NCP and its attached resources. This trace is hierarchical when started for a physical unit or a line. That is, logical units associated with the physical unit are automatically traced when traffic flows to them. Likewise, when the generalized PIU trace is started for a line, physical units and logical units associated with the line are automatically traced when traffic flows to them. When the generalized PIU trace is stopped for a physical unit, the trace is reset for all logical units associated with the physical unit, regardless of how the generalized PIU trace was started.

When to Use the Generalized PIU Trace

Use the generalized PIU trace to trace PIU data at the NCP level and to determine whether the NCP has received or sent PIU data.

Note: VTAM can start a generalized PIU trace only for the resources that it owns. A data host, which does not own any NCPs, cannot start a generalized PIU trace for an NCP or any of the NCP's subordinate resources.

Generalized PIU Trace Operation

Start the generalized PIU trace with the MODIFY TRACE command. For more information on the MODIFY TRACE command for the generalized PIU trace, see *VTAM Operation*.

MVS Make sure that GTF (with TRACE=USR) is active before starting this trace. Use TAP with INPUT=GPT to format and print the trace output.

VM Use CPTRAP (with a format ID of 3D) to collect generalized PIU trace data. Enter the ETRACE command, using the GTRACE and GROUP operands. Use TRAPRED or TAP to format and print the output.

VSE Use TAP with INPUT=GPT to format and print the trace output.

For more information on printing trace output, see "Printing Trace Records" on page 223.

Line Trace

The line trace, a joint function of VTAM and the NCP, records the status of a line each time the NCP receives data from or sends data to that line. Although the trace is controlled by VTAM, the information in the trace records is collected by the NCP. The NCP sends the data to VTAM in a PIU. A trace type indicator in the PIU indicates whether the trace is a byte line trace (Type 2 scanner) or a block line trace (Type 3 scanner).

The line trace records the operating parameters of a line each time a level 2 interruption occurs on that line.

- For a 3705 Communications Controller with a Type 2 communication scanner, a level 2 interruption occurs each time a byte of data is sent or received across the line.
- For a 3705 Communications Controller with a Type 3 communication scanner, a level 2 interruption occurs each time an NCP buffer is filled and the buffer data is sent or received across the line.
- For a 3720, 3725, or 3745 Communication Controller, a level 2 interruption occurs each time a message (an entire PIU) is sent or received.

You can use the line trace only for lines attached to a communication controller and operating in network control mode.

For each 3705 Communications Controller, as many as eight line traces can be active at a time.

For each 3720, 3725, or 3745 Communication Controller, a combination of eight line or scanner interface traces can be active at one time.

The number of active line traces to be allowed is specified during NCP generation. In a cross-domain network in which the communication controller is connected to more than one host processor, the number of active traces allowed is distributed among the connected host processors on a first-come, first-served basis.

Note: As the number of active line traces increases, the system becomes less efficient.

When to Use the Line Trace

You might use this trace if you suspect a problem with a device attached to a communication controller. If data appears correctly in the line trace but the terminal or printer does not react appropriately, the device itself is probably causing the failure.

Before using a line trace, you should use buffer and I/O traces to verify that the problem is not in VTAM or an application program. You may also want to use the scanner interface trace (for 3720, 3725, and 3745 Communication Controllers only), which traces data after it has been processed by the NCP and before it leaves the communication controller. Therefore, it can help determine whether the problem is in the NCP or in the line.

Note: VTAM can start a line trace only for the resources that it owns. A data host, which does not own any NCPs, cannot start a line trace for an NCP or any of the NCP's subordinate resources.

You might want to use a transmission group (TG) trace instead of or in addition to this trace. If there is more than one active line in a transmission group, and you do not know which line is causing the problem, use the transmission group trace. The transmission group trace also shows more data than the line trace.

Line Trace Operation

Start the line trace with the MODIFY TRACE command or the TRACE start option. In either case, specify TYPE=LINE. For more information on the MODIFY TRACE command, see *VTAM Operation.* For more information on the TRACE start option, see *VTAM Installation and Resource Definition.*

MVS Make sure that GTF (with TRACE=USR) is active before starting this trace. To format and print the line trace:

- For a 3705, use TAP or PRDMP. For PRDMP, specify USR=(LINE) or USR=(FF2) on the EDIT option.
- For a 3720, 3725, or 3745, use TAP with INPUT=LINE specified.

VM Use TRAPRED or TAP to format and print the output.

- For a 3705, use TAP or TRAPRED. For TRAPRED, specify the EFF2 option.
- For a 3720, 3725, or 3745, use TAP with INPUT=LINE specified.

VSE Use TPRINT or TAP to format and print the line trace.

- For a 3705, use TAP or TPRINT. For TPRINT, specify LINE=name or LINE=ALL.
- For a 3720, 3725, or 3745, use TAP with INPUT=LINE specified.

For more information on printing trace output, see "Printing Trace Records" on page 223.

Line Trace Output (CS Type 2)

Figure 47 shows an example of line trace output for a 3705 Communications Controller with a Type 2 communication scanner. This example was created with PRDMP, the MVS print utility, but the output looks the same for VM and VSE, except as noted below.

The trace fields are explained after the figure. In addition to the fields described here, additional operating-system-dependent fields may appear (in MVS and VM). For a description of these fields, see "MVS Fields in VTAM Trace Output" on page 225 and "VM VM-Only Fields in VTAM Trace Output" on page 226.

```
LINE LINE01 LRC(000,000) INBOUND ACTIVE RNTIME=1D
  LCD C PCF A TIME 16 SCF 42 PDF D4   LCD C PCF A TIME 16 SCF 42 PDF 40
  LCD C PCF A TIME 16 SCF 42 PDF E5   LCD C PCF A TIME 16 SCF 42 PDF 40
  LCD C PCF A TIME 16 SCF 42 PDF E3   LCD C PCF A TIME 16 SCF 42 PDF 40
  LCD C PCF A TIME 16 SCF 42 PDF C1   LCD C PCF A TIME 16 SCF 42 PDF 40
```

Figure 47. Example of Line Trace Output (CS Type 2)

The following fields appear in the line trace for CS Type 2.

The header portion of the line trace record contains these fields:

Field Header	Field Contents
LINE	The name of the node being traced. In Figure 47, the line name is LINE01.
LRC(*xxx*,*yyy*)	The number of records lost since the last trace record was written because the trace facility could not get a VTAM buffer. *xxx* is the destination's lost record count, and *yyy* is the source's lost record count.
INBOUND	MVS VM The direction of the data with respect to this host. It always says INBOUND because VTAM always receives the trace records from the NCP. VSE This field is replaced by a time stamp.
ACTIVE	The status of the line being traced. In Figure 47, the line status is ACTIVE. The line status may also appear as DEACTIVATE or SLOWDOWN. • ACTIVE means that the line trace is active. • DEACTIVATE means that the line trace is not active. • SLOWDOWN means that the line trace is not active because the NCP is in slowdown mode. A status of DEACTIVATE or SLOWDOWN appears only in the last record to be sent for that line. It means that no more data will be sent until the trace is activated again for that line.
RNTIME=*hh*	Is a timer field. *hh* is a hexadecimal value indicating in tenths of a second the time at which the communication controller sent the completed line trace record to VTAM. This value is taken from a timer that is reset to 0 after 25.5 seconds. This value can be compared with the TIME value for the first level 2 interruption data contained in the second part of the line trace record. This comparison shows the elapsed time covered by this trace record. Comparing this with the number of level 2 interruptions contained in the trace record indicates the I/O activity during the elapsed time.

The data portion of the line trace record contains a timer value and values from the interface control word (ICW) and communication controller hardware registers at each level 2 interruption. Each printed line contains up to two sets of data, one for each of two level 2 interruptions. For detailed descriptions of these additional fields in the line trace records, see the *Principles of Operation* manual for your communication controller and the *NCP, SSP, and EP Diagnosis Guide*.

Field Header	Field Contents
LCD	The line control definer (LCD) defines the type of protocol being used over the line. The LCD for a line is generally set when the NCP is loaded and remains unchanged throughout normal operation.
PCF	The primary control field (PCF) defines the state of the line interface at any particular time.
TIME	This hexadecimal value indicates, in tenths of a second, the time elapsed between level 2 interrupts as they occur on the line interface as data is received. Level-two interrupts are represented by the data (PDF). The value of this timer is reset to 0 every 25.5 seconds.
SCF	The secondary control field (SCF) is used as a status and operation modifier by the communication scanner and the control program.
PDF	The parallel data field (PDF) is used as a character buffer. For a transmission, the program places the characters to be sent in the PDF, and transmits them to the line interface. For a receive operation, the character is assembled, then transferred by hardware to the PDF.

Line Trace Output (CS Type 3)

Figure 48 shows an example of line trace output for a 3705 Communications Controller with a Type 3 communication scanner. This trace record shows the operating status of a line each time an NCP buffer is filled, instead of each time a character is transmitted.

This example was created with PRDMP, the MVS print utility, but the output looks the same for VM and VSE, except as noted below.

The trace fields are explained after the figure. Most fields in this record are similar to the last example, but the Type 3 scanner has some additional fields. For more detailed descriptions of these additional fields see the *Principles of Operation* manual for your communication controller and the *NCP and EP Reference Summary and Data Areas*.

```
LINE LINE01 LRC(000,000) INBOUND ACTIVE RNTIME=1D
  STATUS SCF LCD PCF EPCF STAT1 STAT2 TIME ADDR CNTL IOBLXB DISP ICW CCBLV
          44  9   9  00    00    01    80   03   71    30     AA  11  1BB2
  TEXT C0B4
```

Figure 48. Example of Line Trace Output (CS Type 3)

In addition to several fields shown in the line trace for CS Type 2, the following fields appear in the line trace for CS Type 3:

Field Header	Field Contents
EPCF	The extended primary control field. These bits extend the meaning of the PCF.
STAT1	Status byte 14 of the ICW.
STAT2	Status byte 15 of the ICW.
TIME	This hexadecimal value indicates, in tenths of a second, the time elapsed between level 2 interrupts as they occur on the line interface as data is received. Level-two interrupts are represented by the data. The value of this timer is reset to 0 every 25.5 seconds.
ADDR	The SDLC frame address field.
CNTL	The SDLC frame control field.
IOBLXB	The 1-byte command field from the IOB/LXB control block.
DISP or XDISP	MVS VM DISP or VSE XDISP is byte 0 from the communication scanner display register.
ICW	Byte 13 of the interface control word.
CCBLV or LVL2	MVS VM CCBLV or VSE LVL2 is the address of the level 2 interruption processor routine, taken from field CCBL2 in the NCP CCB control block.
TEXT	The contents of the NCP buffer being transmitted in both hexadecimal and EBCDIC.

MVS Network Controller Line Trace (3710 Only)

The network controller line trace traces SDLC, BSC, and START/STOP data link control frames sent or received by an IBM 3710 Network Controller.

The 3710 physical unit provides a network controller line trace on a physical unit type 2 node. The network controller line trace traces data link control (DLC) protocols, such as DLC frames exchanged between a network controller and an NCP, or DLC frames exchanged between a network controller and a control unit.

The network controller line trace facility also traces the synchronous data link control (SDLC), binary synchronous control (BSC), and start/stop DLC frames that are sent or received by a line adapter.

When tracing an SDLC or BSC line, you can choose to trace all the frames (control and data) or just the data frames. All frames are traced on a start/stop line.

MVS When to Use the Network Controller Line Trace

Run this trace if you have an incorrect output problem, a performance problem, or you suspect a network controller problem.

MVS Network Controller Line Trace Operation

Start the network controller line trace with the MODIFY TRACE command. For more information on the MODIFY TRACE command for the network controller line trace, see *VTAM Operation.*

Active traces stop if:

- The network controller is reset (for example, during a failure or a configuration load)
- The network controller enters slowdown (shown in the header of the RECTRD RU)
- The line fails during the trace.

If the network controller recovers after a failure, and it is not the only one on the line, the trace will resume.

Make sure that GTF is active before you start this trace. Use TAP with NETCTLR specified to format and print the output.

For more information on printing trace output, see "Printing Trace Records" on page 223.

MVS Network Controller Line Trace Output

VTAM receives the trace data from a network controller in a series of RECTRD RUS, which are put into trace records. These records are passed to GTF, which stores them in a trace output data set, SYS1.TRACE. Use TAP to print the trace output.

Scanner Interface Trace (3720, 3725, and 3745 Only)

The scanner interface trace, a joint function of VTAM and the NCP, records the operating parameters of a line whenever a 3720, 3725, or 3745 Communication Controller sends or receives a message. Although the trace is controlled by VTAM, the information in the trace records is collected by the NCP. The scanner interface trace records the operating status of a line whenever the communication controller sends or receives a message (PIU).

When you start the trace, you can choose how many bytes of data you want to trace. The default is the entire PIU. You can only use the scanner interface trace on lines operating in network control mode.

For each communication controller, a combination of eight line traces or scanner interface traces can be active at a time. The number of active scanner interface traces to be allowed is specified during NCP generation. In a multisystem network in which the communication controller is connected to more than one host processor, the number of active traces allowed is distributed among the connected host processors on a first-come, first-served basis.

Note: As the number of active scanner interface traces increases, the system becomes less efficient.

When to Use the Scanner Interface Trace

Use the scanner interface trace to determine whether the NCP or a line is causing a problem. The scanner interface trace records inbound data before the NCP processes it, and records outbound data after the NCP processes it. Therefore, if outbound data is correct in a scanner interface trace, but the device does not react properly, the problem is in either the line or the device.

Scanner Interface Trace Operation

Start the scanner interface trace with the MODIFY TRACE command or the TRACE start option.

The COUNT option allows you to choose how many bytes of data you want traced (0 to 254 or ALL). The default is the entire PIU.

For more information on the MODIFY command for the scanner interface trace, see *VTAM Operation*. For more information on the TRACE start option, see *VTAM Installation and Resource Definition.*

MVS Make sure that GTF (with TRACE=USR) is active before starting this trace. Use TAP with INPUT=LINE to format and print the output.

VM Use CPTRAP (with a format ID of 3D) to collect scanner interface trace data. Enter the ETRACE command, using the GTRACE and GROUP operands. Use TRAPRED or TAP to format and print the output.

VSE Use TAP to format and print the output.

For more information on printing trace output, see "Printing Trace Records" on page 223.

Transmission Group Trace

The transmission group trace, a joint function of VTAM and the NCP, records the sequence of PIUs being sent through a transmission group. The transmission group trace shows PIU traffic on a transmission group as if the transmission group were a single line. The sequence of PIUs traced is the sequence of their entry to and exit from the transmission group, not the sequence of actual transmission along the physical lines in the transmission group. The line trace shows the sequence of PIUs transmitted and received on a physical line.

When to Use the Transmission Group Trace

Use the transmission group trace instead of a line trace if your failure is restricted to sessions using a specific transmission group and you have more than one active line in a TG.

Note: VTAM can start a transmission group trace only for the resources that it owns. A data host, which does not own any NCPs, cannot start a transmission group trace.

Transmission Group Trace Operation

Start the transmission group trace with the MODIFY TRACE command. The name specified in the ID operand is the line "associated" with the transmission group trace.

The trace is started for the transmission group in which the specified line resides. This line and its associated link station must be active before the transmission group trace is started.

If a line trace is already active for the chosen line within a transmission group, you must stop it before starting a transmission group trace.

Once activated, the transmission group trace remains active until one of the following occurs:

- The operator stops the trace.
- The associated line or link station is deactivated or fails.
- The NCP goes through Auto Network Shutdown.
- The NCP goes into slowdown mode.

If the transmission group trace is ended because the associated line or link station fails or is deactivated, the operator can restart it by issuing the MODIFY command for another line in the TG.

For more information on the MODIFY TRACE command for the transmission group trace, see *VTAM Operation*.

MVS GTF (with TRACE=USR) must be active to use this trace. Use PRDMP or TAP to format and print these trace records. For PRDMP, specify USR=(LINE) or USR=(FF2) on the EDIT option. For TAP specify INPUT=LINE.

VM Use CPTRAP (with a format ID of 3D) to collect transmission group trace data. Enter the ETRACE command, using the GTRACE and GROUP operands. Use TRAPRED or TAP to format and print the output.

VSE Use TPRINT or TAP.

For more information on printing trace output, see "Printing Trace Records" on page 223.

Transmission Group Trace Output

MVS VM Figure 49 is an example of transmission group trace records.

VSE Figure 50 on page 255 is an example of transmission group trace records.

The trace fields are explained after each figure. In addition to the fields described here, additional operating system dependent fields may appear (in MVS and VM). For a description of these fields, see "MVS Fields in VTAM Trace Output" on page 225 and "VM VM-Only Fields in VTAM Trace Output" on page 226.

```
LINE   LINE23    LRC(000,000)   INBOUND       ACTIVE       RNTIME=00
       TEXT 42000000 2000009A 00000004 0000000C 0E00000B
            0005808D 000ACB80 00A0
       TEXT 6E000000 F000009A 00000004 00000001 00000000
            00000032 20000000 00000823 00000000 00000000
            00000123 00000040
```

Figure 49. MVS VM Example of Transmission Group Trace Output

The following fields appear in the transmission group trace for MVS and VM:

Field Header	Meaning
LINE *linename*	The name of the line "associated" with the transmission group trace. This line name was specified in the MODIFY TRACE command that started the transmission group trace.
LRC(*xxx*,*yyy*)	The number of records lost since the last trace record was written because the trace facility could not get a VTAM buffer. *xxx* is the destination's lost record count, and *yyy* is the source's lost record count.
INBOUND	The direction of the data with respect to this host. It always says INBOUND because the data is always received from the NCP.
ACTIVE	The status of the line being traced. In Figure 49, the line status is ACTIVE. The line status may also appear as DEACTIVATE or SLOWDOWN. • ACTIVE means that the line trace is active. • DEACTIVATE means that the line trace is not active. • SLOWDOWN means that the line trace is not active because the NCP is in slowdown mode. A status of DEACTIVATE or SLOWDOWN appears only in the last record to be sent for that line. It means that no more data will be sent until the trace is activated again for that line.

Field Header	Meaning
RNTIME=*hh*	Is a timer field, where *hh* is a hexadecimal value indicating, in tenths of a second, the time at which the communication controller sent the completed line trace record to VTAM. This value is taken from a timer that is reset to 0 every 25.5 seconds. This value can be compared with the TIME value for the first level 2 interruption data contained in the second part of the line trace record. This comparison shows the elapsed time covered by this trace record. Comparing this with the number of level 2 interruptions contained in the trace record indicates the I/O activity during the elapsed time.
TEXT	Shows the TH and RH for each PIU that traversed the data path through the transmission group. If the PIU is for data flow control, session control, or network control, the full RU portion of the PIU is also included in the trace record. The RU may contain sense data. If the PIU is function management data (FMD) and contains an FM header, six bytes of the RU are included in the transmission group trace record. If an FMD PIU without an FM header is traced, the RU is not included.

```
LINE   LINE23    87.150/09:15:06.18                          ACTIVE      RNTIME=00
 TEXT 42000000 2000009A 00000004 0000000C 0E00000B 0005808D 000ACB80 00A00000
      00000032 20000000 00050101 0000000B 0040
 TEXT 6E000000 F000009A 00000004 00000001 00000000 0005808D 000ACB80 00A00000
      00000123 00000040 00050101 0000000B 0000
```

Figure 50. VSE Example of Transmission Group Trace Output

The following fields appear in the transmission group trace for VSE:

Field Header	Meaning
LINE *linename*	Is the name of the line associated with the transmission group trace. This line name was specified in the MODIFY TRACE command that started the transmission group trace.
yy.jjj/hh:mm:ss.hh	Is the year (yy), the Julian date (jjj), and the time to hundredths of a second (hh:mm:ss.hh) that trace record was written. To find this field in Figure 50, look for "87.150/09:15:06.18" in the first trace record.
LRC(xxx,yyy)	Is the number of records lost since the last trace record was written because of the inability of the trace facility to get VTAM buffers. *xxx* is the destination's lost record count, and *yyy* is the source's lost record count. This field is not present in the trace record unless records are lost.

Field Header	Meaning
TRANSMIT or RECEIVE	Indicates, for full-duplex lines, the side of the line used for the transmission.
ACTIVE	The status of the line being traced. In Figure 50, the line status is ACTIVE. The line status may also appear as DEACTIVATE or SLOWDOWN. • ACTIVE means that the line trace is active. • DEACTIVATE means that the line trace is not active. • SLOWDOWN means that the line trace is not active because the NCP is in slowdown mode. A status of DEACTIVATE or SLOWDOWN appears only in the last record to be sent for that line. It means that no more data will be sent until the trace is activated again for that line.
RNTIME=*hh*	Is a timer field, where *hh* is a hexadecimal value indicating, in tenths of a second, the time at which the communication controller sent the completed line trace record to VTAM. This value is taken from a timer that wraps around to 0 after 25.5 seconds. This value can be compared with the TIME value for the first level 2 interruption data contained in the second part of the line trace record. This comparison will show the elapsed time covered by this trace record. Comparing this with the number of level 2 interruptions contained in the trace record indicates the I/O activity during the elapsed time.
TEXT	Shows the TH and RH for each PIU that traversed the data path through the transmission group. If the PIU is data flow control, session control, or network control, the full RU portion of the PIU is also included in the trace record. The RU may contain sense data. If the PIU is function management data (FMD) and contains an FM header, six bytes of the RU are included in the transmission group trace record. If an FMD PIU without an FM header is traced, the RU is not included.

VSE Dynamic Trace Utility for Communication Adapters

This trace facility of VSE applies to the 4361 communication adapter, and to the 9370 telecommunications subsystem. It can record communication adapter line traces on tape while the line and the communication adapter are both active. It can also print selected trace entries after a line-related error has occurred.

The communication adapter line trace should be used when line-related errors are suspected on lines attached to the communication adapter, for example:

- Frequent unexplained line outages
- Unusual sense or status in I/O error messages
- Device outages that cannot be explained with I/O or buffer traces.

For more information on using this trace facility, see the *VSE Service Aids* manual.

Operator Commands

You can control and monitor the VTAM network with both predefined specifications (start options) and operator commands. These commands include the DISPLAY command and the MODIFY command.

- DISPLAY
  - "Display Buffer Pool Use" on page 258
  - "Display NCP Storage" on page 263
  - "Display Network Status" on page 263
  - "Display Route Status" on page 265
  - "Display Route Test" on page 266.
- MODIFY
  - "Modify Input/Output Problem Determination" on page 270
  - "Modify Message Module Identification" on page 270
  - "Modify NCP Intensive Mode Recording" on page 271
  - "Modify SDLC Link Level 2 Test" on page 271
  - "Modify Tuning Statistics" on page 272.
- For more information on problem determination, see the other parts of this chapter or the table of contents.

For information on VTAM start options, see *VTAM Installation and Resource Definition*. For information on commands, see *VTAM Operation*.

Note: Several other licensed IBM programs can also be used to monitor and collect error statistics from the VTAM network: the NetView program, NCCF, NPDA, and NLDM. The *Network Program Products General Information* manual describes these licensed programs.

Display Buffer Pool Use

You can use the DISPLAY BFRUSE command to display information about buffer use. In response to this command, VTAM indicates that the display is for buffer use and issues a series of messages that contain monitoring information. For each buffer pool, this information includes:

- Buffer pool ID
- Flags (Q or F): Q shows a request is queued for this pool; F shows dynamic buffering has failed for this pool
- Size of each buffer in this pool
- Current total number of buffers in this pool
- Current count of buffers available (the number not in use)
- Largest number of buffers this pool has expanded to at any time
- Largest number of buffers in use at any time
- Cumulative count of the number of times each buffer pool has expanded
- Expansion and contraction thresholds
- The expansion increment (the number of buffers to be added to a buffer pool during dynamic expansion)
- VTAM's intermediate routing node buffer use limit (IRNLIMIT), current buffer use, and maximum buffer use
- MVS VM VTAM's CSA buffer use limit (CSALIMIT), current buffer use, and maximum buffer use
- MVS VM Maximum amount of CSA in use since VTAM was started.

If the DISPLAY BFRUSE command is used while an SMS (buffer use) trace is running, the fields MAX TOTAL, MAX USED, and TIMES EXP reflects buffer usage only since the last trace record was written, because the SMS trace resets these fields.

For more information about the DISPLAY BFRUSE command, see *VTAM Operation*.

MVS VM Sample Output

Figure 51 on page 259 and Figure 52 on page 260 are examples of the output from a DISPLAY BFRUSE command in MVS and VM.

For example, in Figure 51, for the buffer pool with ID IO00, each buffer has 183 bytes, 200 buffers are currently in the pool, and 166 of those are available. The largest number of buffers the pool has ever contained is 200, and the maximum ever used at one time is 49. The buffer pool has not been expanded.

If the number of available buffers drops to 5 (the expansion threshold in this example), VTAM expands the pool by adding 20 buffers (the expansion increment). If the number of available buffers reaches the contraction threshold, VTAM frees buffers (in pages) until the number of buffers in the pool is less than the contraction threshold. VTAM frees buffers only if there is a full page in which none of the buffers is being used.

An N/A (not applicable) in the expansion/contraction column indicates that the user did not set an expansion increment.

If dashes (-----) appear in place of the contraction threshold, the buffer pool is not currently expanded (although it might have been expanded previously).

```
IST350I  VTAM DISPLAY - DOMAIN TYPE = BUFFER POOL DATA
IST632I  BUFF   BUFF  CURR   CURR    MAX    MAX  TIMES     EXP/CONT    EXP
IST633I   ID    SIZE TOTAL  AVAIL  TOTAL   USED   EXP     THRESHOLD   INCR
IST356I  I000  00183 00200  00166  00200 00049  00000  00005/----- 00020
IST356I  LP00  01344 00016  00013  00016 00006  00000  00001/----- 00003
IST356I  WP00  00184 00350  00344  00350 00006  00000  00001/----- 00020
IST356I  LF00  00068 00048  00042  00048 00006  00001  00001/00093 00046
IST356I  CRPL  00116 00500  00497  00500 00004  00000  00004/----- 00030
IST356I  SF00  00064 00051  00051  00051 00000  00000  00001/----- 00051
IST356I  SP00  00112 00002  00002  00002 00000  00000  00001/----- 00032
IST449I  CSALIMIT = NOLIMIT, CURRENT = 0001887K, MAXIMUM = 0001887K
IST449I  MAXIMUM CSA USED = 0001887K
IST595I  IRNLIMIT = NOLIMIT, CURRENT = 0000000K, MAXIMUM = 0000000K
IST314I  END
```

Figure 51. **MVS VM** Example of DISPLAY BFRUSE Output for VTAM Pre-V3R2

```
IST350I VTAM DISPLAY - DOMAIN TYPE = BUFFER POOL DATA
IST920I I000 Q F BUFF SiZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I ------------------------------------------------------------
IST920I LP00 Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I ------------------------------------------------------------
IST920I WP00 Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I ------------------------------------------------------------
IST920I LF00 Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I ------------------------------------------------------------
IST920I CRPL Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I ------------------------------------------------------------
IST920I SF00 Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I ------------------------------------------------------------
IST920I SP00 Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I ------------------------------------------------------------
      .
      .
IST314I END
```

Figure 52. **MVS VM** Example of DISPLAY BFRUSE Output for VTAM V3R2

DISPLAY BFRUSE output can help you identify possible sources of problems. The following chart shows some problem symptoms and the corresponding buffers to check in SMS trace output:

For this symptom:	Check this buffer pool:
I/O hang	IOBUF
Session failure	CRPLBUF and LPBUF
VTAM hang	LPBUF

VSE Sample Output

Figure 53 and Figure 54 on page 262 are examples of the output of a DISPLAY BFRUSE command in VSE.

For example, in Figure 53, for the buffer pool with ID LF, each buffer has 407 bytes, 92 buffers are currently in the pool, and 54 of those are available. The most buffers the pool has ever contained is 140, and the maximum ever used at one time is 127. The buffer pool has been expanded 24 times.

If the number of available buffers equals 7, VTAM will expand the pool by adding 12 buffers. If the number of available buffers becomes greater than 31, VTAM will attempt to free enough of the extents to reduce the number of available buffers below 31 (provided these buffers are on the same page or pages).

An N/A (not applicable) in the expansion/contraction column indicates that the user did not set an expansion increment.

If dashes (-----) appear in place of the contraction threshold, the buffer pool is not currently expanded (although it might have been expanded previously).

```
5D50I  VTAM DISPLAY - DOMAIN TYPE = BUFFER POOL DATA
5G32I  BUFF    BUFF  CURR   CURR   MAX    MAX   TIMES    EXP/CONT     EXP
5G33I   ID     SIZE TOTAL   AVAIL  TOTAL  USED   EXP    THRESHOLD    INCR
5D56I  VF     02048 00020P 00008P  N/A  00012P  N/A        N/A       N/A
5D56I  VP     02048 00512P 00170P  N/A  00456P  N/A        N/A       N/A
5D56I  SF     00356 00022  00020  00022 00016  00000  00003/----- 00005
5D56I  LF     00407 00092  00054  00140 00127  00024  00007/00031 00012
5D56I  SP     00112 00022  00022  00022 00000  00000  00003/----- 00017
5D56I  LP     01016 00013  00009  00015 00001  00001  00003/00007 00002
5D56I  WP     00160 00075  00074  00135 00004  00004  00003/----- 00012
5F95I  IRNLIMIT = NOLIMIT, CURRENT = 0000000K, MAXIMUM = 0000000K
5D14I  END
```

Figure 53. VSE Example of DISPLAY BFRUSE Output for VTAM Pre-V3R2

```
IST350I VTAM DISPLAY - DOMAIN TYPE = BUFFER POOL DATA
IST920I VF   Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  ****N/A***  EXP/CONT THRESH   *****N/A***
IST922I          CURR TOTAL 0123456789P CURR AVAILABLE    0123456789P
IST923I          MAX TOTAL  ****N/A***  MAX USED          0123456789P
IST924I -----------------------------------------------------------
IST920I VP   Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  ****N/A***  EXP/CONT THRESH   *****N/A***
IST922I          CURR TOTAL 0123456789P CURR AVAILABLE    0123456789P
IST923I          MAX TOTAL  ****N/A***  MAX USED          0123456789P
IST924I -----------------------------------------------------------
IST920I SF   Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I -----------------------------------------------------------
IST920I LF   Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I -----------------------------------------------------------
IST920I SP   Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I -----------------------------------------------------------
IST920I LP   Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I -----------------------------------------------------------
IST920I WP   Q F BUFF SIZE  0123456789  EXP INCREMENT     01234
IST9211          TIMES EXP  0123456789  EXP/CONT THRESH   012345/6789
IST922I          CURR TOTAL 0123456789  CURR AVAILABLE    0123456789
IST923I          MAX TOTAL  0123456789  MAX USED          0123456789
IST924I -----------------------------------------------------------
IST595I IRNLIMIT = NOLIMIT, CURRENT = 0000000, MAXIMUM = 0000000
IST314I END
```

Figure 54. VSE Example of DISPLAY BFRUSE Output for VTAM V3R2

DISPLAY BFRUSE output can help you identify possible sources of problems. The following chart shows some problem symptoms and the corresponding buffers to check in SMS trace output:

For this symptom:	Check this buffer pool:
I/O hang	LFBUF
Session failure	LPBUF
VTAM hang	LPBUF

Display NCP Storage

You can dynamically request and display any selected portion of NCP storage by using a DISPLAY NCPSTOR command. Up to 256 bytes can be displayed for each command. For the 3720 or 3745 communication controller, you can use the DISPLAY NCPSTOR command to display up to 256 bytes of an NCP dump or state vector stored on the disk.

VTAM issues message IST245I (or 5C45I, for VTAM V3R1 or V3R1.2 running under VSE) to display the storage contents.

Notes:

1. The NCP storage to be displayed may change while it is being formatted for transmission.
2. You cannot display NCP storage at a data host, because a data host does not own any NCPS.

For more information on the DISPLAY NCPSTOR command, see *VTAM Operation*.

Display Network Status

You can use the DISPLAY command to display status information about the VTAM network. For example, a request for DISPLAY of a physical unit or of a logical unit indicates whether that node has been added by dynamic reconfiguration. DISPLAY also indicates whether a logical unit, a physical unit, or a link is supported by the Network Terminal Option (NTO).

Note: From a data host you cannot display the status of either an NCP or the NCP's subordinate resources, because a data host does not own any NCPS.

Two types of node status are displayed when you use the DISPLAY command: The STATUS field shows the *current state* of the node; the DESIRED STATE field shows the *desired state*. Both states are taken from the RDTE for the node.

The desired state is the condition that VTAM processing is attempting to establish for the node. Previously entered operator commands or recovery processing may establish the desired state for a node. When processing is completed, the desired state and the current state should be equal. See *VTAM Messages and Codes* for descriptions of the resource status codes used in response to the DISPLAY command.

The DISPLAY command displays the status of the following:

- Any or all application programs
- Any or all host physical units
- Any or all active major nodes
- An NCP in a channel- or link-attached communication controller
- A channel-attachment (CA) major node and its subordinate links and link stations
- Any or all lines and the traces associated with them. This display includes the line control for each line.
- Any or all line groups

- Any or all cross-subarea link stations
- Any or all physical units or BSC cluster controllers
- Any or all logical units
- Any or all LU-LU session states
- Routes used to access other subareas (which are defined by active path definitions)
- Any or all cross-domain resource managers (CDRMS).
- Any or all cross-domain resources (CDRSCS)
- MVS VSE The adjacent SSCPS that can be used on a path to a given destination
- MVS Any or all TSO users, by user identification. This display indicates whether the TGET/TPUT trace is active for a TSO user.

MVS VM Displays for SNA Network Interconnection

If SNA network interconnection is in use, the DISPLAY command shows:

- The network ID associated with a resource (if any)
- For a cross-network CDRSC, the real resource name.
- For a cross-network CDRM, the real name of the gateway node through which the SSCP-SSCP session passes, and the network address as known in the requesting host's network.

Trace Status

The DISPLAY command also shows a 5-character resource status code (xxxxx) for the transmission group trace, SIT trace, generalized PIU trace, or network controller line trace when applicable.

- Line trace status, transmission group trace status, and SIT trace status are part of the response to the DISPLAY ID=*linename* command. They appear in the messages shown in the following example.

  Note: In this example, both the ISTxxxI and 5xxxI message IDS are shown. (Messages beginning with the number 5 pertain only to VTAM V3R1 or V3R1.2 running under VSE.) In the actual output only one of these message IDS will appear, according to the operating system used.

  ```
            V3R1 &
  MVS VM    V3R1.2
  V3R2 VSE  VSE
  IST655I   5F55I   LINE TRACE STATUS=xxxxx  TG TRACE STATUS=xxxxx
  IST588I   5E88I   SIT  TRACE STATUS=xxxxx
  ```

- Generalized PIU trace status is part of the the response to the DISPLAY ID=*nodename* command. It appears in the messages shown in the following example.

  Note: In this example, both the ISTxxxI and 5xxxI message IDS are shown. (Messages beginning with the number 5 pertain only to VTAM V3R1 or V3R1.2 running under VSE.) In the actual output only one of these message IDS will appear, according to the operating system used.

```
           V3R1 &
MVS VM     V3R1.2
V3R2 VSE   VSE
IST752I    5H52I   GPT TRACE STATUS=xxxxx
```

- MVS The status of the network controller line trace is part of the response to the DISPLAY TRACES command. This command tells you if a 3710 network controller is tracing and gives you the names of the resources being traced. The status appears in this message:

```
IST838I TRACE STATUS DISPLAY FOR ID=network controller name
IST839I PU NAME      DOWNSTREAM LINE NAME
IST840I nnnnnnn1          nnnnnnn2
IST840I nnnnnnn1          nnnnnnn2
IST314I END
```

For more information about the DISPLAY command and the status it displays, see *VTAM Operation*.

See *VTAM Messages and Codes* for more information on resource status codes.

Display Route Status

The DISPLAY ROUTE command shows the status and availability of virtual and explicit routes. For more information on the DISPLAY ROUTE command, see *VTAM Operation*.

VTAM displays the status of selected routes and, if TEST=YES is specified, does a route test on the routes selected in the DISPLAY command. (See "Display Route Test" on page 266.)

The VTAM operator may select the origin of the routes to be displayed or tested. The origin may be either a host processor or an NCP.

The display of status for the routes selected is formatted as shown in Figure 55.

Note: In this example, both the ISTxxxI and 5xxxI message IDs are shown. (Messages beginning with the number 5 pertain only to VTAM V3R1 or V3R1.2 running under VSE.) In the actual output only one of these message IDs will appear, according to the operating system used.

```
          V3R1 &
MVS VM    V3R1.2
V3R2 VSE  VSE
IST535I   5F35I   ROUTE DISPLAY nnn FROM SA sa1 TO SA sa2
IST808I   5I08I   ORIGIN PU = origpu DEST PU = destpu
IST536I   5F36I   VR   TP    STATUS        ER   ADJSUB   STATUS
IST537I   5F37I   x    y     status        z     aaa     status
   .        .     .    .       .           .      .         .
   .        .     .    .       .           .      .         .
   .        .     .    .       .           .      .         .
IST537I   5F37I   x    y     status        z     aaa     status
IST314I   5D14I   END
```

Figure 55. DISPLAY Route Status

MVS VM If you are using SNA network interconnection, the DISPLAY ROUTE command can be used to show the status and availability of adjacent VTAM networks. The resulting display is the same as shown in the previous example, except that message IST808I contains an additional field, NETID=netid, to show the ID of the adjacent network. See the *VTAM Operation* manual for information on displaying routes in adjacent networks.

See *VTAM Messages and Codes* for a complete description of the variable data contained in the messages that result from the DISPLAY ROUTE command.

It is possible for the test results for an explicit route to be lost before they are displayed. For example, if a node or a link along the explicit route fails between the time the test request flows outbound and the time the test results flow inbound, the results will be lost. If this occurs, re-enter the DISPLAY ROUTE command for that explicit route.

Note: You can use the session monitor of the NetView or NLDM licensed program to collect more information about routes. If all the required session monitors or NLDMS along the route are in session, you can test the entire route, from one session end to the other. For more information, see *NetView Operation* or *NLDM Operation*.

Display Route Test

If a route test was requested, results of the test are sent asynchronously to the console of the operator requesting the display. If the route test failed, the results are also sent to the console of the host that owns the rejecting subarea node. If the host owning the rejecting subarea is the same host that initiated the route test, that host will receive the test results twice.

To be tested, the explicit route must be known to VTAM. This means the explicit route must be defined to VTAM, or must at some time have been operative.

Successful Route Test

If TEST=YES is set and the route test is successful, the following asynchronous messages follow the route status display messages previously described.

The test results are formatted as shown in Figure 56.

Note: In this example, both the ISTxxxI and 5xxxI message IDS are shown. (Messages beginning with the number 5 pertain only to VTAM V3R1 or V3R1.2 running under VSE.) In the actual output only one of these message IDS will appear, according to the operating system used.

```
         V3R1 &
MVS VM   V3R1.2
V3R2 VSE VSE
IST538I  5F38I ROUTE TEST ### IN PROGRESS
IST533I  5F33I ER n SUCCEEDED IN ROUTE TEST ###
IST797I  5H97I          FROM  VIA    ADJACENT      DEST   ER LENGTH
IST644I  5G44I      fffffff    TG    aaaaaaaa  dddddddd
IST534I  5F34I           sss    t         xxx       yyy           1
IST798I  5F98I          nnnn
```

Figure 56. Output of a Successful Route Test

In this example,

- fffffff is the name of the origin physical unit
- aaaaaaaa is the adjacent physical unit

- dddddddd is the name of the destination physical unit
- sss is the subarea number of ffffffff
- t is the transmission group number
- xxx is the subarea number of aaaaaaaa
- yyy is the subarea of dddddddd
- 1 is the explicit route length
- nnnn is the network ID of the node being displayed.

Failed Route Test

If the explicit route test fails because VTAM is unable to send the Explicit Route Test RU into the network, a message tells why the test cannot be performed. This message is shown in the following example.

Note: In this example, both the ISTxxxI and 5xxxI message IDs are shown. (Messages beginning with the number 5 pertain only to VTAM V3R1 or V3R1.2 running under VSE.) In the actual output only one of these message IDs will appear, according to the operating system used.

```
          V3R1 &
MVS VM    V3R1.2
V3R2 VSE  VSE
IST510I   5F10I    ROUTE TEST ### FAILED - reason
```

If the explicit route test is initiated by VTAM but fails, the messages in Figure 57 will show the reason for the test failure.

```
          V3R1 &
MVS VM    V3R1.2
V3R2 VSE  VSE
IST533I   5F33I   ER 0 FAILED IN ROUTE TEST 8
IST797I   5H97I              FROM   VIA     ADJACENT       DEST   ER LENGTH
IST644I   5G44I          ffffffff    TG     aaaaaaaa   dddddddd
IST534I   5F34I               sss     t          xxx        yyy
IST798I   5H98I              nnnn
IST572I   5F72I         REJECTING    TG     ADJACENT              ER MASK
IST816I   5I16I               rrr     g          zzz                 mmmm
IST573I   5F73I     <ER NOT DEFINED,>
                    <A REQUIRED TG IS NOT ACTIVE,>
                    <ER NOT REVERSIBLE,>
                    <ER EXCEEDS MAXIMUM LENGTH,>
                    <UNEXPECTED TYPE BYTE X'##',>
```

Figure 57. Output of a Failed Route Test

In this example,

- ffffffff is the name of the physical unit which originated the ER_TEST
- aaaaaaaa is the adjacent physical unit
- dddddddd is the name of the destination physical unit
- sss is the subarea number of ffffffff
- t is the transmission group number
- xxx is the subarea number of aaaaaaaa
- yyy is the subarea of dddddddd
- nnnn is the network ID of the node being displayed
- rrr is the rejecting subarea
- zzz is the adjacent subarea

- mmmm is the explicit route mask
- g is the transmission group number.

Location of Failure in a Route Test

The variable text in message 573I can help you determine which direction the route test was going when it failed. If the reason is "TG NOT ACTIVE" or "MIGRATION," the adjacent subarea **follows** the rejecting subarea in the route being tested. Therefore, the problem is in the adjacent subarea or the link to the adjacent subarea from the rejecting subarea. (See Figure 58.)

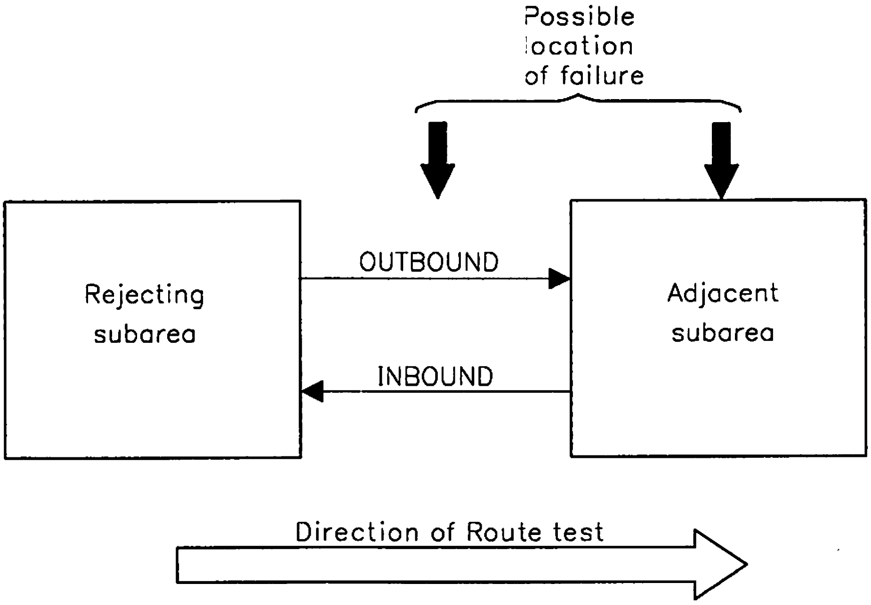


Figure 58. Route Test Failure (TG NOT ACTIVE or MIGRATION)

If the reason is "NOT REVERSIBLE," "MAX ER LENGTH," or "NOT DEFINED," the adjacent subarea **precedes** the rejecting node in the route being tested. (See Figure 59.)

Check to see if the problem is a path definition error. If not, it may be a VTAM error.

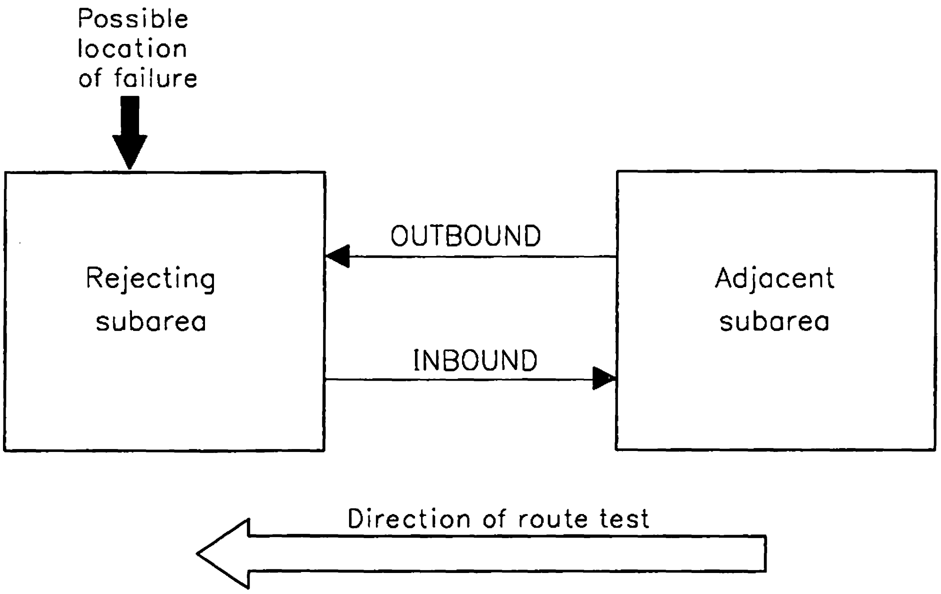


Figure 59. Route Test Failure (NOT REVERSIBLE, MAX ER LENGTH, or NOT DEFINED)

The ER MASK field indicates which ER numbers the rejecting subarea can use to send data back to the host that requested the test. (These explicit routes are called *reverse explicit routes.*) For example, an ER MASK field of hex 88 (binary 10001000) means that the "0" and "4" bits of the mask are turned on, so ER numbers 0 and 4 can be used to send data back to the host that requested the test. If message IST573I or 5F73I says an explicit route is "NOT REVERSIBLE", that means the rejecting subarea does not have the reverse explicit routes in its path definitions.

It is possible for the test results for an explicit route to be lost before they are displayed. For example, if a node or link along the explicit route fails between the time VTAM sends the test request and the time VTAM receives the test results, the results will be lost. If this occurs, re-enter the DISPLAY ROUTE command for that explicit route.

See *VTAM Messages and Codes* for complete explanations of the messages resulting from the DISPLAY ROUTE command.

Modify Input/Output Problem Determination

The input/output problem determination (IOPD) facility detects pending I/O operations for the SSCP, PUS, NOS, and LUS components of VTAM, and reports those operations to the VTAM operator. These operations are represented by a waiting request element (WRE) queued to an I/O locked queue anchor block (LQAB).

The IOPD facility searches the I/O LQAB queues at a user-defined time interval, specified with the IOINT option. The IOPD facility then writes message pair IST530I/IST532I (or 5F30I/5F32I, for VTAM V3R1 or V3R1.2 running under VSE) for each operation that has been pending longer than the specified time interval.

The message pair includes a 10-character I/O operation identifier (usually an RU name or operator command), the names of the origin and destination nodes, the name of the node for which I/O is pending, and the event ID associated with the I/O operation.

For more information on event IDS, and for a complete description of message pair IST530I/IST532I (or 5F30I/5F32I), see *VTAM Messages and Codes.*

These messages are only an indication that a problem *may* exist. The longer an operation remains pending (for example, the more a message is written for the same request unit), the more likely it is that a problem exists. See "Wait Procedure" on page 29 for more information on identifying pending I/O problems.

To use the IOPD facility, use the MODIFY IOPD command or the IOINT start option. The MODIFY IOPD command is described in *VTAM Operation.* The IOINT start option is described in *VTAM Installation and Resource Definition.*

Note: During initialization of a large VTAM network, you may see more pending I/O operations than usual. If you are using the IOPD facility to track I/O problems during initialization, the number of message pairs written can degrade your network's performance.

You can use the RACNTWRE field of ISTRACON (the VTAM constants module) to instruct the IOPD facility to write just one message pair for each *type* of pending I/O operation, rather than one pair for *each* operation. The resulting reduction in the number of messages written can improve your network's performance during initialization.

See *VTAM Customization* for a complete description of RACNTWRE and for information on using ISTRACON.

Modify Message Module Identification

You can choose to include in VTAM messages the last five characters of the VTAM module that issued the message. The module name abbreviation appears between the message ID and the beginning of the message text. To insert or delete module name abbreviations, use the MODIFY MSGMOD command or use the MSGMOD start option.

Specifying MSGMOD=NO stops the insertion of module IDS into messages.

MSGMOD=NO is the default; the issuing module name does not appear in messages unless MSGMOD=YES is specified in either the MODIFY command or the MSGMOD start option.

For more information on the MODIFY MSGMOD command, see *VTAM Operation*. For more information on the MSGMOD START OPTION, see *VTAM Installation and Resource Definition*.

For example, VTAM message xxxxx (where xxxxx is an operating system-unique message number) normally appears as:

```
xxxxx   DUMP OF ncpname COMPLETE
```

If MSGMOD=YES were specified in either the MODIFY command or start option, the message would appear as:

```
xxxxx   INFXI   DUMP OF ncpname COMPLETE
```

Notes:

1. Any message that exceeds the maximum message length after the insertion of the module ID will be truncated.
2. If your installation has changed the message text and omitted the message ID, the module name will be the first item in the message.

Modify NCP Intensive Mode Recording

When you receive a large number of temporary errors for a line, but not enough to create a permanent error, you can use intensive mode recording to find the cause of the error.

Intensive mode error recording allows each temporary error over a link (and its cause) to be recorded and signaled to VTAM.

Start intensive mode recording with the MODIFY IMR command. For more information on this command, see *VTAM Operation*.

The MODIFY IMR command causes the owning SSCP to send a request to the NCP. The NCP then builds and sends RECMS RUS to the SSCP each time an error occurs. The SSCP writes these error records on the LOGREC or SYSREC file. It may also optionally pass them to a user-defined communication network management (CNM) application program.

Modify SDLC Link Level 2 Test

Use the SDLC link level 2 test to test an SDLC link between an NCP and a physical unit attached on a multipoint line, between two NCPS, or between VTAM and any secondary physical unit attached via ICA SDLC lines (VM and VSE only). By running this test over an extended time, you can increase the possibility of repeating an intermittent error that is hard to re-create.

Note: If you just want to test connectivity to a physical unit, activate the physical unit instead of using a link level 2 test.

To test the link, VTAM sends test data over the link from the controlling NCP to the remote station (NCP or peripheral physical unit). The data is then echoed back to the sending NCP. This NCP then compares the data received with the data sent and forwards the results to VTAM.

Notes:

1. For ICA-attached devices, VTAM sends test data over the ICA attachment to the physical unit. The data is then echoed back to VTAM.
2. VTAM may also receive an LL2 request from an ICA-attached NCP, in which case VTAM echoes the data back to the NCP.

When VTAM receives the test results, it sends message IST549I (or 5F49I, for VTAM V3R1 or V3R1.2 running under VSE) to the initiating console to indicate how the link level 2 test ended: data returned without errors, data returned with errors, or no data returned because of an inoperative link or initialization error. See *VTAM Messages and Codes* for a complete message description.

When testing the link between an NCP and a peripheral physical unit, the physical unit name in the ID operand must be inactive ("dedicated" to the test). Other physical units on the same link, however, may remain active.

When testing the link between two NCPs, the primary link station is the one named in the LL2 command, and the secondary link station is the one responding to the test. You must start the test from the primary link station. The primary link station must be inactive; the secondary link station must be active; and the link must be active.

Start this test with the MODIFY LL2 command. See *VTAM Operation* for more information about the MODIFY LL2 command.

This command causes the SSCP to send a test RU to the NCP to which the test terminal is connected. The NCP returns test results to the requesting SSCP in a Record Test Results RU.

The test results include:

- The number of test frames transmitted by *nodename*
- The number of test frames received by *nodename*
- The number of test frames received without error by *nodename* (that is, successfully returned by the test station).

If the above three numbers are not the same, that indicates an error in the link or a physical unit. Using a line trace might help you further isolate the problem.

Modify Tuning Statistics

To record tuning statistics for VTAM, issue the MODIFY TNSTAT command or use the TNSTAT start option.

For detailed information about recording tuning statistics and using these statistics to improve communication between VTAM and a channel-attached communication controller or host processor, see *VTAM Customization.*

Dumps

The following dumps may be used for problem determination:

- "Network Control Program (NCP) Dump" on page 274.
- "Communication Scanner Processor (CSP) Dump (3720, 3725, and 3745 Only)" on page 277.
- "Maintenance and Operator Subsystem (MOSS) Dump (3720, 3725, and 3745 Only)" on page 277.
- MVS Dumps
  - "MVS Abend Dump" on page 284
  - "MVS SNAP Dump" on page 284
  - "MVS Stand-Alone Dump" on page 284
  - "MVS SVC Dump" on page 284.
- VM Dumps
  - "VM SDUMP Dump" on page 286
  - "VM GDUMP Dump" on page 286
  - "VM DUMP Command" on page 286
  - "VM VMDUMP Command" on page 286.
- VSE Dumps
  - "VSE Program-Initiated Dump" on page 287
  - "VSE SDAID Dump Facility" on page 287
  - "VSE Stand-Alone Dump Utility" on page 287
  - "VSE DUMP Command" on page 287.

Network Control Program (NCP) Dump

You can get either dynamic or static dumps of NCP storage in a communication controller.

You can use the DISPLAY NCPSTOR command (described on page 263) to dynamically dump up to 256 bytes of NCP storage. When you request a dynamic dump, VTAM sends repeated DISPSTOR RUS to the NCP until the entire contents of NCP storage have been transmitted. The NCP continues to operate during this period. As a result, the dump represents NCP storage over a period of time.

You can get a static dump with the VTAM dump facility (for a channel-attached or link-attached communication controller). You can also get a static dump with the independent dump utility (for a channel-attached communication controller only). Figure 60 on page 275 summarizes the methods and requirements for dumping the NCP.

MVS NCP dumps are not allowed on lines or data links (including multipoint subarea links) for which IPL=NO was specified in the NCP definition deck. Therefore:

- Code IPL=YES for any line or link which might be used to IPL or dump any attached type 4 physical unit.
- If IPL=YES was not specified for a line with one or more type 4 physical units attached, do not try to load or dump over that line.

Note: If you do not specify a value for IPL, the default is IPL=NO.

If you have a 3720 or 3745 Communication Controller with a hard disk in your network, you can transfer NCP, MOSS, and CSP dumps from the communication controller disk to the host. You can also transfer the NCP load module from the host to the disk. If you need to restart the NCP, you can load the NCP load module from the disk.

Specifically, the following tasks are allowed with the 3720 or 3745 Communication Controller with hard disk:

- You can save the NCP load module to the disk.
- You can load the NCP load module from the disk.
- You can transfer an NCP, MOSS, or CSP dump stored on the disk to the host.
- You can purge an NCP, MOSS, or CSP dump stored on the disk.
- You can control automatic loading and dumping of the NCP to or from the disk.
- You can display up to 256 bytes of NCP dump or state vector stored on the disk.

When to Use the NCP Dump

A dump of the NCP should be taken whenever the NCP abnormally terminates or when an error is suspected in the NCP. It may be possible to determine that a problem exists in the NCP by using the VTAM I/O trace to determine what PIUS are being sent to and received from the communication controller and by using the NCP line trace to determine what is happening on the lines between the communication controller and the link-attached logical unit.

Using the VTAM Dump Facility

To dump the NCP, use the MODIFY DUMP command with TYPE=NCP specified. For more information on this command, see *VTAM Operation*. You must execute a separate job to print the dump.

Note: A data host cannot load or dump an NCP.

You can use the DUMPDS operand of the MODIFY DUMP command to specify the file into which you want the dump transferred. If you omit this operand, VTAM uses the file specified on the PCCU definition statement for the NCP.

Note: If you use the same file on the MODIFY DUMP command as was named on the PCCU definition for the NCP, an earlier NCP dump may be overwritten.

If you omit the DUMPSTA operand, VTAM uses the link station specified on the VARY ACT command or the PCCU definition for the NCP, in that order.

MVS VM If you are using SNA network interconnection, and you want a gateway NCP to perform a dump, you can set its link station name on the DUMPSTA operand only if that NCP is in the same network as the host processor requesting the dump.

If you set AUTODUMP=YES on the PCCU definition statement, a dump is taken automatically if the NCP abnormally terminates. This dump is written to the dump file named on the DUMPDS option of the PCCU definition statement. If AUTOIPL=YES is specified on the PCCU definition, the NCP is automatically reloaded after the dump is taken. With the 3720 or 3745 Communication Controller with hard disk, you can specify DUMPLOAD=YES on the VARY ACT command, which automatically stores the dump on the disk and loads the NCP load module from the disk. (For more information about dump commands, see *VTAM Operations*.)

Note: For the 3705 only, the NCP is partially overwritten in the communication controller storage when the dump is taken. If the NCP is running with the partition emulation program (PEP), the emulation routines are included in the dump and are also partially overwritten. The 3705 must be reactivated after the dump is taken.

Method of Starting NCP Dump	Channel or Link Attached	Requirements	NCP Status	Printing Dump
MODIFY DUMP command, OPTION=DYNA	Both	DUMPDS must be specified	Active throughout dump, no reactivation required.	Execute SSP dump formatting program IFLDUMP (MVS and VM) or IFUDUMP (VSE)
MODIFY DUMP command, OPTION=STATIC	Both	DUMPDS must be specified	Deactivated when dump is completed; operator must reactivate the NCP.	Execute SSP dump formatting program IFLDUMP (MVS and VM) or IFUDUMP (VSE)

Figure 60 (Part 1 of 2). Dumping the NCP

Method of Starting NCP Dump	Channel or Link Attached	Requirements	NCP Status	Printing Dump
VTAM Error Recovery Procedures (automatic)	Both	PCCU definition requirements: DUMPDS must be specified AUTODMP=YES must be specified AUTOIPL may be specified	Activated after dump is completed if AUTOIPL=YES and restart is successful, or if AUTOIPL=NO and operator requests communication controller IPL.	Execute SSP dump formatting program IFLDUMP (MVS and VM) or IFUDUMP (VSE)
VTAM Error Recovery Procedures (with operator intervention)	Both	PCCU definition requirements: DUMPDS must be specified AUTODMP=NO must be specified AUTOIPL may be specified	Activated after dump is completed if AUTOIPL=YES and restart is successful, or if AUTOIPL=NO and operator requests communication controller IPL.	Execute SSP dump formatting program IFLDUMP (MVS and VM) or IFUDUMP (VSE)
SSP independent dump utility: IFLREAD (MVS and VM) or IFUREAD (VSE)	Channel attached only	See the *NCP and SSP Generation and Loading Guide* for information about using independent dump utility.	Deallocate communication controller from VTAM and allocate to the independent dump utility	Done automatically by IFxREAD using IFxDUMP.
Controller-detected error (3720 or 3745 with disk only)	Both	VARY ACT requirements: DUMPLOAD=YES must be specified Prior dump must have been transferred to host	Activated after dump is completed if AUTOIPL=YES and restart is successful, or if AUTOIPL=NO and operator requests communication controller IPL.	Execute SSP dump formatting program IFLDUMP (MVS and VM) or IFUDUMP (VSE). Transfer to host using MODIFY DUMP with ACTION=TRANSFER.

Figure 60 (Part 2 of 2). Dumping the NCP

Using the Independent Dump Utility (Channel-Attached Controller Only)

To use the independent NCP dump, the communication controller must be inactive. For a description of the job control language needed to invoke the independent dump utility, see the *NCP, SSP, and EP Diagnosis Guide*. NCP data areas, registers, and codes found in an NCP storage dump are explained in the *NCP and EP Reference Summary and Data Areas*. For more information on the contents of NCP storage, see the *Principles of Operation* manual for your communication controller.

Note: To use this utility in VSE, a communication controller must be assigned to a programmer logical unit for the partition in which it is to run. When the dump is complete, you must take away the assignment to make the device available to VTAM again.

Communication Scanner Processor (CSP) Dump (3720, 3725, and 3745 Only)

The communication scanner processor (CSP) automatically dumps its contents when it detects an error. CSP stores the dump on the communication controller disk, or on the MOSS diskette, and the NCP sends an alert message to the host to inform it of the error. You can use the MODIFY DUMP command to transfer this dump to a dump file in the host processor.

If you have a 3720 or 3745 Communication Controller with hard disk in your network, you can also purge the CSP dump from the communication controller before sending the dump to the host.

To transfer the contents of the CSP dump, use the MODIFY DUMP command with TYPE=CSP specified. You can use the DUMPDS operand of the MODIFY DUMP command to specify the file into which you want the dump transferred. If you omit this operand, the dump is put into one of two dump files:

- The file specified by the CDUMPDS operand of the PCCU definition statement for the NCP
- The file specified by the DUMPDS operand of the PCCU definition statement for the NCP (if CDUMPDS was not specified on the PCCU definition).

After the dump has been transferred to the host, you must then run a separate job to print the dump.

For more information on the MODIFY DUMP command, see *VTAM Operation*.

Maintenance and Operator Subsystem (MOSS) Dump (3720, 3725, and 3745 Only)

The maintenance and operator subsystem (MOSS) automatically dumps its contents when it detects an error. MOSS stores the dump on the communication controller disk, or on the MOSS diskette, and the NCP sends an alert message to the host to inform it of the error. You can use the MODIFY DUMP command to transfer this dump to a dump file in the host processor. After the dump has been transferred to the host, you must then run a separate job to print the dump.

To transfer the contents of the MOSS dump, use the MODIFY DUMP command with TYPE=MOSS specified. You can use the DUMPDS operand of the MODIFY DUMP command to specify the file into which you want the dump transferred. If you omit this operand, the dump is put into one of two dump files:

- The file specified by the MDUMPDS operand of the PCCU definition statement for the NCP
- The file specified by the DUMPDS operand of the PCCU definition statement for the NCP (if MDUMPDS was not specified on the PCCU definition).

For more information on the MODIFY DUMP command, see *VTAM Operation*.

MVS VM Formatting and Printing Dump Output

The service aids described in this section are available for formatting and printing dump output.

MVS ABDUMP Service Aid

ABDUMP operates as part of the operating system's abnormal termination (abend) procedure. It automatically formats and prints abend dumps and SNAP dumps. (See "MVS Abend Dump" on page 284 and "MVS SNAP Dump" on page 284.)

During ABDUMP processing, VTAM formats control blocks related to the abending task and prints them as part of the dump created by ABDUMP.

ABDUMP formats the control blocks listed below.

Note: This list is in alphabetical order; the control blocks may be ordered differently in a dump.

Control Block	Description
ACDEB	VTAM data extent block for the abending task
APPCB	LU 6.2 control block
COPR	Control operator control block associated with the abending task
CRA	Component recovery area for the abending task
FMCB	Function management control block and extensions for the abending task
HSICB	Half session information control block for the abending task
LUCB	Logical unit control block associated with the abending task
MPST	Memory process scheduling table for the abending task
NSICB	Logical network services information control block for the abending task
NSSCB	Logical network services storage control block for the abending task
PST	Process scheduling table for the abending task
RAB	LU 6.2 resource allocation block for the abending task
RDTE	Resource definition table application program entry for the abending task
SAB	Logical resource manager session allocation block for the abending task

The following information appears for each control block:

- A header line with the name and hexadecimal address of the beginning of the control block.
- Under the header, the name of each selected field (as it appears in that control block's mapping DSECT) and the contents of the field (listed sequentially).
- After the formatted printout, a hexadecimal dump of the entire control block.

Note: If the VTAM formatted dump routine cannot access a field (either in the control block or in the chain of pointers to the control block), a note of the condition is made on the dump output.

See *VTAM Data Areas* for a description of the fields in each formatted VTAM control block.

VM PRTDUMP Service Aid

You can format VTAM control blocks from GDUMP output, as follows:

1. Load the dump with IPCSDUMP and use the PRTDUMP command to format it or print it. (Figure 29 on page 157 shows an example of using PRTDUMP.) VTAM asks whether you want VTAM control blocks to be formatted.
2. Answer yes to format the dump. VTAM then asks about VSCS formatting.

   Note: Answering yes provides only the formatted portion of the dump.
3. Answer no to print the unformatted dump.
4. Enter the formatting options you want for VSCS. VTAM then asks about VTAM formatting.
5. Enter the formatting options you want for VTAM. (The formatting options are the same as for VTAMMAP; see page 282). If you request formatting, the VTAM control blocks listed below may be formatted, depending on which virtual machine was dumped, and which options you selected.

   Note: This list is in alphabetical order; the control blocks may be ordered differently in a dump.

Control Block	Description
ACDEB	VTAM data extent block
APPCB	LU 6.2 control block
ATCVT	VTAM communication vector table
BPCB	Buffer pool control block
BPDTY	Buffer pool directory
CONFT	VTAM configuration table
COPR	Control operator control block
CRA	Component recovery area (active)
ERTE	Explicit route table entry
FMCB	Function management control block (chained to FMCB extensions)
FMCBE	FMCB extensions (queued to LUCB)
FMCBEXT	Function management control block extension
HSCIB	Half session information control block
ITTRC	Internal trace table
LUCB	Logical unit control block (chained to ACDEB)
MPST	Memory process scheduling table
NCB	Node control blocks (pointed to by RDTE)

NSICB	Logical network services information control block
NSSCB	Logical network services storage control block
PAGTB	Storage pool page table
PST	Process scheduling table
PTE	Storage pool page table entry
PXB	Pool expansion blocks (chained to BPCB)
QAB	Queue anchor block for the RDT
RAB	LU 6.2 resource allocation block
RDTE	Resource definition table entry
SAB	Logical resource manager session allocation block
SIB	Session information block
SIBIX	Session information block initiation extension
SIBRX	Session information block resource extension
SIBX	Session information block cross-network extension
SPANC	Storage pool anchor block
VRBLK	Virtual route block
WRE with EIDS	Waiting request element with event IDS

The following information appears for each control block:

- A header line with the name and hexadecimal address of the beginning of the control block.
- Under the header, the name of each selected field (as it appears in that control block's mapping DSECT) and the contents of the field (listed sequentially).
- After the formatted printout, a hexadecimal dump of the entire control block.

Note: If the VTAM formatted dump routine cannot access a field (either in the control block or in the chain of pointers to the control block), a note of the condition is made on the dump output.

See *VTAM Data Areas* for a description of the fields in each formatted VTAM control block. See "VM GDUMP Dump" on page 286 for more information on this dump.

MVS PRDMP Service Aid

PRDMP formats and prints SVC dumps and high-speed stand-alone dumps. (High-speed stand-alone dumps are not ordinarily formatted, but they can be formatted with this service aid.) Use PRDMP and supply the VTAMMAP control card in the SYSIN data. (The VTAMMAP options are shown below.) Use the SUMDUMP option on the PRDMP command if you want to print the summary information contained in the dump. During PRDMP processing, VTAM formats and prints certain VTAM control blocks and the VTAM internal trace table. See the diagnostic manuals for your operating system for more information on PRDMP.

You can choose one or more of the following options on the VTAMMAP control card. The default is ALL.

Option Specified **PRDMP Action**

ALL
Formats the VTBASIC, RDTFULL, VTAM, STORAGE, ROUTES, and SES options.

RDTFULL(*name*)
Formats all resource definition table entries (RDTES) and node control blocks (NCBS). If *name* is specified, formats only the RDTE specified by *name*.

RDTHIER(*name*)
Formats the specified RDTE and all RDTES below it in the RDTE hierarchy. If *name* is omitted, formats all RDTES.

RDTSUM(*name*)
Formats the specified RDTE in short form (one line for each RDTE). If *name* is omitted, formats all RDTES in short form.

ROUTES
Formats explicit route table entries (ERTES), virtual route block (VRBLK).

SES(*name*)
Formats the RDTE specified by *name*, and formats all SIBS and RDTES in session with *name*. The specified *name* can be any session endpoint, such as a logical unit, terminal, or application program. It also formats ACDEBS, APPCBS, COPRS, FMCBS, FMCBEXTS, HSICBS, LUCBS, NSICBS, NSSCBS, RABS, SABS, SIBIXS, SIBRXS, and SIBXS.

If *name* is not specified, it formats all SIBS and RDTES, as well as ACDEBS, APPCBS, COPRS, FMCBS, FMCBEXTS, HSICBS, LUCBS, MPSTS, NSICBS, NSSCBS, PSTS, RABS, SABS, SIBIXS, SIBRXS, and SIBXS.

STORAGE
Formats BPCBS, PXBS, SPANCS, PAGTBS, and PTES.

VTAM
Formats the following:

- RDT and RDTES
- Memory process scheduling table (MPST) and process scheduling table (PST).
- ACDEBS, APPCBS, COPRS, FMCBS, FMCBEXTS, HSICBS, LUCBS, NSICBS, NSSCBS, RABS, and SABS.
- NCBS (LDNCB, ICNCB and XCNCB).
- Buffer pool control blocks (BPCBS), extension blocks (PXBS), storage pool anchor block (SPANC), storage pool page table (PAGTB), and storage pool page table entries (PTES).
- Locked queue anchor block (LQAB).
- Waiting request elements (WRES) and event identifiers (EIDS).
- Modules from the ATCVT, in the form *modname address modname address*, sorted by module name.

VTBASIC
Formats the ATCVT, the configuration table (CONFT), the VTAM internal trace (VIT), and the component recovery areas (CRAS).

For example, to format the VIT, ATCVT, CONFT, CRA, RDT, and RDTES, specify the VTBASIC and RDTFULL options, like this:

```
VTAMMAP VTBASIC,RDTFULL
```

PRTDUMP formats the same control blocks as those listed in " VM PRTDUMP Service Aid" on page 280.

See " MVS Stand-Alone Dump" on page 284 and " MVS SVC Dump" on page 284 for more information on these dumps.

MVS SADMP Service Aid

SADMP formats and prints low-speed stand-alone dumps. During SADMP processing, VTAM formats selected control blocks and prints them as part of the dump created by SADMP.

See the diagnostic manuals for your operating system for more information on SADMP. See " MVS Stand-Alone Dump" on page 284 for more information on this dump.

VM SDUMP Service Aid

You can format VTAM control blocks for SDUMP output. The procedure is the same as that given in " VM PRTDUMP Service Aid" on page 280.

With SDUMP, the same VTAM control blocks as those listed in " MVS ABDUMP Service Aid" on page 279 may be formatted, depending on which virtual machine was dumped.

See " VM SDUMP Dump" on page 286 for more information on this dump.

MVS Dumps

Several kinds of dumps can be produced in an MVS system, depending on the type of failure and operator action:

- "MVS Abend Dump"
- "MVS SNAP Dump"
- "MVS Stand-Alone Dump"
- "MVS SVC Dump."

"MVS VM Formatting and Printing Dump Output" on page 279 tells you how to format and print the output.

MVS Abend Dump

If the appropriate DD card exists, an abend dump is produced when:

- The operator enters a CANCEL command
- An abend macroinstruction is issued
- A job abnormally ends.

To get an abend dump, the input stream for VTAM must contain a DD statement with ddname SYSABEND. The resulting dump is written to the data set specified on the SYSUDUMP or SYSABEND DD card. The contents of the dump depend on user specifications. For more information, see the *Supervisor* manual for your operating system.

MVS SNAP Dump

To get a SNAP dump, issue a SNAP macroinstruction. The resulting dump is written to the data set specified on the DCB operand of the SNAP macroinstruction. The dump data set must be opened before SNAP is issued. A separate DD statement is required for each job step in which SNAP is issued. The ddname can be any unique name except SYSABEND, SYSUDUMP, or SYSMDUMP. The contents of the SNAP dump vary depending on the options you specify in the SNAP macroinstruction. The SDATA=ALL or SDATA=CB options include major operating-system and selected VTAM control blocks in the dump. For more information, see the *Supervisor Services and Macros* manual for your operating system.

MVS Stand-Alone Dump

The stand-alone dump is produced when the operator invokes the stand-alone dump program. This program can be invoked when the operating system is in a disabled wait state or looping. The stand-alone dump may be a high-speed dump, which is not formatted, or a low-speed dump, which is formatted. The dump output is written to the tape or printer (low-speed only) specified on the output operands. The output for a high-speed dump can be formatted by PRDMP. For more information on the stand-alone dump, see *MVS Service Aids*.

MVS SVC Dump

SVC dumps are produced under these conditions:

- VTAM produces an SVC dump automatically when a program exception occurs. VTAM may or may not be terminated as part of this process. The system log indicates the location of the dump output and whether the dump was successful.
- An operator may request a dump with the operating system DUMP command. This will not cause VTAM to terminate.

- An operator uses a SLIP command with ACTION=SVCD specified, and an event occurs that matches the trap indicated in the SLIP.
- A macroinstruction issues an abend, and there is a DD statement with ddname=SYSMDUMP.
- An SDUMP macroinstruction is issued.
- System recovery routines produce an SVC dump if VTAM causes an error, such as a program exception or abend.

An SVC dump is written to a SYS1.DUMPnn data set (if allocated), the SYSMDUMP output data set, or the data set specified on the DCB operand of the SDUMP macroinstruction. An SVC dump may be printed with the PRDMP system utility. For more information on SVC dump, see the *Supervisor* manual for your operating system.

VM Dumps

Several kinds of dumps can be produced in a VM system, depending on the type of failure and operator action:

- "VM SDUMP Dump"
- "VM GDUMP Dump"
- "VM DUMP Command."
- "VM VMDUMP Command."

For dumping CP storage, see the *VM/SP Operator's Guide*.

"MVS VM Formatting and Printing Dump Output" on page 279 tells you how to format and print the output.

VM SDUMP Dump

An SDUMP dump is produced automatically when a machine program check or GCS internal error occurs. The resulting dump can be formatted and printed with the PRTDUMP command of IPCS. The VTAM control blocks shown in "MVS ABDUMP Service Aid" on page 279 are formatted in an SDUMP.

VM GDUMP Dump

The GDUMP command can be entered by the VM system operator. The resulting dump can be formatted and printed with the PRTDUMP command of IPCS. The VTAM control blocks shown in "VM PRTDUMP Service Aid" on page 280 are formatted in a GDUMP.

Note: For a loop problem, do not use the GDUMP command. Instead, use the VMDUMP command.

VM DUMP Command

The DUMP command can be entered by the VM system operator. This command produces a formatted hardcopy dump.

VM VMDUMP Command

The VMDUMP command can be entered by the VM system operator. The resulting dump can be formatted and printed with the PRTDUMP command of IPCS. The VTAM control blocks shown in "VM PRTDUMP Service Aid" on page 280 are formatted in a VMDUMP.

CP should have control when you issue this command. That is, you should either press PA1 or enter #CP before issuing the VMDUMP command.

VSE Dumps

Several kinds of dumps can be produced in a VSE environment, depending on the type of failure and operator action:

- "VSE Program-Initiated Dump"
- "VSE SDAID Dump Facility"
- "VSE Stand-Alone Dump Utility"
- "VSE DUMP Command."

Note: Dumps produced by canceling VTAM are of limited diagnostic value. Use other dump facilities to document a problem before canceling VTAM.

VSE Program-Initiated Dump

Customer programs can produce dumps by using system dump macroinstructions (PDUMP, JDUMP, DUMP). Other IBM licensed programs, such as CICS/VS, also have their own dumping facilities, which may be useful in isolating some VTAM problems.

VSE SDAID Dump Facility

The dump facility of SDAID is useful when debugging requires a dump of one or more specific storage areas on the occurrence of a specific event. The output device can be a tape drive or a printer. SDAID output on tape can be printed with DOSVSDMP. For more information on VSE dumps, including DOSVSDMP and SDAID, see *VSE Service Aids*.

VSE Stand-Alone Dump Utility

A stand-alone dump, created with the DOSVSDMP utility, is useful when the severity of a system malfunction (such as a hard wait state) prevents using other methods. The output of a stand-alone dump is printed by DOSVSDMP.

VSE DUMP Command

The DUMP command causes VSE to produce a dump of the storage area specified in the command. The operator can direct the dump either to tape or to a SYSLST device. To retrieve the dump information, you can do either of the following:

- Use DOSVSDMP to have the dump information printed on the printer assigned to SYSLST.

- Load the dumps recorded on the tape into your system's dump library and use Info/Analysis to retrieve the dump information.

See *VSE Service Aids* for more information about retrieving dump information from tape.

Other Service Aids

Alert Messages

The NCP in a 3720, 3725, or 3745 communication controller sends alert messages to VTAM whenever a serious or permanent error occurs in the communication controller. The NCP sends hardware error records to the maintenance operator subsystem (MOSS). If the MOSS determines that they are permanent errors, it sends them back to the NCP, which forwards them to all owning host processors.

If a communication network management (CNM) application program, such as the NetView program, is active and authorized to receive alert messages, VTAM forwards the alert messages to that program. Otherwise, VTAM sends a message to the operator's console.

Recording NMVT Alerts in LOGREC or SYSREC (V3R2)

A network management vector transport (NMVT) is a Systems Network Architecture (SNA) request unit (RU) that contains solicited or unsolicited data, such as line statistics and generic alerts. MVS VM LOGREC and VSE SYSREC are host data sets that contain records of various types of system failures, both hardware and software.

VTAM records all *unsolicited* NMVT alerts from 3745s, and *all* NMVT alerts from LANS and 9370s, in either MVS VM LOGREC or VSE SYSREC. If you have the NetView program, VTAM also forwards the NMVT alerts to the NetView hardware monitor for recording.

VTAM identifies and records an NMVT alert as follows:

- Each NMVT has an SNA network services (NS) header of X'41038D'.
- Each NMVT that is an alert has a management services (MS) major vector of X'0000'.
- To determine the type of device that sent the NMVT alert, VTAM checks the Product ID subvector (X'11') of the NMVT for the hardware machine type. Each type of device has its own unique machine type; for example, each NMVT alert that comes from a 3745 has a machine type of C'3745'.
- VTAM records the NMVT alert in LOGREC or SYSREC as a miscellaneous data record (MDR;type=X'91') with a device type of NMVT (X'30').

You can format and print NMVT alerts from LOGREC or SYSREC using the Environmental Record Editing and Printing program (EREP). For information on how to use the EREP program, see the *EREP* book or the *Service Aids* book for your operating system.

For more information on NMVTS, see the *NetView Hardware Problem Determination Reference*, SC30-3366.

VM VSE Alerts (V3R1.2 and Later)

VTAM V3R1.2 and later uses the alert function to notify an operator of an actual or impending loss of availability of a resource. VTAM builds a network management vector transport (NMVT) for the alert. The NMVT can be traced with the PIU option of the VIT. The PIU trace of the NMVT contains several extensions. Figure 61 on page 289 is an example PIU trace entry of an alert NMVT.

```
D7C9E440  43990000  00863DD8  40000000     *PIU .......Q ...*
20000000  00000001  00000001  1C000001     *................*
D7C9E4F2  00000001  00860B00  0041038D     *PIU2............*
00001000  10007B00  000A0401  80000000     *......#.........*
D7C9E4F2  01000C0A  01081057  061A0C0F     *PIU2............*
153C1000  2311040E  02F5F6F6  F4F2F8F0     *.........5664280*
D7C9E4F2  F0F1F3F1  F10804F0  F3F0F1F0     *PIU201311..03010*
F10A06C1  C3C661E5  E3C1D416  11011300     *1..ACF/VTAM.....*
D7C9E4F2  11F9F3F7  F5000000  0000F0F0     *PIU2.9375.....00*
F0F0F0F5  F60E5104  02000508  03400000     *00056........ . *
D7C9E4F2  00003105  52030404  0D910001     *PIU2............*
0F008000  C100C100  0004A108  4003A105     *....A.A.........
```

Figure 61. VM VSE PIU Trace Entry Example

The NMVT RU that is traced contains the alert (X'0000') management services major vector. The alert major vector and its subvectors are described in *SNA Formats* and in the *NetView Hardware Problem Determination Reference*.

In the example, the alert major vector has three subvectors that contain information to be used with "VM VSE Alert Description Codes." The subvectors are the Basic Alert subvector X'**91**' and two Detail Qualifier subvectors X'**A1**'. (There may be as many as three Detail Qualifier subvectors of the alert major vector in a single alert NMVT). The Basic Alert subvector has an alert description code of C1 and the Detail Qualifier subvectors have values of 0840 and 05.

VM VSE Alert Description Codes

The list that follows describes each alert by description code and provides the list of detail qualifier items that accompany the alert.

Code Failure

66 Open Failure: Token-Ring Lobe

Description: The adapter detected a problem on its lobe during the wrap-test portion of the insertion process. The insertion process did not complete.

Detail data:

1. Channel unit address
2. Command
3. Error code.

67 Open Failure: Token-Ring Fault Domain

Description: The adapter detected a beaconing condition on the ring during the insertion process. The insertion process did not complete.

Detail data:

1. Channel unit address
2. Command
3. Error code.

68 Open Failure: Token-Ring Duplicate Station Address

Description: The adapter detected the presence of a station with its address on the ring during the insertion process. The insertion process did not complete.

Detail data:

1. Channel unit address
2. Command
3. Error code.

69 Open Failure: Token-Ring Remove Command Received

Description: The adapter received a Remove Ring Station MAC frame during the insertion process. The insertion process did not complete.

Detail data:

1. Channel unit address
2. Command
3. Error code.

6A Open Failure: Token-Ring Lobe

Description: An error was detected during the insertion process that is not defined in the previous alerts. These conditions are not expected to occur, so they are included within one alert definition. The insertion process did not complete.

Detail data:

1. Channel unit address
2. Command
3. Error code.

6B Adapter Error: Local Token-Ring Adapter

Description: A hardware error in the reporting station's adapter has occurred. The adapter is now inoperable.

Detail data:

1. Channel unit address
2. Command
3. Error code.

6C Wire Fault: Token-Ring Lobe

Description: The reporting station's adapter detected a wire-fault condition on the ring.

Detail data:

1. Channel unit address
2. Command
3. Error code.

6D **Auto Removal: Token-Ring Lobe**

Description: The reporting station's adapter has left the ring as part of the beacon automatic-recovery process. That is, the reporting station's adapter was a member of the beacon fault domain. The station removed itself from the token ring and ran a self test, which was unsuccessful.

Detail data:

1. Channel unit address
2. Command
3. Error code.

6E **Remove Command Received: Network Operator**

Description: The reporting station's adapter received a Remove Adapter command from a LAN manager and, as a result, left the ring.

Detail data:

1. Channel unit address
2. Command
3. Error code.

70 **Token-Ring Inoperative: Token-Ring Fault Domain**

Description: The ring has been beaconing for more than 52 seconds.

Detail data:

1. Channel unit address
2. Command
3. Error code.

72 **Token-Ring Temporary Error: Token-Ring Fault Domain**

Description: The ring was in a beaconing state for less than 52 seconds and then recovered. The sender of this alert does not know whether a station removed itself from the token ring to bypass the fault, or whether the fault was temporary.

Detail data:

1. Channel unit address
2. Command
3. Error code.

83 **Link Error: LAN LLC Communications/Remote Node**

Description: The remote link station does not respond. The response timer (t1) has expired, causing the remote station to be polled. The retry count (n2) for the number of polls has been exceeded.

Detail data:

1. Channel unit address
2. Command
3. Error code.

84 Link Error: LAN LLC Communications

Description: The remote link station sent a Disconnect Mode (DM) to the local link station.

Detail data:

1. Channel unit address
2. Command
3. Error code.

87 Software Program Error: LAN LLC Communications

Description: The remote link station sent an SABME (Set Asynchronous Balanced Mode-Extended) to the local link that had been previously initialized through an SABME-UA (Unnumbered Acknowledgement) exchange.

Detail data:

1. Channel unit address
2. Command
3. Error code.

A0 Software Program Error: LAN LLC Communications

Description: The local link station sent an invalid or unsupported frame to the remote link station. That resulted in the remote link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

A1 Software Program Error: LAN LLC Communications

Description: The local link station sent an I-field when not permitted. That resulted in the remote link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

A2 Software Program Error: LAN LLC Communications

Description: The local link station sent an invalid send sequence number. That resulted in the remote link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

A3 Software Program Error: LAN LLC Communications

Description: The local link station sent an I-field that was too long. That resulted in the remote link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

AA Software Program Error: LAN LLC Communications

Description: The remote link station sent an invalid or unsupported frame to the local link station. That resulted in the local link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

AB Software Program Error: LAN LLC Communications

Description: The remote link station sent an I-field when not permitted. That resulted in the local link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

AC Software Program Error: LAN LLC Communications

Description: The remote link station sent an invalid receive sequence number. That resulted in the local link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

AD Software Program Error: LAN LLC Communications

Description: The remote link station sent an I-field that was too long. That resulted in the local link station returning a frame reject.

Detail data:

1. Channel unit address
2. Command
3. Error code.

BE Configuration or Customization Error: Local System Operator

Description: The local system operator has made an error in the VTAM definition. One of the following situations has occurred:

- A physical unit has dialed in, and there are no more available lines defined in the LAN major node.
- The activation of a line in a LAN major node has caused the total number of active lines to exceed the maximum stations coded for the LAN major node.
- A dial out has been attempted that will cause VTAM to exceed the maximum stations defined for the adapter. If more than one user is connected to the adapter, then the maximum number of stations must include the total number of stations for all connections.

Detail data:

1. Channel unit address
2. Reason code:

   04 Dial in line not available

   08 Maximum number of stations exceeded (line activation)

   09 Maximum number of stations exceeded (dial out).

BF Communications Subsystem Failure: Communications Subsystem Controller Interface

Description: One of the channel programs for the CETI group failed.

Detail data:

1. Channel unit address
2. Channel status
3. Sense.

C0 Communications Subsystem Failure: Communications Subsystem Controller Interface

Description: A command to the adapter has reported an unsuccessful completion or an asynchronous indication has been received from the adapter.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C1 Communications Subsystem Failure: Communications Subsystem Controller Interface

Description: Local area network support code detected an error. The error is not the result of a user definition.

Detail data:

1. Channel unit address
2. Reason code:

   01 MAXDATA segmenting error

   02 Sequence error

03 Invalid PIU

05 Adapter input is discarded

06 CETI architecture violation

07 Unexpected SABME or XID received.

C2 Configuration or Customization Error: Local System Operator

Description: The local system operator has made an error in the VTAM definition. An invalid local or remote node address was specified. Either the SAP address or the MAC address is invalid or not recognized or the station is not on the ring.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C3 Configuration or Customization Error: Local System Operator

Description: The local system operator has made an error in the VTAM definition. The maximum data size is too large or invalid.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C4 Configuration or Customization Error: Local System Operator

Description: The local system operator has made an error in the VTAM definition. The service access point (SAP) address on the PORT statement is already in use.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C5 Configuration or Customization Error: Local System Operator

Description: The local system operator has made an error in the VTAM definition. An invalid number of stations was specified on the PORT statement or the number of lines defined in the LAN major node is too large for the adapter.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C6 Warning: Single Station on the Ring

Description: The adapter has opened and it is the only station on the ring.

Detail data:

1. Channel unit address
2. Command
3. Error code.

C7 Configuration or Customization Error: Local System Operator

Description: The local system operator has made an error in the VTAM definition. The station that is being activated as a LAN resource is not defined to VTAM as LAN-capable.

Detail data:

1. Channel unit address.

VM VSE Description of Detail Qualifier Items

The following list describes the items in the detail qualifier data fields of an NMVT.

Channel unit address This is the channel unit address specified on the PORT statement that defines a CETI group.

Command This is the command that was processed when the alert was reported. Commands are used by VTAM LAN support to communicate with the adapter. See "**VM VSE** Command Values Reported in Trace Entries" for more about commands used by LAN support.

Error code This is the completion status code that is associated with the command.

Channel status This is the channel status that is reported for a failing CETI channel program.

Sense This is the sense that is reported for a failing CETI channel program. This value may be 0 if no sense data is available.

Reason code This is the reason code for the failure reported by the alerts with description codes of BE and C1. See the listing of those description codes for explanations of the values reported.

VM VSE Command Values Reported in Trace Entries

Figure 62 on page 298 describes the command values reported in trace entries. These are the commands used by VTAM LAN support and the Token-Ring Subsystem when they communicate.

Command Value	Description
810000	Set network parameters
040C0D 04CC0D	Activate VTAM's service access point
040C0E 04CC0E	Deactivate VTAM's service access point
040C0A 04CC0A	Cause the subsystem to send test frames for remote station address resolution and to establish a logical connection with a remote station
040C0B 04CC0B	Cause the subsystem to break a logical connection with a remote station
040C20 04CC20 044C20 048C20	Perform SNA exchange identification (XID)
040D00 04CD00 044D00 048D00	Perform SABME - UA exchange
040D01 04CD01 044D01	Perform DISC - UA/DM exchange
040D10 04CD10 044D10	Send and receive normal data
040D50 04CD50 044D50	Control the flow of data to and from a remote station
040660 04C660	Send and receive test frames to perform VTAM link level 2 (LL2) test
044D1A	Pass asynchronous status and statistics
024D1A	Pass asynchronous status and statistics

Figure 62. Command Value Descriptions

The commands in the table are in the form:

xxyxxx

where

xx_xxx is the command identifier and
y is the command type.

The command types for *y* and the direction of flow are:

y	Command Type	Direction of Flow
0	Request	From VTAM LAN support to the subsystem
4	Indication	From the subsystem to VTAM LAN support
8	Response to an indication	From VTAM LAN support to the subsystem
C	Confirmation of a request	From the subsystem to VTAM LAN support

Note: The set network parameters command flows as both a request and a confirmation. It is always shown as 810000 in the VTAM trace command fields.

MVS Messages Issued for 3745 Bus Switching (V3R2)

A 3745 can be configured with one or two CCUs. For twin-CCU 3745s, NCP V5R1 supports the two CCUs as independent communication controllers.

A twin-CCU 3745 can be operated such that, if one of the CCUs fails, the maintenance and operator subsystem (MOSS) switches the IOC buses from the failing CCU to the active CCU. The operator can then activate all or part of the resources of the failing CCU to the other CCU.

The bus-switch can also be performed at the operator's request.

If the bus-switch occurs in the middle of a VTAM channel operation, such as a Read or a Write, VTAM issues messages IST881I and IST882I.

MVS **Message IST881I:** Message IST881I tells the operator that VTAM either was unable to contact a link station, or lost contact to a link station.

This message is issued for one of the following reasons:

- A bus is being switched from one CCU to the other.
- A dump or load is being done on another channel.
- A dump or load is being done by a local disk.

When the link station becomes available (as indicated in other messages), VTAM resumes CONTACT processing.

To terminate CONTACT processing before the link station becomes available, issue a VARY INACT,F command to deactivate the link station.

MVS **Message IST882I:** Message IST882I tells the operator that VTAM is waiting for a device end from the link station identified in message IST881I.

You should check to see if the link station is online. If it is, then there is a possibility that NCP is being dumped or loaded over another channel adapter, and no further action is necessary.

Note: If the link station is not operating, not physically connected, or not online, VTAM never receives the device end. In those cases, you should issue a VARY INACT,F command to deactivate the link station.

Hardware Error Recording

During error recovery processing (ERP), VTAM writes outboard recorder (OBR) records and miscellaneous data records (MDRS) to LOGREC or SYSREC. OBR records are written for hardware errors on channel-attached devices. (**VSE VM** OBR records are written for ICA-attached devices as well.) MDRS are written for hardware errors on NCP-attached devices.

VTAM collects the error data and puts it in LOGREC or SYSREC. EREP formats and prints the LOGREC or SYSREC data set.

OBR records contain information about the following:

- Statistics on the overflow of error counters for devices
- Sense and status data on all channel-attached devices
- Failures on teleprocessing devices
- Temporary or intermittent failures on I/O devices
- End-of-day requests
- Permanent channel and device errors (unrecoverable errors and unit checks).

Permanent error records show the date, time, logical unit name, type of record, contents of counters, failing CCW, channel device name, CSW, sense information, device type, and flags. The time field shows the time at which the permanent error occurred.

Counter overflow and end-of-day records show the date, time, logical unit name, type of record (counter overflow or end-of-day), contents of counters, channel or unit address, and device type. The time field shows the time at which the counter overflow or end-of-day error occurred.

Counter overflow records are written when the SIO counter, the temporary error counter, or a device statistics table counter is about to overflow. (VTAM maintains two counters for each channel-attached device. One counter keeps track of the number of SIO commands issued to the device; the other counter keeps track of temporary errors. Counters of unit check errors by error type are also maintained in the device statistics table.)

End-of-day records are written whenever a VARY INACT command is entered for a link or channel.

Miscellaneous data records contain the following information:

- Statistics on the overflow of error counters for communication controllers
- Record Maintenance Statistics (RECMS) RUS
- Permanent errors on NCP-attached devices.

See the *EREP* manual for information on printing and interpreting these records.

Logical Unit Connection Test (IBMTEST)

You can enter the IBMTEST command from a terminal to find out whether that terminal can communicate with its owning SSCP. When you use the IBMTEST command, an unformatted RU is sent through the network path supporting the LU-SSCP session. This RU contains the IBMTEST command followed by the number of times the SSCP is to return (echo) the data to the logical unit and optional data (up to 247 bytes) being sent to the SSCP.

You can increase the possibility of repeating an intermittent error that is hard to re-create by using IBMTEST, since you can request up to 255 echoes. You can also use it to determine whether a suspended LU-LU session is caused by either a hardware problem or by a problem with VTAM or an application program.

Start this test with the following command:

IBMTEST [*n*][,*data*]

n

specifies the number of times the test data should be returned to the terminal. Specify *n* as a decimal number in the range 1 through 255. If no value is specified, a value of 10 is used by default.

data

specifies the test data to be returned. Specify a character string of up to 247 characters, or the maximum message length of the terminal, whichever is smaller. If no test data is supplied, VTAM returns the alphanumeric sequence:

'ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789'

The IBMTEST command is valid only for terminals that use the USS LOGON format. The IBMTEST command must be defined in the USS table for that terminal. The SSCPFM option for that terminal must specify something that is supported and starts with USS.

Note: This echo check does not verify that a terminal can establish a session with an application program in the host, because the session request may specify a different network path than the one supporting the terminal's LU-SSCP session. If the requested path is unavailable, the session request is rejected, even though another path is available.

If there are any errors, the VTAM operator receives a message that contains the logical unit name associated with the terminal, the number of echoes that took place before the I/O error, and the error sense code.

NCP Error Recording

NCP error recording procedures create Record Maintenance Statistics (RECMS) RUS that contain:

- The initial error status that began the recovery process
- The final error status that caused the permanent RECMS RU to be generated.

RECMS RUS are created for each adapter check, program check, unresolved interruption, counter overflow, I/O operation, and permanent line error. The RECMS RUS, which contain the miscellaneous data record (MDR), are sent to the VTAM host that owns the failing component. VTAM then records the MDR and writes the error information to the LOGREC or SYSREC file.

Patch Areas

Patch areas are available for VTAM, TSO/VTAM, and VSCS.

VTAM Patch Area

VTAM supplies a fixed patch area as a separate module. You can insert short service programs in this area to do maintenance-related functions. ISTPATCH is loaded into LPALIB during VTAM initialization and is pointed to by the ATCPTCHA field in the ATCVT. The initial size of the patch area is 64 bytes, but you can change the size by link-editing a module named ISTPATCH of the desired size.

Code the necessary change in the patch area. (ISTPATCH follows the same coding rules as other modules in LPALIB.) Then replace part of the failing code with a branch to the patch area, allowing you to bypass the failing code.

MVS TSO/VTAM Patch Area

TSO/VTAM maintains a patch area in each module. The size of the patch area varies from module to module. If you need more information on using these patch areas, contact the IBM Support Center.

VM VSCS Patch Area

VSCS supplies a fixed patch area as a separate module. You can insert short service programs in this area to do maintenance-related functions. DTIPATCH is addressable from all VSCS code modules and is pointed to in the VSCS internal trace table header.

Using a Dump to Follow the Execution Sequence of VTAM Components

VTAM traces the flow of the execution of three VTAM components, SSCP, PUS, and LUS, by saving the work areas of modules in these components. The addresses of the module work areas are stored in either of these control blocks:

- Network configuration services parameter list (NCSPL)
- Request/response unit processing element (RUPE).

In the RUPE, the work area address can be found at RUPEDAP. In the NCSPL, the work area address can be found at NCSPLWKA. For the hex offsets of these fields, see *VTAM Data Areas*.

The NCSPL or RUPE work area contains the work and save areas for each module invoked for the command that the NCSPL or RUPE represents. The module work and save areas provide status information that pertains to both the processing of that command and any interruptions in the processing.

This status information includes a record of which modules were entered, which modules returned to their callers, and which modules returned with a return code. Each module save area contains the 4th, 5th, 7th, and 8th characters of the module name and the register 15 value that includes a pointer to the last module called by this module.

The high-order byte of the register 15 save area also indicates the status of the last module called, as shown below. (MVS/XA In 31-bit mode, the address fills register 15, so that the status then overlays the high-order byte of the address in the register 15 save area.)

Byte Value	Status Indicated
FE	The called module has returned to this module without a return code.
FF	The called module has not returned to this module.
nn	The called module has returned to this module with a return code of nn.

Figure 63 on page 304 is an example of what the NCSPL or RUPE work area might contain for modules invoked for a VTAM process.

VTAM Error Recording for Communication Adapter Lines

VTAM error recording procedures create Record Maintenance Statistics (RECMS) RUs to record information about the following conditions:

- Recovered (temporary) SDLC link and station errors
- Recovered (temporary) BSC link and station errors
- Unrecoverable (permanent) SDLC link and station errors
- Unrecoverable (permanent) BSC link and station errors
- BSC 3270 sense/status data.

RECMS RUs are sent to a communication network management (CNM) application program, if one is designated to receive these RUs.

VSE VTAM also writes the RECMS records on SYSREC.

Formats of the RECMS RUs created by VTAM are mapped by the ISTRMD control block; ISTRMD is described in *VTAM Data Areas*.

Use EREP to edit and print the error data. (See the *EREP* manual.)

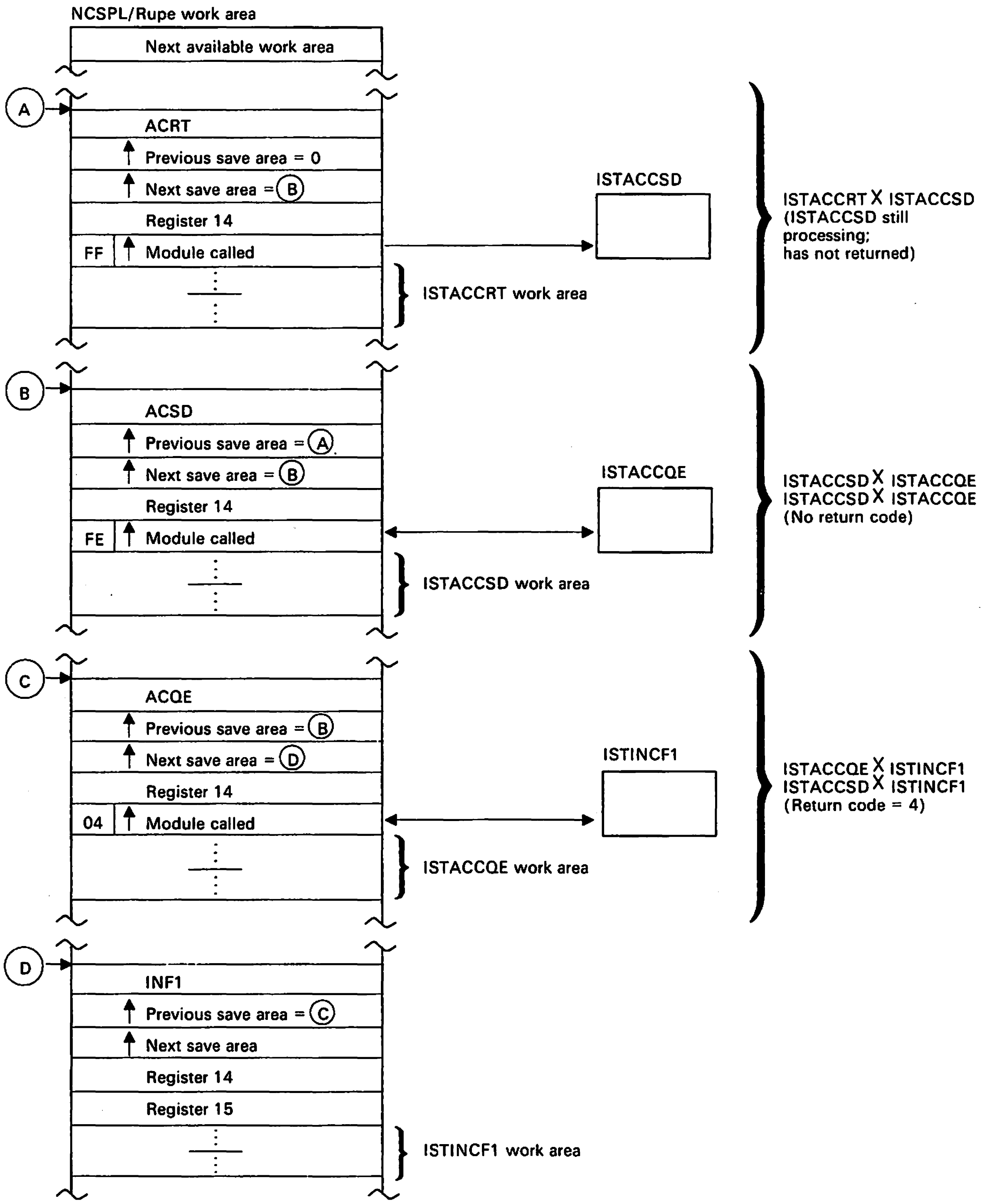


Figure 63. Save Area Conventions

Chapter 8. Using the VTAM Internal Trace

Chapter 8. Using the VTAM Internal Trace

Most VTAM traces show information flow between VTAM and other network components. However, the VTAM internal trace (VIT) provides a record of the sequence of events *within* VTAM. These internal events include the scheduling of processes (for example, POST. WAIT, and DISPATCH), the management of storage (for example, VTALLOC), and the flow of internal PIUS between VTAM components.

Note: VTAM disables the VIT when it issues SDUMP to prevent the VIT table from being overwritten.

Together with the operator console listing and a dump, output from the VIT can help you reconstruct sequences of VTAM events and find internal VTAM problems more easily.

Activating the Internal Trace

Use one of the following methods to start the VIT:

- You can use the TRACE start option, with TYPE=VTAM specified, to start the VIT when you first start VTAM. See *VTAM Installation and Resource Definition* for more information on how to use the TRACE start option.
- You can use the MODIFY TRACE command, with TYPE=VTAM specified, to start the VIT after you have started VTAM. See *VTAM Operation* for more information on how to use the MODIFY TRACE command.

Selecting Internal Trace Options

Both the TRACE start option and the MODIFY TRACE command have an **OPTION** operand you can use to select VIT options. You select one or more of these options to indicate the VTAM functions to be traced:

- The API option, for application program interfaces.

  This option helps you determine whether an application program is causing a problem. API entries are written for RPL macros, RPL exit routines, user exit routines, and user posts.

- The APPC option, for LU 6.2 application program interfaces.

  This option helps you determine whether an LU 6.2 application is causing a problem. LU 6.2 entries are written for APPCCMD macro invocations, user posts and exit scheduling by LU 6.2 code, and message unit transmissions between LU 6.2 components.

- The CIO option, for channel input and output.

  This option helps you isolate problems related to channel I/O. CIO entries are written for attentions, error recovery, interruptions, HALT I/O SVC, and START I/O SVC.

- The ESC option, for execution sequence control.

  This option helps you track in detail the flow of requests for a given process.

- The LOCK option, for locking and unlocking.

  This option helps you determine when VTAM modules get and release locks.

- The MSG option, for messages.

  This option helps you correlate other VIT entries with the console messages even if the console sheet is lost. MSG entries are written for all messages to the VTAM operator.

- The NRM option, for network resource management.

  This option helps you follow the services of the Network Resource Management component. These include the assignment of, references to, and the deletion of certain VTAM resources such as node names, network addresses, and control blocks. NRM entries are written for SRT macros issued by VTAM modules.

- The PIU option, for path information unit flows.

  This option, like the I/O and buffer contents traces, helps you isolate problems to hardware, to the NCP, or to VTAM. PIU entries are written for all I/O to and from VTAM, which can be an advantage over I/O and buffer contents traces.

- The PSS option, for process scheduling services.

  This option helps you track the flow of requests through VTAM. PSS entries are written for the VTAM macros that invoke and control PSS, scheduling and dispatching VTAM routines.

- The SMS option, for storage management services.

  This option helps you isolate problems caused by storage shortages. When used with the SSCP or PSS trace options, it can also help you isolate internal VTAM problems. SMS entries are written when SMS macros are used to request or free fixed- or variable-length buffers.

- The SSCP option, for system services control point request scheduling and response posting.

  This option helps you isolate a VTAM problem to a specific VTAM component or module. SSCP entries are written for the request/response units (RUs) sent between VTAM components.

The VIT always traces the following exception conditions, regardless of the options you specify:

Option	Exception conditions traced
API option	RE, UE, or UP entries with a non-zero RPL return code or feedback
APPC option	ACA and ACI entries when issuing the following commands: • SEND ERROR • DEALLOC ABND • REJECT Other entries with non-zero return codes (except RPLGRCSC)

Option	Exception conditions traced
PIU option	Entries for a response with a non-zero sense code All NRSP and DSCD entries
SMS option	Entries with non-zero return codes
SSCP option	Entries with non-zero return or sense codes All CPRC entries
(No option)	All SNAP entries.

Note: Even if you do not specify *any* options, the VIT still traces the above exception conditions, if it is active.

Figure 64 is a summarization of the VIT options and the records they create.

Note: The ABND and LOST trace records are not created by specific VIT options. They are created as a result of exception conditions.

API	APPC	CIO	ESC	LOCK	MSG	NRM	PIU	PSS	SMS	SSCP
AIx	ACAx	ADPx	ESC	LKEX	MSGx	RCEx	DSCx	ADSP	AREL	CCx
IOx	ACIx	ATTx		LKSH	OPEx	SRTx	PIUx	AXIT	FBLK	CIx
RE	ACPx	CONN		ULKA			NRSx	DISx	GBLK	COx
UEx	ACRx	DISC		UNLK				EXIT	QREQ	CPPx
UP	ACUx	ERPx						IRBx	RELS	CPRx
	MUx	HIOx						POST	REQS	CPWx
		INTx						QUEx	VTAL	
		LICx						RESM	VTFR	
		PICx						SCHx		
		PKTx						SRBx		
		SIOx						WAIT		

Figure 64. VIT Options and the Records They Create

Selecting Internal or External Trace Recording

The VIT can record the trace data either internally in a trace table or externally in a trace file. You specify which using the **MODE** operand of the TRACE start option or the MODIFY TRACE command.

Recording Trace in Internal Table (MODE = INT)

If you set MODE = INT on the MODIFY TRACE command, or if you let MODE default to INT, the VIT writes the trace records in an internal trace table. The table is allocated and initialized in a storage area.

- MVS The storage area is in common service area (CSA).
- VM The storage area is in common storage.
- VSE The storage area is in the VTAM VFBUF buffer pool, located in the VTAM partition.

Examine the internal trace table by taking a dump of VTAM with the appropriate storage area.

Note: The internal trace records contain exactly the same information as an external file (MODE=EXT).

Find the internal trace table in the dump by first locating the ATCVT. Field ATCITTBL contains a pointer to the trace table. The internal trace table contains a 32-byte header followed by 32-byte trace records.

The **SIZE** operand of the MODIFY TRACE command specifies the number of pages (1 to 999) in storage to be allocated for the table. Because it is a wraparound table, specify enough pages to ensure that the VIT will not overwrite important trace records when the table fills and begins to wrap around.

The SIZE operand should also reflect the amount of storage available for the internal trace table. If there is not enough storage available for the number of pages specified, you will receive a message indicating that internal trace activation failed.

You can change the size of the internal trace table while the VIT is running by issuing a MODIFY TRACE or MODIFY NOTRACE command with a new SIZE operand.

Note: Issuing another MODIFY command while the VIT is running frees the current internal trace table and begins a new one. The trace information that the VIT has already written in the current table is lost.

Figure 65 shows an example of internal trace records written with MODE=INT and printed in dump output.

```
V01AAEB20  C4C9E2D7 0A990048 01AC8EE0 00B69720  80000000 80000000 E3E2D5C9 01AE1560     *DISP....................TSNI...-*
V01AAEB40  C4C9E2F2 01AF7120 30000000 00000000  00000000 00000000 00000000 01AF7120     *DIS2............................*
V01AAEB60  D3D2C5E7 0A000100 00B69738 00000000  819E58FA 00000000 00000000 01AE1560     *LKEX...........................-*
V01AAEB80  E4D5D3D2 0A000100 00B69738 00000000  819E59C8 00000000 01800000 01AE1560     *UNLK..............H............_*
V01AAEBA0  C5E2C340 0A000048 81B2F228 00B69720  81A16222 01AF7120 C9D5C9D7 01AE1560     *ESC ......2.............INIP...-*
V01AAEBC0  E2D9E3C6 0A000000 00000000 00010001  81B2F33E 01C25FD0 40404040 40404040     *SRTF..............3..B¬.        *
V01AAEBE0  C7C2D3D2 0A000000 01C27E68 01ACBC90  819F0304 00000088 81B30206 00000000     *GBLK.....B=.....................*
V01AAEC00  E5E3C1D3 0A000000 01B13008 000000FA  819F0528 00000005 81B30300 00000000     *VTAL............................*
V01AAEC20  C3C3D600 0A540000 01'AE179C C9D5C9D7  81B2F716 00000001 00010000 0001003E     *CCO.........INIP..7.............*
V01AAEC40  C3C3F254 0E000000 00000000 00000000  00000000 00000000 00000000 00000000     *CC2.............................*
V01AAEC60  D8E4C540 0A540080 01AC8EE0 00B69758  01C27E68 00000000 D3E4D9E3 00000000     *QUE ............B=.....LURT.....*
V01AAEC80  D8E4C5F2 819F118C 30480000 00000000  00000000 00000000 00000000 00000000     *QUE2............................*
V01AAECA0  D3D2C5E7 0A000100 00B69770 00000000  819EA24C 00000000 00000000 01AE1560     *LKEX..............<...........-*
V01AAECC0  E4D5D3D2 0A000100 00B69770 00000100  819EA3A4 00000000 01800000 01AE1560     *UNLK...........................-*
```

Figure 65. Internal Trace Records in Dump Output (MODE=INT)

The first line of the internal trace table is the header, which contains the following status information:

Byte (hex)	**Contents**
00–07	Current entry time stamp (the time the current record was entered)
08–0F	Present wraparound time stamp
10–17	Last wraparound time stamp
18–1B	Address of the most recent entry in the table
1C–1F	Address of the last entry in the table.

Recording Trace in External File (MODE = EXT)

If you set MODE = EXT on the MODIFY TRACE command, the VIT writes the trace records in an external trace file, rather than an internal trace table. You should use this option when you must collect large amounts of trace data, to lessen the chance of wraparound and lost data.

Note: The trace data can still fill the external trace file, wrap around, and begin overwriting itself if you do not ensure that the external trace file is large enough.

The external trace file contains the same information as the internal trace table. However, unlike the internal trace table, the external trace file does not contain a first-line header record.

The external trace file is produced by GTF, and the default file name is SYS1.TRACE. You can print the internal trace data with GTF, PRDMP or TAP. Use the EDIT option of PRDMP and set the USR = (FE1) keyword. For more information on GTF and PRDMP, see "Activating VTAM Traces" on page 222 and " Using PRDMP" on page 224. For more information on TAP, see the *NCP, SSP, and EP Diagnosis Guide*. Figure 66 is an example of internal trace output printed with PRDMP.

```
USRFD FE1 ASCB 00FA2080            JOBN HSTC016
 AI1     ASID 11   TYPE  15       EXTDS 40       OPT1  20       RPL   00164DD4 ARG   00000000 RH3   00
                                  SRTYP 00       VTFL  200010   CNTRL 800000   OPTC  10309450 80002010
 AI2               RTRN  00000000 LOGMD          PLUNM          SLUNM
USRFD FE1 ASCB 00FA2080            JOBN HSTC016
 CC0     ASID 11   TYPE  00       CBID  54       SAVE  01DC7CF8 SVID           ISSR  81AE4474
                                                 ORIG  0000000A000C            DEST  0000000A000C
 CC2               CBID  54       RU    00000000 63000024 01DB3450 01DAB018 00000000 00000000 SENSE 00000000
USRFD FE1 ASCB 00F97080            JOBN HSVTAM
 SRTF    ASID      RC    00       PARM  0000000000A000C         ISSR  81E4403E SRTE  01F35E80 NETID
USRFD FE1 ASCB 00F97080            JOBN HSVTAM
 CCI     ASID 66   TYPE  00       CBID  54       SAVE  01F5C950 SVID  INS1     ISSR  81E3A20C
                                                 ORIG  0000000A000C            DEST  0000000A000C
 CC2               CBID  54       RU    0C810A01 63000024 01DB3450 01DAB018 00000000 0A000C00 SENSE 00000000
USRFD FE1 ASCB 00FA2080            JOBN HSTC016
 UP      ASID 11   TYPE  15       EXTDS 40       CODES 000000   RPL   00164DD4 ECB   00000000 AREA  00000000
                                                                RLEN  00000000 CID   00000000 FBK2  00000000
USRFD FE1 ASCB 00F97080            JOBN HSVTAM
 CCI     ASID 66   TYPE  00       CBID  58       FLAGS 80       RC    00       SAVE  01DC7C58 SVID
                                                 ISSR  81E10890 CPCB  00000000 58488000 00000000
SVC   020 ASCB 00FA2080  CPU 0000  JOBN HSTC016   OLD PSW 075C0014 00961F5E  TCB 005FF1F8  MODN SVC-T2
          R15  00000000  R0  00000000  R1 005A0270
          PLIST 805A0308
USRFD FE1 ASCB 00F97080            JOBN HSVTAM
 CCI     ASID 66   TYPE  00       CBID  58       FLAGS 80       RC    00       SAVE  01DC7C58 SVID
                                                 ISSR  81E10890 CPCB  00000000 58488000 00000000
```

Figure 66. Internal Trace Records in GTF Output (MODE = EXT)

The trace file is produced by CPTRAP. You can format the file with TRAPRED. For a full explanation of printing trace records, see " Using CPTRAP and TRAPRED" on page 225. Figure 67 on page 312 is a sample of the output.

```
3D OE VTAM        VM/SI USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 97F6F2B3CB2BAA40
       LENGTH OF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = FD
       EVENT IDENTIFICATION = EFE1
VTFR ISSR     0AD1D4   AREA 00105308 SP#  000000E5 LENGTH  00000368 RC     00000000 INIT  00000000 00000000
3D OE VTAM        VM/SI USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 97F6F2B3CEA29E00
       LENGTH CF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = FD
       EVENT IDENTIFICATION = EFE1
CCI  FLGS 00 ISSR 09D2EA CBID 54 SAV 106938 SVID INS1 CID  10011001 RU 00000000 00000000 00000000
3D OE VTAM        VM/SI USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 97F6F2B3D1131C80
       LENGTH OF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = FD
       EVENT IDENTIFICATION = EFE1
QUE  ISSR     A1289E   PST  5485C008 PAB  108436D8 WG 00 ELM 104F40 PABCHN 00000000 NAME NORT
     RPH      20000000
3D OE VTAM        VM/SI USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 97F6F2B3D3807860
       LENGTH OF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = FD
       EVENT IDENTIFICATION = EFE1
FBLK ISSR     A10EEA   AREA 00104E20 ANCH 00849928 LENGTH  00000060 RC     00000000 INIT  500AA664
              00000000
3D OE VTAM        VM/SI USER REQUESTED GTRACE
       TIME OF DAY CLOCK = 97F6F2B3D5E01C80
       LENGTH OF GTF HEADER AND TRACE DATA = 0030
       FORMAT ROUTINE ID = FD
       EVENT IDENTIFICATION = EFE1
```

Figure 67. VM Internal Trace Records in CPTRAP Output (MODE = EXT)

VSE The external trace file is produced by TPRINT, and the default file name is VTAM.TRACE.FILE. Use TPRINT or TAP to print the internal trace data. See "VSE Using TPRINT" on page 227 for more information on using TPRINT. For more information on TAP, see the *NCP, SSP, and EP Diagnosis Guide*. Figure 68 is an example of internal trace output printed with TPRINT.

```
VTAM INTERNAL TRACE          85.295/18:35:54.01    LRC(62)
 VTAL  E5E3C1D3  34000000  0088C058  FFFFE5D7  60718532  00000078  00000000  00000000  VTAL.....H....VP-.E.............
 QUE   D8E4C540  340E0010  3297F010  00980BE0  009CD010  00000000  D9C1E3E6  00000001  QUE .....PO..Q..........RATW....
 QUE2  D8E4C5F2  6087DC70  04400000  00000000  00000000  00000000  00000000  00000000  QUE2-G... ......................
 VTAL  E5E3C1D3  34000000  0088C660  FFFFE5D7  607185B0  00000140  00000000  00000000  VTAL.....HF-..VP-.E.... ........
 QUE   D8E4C540  34490010  2497F9A8  0072BF18  0088C660  00000000  C3C6C6C5  00000000  QUE .....P9Y.....HF-....CFFE....
 QUE2  D8E4C5F2  6071886E  04400000  00000000  00000000  00000000  00000000  00000000  QUE2-.H>. ......................
 VTFR  E5E3C6D9  34000000  0088C058  FFFFE5D7  60718AB0  00000078  00000000  00000000  VTFR.....H....VP-..............
 ADSP  C1C4E2D7  32000000  0097F010  00000000  00000000  00000000  80000000  00980BE0  ADSP.....PO..................Q..
 DISP  C4C9E2D7  320E0010  0097F010  00980BE0  809CD010  80000000  D9C1E3E6  00A31360  DISP.....PO..Q..........RATW.T.-
 DIS2  C4C9E2F2  009CD010  04000000  00000000  00000000  00000000  00000000  00000000  DIS2............................
 EXIT  C5E7C9E3  32000010  3297F010  00980BE0  50A439AC  80000000  00A43C00  00A31360  EXIT.....PO..Q..&U.......U...T.-
 AXIT  C1E7C9E3  32000000  3297F010  00000000  00000000  00000000  00000000  00000000  AXIT.....PO.....................
```

Figure 68. VSE Internal Trace Records in TPRINT Output (MODE = EXT)

Module Names in Internal Trace Records

Many VTAM internal trace records include the associated module names in EBCDIC, without the IST prefix and the sixth letter. For example, you would see TSSR for module ISTTSCSR. You can save time by scanning for these module names when you are following the logic flow through VTAM. You can sometimes isolate a VTAM problem to a specific component or module without even looking at a dump.

The first two letters in the module name, as it appears in a trace record, are the component ID for the module. VTAM component IDs are listed below:

ID	VTAM Component
AC	Activation (configuration services)
AI	Application program interface (API)
AP	Process scheduling services (PSS)
AT	PSS termination
AU	Authorization or accounting exit routine
CD	Cross-domain resource manager (CDRM)
CF	Display, network operator command facilities
CO	LU 6.2 control operator
CP	SSCP functions
DE	Deactivation (configuration services)
ES	Execution sequence control (ESC)
HS	LU 6.2 half-session
IN	Initialization/Termination
LU	Logical unit services (LUS)
MA	Maintenance services
MG	Management services
NA	Network address management
NO	Network operator command facilities (NOCF)
NR	Network resource manager (NRM)
NS	LU 6.2 logical network services
OC	OPEN/CLOSE
OR	Storage management services (SMS)
PD	Problem determination
PS	LU 6.2 presentation services
PU	Physical unit services (PUS)
RA	Trace, reliability, availability, serviceability (RAS)
RM	LU 6.2 logical resource management
SD	System definition
SQ	Inquire
SS	Session Services
TS	Transmission subsystem component (TSC)
ZB	Error recovery

SNAP Trace Record

Besides recording the previously mentioned functions, you can use the VIT to record 32, 64, 96, or 128 bytes of user-generated information in a SNAP trace. This SNAP trace can place additional information into a VTAM module that might be useful in resolving a VTAM problem. The VIT records the specified bytes and enters the data as a SNAP trace entry in the internal trace table or in the external trace file.

Unlike other trace entries, you do not specify the SNAP trace as an option of the MODIFY command. Instead, you must follow these steps:

1. Insert a record to check that ATCLCPTR is not 0.
2. Insert a record to check that ATCRACTR is not 0.
3. Insert a call to the internal trace recording routine (ISTRACTR) in a VTAM module. (Field ATCRACTR in the ATCVT points to the trace recording routine. See *VTAM Data Areas* for the hex offset.)

Note: SNAP records cannot be taken during the early stages of VTAM initialization, nor can they be taken during the late stages.

The SNAP trace requires the following information in registers 1, 14, and 15 on entry to the trace routine:

Register 1
Points to the 32, 64, 96, or 128 bytes to be recorded.

Register 14
Points to a 2-byte input flag field containing:

- X'141C' for a 32-byte SNAP entry
- X'142C' for a 64-byte SNAP entry
- X'143C' for a 96-byte SNAP entry
- X'144C' for a 128-byte SNAP entry.

These flags tell the trace recording routine to record a SNAP trace.

Two Bytes Past the Address in Register 14
Is the return address to the VTAM module that called the trace recording routine.

Register 15
Points to the SNAP trace recording routine (ISTRACTR).

When the trace recording routine returns control to the calling module, the contents of register 6 are unpredictable. The contents of the other registers, however, are not changed.

The SNAP trace record is unformatted. You should format the first 4 bytes of the SNAP record for a 1-to 4-character EBCDIC record identifier (for example, C'ABCD').

Note: The 1-to 4-character EBCDIC record identifier should not match any existing VIT record identifier.

Deactivating the Internal Trace

You can stop one or more of the active VIT options by issuing the MODIFY NOTRACE command with TYPE=VTAM specified. You can specify which traces are to be stopped using the OPTION operand.

If you specify OPTION=ALL, all active VIT options are stopped, but the internal trace table is not freed. Exceptions are still recorded during this idle period.

If you specify OPTION=END, all active VIT options are stopped, and the internal trace table is freed.

For complete descriptions of the MODIFY TRACE and MODIFY NOTRACE commands, and start options for the VIT, see *VTAM Operation*.

Internal Trace Record Summary

The following chart lists all VIT entries alphabetically by record ID. For detailed descriptions of the trace records, see "Internal Trace Record Descriptions" on page 338.

Note: A zero (0) in a trace record field means that field is not used by VTAM.

<table>
<tr><th></th><th>00</th><th>01</th><th>02</th><th>03</th><th>04</th><th>05</th><th>06</th><th>07</th><th>08</th><th>09</th><th>0A</th><th>0B</th><th>0C</th><th>0D</th><th>0E</th><th>0F</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th><th>18</th><th>19</th><th>1A</th><th>1B</th><th>1C</th><th>1D</th><th>1E</th><th>1F</th></tr>
<tr><td>VM Abend SNAP</td><td colspan="4">ABND</td><td colspan="4">CODE</td><td colspan="4">PST ADDRESS</td><td colspan="4">RPH ADDRESS</td><td colspan="4">SDWA ADDRESS or 0</td><td colspan="4">SDWA FLAGS or 0</td><td colspan="8">ABEND PSW or 0</td></tr>
<tr><td>LU 6.2 Authorized IO or LU 6.2 TPIO (Part 1)</td><td colspan="4">ACA1 or ACI1</td><td>ID</td><td>EXTDS</td><td>OPT1</td><td>OPT6</td><td colspan="4">RPL ADDRESS</td><td>CNTRL</td><td>QUAL</td><td colspan="2">0</td><td colspan="4">RPL AREA</td><td colspan="4">APPCCMD FLAGS</td><td colspan="4">CONV ID or USR FLD or CID or 0</td><td colspan="4">SENSE DATA or 0</td></tr>
<tr><td>LU 6.2 Authorized IO or LU 6.2 TPIO (Part 2)</td><td colspan="4">ACA2 or ACI2</td><td colspan="4">RETURN ADDRESS</td><td colspan="8">MODE NAME or 0</td><td colspan="8">LOCAL LU NAME or 0</td><td colspan="8">REMOTE LU NAME or 0</td></tr>
<tr><td>LU 6.2 User Post or LU 6.2 RPL Exit (Part 1)</td><td colspan="4">ACP1 or ACR1</td><td>ID</td><td>RTUN</td><td>RTNCD</td><td>FDB2</td><td colspan="4">RPL ADDRESS</td><td>CNTRL</td><td>QUAL</td><td colspan="2">0</td><td colspan="2">RCPRI</td><td colspan="2">RCSEC</td><td colspan="4">APPCCMD FLAGS</td><td colspan="4">CONV ID or 0</td><td colspan="4">SENSE DATA or 0</td></tr>
</table>

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
LU 6.2 User Post or LU 6.2 RPL Exit (Part 2)	ACP2 or ACR2				ECB ADDRESS or EXIT ADDRESS				RPL6 ADDRESS				CNTRL	0	RCVIND		RPL AREA				RPL RECORD LENGTH				0							
LU 6.2 User Exit (Part 1)	ACU1				ID	CODE	0		EXIT TYPE DEPEND INFO				0				PLU NAME or APPLICATION ID								SLU NAME or 0							
LU 6.2 User Exit (Part 2)	ACU2				EXIT ADDRESS				MODE NAME or 0								0															
VM VSE Adapter (Part 1, for V3R1.2 and later)	ADPA ADPI ADPO				ID	CUA			TSCB ADDRESS				NCB ADDRESS				CBID	ACK	INDX	SLPT	FRNT	END	0		DATA FIELD							
VM VSE Adapter (Part 2, for V3R1.2 and later)	ADP2				DATA FIELD																											
VM VSE Asynchronous Dispatch	ADSP				ID	0			PST ADDRESS				0				SYNC TPPOSTD QUEUE				SYNC NORMAL QUEUE				ASYNC TPPOSTD QUEUE				ASYNC NORMAL QUEUE			
MVS VM Authorized IO (Part 1)	AI1			REQTY	ID	EXTDS	0	OPT1	RPL ADDRESS				NIB ADDR or CID				RH3	SRTYP	VTFL1	VTFL2	CHN	RU CNTL CODE			OPTC2				OPTC3			
MVS VM Authorized IO (Part 2)	AI2			0	RETURN ADDRESS				LOGON MODE NAME or 0								PLU NAME (if available)								SLU NAME (if available)							

<table>
<tr><td></td><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td>MVS VM Authorized IO (Part 3)</td><td colspan="3">AI3</td><td>0</td><td colspan="4">ADDRESS
OF USER
DATA,
BIND,
or 0</td><td colspan="24">0</td></tr>
<tr><td>MVS VM Abend RELSTORE</td><td colspan="4">AREL</td><td>ID</td><td>CBID</td><td colspan="2">0</td><td colspan="4">PST
ADDRESS</td><td colspan="4">BUFFER
ADDRESS</td><td colspan="4">RETURN
ADDRESS</td><td colspan="12">0</td></tr>
<tr><td>MVS Attention</td><td colspan="3">ATT</td><td>TYPE</td><td>ID</td><td>STATE</td><td colspan="3">0</td><td colspan="3">CUA</td><td colspan="4">NCB
ADDRESS</td><td colspan="4">FLAG
BYTES</td><td>0</td><td>CODE</td><td colspan="2">SENSE</td><td colspan="8">CSW</td></tr>
<tr><td>VM VSE Asynchronous Exit</td><td colspan="4">AXIT</td><td>ID</td><td colspan="3">0</td><td colspan="4">PST
ADDRESS</td><td colspan="4">0</td><td colspan="4">SYNC
TPPOSTD
QUEUE</td><td colspan="4">SYNC
NORMAL
QUEUE</td><td colspan="4">ASYNC
TPPOSTD
QUEUE</td><td colspan="4">ASYNC
NORMAL
QUEUE</td></tr>
<tr><td>SSCP (RUPE — Part 1)</td><td colspan="3">CCI
or
CCO</td><td>FLAGS</td><td>ID</td><td>CBID</td><td colspan="2">0</td><td colspan="4">SAVE
AREA
ADDRESS</td><td colspan="4">SAVE
AREA
ID</td><td colspan="4">RETURN
ADDRESS</td><td colspan="6">RUPE
ORIGIN
NETWORK
ADDRESS</td><td colspan="6">RUPE
DESTINATION
NETWORK
ADDRESS</td></tr>
<tr><td>SSCP (RUPE — Part 2)</td><td colspan="3">CC2</td><td>CBID</td><td colspan="24">FIRST 24 BYTES OF RU</td><td colspan="4">SENSE
DATA
or 0</td></tr>
<tr><td>SSCP (NCSPL — Part 1)</td><td colspan="3">CCI
or
CCO</td><td>FLAGS</td><td>ID</td><td>CBID</td><td>FLAGS</td><td>RC</td><td colspan="4">SAVE
AREA
ADDRESS</td><td colspan="4">SAVE
AREA
ID</td><td colspan="4">RETURN
ADDRESS</td><td colspan="4">CPCB
OPCODE</td><td>TYPE</td><td>0</td><td colspan="6">DESTINATION
NETWORK
ADDRESS</td></tr>
</table>

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SSCP (NCSPL — Part 2)	CC2			CBID	RDTE STATE				CPCB WTD				0																			
SSCP (not RUPE or NCSPL)	CCI or CCO			FLAGS	ID	CBID	CPCBF	RC	SAVE AREA ADDRESS				SAVE AREA ID				RETURN ADDRESS				CPCB OPCODE				CPCB WORD 3				CPCB WORD 4			
SSCP (RUPE)	CI1 or CO1			FLAGS	ID	CBID	CPCBF	RC	SAVE AREA ADDRESS				SAVE AREA ID				RETURN ADDRESS				RUPE ORIGIN NETWORK ADDRESS						RUPE DESTINATION NETWORK ADDRESS					
SSCP (NCSPL)	CI1 or CO1			FLAGS	ID	CBID	FLAGS	RC	SAVE AREA ADDRESS				SAVE AREA ID				RETURN ADDRESS				CPCB OPCODE				TYPE	0	RUPE DESTINATION NETWORK ADDRESS					
SSCP (not RUPE or NCSPL)	CI1 or CO1			FLAGS	ID	CBID	CPCBF	RC	SAVE AREA ADDRESS				SAVE AREA ID				RETURN ADDRESS				CPCB OPCODE				CPCB WORD 3				CPCB WORD 4			
SSCP (RUPE)	CI2 or CO2			CBID	FIRST 24 BYTES OF RU																								SENSE DATA or 0			

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F

SSCP (NCSPL)

CI2 or CO2	CBID	RDTE STATE	CPCB WTD	0

SSCP (not RUPE or NCSPL)

CI2 or CO2	CBID	0

SSCP (RUPE, NCSPL, or not RUPE or NCSPL)

CI3 or CO3	0	SIB ADDRESS	PCID FOR THIS REQUEST	PLU NAME	SLU NAME

SSCP (RUPE, NCSPL, or not RUPE or NCSPL)

CI4 or CO4	0	PCID IN PREVIOUS REQUEST	PLU NETID	SLU NETID

VSE SHM Connect

CONN	TIK	CFSMM	PMFSM	SHSI	0	CUA	GROUP NCB ADDRESS	HALCB ADDRESS	PUT ADDRESS	0	PEA	0

CPPURGE or CPPOST with EID or URC (Part 1, for V3R1, V3R1.1, and V3R1.2)

CPPG or CPPT	ID	CODE	TYPE	0-LEN	LQAB ADDRESS	DATA	RETURN ADDRESS	FIRST 12 BYTES OF EID or URC

CPPURGE or CPPOST (Part 2, for V3R1, V3R1.1, and V3R1.2)

CPP2	ADDITIONAL EVENT IDENTIFIER FOR CPPOST/CPPURGE (if needed)

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F

CPPURGE or CPPOST for PVI or Non-PVI Event (Part 1, for V3R2)

CPPG or CPPT	ID	CODE	TYPE	OPT	LQAB ADDRESS	WREDATA IN WRE	WORK ELEMENT ADDRESS	OPC IN WORK ELEMENT or 0	SAVE AREA ID or 0	SENSE DATA or 0

CPPURGE or CPPOST (Part 2, for V3R2)

CPP2	RETURN ADDRESS	RUPEURC or 0	0

CPPURGE or CPPOST (Part 3, for V3R2)

CPP3	0	EIDLN	FIRST 24 BYTES OF EID

CPPURGE or CPPOST (Part 4, for V3R2)

CPP4	FIRST 24 BYTES OF RU or 0	0

CPRC (Part 1)

CPRC	ID	0	RUPE ADDRESS or 0	OPC IN THIS RUPE	SENSE CODE	RUPE ORIGIN NETWORK ADDRESS	RUPE DESTINATION NETWORK ADDRESS

CPRC (Part 2)

CPR2	RETURN ADDRESS	RUPEURC	0

CPWAIT (Part 1, for V3R1, V3R1.1, and V3R1.2)

CPWT	ID	0	EIDLN	LQAB ADDRESS	WREDATA IN WRE	WORK ELEMENT ADDRESS	OPC IN WORK ELEMENT	FIRST 8 BYTES OF CPCBURC

CPWAIT (Part 2, for V3R1, V3R1.1, and V3R1.2)

CPW2	RETURN ADDRESS	FIRST 24 BYTES OF EID

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

CPWAIT for PVI or Non-PVI Events (Part 1, for V3R2)

CPWT	ID	0	TYPE	0	LQAB ADDRESS	WREDATA IN WRE	WORK ELEMENT ADDRESS	OPC IN WORK ELEMENT or 0	SAVE AREA ID or 0	0

CPWAIT (Part 2, for V3R2)

CPW2	RETURN ADDRESS	RUPEURC or 0	0

CPWAIT (Part 3, for V3R2)

CPW3	0	EIDLN	FIRST 24 BYTES OF EID

CPWAIT (Part 4, for V3R2)

CPW4	FIRST 24 BYTES OF RU or 0

VSE SHM Disconnect

DISC	TIK	CFSM	PMFSM	SHSI	0	CUA	GROUP NCB ADDRESS	HALCB ADDRESS	PUT ADDRESS	0

PAB Dispatch (Part 1, for V3R1, V3R1.1, and V3R1.2)

DISP	ID	CBID	PABOF	PST ADDRESS	PAB ADDRESS	WORK ELEMENT QUEUE	PAB CHAIN FIELD	MODULE NAME or DVT ADDRESS	RPH ADDRESS

PAB Dispatch (Part 2, for V3R1, V3R1.1, and V3R1.2)

DIS2	WORK ELEMENT ADDRESS or 0	FLAGS	0	PAB EXTENSION or 0

<table>
<tr><th></th><th>00</th><th>01</th><th>02</th><th>03</th><th>04</th><th>05</th><th>06</th><th>07</th><th>08</th><th>09</th><th>0A</th><th>0B</th><th>0C</th><th>0D</th><th>0E</th><th>0F</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th><th>18</th><th>19</th><th>1A</th><th>1B</th><th>1C</th><th>1D</th><th>1E</th><th>1F</th></tr>
<tr><td>PAB Dispatch (V3R2)</td><td colspan="4">DISP</td><td>ID</td><td>CBID</td><td>FLAG</td><td>0-WEQ</td><td colspan="4">PST ADDRESS</td><td colspan="4">PAB ADDRESS</td><td colspan="4">LAST WORK ELEMENT ADDRESS</td><td colspan="4">CURRENT WORK ELEMENT ADDRESS</td><td colspan="4">MODULE NAME or DVT ADDRESS</td><td colspan="4">RPH ADDRESS</td></tr>
<tr><td>Discarded PIU Buffer (Part 1)</td><td colspan="4">DSCD</td><td>ID</td><td colspan="3">REASN CODE</td><td colspan="4">TSCB ADDRESS</td><td colspan="20">20 BYTES OF PIU</td></tr>
<tr><td>Discarded PIU Buffer (Part 2)</td><td colspan="4">DSC2</td><td colspan="24">24 MORE CHARACTERS OF THE PIU</td><td colspan="4">MODULE ID</td></tr>
<tr><td>MVS Error Recovery Procedures</td><td colspan="4">ERPI or ERPL</td><td>ID</td><td>STATE</td><td colspan="3">0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="4">FLAG BYTES</td><td>FLAG</td><td>CODE</td><td colspan="2">SENSE</td><td colspan="8">CSW</td></tr>
</table>

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
</table>

VM Error Recovery Procedures (Part 1)

<table>
<tr><td colspan="3">ERP</td><td>TYPE</td><td>ID</td><td colspan="3">CUA</td><td>0</td><td colspan="3">NCB ADDR</td><td colspan="4">VIRTUAL CAW</td><td colspan="4">FLAG BYTES</td><td colspan="2">ERCOD</td><td colspan="2">SENDE</td><td colspan="8">0</td></tr>
<tr><td colspan="4">ERPB</td><td>ID</td><td>LFSM</td><td colspan="3">0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="4">FLAG BYTES</td><td>ERCOD</td><td>0</td><td colspan="2">SENSE</td><td colspan="8">CSW</td></tr>
<tr><td colspan="4">ERPH</td><td>ID</td><td>LFSM</td><td colspan="3">0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="4">FLAG BYTES</td><td colspan="4">0</td><td colspan="8">CSW</td></tr>
<tr><td colspan="4">ERPI</td><td>ID</td><td>LNKST</td><td colspan="3">0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="4">FLAG BYTES</td><td>ERCOD</td><td>0</td><td colspan="2">SENSE</td><td colspan="8">CSW</td></tr>
<tr><td colspan="4">ERPL
or
ERPX</td><td>ID</td><td>LNKST</td><td colspan="3">0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="4">FLAG BYTES</td><td colspan="2">0</td><td colspan="2">SENSE</td><td colspan="8">CSW</td></tr>
</table>

VM Error Recovery Procedures (Part 2)

<table>
<tr><td colspan="4">ERP2</td><td colspan="4">0</td><td colspan="21">DEVICE-DEPENDENT INFORMATION</td><td colspan="3">0</td></tr>
</table>

VSE Error Recovery Procedures (Part 1)

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ERP			TYPE	ID	CUA			0	NCB ADDR			VIRTUAL CAW				FLAG BYTES				ERCOD		SENDE		0							
ERPB				ID	LFSM	0			CUA			NCB ADDRESS				FLAG BYTES				ERCOD	0	SENSE		CSW							
ERPH				ID	LFSM	0			CUA			NCB ADDRESS				FLAG BYTES				0	RCODE	0		CSW							
ERPI				ID	LNKST	ERACT	0		CUA			NCB ADDRESS				FLAG BYTES				ERCOD	RCODE	SENSE		CSW							
ERPL				ID	LNKST	ERACT	0		CUA			NCB ADDRESS				FLAG BYTES				0	RCODE	SENSE		CSW							

VSE Error Recovery Procedures (Part 2)

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ERP2				0				DEVICE-DEPENDENT INFORMATION																					0		

TPESC

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ESC				ID	0	PABOF		DISP ADDRESS				PAB ADDRESS				RETURN ADDRESS				RPH WORK ELEMENT				MODULE NAME or DVT ADDRESS				RPH ADDRESS			

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F

TPEXIT

EXIT	ID	APNOP	PABOF	PST ADDRESS	PAB ADDRESS	RETURN ADDRESS	WORK ELEMENT QUEUE	MODULE NAME or DVT ADDRESS	RPH ADDRESS

FREEBLK

FBLK	ID	0	POOL	0	STORAGE ADDRESS or 0	ADDRESS OF SPANC	RETURN ADDRESS	LENGTH OF STORAGE FREED	CALLER OF UTILITY or 0	RETURN CODE

GETBLK

GBLK	ID	0	POOL	0	STORAGE ADDRESS or 0	ADDRESS OF SPANC	RETURN ADDRESS	LENGTH OF STORAGE REQUEST	CALLER OF UTILITY or 0	RETURN CODE

Halt I/O

HIOB HIOH HIOX	ID	STATE	MODID	0	CUA	NCB ADDRESS	FLAG BYTES	0

VM VSE Halt I/O (V3R1.2 and later)

HIOP	ID	CPFSM	MODID	0	CUA	NCB ADDRESS	PDATA	0

MVS Interrupt

INTI INTL INTX	ID	STATE	EOPCD	0	CUA	NCB ADDRESS	FLAG BYTES	FLAG	CODE	SENSE or 0	CSW

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F

VM Interrupt (Part 1)

INT	TYPE	ID	CUA	CSW	SENSE_STATUS CCW	NCB ADDR

INTB or INTH	ID	LFSM	0	CUA	NCB ADDRESS	FLAG BYTES	0	CODE	SENSE	CSW

INTI	ID	LNKST	0	CUA	NCB ADDRESS	FLAG BYTES	0	CODE	SENSE	CSW

INTL	ID	LNKST	0	CUA	NCB ADDRESS	FLAG BYTES	0	CODE	0	CSW

INTP	ID	CPFSM	0	CUA	NCB ADDRESS	PDATA	0	STIM	CODE	SENSE	0	ENDING CSW

INTX	ID	STATE	EOPCD	0	CUA	NCB ADDRESS	FLAG BYTES	FLAG	CODE	SENSE or 0	CSW

VM Interrupt (Part 2)

INT2	0	DEVICE-DEPENDENT INFORMATION	0

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
VSE Interrupt (Part 1)	INT			TYPE	ID	CUA			CSW		SENSE_STATUS CCW																		NCB ADDR			
	INTB or INTH				ID	LFSM	0			CUA		NCB ADDRESS				FLAG BYTES				0	CODE	SENSE		CSW								
	INTI				ID	LNKST	ERACT	0		CUA		NCB ADDRESS				FLAG BYTES				ERCOD	CODE	SENSE		CSW								
	INTL				ID	LNKST	ERACT	0		CUA		NCB ADDRESS				FLAG BYTES				0	CODE	0		CSW								
	INTP				ID	CPFSM	0			CUA		NCB ADDRESS				PDATA		0		STIM	CODE	SENSE	0	ENDING CSW								
VSE Interrupt (Part 2)	INT2				0				DEVICE-DEPENDENT INFORMATION												0											
TPIO Request (Part 1)	I01			REQTY	ID	EXTDS	0	OPT1	RPL ADDRESS				NIB ADDRESS or CID				RH3	SRTYP	VTFL1	VTFL2	CHN	RU CNTL CODE			OPTC2				OPTC3			
MVS VSE TPIO Request (Part 2)	I02			0	RETURN ADDRESS				LOGON MODE NAME or 0								PLU NAME (if available)								SLU NAME (if available)							

0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	1 9	1 A	1 B	1 C	1 D	1 E	1 F

MVS VSE TPIO Request (Part 3)

I03	0	ADDRESS OF USER DATA, BIND, or 0	0

VM TPIO Request (Part 2)

I02	0	RETURN ADDRESS	LOGON MODE NAME or 0	PLU NAME (if available)	SLU NAME (if available)

VM TPIO Request (Part 3)

I03	0	ADDRESS OF BIND or 0	0

MVS IRB Dispatch

IRBD	ID	0	PST ADDRESS	PST SYNC TPPOSTD QUEUE	PST SYNC NORMAL QUEUE	IRB ADDRESS	TCB ADDRESS	FLG1	0

MVS IRB Exit

IRBX	ID	0	PST ADDRESS	NEW TCB ADDRESS	OLD TCB ADDRESS	NEW ASCB ADDRESS	OLD ASCB ADDRESS	IRB ADDRESS

VM VSE LAN Interface Control (V3R1.2 and later)

LICI LICO	ID	CUA	TSCB ADDRESS	NCB ADDRESS	CBID	CMD	TFSMS	VFSMS	FLAGS	TSCB CONT ADDRESS
LICS	ID	CUA	TSCB ADDRESS	NCB ADDRESS	CBID	CMD	TFSMS	REASON		

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F

TPLOCK Exclusive

LKEX	ID	0	LOCK	LEVEL	LOCK ADDRESS	CRA LKACT	RETURN ADDRESS	LOCKWORD	RPH ADDRESS

TPLOCK Shared

LKSH	ID	0	LOCK	LEVEL	LOCK ADDRESS	CRA LKACT	RETURN ADDRESS	LOCKWORD	RPH ADDRESS

Lost Trace Record

LOST	ID	0	REG 14	LOST TRACE RECORD

TPMSG or CPMSG (Part 1)

MSG	ID	0	SorU	SAVE AREA ADDRESS	MSG ID or 'USS '	RETURN ADDRESS	MODULE ID	DESTINATION

Message Sent Record (Part 1)

MSGS	ID	0	DESTINATION	MESSAGE HEADER	FIRST 12 CHARACTERS OF MESSAGE TEXT

TPMSG or CPMSG (Part 2 for both MSG and MSGS)

MSG2	VARIABLE DATA or MORE MESSAGE TEXT

LU 6.2 Message Unit (Part 1)

MU1	ID	MU ID	MU ADDRESS	HS ID	CORR ADDRESS	RETCD	0	AMU FLAGS	SENSE DATA or 0

LU 6.2 Message Unit (Part 2)

MU2	0	MU ID	DATA ADDRESS or 0	DATA LENGTH or 0	RAB ADDRESS or 0	SES CNT or 0	SES LIM or 0	CONV ID or 0	VARIABLE OVERLAY FIELD

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F

LU 6.2 Message Unit (Part 3)	MU3	0	MU ID	MODE NAME or 0	LOCAL LU NAME	REMOTE LU NAME		
Negative Response to a PIU Request (Part 1)	NRSP	ID	CBID	FLAGS	0	TSCB ADDRESS	20 BYTES OF PIU	
Negative Response to a PIU Request (Part 2)	NRS2	24 MORE CHARACTERS OF THE PIU	RETURN ADDRESS					
Operator Command (Part 1)	OPER	ID	FLAG	0	FIRST 24 CHARACTERS OF OPERATOR COMMAND			
Operator Command (Part 2)	OPE2	UP TO 28 MORE CHARACTERS OF THE OPERATOR COMMAND						
VM VSE Packet Interface (V3R1.2 and later)	PIC	INTERNAL TRACE DATA						
PIU Record (Part 1)	PIU	ID	CBID	FLAGS	0	TSCB or RUPE ADDRESS	20 BYTES OF PIU	
PIU Record (Part 2)	PIU2	28 MORE BYTES OF PIU						
VM VSE Packet Inbound or Packet Outbound (V3R1.2 and later)	PKTI PKTO	ID	CUA	PFSM	INDX	LNGTH	FIRST 20 BYTES OF THE PACKET	

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F

VM VSE Packet Extension (V3R1.2 and later)

PKTX	ID	ADDITIONAL BYTES OF PACKET (if needed)

TPPOST

POST	ID	0	PABOF	PST ADDRESS	PAB ADDRESS	RETURN ADDRESS	WORK ELEMENT ADDRESS	DVT ADDRESS	RPH ADDRESS

MVS VM Queued REQSTORE

QREQ	ID	CBID	0	PST ADDRESS	BUFFER ADDRESS	0	NUM BUF REQ	0	RPH ADDRESS

TPQUE (Part 1, for V3R1, V3R1.1, and V3R1.2)

QUE	ID	CBID	PABOF	PST ADDRESS	PAB ADDRESS	WORK ELEMENT ADDRESS	PAB FIELD	MODULE NAME or DVT ADDRESS	RPH ADDRESS or 0

TPQUE (Part 2, for V3R1, V3R1.1, and V3R1.2)

QUE2	RETURN ADDRESS	FLAGS	APNOP	0	PAB EXTENSION or 0

TPQUE (V3R2)

QUE	CBID	ID	FLAG	NOP	GATE	PST ADDRESS	PAB ADDRESS	RETURN ADDRESS	WORK ELEMENT ADDRESS	MODULE NAME or DVT ADDRESS	RPH ADDRESS or 0

TPQUE None

QUEN	ID	CBID	0	PST ADDRESS or 0	QUEUE ADDRESS	RETURN ADDRESS	WORK ELEMENT ADDRESS	0	RPH ADDRESS or 0

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F

RCE (V3R2)

RCEA RCEC RCED RCEF	ID	RC	TYPE	MODE	KEY VALUE	RETURN ADDRESS	OUT1 or NEW TABLE NAME	OUT2 or NEW TABLE NAME	DATA ADDRESS

RPL Exit

RE	REQTY	EXTDS	ID	RTNCD	FDB2	FDB3	RPL ADDRESS	EXIT ADDRESS	RPL AREA	RPL RLEN	CID or 0	RPL FDBK2

RELSTORE

RELS	ID	CBID	POOL	ID	PST ADDRESS	BUFFER ADDRESS	RETURN ADDRESS	NEXT BUFFER ADDRESS or 0	REG 1	CALLER OF UTILITY or 0

REQSTORE

REQS	ID	CBID	0	PST ADDRESS or 0	BUFFER ADDRESS or 0	RETURN ADDRESS	NUM BUF REQ	0	REG 1	RETURN CODE

Resume from TPWAIT

RESM	ID	CBID	PABOF	PST ADDRESS	PAB ADDRESS	WORK ELEMENT ADDRESS	WORK ELEMENT QUEUE	DVT ADDRESS	RPH ADDRESS

TPSCHED (Part 1, for V3R1, V3R1.1, and V3R1.2)

SCHD	ID	CBID	PABOF	PST ADDRESS	PAB ADDRESS	WORK ELEMENT QUEUE	PAB CHAIN FIELD	MODULE NAME or DVT ADDRESS	RPH ADDRESS

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F

TPSCHED (Part 2, for V3R1, V3R1.1, and V3R1.2)

SCH2	RETURN ADDRESS	0	APNOP	0	PAB EXTENSION or 0

TPSCHED (V3R2)

SCHD	ID	FLAG	NOP	GATE	PST ADDRESS	PAB ADDRESS	RETURN ADDRESS	WORK ELEMENT QUEUE	MODULE NAME or DVT ADDRESS	RPH ADDRESS

MVS Start I/O (Part1)

SIOI or SIOL	ID	STATE	MODID	WRITE	CUA	NCB ADDRESS	FLAG BYTES	CAW	CCW or DEVICE-DEPENDENT INFORMATION
SIOX	ID	STATE	MODID	0	CUA	NCB ADDRESS	FLAG BYTES	CAW	CCW or OUTPUT CONTROL AREA (STATUS INFORMATION)

MVS Start I/O (Part 2)

SIO2	VIRTUAL BUFFER ADDRESS	0

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
</table>

VM Start I/O (Part 1)

<table>
<tr><td colspan="3">SIO</td><td>TYPE</td><td>ID</td><td colspan="3">CUA</td><td>PFSM</td><td>0</td><td colspan="22">DUMP OF WRITE CONTROL AND READ CONTROL DATA AREAS</td></tr>
<tr><td colspan="4">SIOB or SIOH</td><td>ID</td><td>LFSM</td><td colspan="2">MODID</td><td>0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="4">FLAG BYTES</td><td colspan="4">CAW</td><td colspan="8">0</td></tr>
<tr><td colspan="4">SIOI or SIOL</td><td>ID</td><td>LNKST</td><td colspan="2">MODID</td><td>0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="4">FLAG BYTES</td><td colspan="4">CAW</td><td colspan="8">0</td></tr>
<tr><td colspan="4">SIOP</td><td>ID</td><td>CPFSM</td><td colspan="2">MODID</td><td>0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="2">PDATA</td><td colspan="2">0</td><td colspan="4">CAW</td><td>CODE</td><td colspan="7">0</td></tr>
<tr><td colspan="4">SIOX</td><td>ID</td><td>STATE</td><td colspan="2">MODID</td><td>0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="4">FLAG BYTES</td><td colspan="4">CAW</td><td colspan="8">CCW or OUTPUT CONTROL AREA (STATUS INFORMATION)</td></tr>
</table>

VM Start I/O (Part 2)

<table>
<tr><td colspan="4">SIO2</td><td>ERACT</td><td>TECT</td><td>ERCOD</td><td>RCODE</td><td colspan="16">DEVICE-DEPENDENT INFORMATION or 0</td><td colspan="8">0</td></tr>
</table>

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F

VSE Start I/O (Part 1)

SIO			TYPE	ID	CUA			PFSM	0	DUMP OF WRITE CONTROL AND READ CONTROL DATA AREAS																					
SIOB or SIOH				ID	LFSM	MODID		0	CUA			NCB ADDRESS				FLAG BYTES				CAW				0							
SIOI or SIOL				ID	LNKST	MODID		0	CUA			NCB ADDRESS				FLAG BYTES				CAW				0							
SIOP				ID	CPFSM	MODID		0	CUA			NCB ADDRESS				PDATA		0		CAW				CODE	0						

VSE Start I/O (Part 2)

SI02				ERACT	TECT	ERCOD	RCODE	DEVICE-DEPENDENT INFORMATION or 0																0							

MVS SRB Dispatch

SRBD				ID	0			PST ADDRESS				PST ASYNC DISP QUEUE (ADSP)				PST ASYNC FLAG FIELD (IAF)				PST ASYNC N-DISP QUEUE (ANDSP)				TCB ADDRESS				FLG1		0	

MVS SRB Exit

SRBX				ID	0			PST ADDRESS				PST ASYNC DISP QUEUE (ADSP)				PST ASYNC FLAG FIELD (IAF)				NEW ASCB ADDRESS				OLD ASCB ADDRESS				0			

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F

SRTCHG (V3R2 only), SRTADD, SRTDEL, SRTFIND

SRTA SRTC SRTD SRTF	ID	RC	0	TYPE	HASH NAME (NAME or NETWORK ADDRESS)	RETURN ADDRESS	SRT ENTRY ADDRESS	NETWORK ID

USER Exit (DFASY or RESP)

UE	CODE	EXTDS	ID	RTNCD	FDB2	FDB3	RPL ADDRESS	EXIT ADDRESS	RPL AREA	RPL RLEN	CID or 0	RPL FDBK2

USER Exit (Part 1, not DFASY or RESP)

UE1	ID	CODE	0	EXIT-TYPE DEPENDENT INFORMATION	PRIMARY (APPL) LU NAME	SECONDARY LU NAME or 0

USER Exit (Part 2, not DFASY or RESP)

UE2	EXIT ADDRESS	0

TPUNLOCK All

ULKA	ID	0	PST ADDRESS	CRA LKACT	RETURN ADDRESS	0	RPH ADDRESS

TPUNLOCK

UNLK	ID	0	LOCK	LEVEL	LOCK ADDRESS	CRA LKACT	RETURN ADDRESS	LOCKWORD	RPH ADDRESS

User Post

UP	REQTY	EXTDS	ID	RTNCD	FDB2	FDB3	RPL ADDRESS	EXIT or ECB ADDRESS	RPL AREA	RPL RLEN	CID or 0	RPL FDBK2

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

VTALLOC

VTAL	ID	0	STORAGE ADDRESS	SUBPOOL NUMBER or POOL ID	RETURN ADDRESS	LENGTH OF STORAGE REQUEST	CALLER OF UTILITY or 0	RETURN CODE

VTFREE

VTFR	ID	0	STORAGE ADDRESS	SUBPOOL NUMBER or POOL ID	RETURN ADDRESS	LENGTH OF STORAGE FREED	CALLER OF UTILITY or 0	0

TPWAIT

WAIT	ID	0	PST ADDRESS	PAB ADDRESS	RETURN ADDRESS	WORK ELEMENT QUEUE	MODULE NAME or DVT ADDRESS	RPH ADDRESS

Internal Trace Record Descriptions

This section describes the VTAM internal trace table records.

On each entry, the description called "Applies to" tells which operating systems the entry is valid for. For MVS and VM, V3R1.1, V3R1.2, and V3R2 are implied unless it is stated specifically that the entry is for one or two releases. For VSE entries, V3R1, V3R1.2, and V3R2 are implied unless it clearly specifies that the entry is for one or two releases.

VM Abend SNAP

Entry:	**VM ABND**
VIT Option:	None (Generated by SNAP routine)
Event:	Abend
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTAPCES
Applies to:	VM

This trace record is written when an abend occurs on a process that has been dispatched by PSS. This entry is not associated with any VIT options, but is recorded as an exception condition when an abend occurs.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ABND				CODE				PST ADDRESS				RPH ADDRESS				SDWA ADDRESS or 0				SDWA FLAGS or 0				ABEND PSW or 0							

Byte (hex)	Contents
00 – 03	Record ID: C'ABND'
04 – 07	Abend completion code
08 – 0B	PST address
0C – 0F	RPH address
10 – 13	SDWA address or zero
14 – 17	SDWA flags or zero
18 – 1F	Abend PSW or zero

LU 6.2 Authorized IO or LU 6.2 TPIO (Part 1)

Entry:	**ACA1 or ACI1**
VIT Option:	APPC
Event:	LU 6.2 authorized IO or LU 6.2 TPIO (Part 1)
VIT Processing Module:	ISTRACAC
Control is Returned to:	ISTAICAR
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

The ACA1 trace record shows LU 6.2 requests under the authorized path or LU 6.2 requests not under the authorized path but issued by programs running under an authorized key. The ACI1 trace record shows LU 6.2 requests running under a normal path.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ACA1 or ACI1				ID	EXTDS	OPT1	OPT6	RPL ADDRESS				CNTRL	QUAL	0		RPL AREA				APPCCMD FLAGS				CONV ID or USR FLD or CID or 0				SENSE DATA or 0			

Byte (hex)	Contents
00 – 03	Record ID: • C'ACA1' for authorized path • C'ACI1' for normal path
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Exit definition (RPLEXTDS)
06	Option code byte 1 (RPLOPT1)
07	Option code byte 6 (RPLOPT6)
08 – 0B	RPL address
0C	APPCCMD CONTROL parameter value: X'10' ALLOC X'20' RESETRCV X'30' DEALLOC X'40' OPRCNTL X'50' PREPRCV X'60' RCVFMH5 X'70' RECEIVE X'80' REJECT X'90' SEND
0D	APPCCMD QUALIFY parameter value: X'01' ABNDPROG X'02' ABNDSERV X'03' ABNDTIME X'04' ABNDUSER X'05' ANY X'06' CNOS

	X'07'	CONFIRM
	X'08'	CONFRMD
	X'09'	DATA
	X'0A'	DATACON
	X'0B'	DATAFLU
	X'0C'	DEFINE
	X'0D'	DISPLAY
	X'0E'	ERROR
	X'0F'	FLUSH
	X'10'	RQSEND
	X'11'	SPEC
	X'12'	ACTSESS
	X'13'	DACTSESS
	X'14'	ALLOCD
	X'15'	IMMED
0E – 0F	Zero	
10 – 13	Address of SEND data or RECEIVE buffer	
14 – 17	APPCCMD flags (RPL6FLGS)	
18 – 1B	User field for LU 6.2 commands where CONTROL equals ALLOC or RCVFMH5. CID for LU 6.2 commands where CONTROL equals OPRCNTL and QUALIFY equals either ACTSESS or DACTSESS. Zero for LU 6.2 commands for which one of the following is true: • CONTROL equals OPRCNTL, and QUALIFY equals CNOS, DEFINE, or DISPLAY • CONTROL equals RECEIVE and QUALIFY equals ANY. Conversation ID for all other LU 6.2 commands.	
1C – 1F	Sense data (RPL6SNSO) or zero	

LU 6.2 Authorized IO or LU 6.2 TPIO (Part 2)

Entry:	**ACA2 or ACI2**
VIT Option:	APPC
Event:	LU 6.2 authorized IO or LU 6.2 TPIO (Part 2)
VIT Processing Module:	ISTRACAC
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

The ACA2 trace record is a continuation of ACA1. The ACI2 trace record is a continuation of ACI1.

<table>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr>
<tr><td colspan="4">ACA2
or
ACI2</td><td colspan="4">RETURN
ADDRESS</td><td colspan="8">MODE NAME or 0</td><td colspan="8">LOCAL LU NAME
or 0</td><td colspan="8">REMOTE LU NAME
or 0</td></tr>
</table>

Byte (hex) Contents

00 – 03 Record ID:

- C'ACA2' for continuation of authorized path for ACA1
- C'ACI2' for continuation of normal path for ACI1

04 – 07 Return address of the issuer of the APPCCMD macro

08 – 0F Mode name for LU 6.2 commands where CONTROL equals ALLOC. Mode name for LU 6.2 commands where CONTROL equals OPRCNTL and QUALIFY equals CNOS, DEFINE, or DISPLAY. Zero otherwise.

10 – 17 Local logical unit name for LU 6.2 commands where CONTROL equals ALLOC. Local logical unit name for LU 6.2 commands where CONTROL equals OPRCNTL, and QUALIFY equals CNOS, DEFINE, or DISPLAY. Zero otherwise.

18 – 1F Remote logical unit name for LU 6.2 commands where CONTROL equals ALLOC. Remote logical unit name for LU 6.2 commands where CONTROL equals OPRCNTL, and QUALIFY equals CNOS, DEFINE or DISPLAY. Zero otherwise.

LU 6.2 User Post or LU 6.2 RPL Exit (Part 1)

Entry:	**ACP1 or ACR1**
VIT Option:	APPC
Event:	LU 6.2 user post (ACP1) or LU 6.2 RPL exit (ACR1) (Part 1)
VIT Processing Module:	ISTRACAC
Control is Returned to:	ISTAICPT and ISTAPCSX for ACP1; ISTAPCSX and ISTAPCUE for ACR1
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

The ACP1 trace record is written when an ECB is posted, signaling that the APPCCMD macro has completed execution. This trace record shows the data returned to the user application program.

The ACR1 trace record is written when an RPL exit is dispatched, signaling that the APPCCMD macro has completed execution. This trace record shows the data returned to the user application program.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ACP1 or ACR1				ID	RTUN	RTNCD	FDB2	RPL ADDRESS				CNTRL	QUAL	0		RCPRI		RCSEC		APPCCMD FLAGS				CONV ID or 0				SENSE DATA or 0			

Byte (hex)	**Contents**
00–03	Record ID: • C'ACP1' for ECB posting • C'ACR1' for RPL exit dispatching
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	APPCCMD returned indicators (RPL6RTUN) Bit 0 FMH5 received indicator Bit 1 LOGON received indicator Bit 2 SIGNAL received indicator
06	VTAM return code (RPLRTNCD)
07	VTAM feedback code (RPLFDB2)
08–0B	RPL address
0C	APPCCMD CONTROL parameter value: X'10' ALLOC X'20' RESETRCV X'30' DEALLOC X'40' OPRCNTL X'50' PREPRCV X'60' RCVFMH5 X'70' RECEIVE X'80' REJECT X'90' SEND

0D	APPCCMD CONTROL parameter value:	
	X'01'	ABNDPROG
	X'02'	ABNDSERV
	X'03'	ABNDTIME
	X'04'	ABNDUSER
	X'05'	ANY
	X'06'	CNOS
	X'07'	CONFIRM
	X'08'	CONFRMD
	X'09'	DATA
	X'0A'	DATACON
	X'0B'	DATAFLU
	X'0C'	DEFINE
	X'0D'	DISPLAY
	X'0E'	ERROR
	X'0F'	FLUSH
	X'10'	RQSEND
	X'11'	SPEC
	X'12'	ACTSESS
	X'13'	DACTSESS
	X'14'	ALLOCD
	X'15'	IMMED
0E – 0F	Zero	
10 – 11	LU 6.2 primary return code (RPL6RCPR)	
12 – 13	LU 6.2 secondary return code (RPL6RCSC)	
14 – 17	APPCCMD flags (RPL6FLGS)	
18 – 1B	Conversation ID or zero	
1C – 1F	Sense data returned (RPL6SNSI) or zero	

LU 6.2 User Post or LU 6.2 RPL Exit (Part 2)

Entry:	**ACP2 or ACR2**
VIT Option:	APPC
Event:	LU 6.2 user post (ACP2) or LU 6.2 RPL exit (ACR2) (Part 2)
VIT Processing Module:	ISTRACAC
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

The ACP2 trace record is a continuation of the ACP1 trace record. The ACR2 trace record is a continuation of the ACR1 trace record.

00 01 02 03	04 05 06 07	08 09 0A 0B	0C	0D	0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B 1C 1D 1E 1F
ACP2 or ACR2	ECB ADDRESS or EXIT ADDRESS	RPL6 ADDRESS	C N T R L	0	R C V I N D	RPL AREA	RPL RECORD LENGTH	0

Byte (hex)	Contents
00 – 03	Record ID: • C'ACP2' for continuation of ACP1 • C'ACR2' for continuation of ACR1
04 – 07	ECB address for ACP2 RPL exit address for ACR2
08 – 0B	RPL6 address
0C	APPCCMD CONTROL parameter value: X'10' for ALLOC X'20' for RESETRCV X'30' for DEALLOC X'40' for OPRCNTL X'50' for PREPRCV X'60' for RCVFMH5 X'70' for RECEIVE X'80' for REJECT X'90' for SEND
0D	Zero
0E – 0F	Indicators for the type of information received, or zero Bit 0 DATA Bit 1 DATA_COMPLETE Bit 2 DATA_INCOMPLETE Bit 3 SEND Bit 4 CONFIRM Bit 5 DEALLOCATE
10 – 13	Address of SEND data or RECEIVE buffer
14 – 17	RPL record length
18 – 1F	Zero

LU 6.2 User Exit (Part 1)

Entry:	**ACU1**
VIT Option:	APPC
Event:	LU 6.2 user exit (Part 1)
VIT Processing Module:	ISTRACAC
Control is Returned to:	ISTAPCUE
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record is written when an LU 6.2 user exit (either ATTN or TPEND) is dispatched. If both the API and APPC trace options are active, and a TPEND user exit is dispatched, VTAM generates user exit trace records for both API and APPC.

00 01 02 03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13 14 15 16 17	18 19 1A 1B 1C 1D 1E 1F
ACU1	ID	CODE	0	EXIT TYPE DEPEND INFO	0	PLU NAME or APPLICATION ID	SLU NAME or 0

Byte (hex)	**Contents**
00 – 03	Record ID: C'ACU1'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Exit code for user exit: • X'0C' for ATTN • X'0D' for TPEND
06 – 07	Zero
08 – 0B	For ATTN exit, the exit sub-type (CNOS, FMH5, or LOSS) For TPEND exit, the reason code
0C – 0F	Zero
10 – 17	PLU name for ATTN exit Application program ID for TPEND exit
18 – 1F	SLU name for ATTN exit Zero for TPEND exit

LU 6.2 User Exit (Part 2)

Entry:	**ACU2**
VIT Option:	APPC
Event:	LU 6.2 user exit (Part 2)
VIT Processing Module:	ISTRACAC
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record is a continuation of the ACU1 trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ACU2				EXIT ADDRESS				MODE NAME or 0								0															

Byte (hex)	Contents
00 – 03	Record ID: C'ACU2'
04 – 07	Exit address
08 – 0F	Mode name for ATTN exit; zero for TPEND exit.
10 – 1F	Zero

VM VSE Adapter (Part 1, for V3R1.2 and Later)

Entry:	VM VSE **ADP**
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRAC6T
Control is Returned to:	ISTTSC6B
Applies to:	VM (V3R1.2 and later), VSE (V3R2)

This trace record is created when a data transfer with the Token-Ring Subsystem is initiated or has completed.

An ADPA record is written when the subsystem acknowledges that an outbound data transfer is complete. This entry always corresponds to a previous ADPO entry. Use the TSCB field to correlate these entries.

An ADPI record is written when an inbound transfer has been received from the subsystem. An ADPI trace entry may be followed by one or more ADP2 entries.

An ADPO record is written when an outbound data transfer is complete. This entry always corresponds to a previous LICO entry. Use the TSCB field to correlate these entries. An ADPO trace entry may be followed by one or more ADP2 entries.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ADPA ADPI ADPO				ID	CUA			TSCB ADDRESS				NCB ADDRESS				CBID	ACK	INDX	SLPT	FRNT	END	0		DATA FIELD							

Byte (hex)	**Contents**
00 – 03	Record ID: C'ADPA' for an acknowledge record C'ADPI' for an inbound record C'ADPO' for an outbound record
04	Machine and task ID
05 – 07	Channel device name in EBCDIC (either a CUA or device number)
08 – 0B	TSCB address
0C – 0F	NCB address
10	Control block ID
11	Index value acknowledged by subsystem
12	Last index value presented to subsystem
13	Last index value acknowledged by subsystem
14	Index for the front pointer of the buffer ring
15	Index for the end pointer of the buffer ring
16 – 17	Zero
18 – 1F	Data field.

VM VSE Adapter (Part 2, for V3R1.2 and Later)

Entry:	VM VSE **ADP2**
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRAC6T
Applies to:	VM (V3R1.2 and later), VSE (V3R2)

This trace record is a continuation of the ADPI or ADPO entries.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ADP2				DATA FIELD																											

Byte (hex)	Contents
00 – 03	Record ID: C'ADP2'
04 – 1F	Data field.

VM VSE Asynchronous Dispatch

Entry:	VM VSE **ADSP**
VIT Option:	PSS
Event:	Asynchronous dispatch
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCAD
Applies to:	VM, VSE

This trace record is written whenever PSS receives control because work has been scheduled (TREADY COND=VTAM).

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ADSP				ID	0			PST ADDRESS				0				SYNC TPPOSTD QUEUE				SYNC NORMAL QUEUE				ASYNC TPPOSTD QUEUE				ASYNC NORMAL QUEUE			

Byte (hex)	**Contents**
00–03	Record ID: C'ADSP'
04	ID • For VM, machine and task ID • For VSE, task ID
05–07	Zero
08–0B	PST address
0C–0F	Zero
10–13	Synchronous TPPOSTed queue (PSTSTPP)
14–17	Synchronous normal queue (PSTSNRM)
18–1B	Asynchronous TPPOSTed queue (PSTATPP)
1C–1F	Asynchronous normal queue (PSTANRM)

MVS VM Authorized IO (Part 1)

Entry:	MVS VM **AI1**	
VIT Option:	API	
Event:	Authorized IO (Part 1)	
VIT Processing Module:	ISTRACAP	
Control is Returned to:	ISTAICIR for:	ISTOCCSM for:
	SEND	INQUIRE
	RECEIVE	INTRPRET
	RESETSR	RCVCMD
	SESSIONC	SENDCMD
		SETLOGON
		SESSIONC (BIND)
		SIMLOGON
		OPNDST
		REQSESS
		OPNSEC
		TERMSESS
		CLSDST
Applies to:	MVS, VM	

The following trace record shows API requests under the authorized path or API requests not under the authorized path but issued by programs running under an authorized key. The API routine ISTAICIR or ISTOCCSM writes the entry and then queues the RPL to the correct PAB. See *VTAM Data Areas* for an explanation of the RPL fields.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
AI1			REQTY	ID	EXTDS	0	OPT1	RPL ADDRESS				NIB ADDR or CID				RH3	SRTYP	VTFL1	VTFL2	CHN	RU CNTL CODE			OPTC2				OPTC3			

Byte (hex)	Contents
00–02	Record ID: C'AI1'
03	RPL request type (RPLREQ)
	X'15' SETLOGON
	X'16' SIMLOGON
	X'17' OPNDST
	X'1A' INQUIRE
	X'1B' INTRPRET
	X'1F' CLSDST
	X'22' SEND
	X'23' RECEIVE
	X'24' RESETSR
	X'25' SESSIONC
	X'27' SENDCMD
	X'28' RCVCMD
	X'29' REQSESS
	X'2A' OPNSEC
	X'2C' TERMSESS

04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID
05	Exit definition (RPLEXTDS)
06	Zero
07	Option code byte 1 (RPLOPT1)
08–0B	RPL address
0C–0F	NIB address or CID
10	Third byte of RH (RPLRH3)
11	Send/receive type (RPLSRTYP)
12	VTAM flags (RPLVTFL1)
13	Post/respond flags (RPLVTFL2)
14	RU chain position (RPLCHN)
15–17	RU control codes (RPLCNTRL)
18–1B	VTAM options (RPLOPTC2)
1C–1F	VTAM options (RPLOPTC3)

MVS VM Authorized IO (Part 2)

Entry:	MVS VM **AI2**
VIT Option:	API
Event:	Authorized IO (Part 2)
VIT Processing Module:	ISTRACAP
Applies to:	MVS, VM

This trace record is a continuation of the AI1 entry. It shows additional information about the PLU and SLU in a session established or terminated by a SIMLOGON, OPNDST, CLSDST, REQSESS, OPNSEC, or TERMSESS macroinstruction. This information includes some of the parameters exchanged and the names of the PLU and SLU.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="3">AI2</td><td>0</td><td colspan="4">RETURN ADDRESS</td><td colspan="8">LOGON MODE NAME or 0</td><td colspan="8">PLU NAME (if available)</td><td colspan="8">SLU NAME (if available)</td></tr>
</table>

Byte (hex)	Contents
00 – 02	Record ID: C'AI2'
03	Zero
04 – 07	Address of the issuer of the macro
08 – 0F	For SIMLOGON, OPNDST, CLSDST, or REQSESS: Logon mode name (NIBLMODE) or zero if the NIB is not available For OPNSEC or TERMSESS: Zero
10 – 17	PLU name or zero
18 – 1F	SLU name or zero

MVS VM Authorized IO (Part 3)

Entry:	MVS VM
VIT Option:	API
Event:	Authorized IO (Part 3)
VIT Processing Module:	ISTRACAP
Applies to:	MVS, VM

This trace record is a continuation of the AI2 entry. It shows additional information in a session established or terminated by a SIMLOGON, OPNDST, CLSDST, REQSESS, or OPNSEC macroinstruction.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
AI3			0	ADDRESS OF USER DATA, BIND, or 0				0																							

Byte (hex)	Contents
00 – 02	Record ID: C'AI3'
03	Zero
04 – 07	For SIMLOGON, CLSDST, or REQSESS: Address of user data (RPLAREA) For OPNDST or OPNSEC: Address of BIND (NIBNDAR) or zero if the NIB is not available
08 – 1F	Zero

MVS VM Abend RELSTORE

Entry:	MVS VM **AREL**
VIT Option:	SMS
Event:	Abend RELSTORE
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTORAPR
Applies to:	MVS VM

This trace record identifies the buffers that are released by VTAM when a VTAM application program is terminated.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
AREL				ID	CBID	0		PST ADDRESS				BUFFER ADDRESS				RETURN ADDRESS				0											

Byte (hex)	**Contents**
00 – 03	Record ID: C'AREL'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID
05	Control block ID index value
06 – 07	Zero
08 – 0B	PST address
0C – 0F	Address of buffer being released
10 – 13	Address of the caller of the abend RELSTORE routine
14 – 1F	Zero

MVS Attention

Entry:	**MVS ATT**
VIT Option:	CIO
Event:	Attention
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCCA or ISTTSCLA
Applies to:	MVS

- ATTI is written when an attention interrupt occurs for a channel-attached communication controller or SNA cluster controller.
- ATTL is written when an attention interrupt occurs for a channel attached non-SNA device.
- ATTX is written when an attention interrupt occurs for a channel-to-channel attachment.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ATT			TYPE	ID	STATE	0			CUA			NCB ADDRESS				FLAG BYTES				0	CODE	SENSE		CSW							

Byte (hex)	Contents
00 – 02	Record ID: C'ATT'
03	NCB type: I for ICNCB, L for LDNCB, X for XCNCB
04	Address space ID (or zero if address space ID is greater than X'FF')
05	Link station state (NCBLNKST) for ICNCB and LDNCB; station state (XCNSSFSM) for XCNCB
06 – 08	Zero
09 – 0B	Channel device name in EBCDIC (either a CUA or device number)
0C – 0F	NCB address
10 – 13	Flag bytes (NCBFLAGS)
14	Zero
15	I/O completion code in field IOSCOD in the IOSB
16 – 17	Sense data in field IOSSNS of the IOSB
18 – 1F	Channel status word from the IOSB

VM VSE Asynchronous Exit

Entry:	VM VSE **AXIT**
VIT Option:	PSS
Event:	Asynchronous exit
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCSX
Applies to:	VM, VSE

This trace record is written whenever PSS exits to the operating system because there is no more work to do under a given task.

<table>
<tr><td>0
0</td><td>0
1</td><td>0
2</td><td>0
3</td><td>0
4</td><td>0
5</td><td>0
6</td><td>0
7</td><td>0
8</td><td>0
9</td><td>0
A</td><td>0
B</td><td>0
C</td><td>0
D</td><td>0
E</td><td>0
F</td><td>1
0</td><td>1
1</td><td>1
2</td><td>1
3</td><td>1
4</td><td>1
5</td><td>1
6</td><td>1
7</td><td>1
8</td><td>1
9</td><td>1
A</td><td>1
B</td><td>1
C</td><td>1
D</td><td>1
E</td><td>1
F</td></tr>
<tr><td colspan="4">AXIT</td><td>I
D</td><td colspan="3">0</td><td colspan="4">PST
ADDRESS</td><td colspan="4">0</td><td colspan="4">SYNC
TPPOSTD
QUEUE</td><td colspan="4">SYNC
NORMAL
QUEUE</td><td colspan="4">ASYNC
TPPOSTD
QUEUE</td><td colspan="4">ASYNC
NORMAL
QUEUE</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'AXIT'
04	ID • For VM, machine and task ID • For VSE, task ID
05 – 07	Zero
08 – 0B	PST address
0C – 0F	Zero
10 – 13	Synchronous TPPOSTed queue (PSTSTPP)
14 – 17	Synchronous normal queue (PSTSNRM)
18 – 1B	Asynchronous TPPOSTed queue (PSTATPP)
1C – 1F	Asynchronous normal queue (PSTANRM)

SSCP (RUPE — Part 1)

Entry:	**CCI or CCO**
VIT Option:	SSCP
Event:	Requests/Responses with a RUPE
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTINCCI for inbound processing, ISTINCCO for outbound processing
Applies to:	MVS, VM, VSE

This trace record provides information about outbound processing, which processes a request, and inbound processing, which sends a response to the requester.

If this entry is associated with an event failure (that is, the sense data is non-zero), this entry is generated whether the SSCP option is in effect or not. It is treated as an exception condition, and therefore, is traced whenever the VIT is active.

00 01 02	03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17 18 19	1A 1B 1C 1D 1E 1F
CCI or CCO	FLAGS	ID	CBID	0	SAVE AREA ADDRESS	SAVE AREA ID	RETURN ADDRESS	RUPE ORIGIN NETWORK ADDRESS	RUPE DESTINATION NETWORK ADDRESS

Byte (hex)	Contents
00 – 02	Record ID: • C'CCI' for inbound processing • C'CCO' for outbound processing
03	Flags • 80 = response RU • 40 = sense traced • 00 = request RU
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID=X'54'
06 – 07	Zero
08 – 0B	Save area address
0C – 0F	Save area ID (bytes 4, 5, 7, and 8 of the module name if available)
10 – 13	Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)
14 – 19	For sender, network address at origin
1A – 1F	For sender, network address at destination

SSCP (RUPE — Part 2)

Entry:	**CC2**
VIT Option:	SSCP
Event:	Requests/Responses with a RUPE (Part 2)
VIT Processing Module:	ISTRACSC
Applies to:	**MVS, VM, VSE**

This trace record is a continuation of the CCI or CCO (RUPE) entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CC2			C B I D	FIRST 24 BYTES OF RU																								SENSE DATA or 0			

Byte (hex)	Contents
00 – 02	Record ID: C'CC2'
03	Control block ID=X'54'
04 – 1B	First 24 bytes of RU
1C – 1F	Sense data or zero

SSCP (NCSPL – Part 1)

Entry:	**CCI or CCO**
VIT Option:	SSCP
Event:	Requests/Responses for a NCSPL
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTINCCI for inbound processing, ISTINCCO for outbound processing
Applies to:	MVS, VM, VSE

This trace record provides information about outbound processing, which processes a request, and inbound processing, which sends a response to the requester.

If this entry is associated with an event failure (that is, the sense data is non-zero), this entry is generated whether the SSCP option is in effect or not. It is treated as an exception condition, and therefore, is traced whenever the VIT is active.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CCI or CCO			FLAGS	ID	CBID	FLAGS	RC	SAVE AREA ADDRESS				SAVE AREA ID				RETURN ADDRESS				CPCB OPCODE				TYPE	0	DESTINATION NETWORK ADDRESS					

Byte (hex)	**Contents**
00 – 02	Record ID: • C'CCI' for inbound processing • C'CCO' for outbound processing
03	Flags • 80 = response RU • 40 = sense traced • 00 = request RU
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID=X'60'
06	Flags (CPCBFL)
07	Return code (CPCBRC)
08 – 0B	Save area address
0C – 0F	Save area ID (bytes 4, 5, 7, and 8 of the module name if available)
10 – 13	Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)
14 – 17	CPCB operation code (see Appendix D, "CPCB Operation Codes" on page 645)
18	WTD type
19	Zero
1A – 1F	Network address at destination

SSCP (NCSPL – Part 2)

Entry:	**CC2**
VIT Option:	SSCP
Event:	Requests/Responses for a NCSPL (Part 2)
VIT Processing Module:	ISTRACSC
Applies to:	MVS, VM, VSE

This trace record is a continuation of the CCI or CCO (NCSPL) entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CC2			CBID	RDTE STATE				CPCB WTD				0																			

Byte (hex)	Contents
00–02	Record ID: C'CC2'
03	Control block ID=X'60'
04–07	RDTE state (in control block RPRE)
08–0B	CPCB WTD (first four bytes)
0C–1F	Zero

SSCP (not RUPE or NCSPL)

Entry:	**CCI or CCO**
VIT Option:	SSCP
Event:	Requests/Responses for neither RUPE nor NCSPL
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTINCCI for inbound processing, ISTINCCO for outbound processing
Applies to:	MVS, VM, VSE

This trace record provides information about outbound processing, which processes a request, and inbound processing, which sends a response to the requester.

If this entry is associated with an event failure (that is, the sense data is non-zero), this entry is generated whether the SSCP option is in effect or not. It is treated as an exception condition, and therefore, is traced whenever the VIT is active.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CCI or CCO			FLAGS	ID	CBID	CPCBF	RC	SAVE AREA ADDRESS				SAVE AREA ID				RETURN ADDRESS				CPCB OPCODE				CPCB WORD 3				CPCB WORD 4			

Byte (hex)	**Contents**
00 – 02	Record ID: • C'CCI' for inbound processing • C'CCO' for outbound processing
03	Flags • Bit 0: 0 = request, 1 = response • Bit 1: 0 = no sense, 1 = sense
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID (Control block IDs are shown in Appendix E, "Storage and Control Block ID Codes" on page 657.)
06	CPCB flag (CPCBFL)
07	Return code (CPCBRC)
08 – 0B	Save area address
0C – 0F	Save area ID (bytes 4, 5, 7 and 8 of the module name if available)
10 – 13	Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)
14 – 17	CPCB operation code (see Appendix D, "CPCB Operation Codes" on page 645)
18 – 1B	CPCB word 3 (field WTDPTR)
1C – 1F	CPCB word 4 (field CPCBPH)

CI1 or CO1 Trace Entries

The following trace record provides information about the inbound and outbound requests sent to begin or end a same-network or cross-network LU-LU session.

The CI1 and CO1 traces have two formats:

- **Format 0** is used for same-network sessions or when VTAM does not know whether the request is for a cross-network session. This format contains parts 1, 2, and 3 of the CIn or COn record, shown below.
- **Format 1** is used for cross-network sessions. This format contains parts 1, 2, 3, and 4 of the CIn or COn record, shown below.

SSCP (RUPE)

Entry:	**CI1 and CO1**
VIT Option:	SSCP
Event:	Requests (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	(Many modules possible)
Applies to:	**MVS, VM, VSE**

Part 1 of the CIn or COn record is for both format 0 and format 1 and contains essentially the same information as the CCI or CCO entry.

If this entry is associated with an event failure (that is, the sense data is non-zero), this entry is generated whether the SSCP option is in effect or not. It is treated as an exception condition, and therefore, is traced whenever the VIT is active.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
CI1 or CO1			FLAGS	ID	CBID	ITFSM		SAVE AREA ADDRESS				SAVE AREA ID				RETURN ADDRESS				RUPE ORIGIN NETWORK ADDRESS						RUPE DESTINATION NETWORK ADDRESS					

Byte (hex) **Contents**

00 – 02 Record ID:
- C'CI1' for inbound processing
- C'CO1' for outbound processing

03 Flags
- Bit 0: 0 = request, 1 = response
- Bit 1: 0 = no sense, 1 = sense
- Bits 6 and 7: 00 = format 0, 01 = format 1

04 ID
- For MVS, address space ID
  (or zero if address space ID is greater than X'FF')
- For VM, machine and task ID
- For VSE, task ID

05 Control block ID=X'54'

06 – 07	Initiation and termination finite state machines (taken from the session information block, SIB)
08 – 0B	Save area address
0C – 0F	Save area ID (bytes 4, 5, 7, and 8 of the module name if available)
10 – 13	Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)
14 – 19	Network address at origin (RUPEOAF)
1A – 1F	Network address at destination (RUPEDAF)

SSCP (NCSPL)

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CI1 or CO1			FLAGS	ID	CBID	FLAGS	RC	SAVE AREA ADDRESS				SAVE AREA ID				RETURN ADDRESS				CPCB OPCODE				TYPE	0	RUPE DESTINATION NETWORK ADDRESS					

Byte (hex)	Contents
00 – 02	Record ID: • C'CI1' for inbound processing • C'CO1' for outbound processing
03	Flags • Bit 0: 0 = request, 1 = response • Bit 1: 0 = no sense, 1 = sense • Bits 6 and 7: 00 = format 0, 01 = format 1
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID=X'60'
06	Flags (CPCBFL)
07	CPCB return code (CPCBRC)
08 – 0B	Save area address
0C – 0F	Save area ID (bytes 4, 5, 7, and 8 of the module name if available)
10 – 13	Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)
14 – 17	CPCB operation code (see Appendix D, "CPCB Operation Codes" on page 645)
18	WTD type
19	Zero
1A – 1F	Network address at destination (NCSPLDAF)

SSCP (not RUPE or NCSPL)

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CI1 or CO1			FLAGS	ID	CBID	CPCBF	RC	SAVE AREA ADDRESS				SAVE AREA ID				RETURN ADDRESS				CPCB OPCODE				CPCB WORD 3				CPCB WORD 4			

Byte (hex)	Contents
00–02	Record ID: • C'CI1' for inbound processing • C'CO1' for outbound processing
03	Flags • Bit 0: 0 = request, 1 = response • Bit 1: 0 = no sense, 1 = sense • Bits 6 and 7: 00 = format 0, 01 = format 1
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID (Control block IDs are shown in Appendix E, "Storage and Control Block ID Codes" on page 657.)
06	CPCB flag (CPCBFL)
07	Return code (CPCBRC)
08–0B	Save area address
0C–0F	Save area ID (bytes 4, 5, 7, and 8 of the module name if available)
10–13	Address of the issuer of the macro (CPCALL, CPEXIT, or CPWAIT)
14–17	CPCB operation code (see Appendix D, "CPCB Operation Codes" on page 645)
18–1B	CPCB word 3 (field WTDPTR)
1C–1F	CPCB word 4 (field CPCBPH)

SSCP (RUPE)

Entry:	**CI2 or CO2**
VIT Option:	SSCP
Event:	Requests (Part 2)
VIT Processing Module:	ISTRACSC
Applies to:	MVS, VM, VSE

The CI2 and CO2 trace records are continuations of the CI1 and CO1 entries.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CI2 or CO2			C B I D	FIRST 24 BYTES OF RU																								SENSE DATA or 0			

Byte (hex)	Contents
00 – 02	Record ID: • C'CI2' for inbound processing • C'CO2' for outbound processing
03	Control block ID = X'54'
04 – 1B	First 24 bytes of RU
1C – 1F	Sense data or zero

SSCP (NCSPL)

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="3">CI2 or CO2</td><td>CBID</td><td colspan="4">RDTE STATE</td><td colspan="4">CPCB WTD</td><td colspan="20">0</td></tr>
</table>

Byte (hex)	Contents
00–02	Record ID: • C'CI2' for inbound processing • C'CO2' for outbound processing
03	Control block ID=X'60'
04–07	RDTE state (in control block RPRE). This value, a resource status code, is explained in *VTAM Messages and Codes* and in control block FSM, which is documented in *VTAM Data Areas*.
08–0B	CPCB WTD (first four bytes)
0C–1F	Zero

SSCP (not RUPE or NCSPL)

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="3">CI2 or CO2</td><td>CBID</td><td colspan="28">0</td></tr>
</table>

Byte (hex)	Contents
00–02	Record ID: • C'CI2' for inbound processing • C'CO2' for outbound processing
03	Control block ID (Control block IDs are shown in Appendix E, "Storage and Control Block ID Codes" on page 657.)
04–1F	Zero

SSCP (RUPE, NCSPL, or Not RUPE or NCSPL)

Entry:	**CI3 or CO3**
VIT Option:	SSCP
Event:	Requests (Part 3)
VIT Processing Module:	ISTRACSC
Applies to:	MVS, VM, VSE

Part 3 of the CIn or COn record is for both format 0 and format 1 and contains the address of the ISTSIB for the session, the ISTPCID, and the PLU and SLU names.

Note: If this is the last CIn or COn entry, these are the PLU and SLU names as known in this network. If this entry is followed by CI4 or CO4, these are just the PLU and SLU names. The CI4 or CO4 entry identifies the network.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CI3 or CO3			0	SIB ADDRESS				PCID FOR THIS REQUEST								PLU NAME								SLU NAME							

Byte (hex)	Contents
00 – 02	Record ID: • C'CI3' for inbound processing • C'CO3' for outbound processing
03	Zero
04 – 07	SIB address
08 – 0F	PCID for this request
10 – 17	PLU name
18 – 1F	SLU name

SSCP (RUPE, NCSPL, or Not RUPE or NCSPL)

Entry:	**CI4 or CO4**
VIT Option:	SSCP
Event:	Requests (Part 4)
VIT Processing Module:	ISTRACSC
Applies to:	MVS, VM, VSE

Part 4 of the CIN or CON entry is for format 1 only.

<table>
<tr><td>0
0</td><td>0
1</td><td>0
2</td><td>0
3</td><td>0
4</td><td>0
5</td><td>0
6</td><td>0
7</td><td>0
8</td><td>0
9</td><td>0
A</td><td>0
B</td><td>0
C</td><td>0
D</td><td>0
E</td><td>0
F</td><td>1
0</td><td>1
1</td><td>1
2</td><td>1
3</td><td>1
4</td><td>1
5</td><td>1
6</td><td>1
7</td><td>1
8</td><td>1
9</td><td>1
A</td><td>1
B</td><td>1
C</td><td>1
D</td><td>1
E</td><td>1
F</td></tr>
<tr><td colspan="3">CI4
or
CO4</td><td colspan="5">0</td><td colspan="8">PCID IN
PREVIOUS
REQUEST</td><td colspan="8">PLU NETID</td><td colspan="8">SLU NETID</td></tr>
</table>

Byte (hex)	Contents
00 – 02	Record ID: • C'CI4' for inbound processing • C'CO4' for outbound processing
03 – 07	Zero
08 – 0F	PCID in previous request
10 – 17	PLU network ID
18 – 1F	SLU network ID

VSE SHM Connect

Entry:	VSE **CONN**
VIT Option:	CIO
Event:	Connection
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSC1R
Applies to:	VSE

This trace record is generated for X.21 short-hold mode attachments. It is written when a port is connected to a station.

00 01 02 03	04	05	06	07	08	09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D	1E 1F
CONN	T I K	C F S M	P M F S M	S H S I	0	CUA	GROUP NCB ADDRESS	HALCB ADDRESS	PUT ADDRESS	0	P E A	0

Byte (hex)	Contents
00–03	Record ID: C'CONN'
04	Task identification key
05	Connection finite state machine (FSM; see HALCFSM in ISTHALCB for values)
06	Port manager FSM (see HALPMFSM in ISTHALCB for values)
07	X.21 short-hold mode status indicator (see PUTFLG1 in ISTPUT)
08	Zero
09–0B	Channel unit address of link
0C–0F	Group network control block address (ISTGRPCB)
10–13	Link network control block address (ISTHALCB)
14–17	Physical unit table address (ISTPUT)
18–1B	Zero
1C–1D	PU element ADD
1E–1F	Zero

CPPURGE or CPPOST with EID or URC (Part 1, for V3R1, V3R1.1, and V3R1.2)

Entry:	**CPPG or CPPT**
VIT Option:	SSCP
Event:	CPPURGE or CPPOST Macro (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTCPCPP
Applies to:	MVS (V3R1.1 and V3R1.2), VM (V3R1.1 and V3R1.2), VSE (V3R1 and V3R1.2)

This trace record gives information about CPPURGE or CPPOST macroinstruction processing. It may help you determine why I/O is outstanding or why an SSCP request is failing.

CPPG and CPPT are the complements of CPWT. Each CPPT is preceded by a CPWT. But a CPPG can post several events at once, as it may be concluding several CPWT entries.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPPG or CPPT				ID	CODE	TYPE	0-LEN	LQAB ADDRESS				DATA				RETURN ADDRESS				FIRST 12 BYTES OF EID or URC											

Byte (hex) Contents

00–03 Record ID:

- C'CPPG' for CPPURGE processing
- C'CPPT' for CPPOST processing

04 ID

- For MVS, address space ID (or zero if address space ID is greater than X'FF')
- For VM, machine and task ID
- For VSE, task ID

05 Return code set by CPPOST or CPPURGE

06 Type byte:

- X'00' = URC was used for this macro
- X'01' = Event ID (EID) was used for this macro

07 If type=X'00' (URC), zero.
If type=X' 01' (EID), length of event ID. (Wait state event IDs are explained in *VTAM Messages and Codes.*)

08–0B Address of LQAB for this macro

0C–0F For MVS and VSE, data operand specified by this macro.
For VM, WREDTA in WRE

10–13 Address of issuer of CPPURGE or CPPOST macro

14–1F Type information:
If type=X'00' (URC), URC up to 12 bytes
If type=X'01' (EID), EID up to 12 bytes. (Wait state event IDs are explained in *VTAM Messages and Codes.*)

CPPURGE or CPPOST (Part 2, for V3R1, V3R1.1, and V3R1.2)

Entry:	**CPP2**
VIT Option:	SSCP
Event:	CPPURGE or CPPOST Macro (Part 2)
VIT Processing Module:	ISTRACSC
Applies to:	MVS (V3R1.1 and V3R1.2), VM (V3R1.1 and V3R1.2), VSE (V3R1 and V3R1.2)

This trace record contains the rest of the event ID for CPPURGE or CPPOST macroinstruction processing.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPP2				ADDITIONAL EVENT IDENTIFIER FOR CPPOST/CPPURGE (if needed)																											

Byte (hex)	Contents
00 – 03	Record ID: C'CPP2'
04 – 1F	Event ID (up to 28 bytes)

CPPURGE or CPPOST for PVI or Non-PVI Event (Part 1, for V3R2)

Entry:	**CPPG or CPPT**
VIT Option:	SSCP
Event:	CPPURGE or CPPOST Macro (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTCPCPP
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record gives information about CPPURGE or CPPOST macroinstruction processing. It may help you determine why I/O is outstanding or why an SSCP request is failing.

CPPG and CPPT are the complements of CPWT. Each CPPT is preceded by a CPWT. But a CPPG can post several events at once, as it may be concluding several CPWT entries.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPPG or CPPT				ID	CODE	TYPE	OPT	LQAB ADDRESS				WREDATA IN WRE				WORK ELEMENT ADDRESS				OPC IN WORK ELEMENT or 0				SAVE AREA ID or 0				SENSE DATA or 0			

Byte (hex)	Contents
00 – 03	Record ID: • C'CPPG' for CPPURGE processing • C'CPPT' for CPPOST processing
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Return code set by CPPOST or CPPURGE
06	Type flags • B'00000000' = URC, non-PVI event • B'00000001' = URC, PVI event • B'00000010' = EID, non-PVI event • B'00000011' = EID, PVI event
07	Optional field • For CPPG: Zero • For CPPT **MVS/370 VSE**: Zero • For CPPT **MVS/XA VM**: 0 for FIND, 1 for DQ, 2 for POST
08 – 0B	LQAB address
0C – 0F	Data field in ISTWRE
10 – 13	Work element address
14 – 17	CPCB operation code that indicates the original request for this work element or zero. CPCB operation codes are shown in Appendix D, "CPCB Operation Codes" on page 645.
18 – 1B	For PVI events, caller's save area ID or 0 For non- PVI events, save area ID
1C – 1F	Sense data or zero

CPPURGE or CPPOST (Part 2, for V3R2)

Entry:	**CPP2**
VIT Option:	SSCP
Event:	CPPURGE or CPPOST Macro (Part 2)
VIT Processing Module:	ISTRACSC
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record is a continuation of the CPPG or CPPT (V3R2) trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPP2				RETURN ADDRESS				RUPEURC or 0																	0						

Byte (hex)	Contents
00–03	Record ID: C'CPP2'
04–07	Return address to issuer of the CPPURGE or CPPOST macroinstruction
08–18	RUPEURC or 0
19–1F	Zero

CPPURGE or CPPOST (Part 3, for V3R2)

Entry:	**CPP3**
VIT Option:	SSCP
Event:	CPPURGE or CPPOST Macro (Part 3)
VIT Processing Module:	ISTRACSC
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record contains part or all of the event ID (EID) for CPPURGE or CPPOST macroinstruction processing. If an EID is less than 24 bytes long, only the bytes in the actual EID will be moved to the corresponding fields in the trace records. The leftover bytes will be set to zero.

Note: This trace record is produced only when the WRE points to an EID.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPP3				0			EIDLN	FIRST 24 BYTES OF EID																							

Byte (hex)	Contents
00 – 03	Record ID: C'CPP3'
04 – 06	Zero
07	Length of EID
08 – 1F	First 24 bytes of EID

CPPURGE or CPPOST (Part 4, for V3R2)

Entry:	**CPP4**
VIT Option:	SSCP
Event:	CPPURGE or CPPOST Macro (Part 4)
VIT Processing Module:	ISTRACSC
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record contains part or all of the RU for CPPURGE or CPPOST macroinstruction processing. It may help you determine why I/O is outstanding or why an SSCP request is failing. If an RU is less than 24 bytes long, only the bytes in the actual RU will be moved to the corresponding fields in the trace records. The leftover bytes will be set to zero.

Note: This trace record is produced only for PVI events where a RUPE is available.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPP4				FIRST 24 BYTES OF RU or 0																								0			

Byte (hex)	Contents
00 – 03	Record ID: C'CPP4'
04 – 1B	First 24 bytes of RU
1C – 1F	Zero

CPRC (Part 1)

Entry:	**CPRC**
VIT Option:	SSCP
Event:	CPRC Macro (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	The module that issued the CPRC macro (many possible)
Applies to:	MVS, VM, VSE

This trace record gives information about CPRC macroinstruction processing. It is written when a VTAM module issues a CPRC macroinstruction to set a non-zero sense code in an RUPE.

This event is treated as an exception condition and is always traced, whether the SSCP option is in effect or not.

00 01 02 03	04	05 06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17 18 19	1A 1B 1C 1D 1E 1F
CPRC	ID	0	RUPE ADDRESS or 0	OPC IN THIS RUPE	SENSE CODE	RUPE ORIGIN NETWORK ADDRESS	RUPE DESTINATION NETWORK ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'CPRC'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05–07	Zero
08–0B	Address of RUPE for this macro, or 0
0C–0F	CPCB operation code for RUPE that indicates the original request for this RUPE
10–13	Sense code
14–19	Network address at origin
1A–1F	Network address at destination

CPRC (Part 2)

Entry:	**CPR2**
VIT Option:	SSCP
Event:	CPRC Macro (Part 2)
VIT Processing Module:	ISTRACSC
Applies to:	**MVS, VM, VSE**

This trace record is a continuation of the CPRC entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPR2				RETURN ADDRESS				RUPEURC																	0						

Byte (hex)	**Contents**
00 – 03	Record ID: C'CPR2'
04 – 07	Address of the issuer of the CPRC macro
08 – 18	RUPEURC
19 – 1F	Zero

CPWAIT (Part 1, for V3R1, V3R1.1, and V3R1.2)

Entry:	**CPWT**
VIT Option:	SSCP
Event:	CPWAIT Macro (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTCPCPW
Applies to:	MVS (V3R1.1 and V3R1.2), VM (V3R1.1 and V3R1.2), VSE (V3R1 and V3R1.2)

This trace record gives information about CPWAIT macroinstruction processing. It may help you determine why I/O is outstanding or why an SSCP request is failing.

This entry is eventually followed by a CPPT entry (to resume processing) or a CPPG entry (to purge the waiting task). Use the EID field to match corresponding entries.

00 01 02 03	04	05 06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B 1C 1D 1E 1F
CPWT	ID	0	EIDLN	LQAB ADDRESS	WREDATA IN WRE	WORK ELEMENT ADDRESS	OPC IN WORK ELEMENT	FIRST 8 BYTES OF CPCBURC

Byte (hex)	Contents
00–03	Record ID: C'CPWT'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05–06	Zero
07	Length of event ID
08–0B	Address of LQAB for this CPWAIT macro
0C–0F	Data field in ISTWRE when CPWAIT is performed
10–13	Work element address: Zero for asynchronous CPWAIT Non-zero for synchronous CPWAIT
14–17	CPCB operation code that indicates the original request for this work element (CPCB operation codes are shown in Appendix D, "CPCB Operation Codes" on page 645.)
18–1F	CPCBURC (first 8 bytes)

CPWAIT (Part 2, for V3R1, V3R1.1, and V3R1.2)

Entry:	**CPW2**
VIT Option:	SSCP
Event:	CPWAIT Macro (Part 2)
VIT Processing Module:	ISTRACSC
Applies to:	MVS (V3R1.1 and V3R1.2), VM (V3R1.1 and V3R1.2), VSE (V3R1 and V3R1.2)

This trace record is a continuation of the CPWAIT trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPW2				RETURN ADDRESS				FIRST 24 BYTES OF EID																							

Byte (hex)	Contents
00 – 03	Record ID: C'CPW2'
04 – 07	Address of issuer of CPWAIT macro
08 – 1F	Event ID (up to 24 bytes). See *VTAM Messages and Codes* for a list of EIDS.

CPWAIT for PVI or Non-PVI Events (Part 1, for V3R2)

Entry:	**CPWT**
VIT Option:	SSCP
Event:	CPWAIT Macro (Part 1)
VIT Processing Module:	ISTRACSC
Control is Returned to:	ISTCPCPW
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record gives information about CPWAIT macroinstruction processing. It may help you determine why I/O is outstanding or why an SSCP request is failing.

This entry is eventually followed by a CPPT entry (to resume processing) or a CPPG entry (to purge the waiting task). Use the EID field to match corresponding entries.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPWT				ID	0	TYPE	0	LQAB ADDRESS				WREDATA IN WRE				WORK ELEMENT ADDRESS				OPC IN WORK ELEMENT or 0				SAVE AREA ID or 0				0			

Byte (hex)	Contents
00–03	Record ID: C'CPWT'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Zero
06	Type flags • B'00000001' = PVI event • B'00000000' = non-PVI event
07	Zero
08–0B	LQAB address for this CPWAIT macro
0C–0F	Data field in ISTWRE when CPWAIT is performed
10–13	Work element address
14–17	CPCB operation code that indicates the original request for this work element or zero (CPCB operation codes are shown in Appendix D, "CPCB Operation Codes" on page 645.)
18–1B	For PVI event, caller's save area ID or zero For non- PVI event, save area ID
1C–1F	Zero

CPWAIT (Part 2, for V3R2)

Entry:	**CPW2**
VIT Option:	SSCP
Event:	CPWAIT Macro (Part 2)
VIT Processing Module:	ISTRACSC
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record is a continuation of the CPWAIT trace record.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">CPW2</td><td colspan="4">RETURN ADDRESS</td><td colspan="17">RUPEURC or 0</td><td colspan="7">0</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'CPW2'
04 – 07	Return address to issuer of the CPWAIT macroinstruction
08 – 18	RUPEURC or 0
19 – 1F	Zero

CPWAIT (Part 3, for V3R2)

Entry:	**CPW3**
VIT Option:	SSCP
Event:	CPWAIT Macro (Part 3)
VIT Processing Module:	ISTRACSC
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record is a continuation of the CPW2 trace record. If an EID is less than 24 bytes long, only the bytes in the actual EID will be moved to the corresponding fields in the trace records. The leftover bytes will be set to zero.

Note: This trace record is produced only when the WRE points to an EID event.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
CPW3				0			EIDLN	FIRST 24 BYTES OF EID																							

Byte (hex)	Contents
00 – 03	Record ID: C'CPW3'
04 – 06	Zero
07	Length of EID
08 – 1F	First 24 bytes of EID

CPWAIT (Part 4, for V3R2)

Entry:	**CPW4**
VIT Option:	SSCP
Event:	CPWAIT Macro (Part 4)
VIT Processing Module:	ISTRACSC
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record is a continuation of the CPW3 trace record. If an RU is less than 24 bytes long, only the bytes in the actual RU will be moved to the corresponding fields in the trace records. The leftover bytes will be set to zero.

Note: This trace record is produced only for PVI events where a RUPE is available.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">CPW4</td><td colspan="28">FIRST 24 BYTES OF RU or 0</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'CPW4'
04 – 1B	First 24 bytes of RU
1C – 1F	Zero

VSE SHM Disconnect

Entry:	VSE **DISC**
VIT Option:	CIO
Event:	Disconnection
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCVI
Applies to:	VSE

This trace record is generated for X.21 short-hold mode attachments. It is written when a port is disconnected from a station.

<table>
<tr><td>0
0</td><td>0
1</td><td>0
2</td><td>0
3</td><td>0
4</td><td>0
5</td><td>0
6</td><td>0
7</td><td>0
8</td><td>0
9</td><td>0
A</td><td>0
B</td><td>0
C</td><td>0
D</td><td>0
E</td><td>0
F</td><td>1
0</td><td>1
1</td><td>1
2</td><td>1
3</td><td>1
4</td><td>1
5</td><td>1
6</td><td>1
7</td><td>1
8</td><td>1
9</td><td>1
A</td><td>1
B</td><td>1
C</td><td>1
D</td><td>1
E</td><td>1
F</td></tr>
<tr><td colspan="4">DISC</td><td>T
I
K</td><td>C
F
S
M</td><td>P
M
F
S
M</td><td>S
H
S
I</td><td>0</td><td colspan="3">CUA</td><td colspan="4">GROUP
NCB
ADDRESS</td><td colspan="4">HALCB
ADDRESS</td><td colspan="4">PUT
ADDRESS</td><td colspan="8">0</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'DISC'
04	Task identification key
05	Connection finite state machine (FSM; see HALCFSM in ISTHALCB for values)
06	Port manager FSM (see HALPMFSM in ISTHALCB for values)
07	X.21 short-hold mode status indicator (see PUTFLG1 in ISTPUT)
08	Zero
09 – 0B	Channel unit address of link
0C – 0F	Group network control block address (ISTGRPCB)
10 – 13	Link network control block address (ISTHALCB)
14 – 17	Physical unit table address (ISTPUT)
18 – 1F	Zero

PAB Dispatch (Part 1, for V3R1, V3R1.1, and V3R1.2)

Entry:	**DISP**
VIT Option:	PSS
Event:	PAB Dispatch (Part 1)
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCPD
Applies to:	MVS (V3R1.1 and V3R1.2), VM (V3R1.1 and V3R1.2), VSE (V3R1 and V3R1.2)

The DISP entry marks the beginning of a PAB dispatch, which is VTAM's major "unit of work." This unit exists until terminated by an EXIT entry. While processing, it may wait for other events to complete (WAIT). It resumes processing with a RESM entry. The unit may be interrupted by a higher priority task and continue when that task is finished.

To correlate all events associated with this unit, look for the RPH address, which is in many trace records, and match it to the RPH address in the DISP trace record. This unit has exclusive use of that RPH until EXIT.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
DISP				ID	CBID	PABOF		PST ADDRESS				PAB ADDRESS				WORK ELEMENT QUEUE				PAB CHAIN FIELD				MODULE NAME or DVT ADDRESS				RPH ADDRESS			

Byte (hex)	Contents
00 – 03	Record ID: C'DISP'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID of work element (Control block IDs are shown in Appendix E, "Storage and Control Block ID Codes" on page 657.)
06 – 07	PAB offset
08 – 0B	PST address
0C – 0F	PAB address
10 – 13	Address of work element currently being dispatched
14 – 17	PAB chain field
18 – 1B	Module name abbreviation (bytes 4, 5, 7, and 8 of the name of the first module to get control), or PAB DVT address (high order bit of X'18' = 0)
1C – 1F	RPH address. The RPH address in location 1C – 1F appears in many other VIT entries associated with this particular dispatch. Use this RPH address to group events associated with this dispatch. The RPH is freed or reused after an EXIT trace record.

PAB Dispatch (Part 2, for V3R1, V3R1.1, and V3R1.2)

Entry:	**DIS2**
VIT Option:	PSS
Event:	PAB Dispatch (Part 2)
VIT Processing Module:	ISTRACPS
Applies to:	MVS (V3R1.1 and V3R1.2), VM (V3R1.1 and V3R1.2), VSE (V3R1 and V3R1.2)

This trace record is a continuation of PAB Dispatch (Part 1).

0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	1 9	1 A	1 B	1 C	1 D	1 E	1 F
DIS2				WORK ELEMENT ADDRESS or 0				F L A G S	0							PAB EXTENSION or 0															

Byte (hex)	Contents
00 – 03	Record ID: C'DIS2'
04 – 07	Address of work element, or zero. Zero means either that there was no work element, or that this process queues work elements somewhere other than the PAB.
08	Flag field (PABFLAGS)
09 – 0F	Zero
10 – 1F	PAB extension (PABXTNSN) if this is an extended PAB.

PAB Dispatch (V3R2)

Entry:	**DISP**
VIT Option:	PSS
Event:	PAB Dispatch
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCPD
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

Each PAB is identified by a unique index number. VTAM uses this number to find the called module and then creates a VIT entry for it.

The DISP entry marks the beginning of a PAB dispatch, which is VTAM's major "unit of work." This unit exists until terminated by an EXIT entry. While processing, it may wait for other events to complete (WAIT). It resumes processing with a RESM entry. The unit may be interrupted by a higher priority task and continue when that task is finished.

To correlate all events associated with this unit, look for the RPH address, which is in many trace records, and match it to the RPH address in the DISP trace record. This unit has exclusive use of that RPH until EXIT.

00 01 02 03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
DISP	ID	CBID	FLAG	0-WEQ	PST ADDRESS	PAB ADDRESS	LAST WORK ELEMENT ADDRESS	CURRENT WORK ELEMENT ADDRESS	MODULE NAME or DVT ADDRESS	RPH ADDRESS

Byte (hex)	**Contents**
00 – 03	Record ID: C'DISP'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID of work element (Control block IDs are shown in Appendix E, "Storage and Control Block ID Codes" on page 657.)
06	Flag field (PABFLAGS)
07	Zero, or PAB work element queue level dispatched for VARY extended PABS
08 – 0B	PST address
0C – 0F	PAB address
10 – 13	Address of work element most recently queued to the PAB
14 – 17	Address of work element currently being dispatched
18 – 1B	Module name abbreviation (bytes 4, 5, 7, and 8 of the name of the first module to get control), or PAB DVT address (high order bit of X'18' = 0)
1C – 1F	RPH address. The RPH address in location 1C – 1F appears in many other VIT entries associated with this particular dispatch. Use this RPH address to group events associated with this dispatch. The RPH is freed or reused after an EXIT trace record.

Discarded PIU Buffer (Part 1)

Entry:	**DSCD**
VIT Option:	PIU
Event:	Discarded PIU buffer (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTTSCUD
Applies to:	VSE (V3R2), MVS, VM

This trace record is written when VTAM's TSC utility discard routine, ISTTSCUD, disposes of PIU buffers because of an error condition. The reason code and module identifier can be used to explain the reason for the discard. This entry is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">DSCD</td><td>ID</td><td colspan="3">REASN CODE</td><td colspan="4">TSCB ADDRESS</td><td colspan="20">20 BYTES OF PIU</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'DSCD'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05 – 07	Discard reason code in hex. (PIU discard reason codes are explained in Appendix F, "PIU Discard Reason Codes" on page 661.)
08 – 0B	Address of TSCB
0C – 1F	First 20 characters of the PIU.

Discarded PIU Buffer (Part 2)

Entry:	**DSC2**
VIT Option:	PIU
Event:	Discarded PIU buffer (Part 2)
VIT Processing Module:	ISTRACOT
Applies to:	VSE (V3R2), MVS, VM

This trace record is a continuation of the DSCD entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
DSC2				24 MORE CHARACTERS OF THE PIU																								MODULE ID			

Byte (hex)	Contents
00 – 03	Record ID: C'DSC2'
04 – 1B	24 more characters of the PIU
1C – 1F	ID of the module that is discarding the PIU

MVS Error Recovery Procedures

Entry:	MVS ERP
VIT Option:	CIO
Event:	Error Recovery
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTZBM0J for LDNCB, or ISTZBM0K for ICNCB
Applies to:	MVS

This trace record is written during error recovery for channel I/O.

- ERPI is generated for communication controllers and local SNA cluster controllers.
- ERPL is generated for local non-SNA cluster controllers.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ERPI or ERPL				ID	STATE	0			CUA			NCB ADDRESS				FLAG BYTES				FLAG	CODE	SENSE		CSW							

Byte (hex)	Contents
00 – 03	Record ID: C'ERPI' for ICNCB C'ERPL' for LDNCB
04	Address space ID (or zero if address space ID is greater than X'FF')
05	Link station state (see NCBLNKST)
06 – 08	Zero
09 – 0B	Channel device name in EBCDIC (either a CUA or device number)
0C – 0F	NCB address
10 – 13	Flag bytes (NCBFLAGS)
14	Flag byte as follows: Bit Meaning ..1. .... Exception condition occurred (IOSEX flag is on) ...1 .... Error routine is in control (IOSERR flag is on)
15	I/O completion code: IOSCOD
16 – 17	Sense data in IOSSNS
18 – 1F	Channel status word from IOSB

VM Error Recovery Procedures (Part 1)

Entry:	VM ERP
VIT Option:	CIO
Event:	Error Recovery (Part 1)
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCZZ
Applies to:	VM

This trace record is written during error recovery for channel I/O.

- ERPB is generated for error recoveries at the channel interface for a BSC line attached to the communication adapter.
- ERPH is generated for error recoveries at the channel interface for an SDLC line attached to the communication adapter.
- ERPI is generated for communication controllers and local SNA cluster controllers.
- ERPL is generated for local non-SNA cluster controllers.
- ERPX is generated for channel-to-channel-attached hosts.

00 01 02 03	04	05	06 07 08	09 0A 0B	0C 0D 0E 0F	10 11 12 13	14	15	16 17	18 19 1A 1B 1C 1D 1E 1F
ERPB	ID	LFSM	0	CUA	NCB ADDRESS	FLAG BYTES	ERCOD	0	SENSE	CSW
ERPH	ID	LFSM	0	CUA	NCB ADDRESS	FLAG BYTES	0			CSW
ERPI	ID	LNKST	0	CUA	NCB ADDRESS	FLAG BYTES	ERCOD	0	SENSE	CSW
ERPL or ERPX	ID	LNKST	0	CUA	NCB ADDRESS	FLAG BYTES	0		SENSE	CSW

Byte (hex)	Contents
00–03	Record ID: C'ERPB' for BSCLB C'ERPH' for HALCB C'ERPI' for ICNCB C'ERPL' for LDNCB C'ERPX' for XCNCB
04	Machine and task ID
05	Link finite state machine (LFSM) or link station state (see NCBLNKST)
06–08	Zero
09–0B	Channel device name in EBCDIC (either a CUA or device number)
0C–0F	NCB address
10–13	Flag bytes (NCBFLAGS)
14	Error code (ERCOD) in NCB or zero
15	Return code or zero
16–17	NCB sense data in NCBSENSE or zero
18–1F	Channel status word from NCB

VM Error Recovery Procedures (Part 2)

Entry:	VM ERP2
VIT Option:	CIO
Event:	Error Recovery (Part 2)
VIT Processing Module:	ISTRACCI
Applies to:	VM

This trace record is written during error recovery for channel I/O. It is a continuation of the ERPB or ERPH trace entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ERP2				0				DEVICE-DEPENDENT INFORMATION																					0		

Byte (hex)	Contents
00 – 03	Record ID: C'ERP2'
04 – 07	Zero
08 – 1C	Device-dependent information shown below
1D – 1F	Zero

For SDLC links (NCBTYPE = 09):

08	Link error retry limit
09	Error retry count
0A	Connection finite state machine
0B	Retry operation code
0C – 0F	CCW address
10	Sense byte 1
11 – 12	NCB sense data area
13 – 1C	Zero

For BSC links (NCBTYPE = 0A):

08 – 0B	Zero
11 – 12	Sense data from NCBSENSE
13	Error code (NCBERCOD)
14 – 1C	Zero

VSE Error Recovery Procedures (Part 1)

Entry:	VSE **ERP**
VIT Option:	CIO
Event:	Error Recovery (Part 1)
VIT Processing Module:	ISTRACCI
Control is Returned to:	$$ABERV1
Applies to:	VSE

This trace record is written during error recovery for channel I/O.

- ERPB is generated for error recoveries at the channel interface for a BSC line attached to the communication adapter.
- ERPH is generated for error recoveries at the channel interface for an SDLC line attached to the communication adapter.
- ERPI is generated for communication controllers and local SNA cluster controllers.
- ERPL is generated for local non-SNA cluster controllers.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ERPB				ID	LFSM	0			CUA			NCB ADDRESS				FLAG BYTES				ERCOD	0	SENSE		CSW							
ERPH				ID	LFSM	0			CUA			NCB ADDRESS				FLAG BYTES				0	RCODE	0		CSW							
ERPI				ID	LNKST	ERACT	0		CUA			NCB ADDRESS				FLAG BYTES				ERCOD	RCODE	SENSE		CSW							
ERPL				ID	LNKST	ERACT	0		CUA			NCB ADDRESS				FLAG BYTES				0	RCODE	SENSE		CSW							

Byte (hex)	Contents
00–03	Record ID: C'ERPB' for BSCLB C'ERPH' for HALCB C'ERPI' for ICNCB C'ERPL' for LDNCB
04	Task ID
05	Link finite state machine (LFSM) or link station state (see NCBLNKST)
06	Error action (ERACT) or zero
07–08	Zero
09–0B	Channel device name in EBCDIC (either a CUA or device number)
0C–0F	NCB address
10–13	Flag bytes (NCBFLAGS)
14	Error code (ERCOD) in NCB or zero
15	Return code or zero
16–17	NCB sense data in NCBSENSE or zero
18–1F	Channel status word from NCB

VSE Error Recovery Procedures (Part 2)

Entry:	VSE **ERP2**
VIT Option:	CIO
Event:	Error Recovery (Part 2)
VIT Processing Module:	ISTRACCI
Applies to:	VSE

This trace record is written during error recovery for channel I/O. It is a continuation of the ERPB or ERPH trace entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ERP2				0				DEVICE-DEPENDENT INFORMATION																					0		

Byte (hex)	Contents
00–03	Record ID: C'ERP2'
04–07	Zero
08–1C	Device-dependent information shown below
1D–1F	Zero

For SDLC links (NCBTYPE=09):

08	Link error retry limit
09	Error retry count
0A	Connection finite state machine
0B	Retry operation code
0C–0F	CCW address
10	Sense byte 1
11–12	NCB sense data area
13–1C	Zero

For BSC links (NCBTYPE=0A):

08–0B	Zero
11–12	Sense data from NCBSENSE
13	Error code (NCBERCOD)
14–1C	Zero

VM VSE CIO Error Recovery (V3R1.2 and Later)

Entry:	VM VSE **ERP**
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTAPCES
Applies to:	VM VSE (V3R1.2 and later)

This trace record provides error recovery information for an X.25 port. An ERP entry is built on every entry to ERP for an X.25 port, whether for an error or for counter overflow or shutdown recording.

The telecommunications subsystem or communication adapter uses two channel programs: a control link channel program and a buffer pool channel program.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ERP			TYPE	ID	CUA			0	NCB ADDR			VIRTUAL CAW				FLAG BYTES				ERCOD		SENDE		0							

Byte (hex)	Contents
00 – 02	Record ID: C'ERPA'
03	Type: 0C = Control link channel program 0D = Buffer pool channel program
04	Machine and task ID
05 – 07	Channel device name in EBCDIC (either a CUA or device number)
08	Zero
09 – 0B	NCB address
0C – 0F	Virtual CAW
10 – 13	Flag bytes (NCBFLAGS)
14 – 15	Error code (ERCOD) in NCB or zero
16 – 17	NCB sense data in NCBSENSE or zero
18 – 1F	Zero

TPESC

Entry:	**ESC**
VIT Option:	ESC
Event:	TPESC Macro
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTESC01
Applies to:	MVS, VM, VSE

This trace record identifies the address of the next routine to get control in a destination vector table (DVT). The PAB address identifies the process. Using the RPH address, you can correlate this trace record to the DISP entry to find when the process was dispatched.

00 01 02 03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
ESC	ID	0	PABOF	DISP ADDRESS	PAB ADDRESS	RETURN ADDRESS	RPH WORK ELEMENT	MODULE NAME or DVT ADDRESS	RPH ADDRESS

Byte (hex)	Contents
00 – 03	Record ID: C'ESC'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Zero
06 – 07	PAB offset
08 – 0B	Dispatch address from DVT
0C – 0F	PAB address
10 – 13	Address of the issuer of the TPESC macro
14 – 17	RPH work element
18 – 1B	Module name abbreviation (bytes 4, 5, 7, and 8 of the name of the next module to get control), or PAB DVT address (high order bit of X'18' = 0)
1C – 1F	RPH address

TPEXIT

Entry:	**EXIT**
VIT Option:	PSS
Event:	TPEXIT Macro
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTX
Applies to:	MVS, VM, VSE

This trace record identifies a VTAM process (PAB) that has finished executing. The PAB is rescheduled if more work elements are waiting to be processed.

In most cases, the RPH (whose address is in location 1C – 1F) is freed or reused after this entry. The two exceptions are the front end message PAB (ATCFEMPB — module ISTCFCFG) and the SMS dynamic expansion DYPAB (ATCPXPAB — module ISTORFPX). These two PABs have dedicated RPHs that are not used for anything else.

This entry is the complement of DISP. An EXIT should eventually follow every DISP. Use the RPH address to correlate the entries.

00 01 02 03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
EXIT	ID	APNOP	PABOF	PST ADDRESS	PAB ADDRESS	RETURN ADDRESS	WORK ELEMENT QUEUE	MODULE NAME or DVT ADDRESS	RPH ADDRESS

Byte (hex)	Contents
00 – 03	Record ID: C'EXIT'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	TPEXIT options (see field APNFLGS in control block ISTAPNOP in *VTAM Data Areas*)
06 – 07	PAB offset, or zero if no PAB (see note)
08 – 0B	PST address
0C – 0F	PAB address, or zero if no PAB (see note)
10 – 13	Address of the issuer of the TPEXIT macro
14 – 17	PAB work element queue, or zero if no PAB (see note)
18 – 1B	Module name abbreviation (bytes 4, 5, 7, and 8 of the name of the next module to get control), or PAB DVT address (high order bit of X'18' = 0), or zero if no PAB (see note)
1C – 1F	RPH address

Note: There is no PAB if the running process has already freed the major control block that contains the PAB.

FREEBLK

Entry:	**FBLK**
VIT Option:	SMS
Event:	FREEBLK Macro
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTCPCSM
Applies to:	MVS, VM, VSE

This trace record provides the status of each FREEBLK request issued by VTAM components.

The FREEBLK macro is the complement of the GETBLK macro. FREEBLK must release the storage obtained by GETBLK. Each GBLK entry should eventually have a corresponding FBLK entry.

If the return code is non-zero, this entry is generated whether the SMS option is in effect or not. This event is treated as an exception condition, and therefore, is traced whenever the VIT is active.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
FBLK				ID	0	POOL	0	STORAGE ADDRESS or 0				ADDRESS OF SPANC				RETURN ADDRESS				LENGTH OF STORAGE FREED				CALLER OF UTILITY or 0				RETURN CODE			

Byte (hex) **Contents**

00—03 Record ID: C'FBLK'

04 ID

- For MVS, address space ID (or zero if address space ID is greater than X'FF')
- For VM, machine and task ID
- For VSE, task ID

05 Zero

06 Storage pool type (in hex):

Pool	Contains
00	Private RUPE
01	Common RUPE
02	VTAM private SIB
03	SSCP FMCB
04	Not used
05	DVT/EPT
06	Dynamic CDRSC
07	ACDEB/APPCB
08	HSQH
09	ERTE
0A	WRE and EID
0B	FMCB extension
0C	SIB extension
0D	RSQE
0E	UECB/VRPL
0F	IOBLOCK

	10	SRTE
	11	NetView trace element
	12	UTILPOOL/UTILPVTS
	13	Message timer blocks (DMTSQ)
	14	VTAM private POWE
	15	Common POWE
	16	PULURDTE
	17	PAQ
	18	RAQ
	19	CPWAPVT
	1A	ERICPOOL
	1B	SIBIX
	1C	CDAJSCP
	1D	GWNAJSCP
	1E	IOSIB
	1F	DSSIB
	20	UTILPVTL
	21	UTILCSAS
	22	UTILCSAL
	23	AMU
	24	HSICB
	25	LMHDR/LME/LMM
	26	SAB/NSI
	27	RAB
	28	PRIDBLK
	29	PRIDQAB
	2A	OCB/NACP
	2B	CPWACSA
07	Zero	
08–0B	Address of block freed (or zero if FREEBLK failed)	
0C–0F	Address of storage pool anchor block (SPANC)	
10–13	Address of the issuer of the FREEBLK macro	
14–17	Length of storage freed	
18–1B	Caller of utility routine or zero. If the FREEBLK macro was issued from a utility routine, then the address of the utility's caller is placed here. A zero address means that the macro was issued directly by the caller (see return address) without a utility routine.	
1C–1F	Return code	

GETBLK

Entry:	**GBLK**
VIT Option:	SMS
Event:	GETBLK Macro
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTCPCSM
Applies to:	**MVS, VM, VSE**

This trace record provides the status of each GETBLK request issued by VTAM components.

The FREEBLK macro is the complement of the GETBLK macro. FREEBLK must release the storage obtained by GETBLK. Each GBLK entry should eventually have a corresponding FBLK entry.

If the return code is non-zero, this entry is generated whether the SMS option is in effect or not. It is treated as an exception condition and is always traced, regardless of the SMS option, if the VIT is active.

00 01 02 03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
GBLK	ID	0	POOL	0	STORAGE ADDRESS or 0	ADDRESS OF SPANC	RETURN ADDRESS	LENGTH OF STORAGE REQUEST	CALLER OF UTILITY or 0	RETURN CODE

Byte (hex) **Contents**

00–03 Record ID: C'GBLK'

04 ID

- For MVS, address space ID
  (or zero if address space ID is greater than X'FF')
- For VM, machine and task ID
- For VSE, task ID

05 Zero

06 Buffer pool type (in hex):

Pool	Contains
00	Private RUPE
01	Common RUPE
02	VTAM private SIB
03	SSCP FMCB
04	Not used
05	DVT/EPT
06	Dynamic CDRSC
07	ACDEB/APPCB
08	HSQH
09	ERTE
0A	WRE and EID
0B	FMCB extension
0C	SIB extension
0D	RSQE
0E	UECB/VRPL
0F	IOBLOCK

	10	SRTE
	11	NetView trace element
	12	UTILPOOL/UTILPVTS
	13	Message timer blocks (DMTSQ)
	14	VTAM private POWE
	15	Common POWE
	16	PULURDTE
	17	PAQ
	18	RAQ
	19	CPWAPVT
	1A	ERICPOOL
	1B	SIBIX
	1C	CDAJSCP
	1D	GWNAJSCP
	1E	IOSIB
	1F	DSSIB
	20	UTILPVTL
	21	UTILCSAS
	22	UTILCSAL
	23	AMU
	24	HSICB
	25	LMHDR/LME/LMM
	26	SAB/NSI
	27	RAB
	28	PRIDBLK
	29	PRIDQAB
	2A	OCB/NACP
	2B	CPWACSA
07	Zero	
08 – 0B	Address of block obtained (or zero if GETBLK failed)	
0C – 0F	Address of storage pool anchor block (SPANC)	
10 – 13	Address of the issuer of the GETBLK macro	
14 – 17	Length of storage request	
18 – 1B	Caller of utility routine or zero. If the GETBLK macro was issued from a utility routine, then the address of the utility's caller is placed here.	
1C – 1F	Return code	

Halt I/O

Entry:	**HIO**
VIT Option:	CIO
Event:	Halt I/O
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCLR
Applies to:	MVS, VM, VSE

This trace record is written when a Halt I/O SVC is issued to end a currently executing channel program. This should be followed shortly by an interrupt (INT) entry. Be sure to match the CUAs in the HIO and INT entries to be sure they are for the same device. See *VTAM Data Areas* for a description of the NCB and XCNCB fields.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">HIOB
HIOH
HIOX</td><td>ID</td><td>STATE</td><td colspan="2">MODID</td><td>0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="4">FLAG BYTES</td><td colspan="12">0</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'HIOB', 'HIOH', or 'HIOX'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	For HIOB, link station state (BSCLKFSM) For HIOH, link station state (HALLFSM) For HIOX, station state (XCNSSFSM)
06 – 07	Module ID (last 2 characters of TSC module name)
08	Zero
09 – 0B	Channel device name in EBCDIC (either a CUA or device number)
0C – 0F	NCB address
10 – 13	Flag bytes (NCBFLAGS)
14 – 1F	Zero

VM VSE Halt I/O (V3R1.2 and Later)

Entry:	VM VSE **HIOP**
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRAC6T
Control is Returned to:	ISTTSC6L or ISTTSC6P
Applies to:	VM (V3R1.2 and later), VSE (V3R2)

This trace record is created when a Halt I/O SVC is issued to end a currently executing channel program. This may be followed by an interrupt (INT) entry. Match the CUAS in the HIO and INT entries to be sure they are for the same device. A Halt I/O SVC is issued against each of four CUAS during initialization or termination of the Token-Ring Subsystem.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
HIOP				ID	CPFSM	MODID		0	CUA			NCB ADDRESS				PDATA		0													

Byte (hex)	Contents
00 – 03	Record ID: C'HIOP'
04	Machine and task ID
05	Channel program FSM
	00 Reset
	01 Sense ID
	02 Interrupt port
	03 Control port
	04 Running
	05 Sense
	06 Orderly halt (HLTC)
	07 Termination
	08 Shutdown
06 – 07	Module ID (last 2 characters of TSC module name)
08	Zero
09 – 0B	Channel device name in EBCDIC (either a CUA or device number)
0C – 0F	NCB address
10	Port state (PDATA)
	00 Reset
	01 Halt
	02 Halt complete
	03 Sense ID
	04 Sense ID complete
	05 Interrupt pending
	06 Active
	07 Sense

11	Port type (PDATA)	
	00	Interrupt
	01	Control
	02	Inbound
	03	Outbound
12–1F	Zero	

MVS Interrupt

Entry:	MVS **INT**
VIT Option:	CIO
Event:	Channel Interrupt
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCLC
Applies to:	MVS

This trace record is written when a channel program interrupt occurs.

Every INT entry has a previously issued SIO entry, although it may have been issued some time ago. Use the CUA field to correlate SIO and INT entries. The SIO entry provides additional information about this completing channel program.

See *VTAM Data Areas* for a description of the NCB fields. See the diagnostic manuals for your operating system for a description of the IOS fields.

00 01 02 03	04	05	06	07 08	09 0A 0B	0C 0D 0E 0F	10 11 12 13	14	15	16 17	18 19 1A 1B 1C 1D 1E 1F
INTI INTL INTX	ID	STATE	EOPCD	0	CUA	NCB ADDRESS	FLAG BYTES	FLAG	CODE	SENSE or 0	CSW

Byte (hex)	Contents
00 – 03	Record ID: C'INTI' for ICNCB C'INTL' for LDNCB C'INTX' for XCNCB
04	Address space ID (or zero if address space ID is greater than X'FF')
05	Link station state (NCBLNKST) for ICNCB and LDNCB Station state (XCNSSFSM) for XCNCB
06	Ending operation code
07 – 08	Zero
09 – 0B	Channel device name in EBCDIC (either a CUA or device number)
0C – 0F	NCB address
10 – 13	Flag bytes (NCBFLAGS)
14	Flag byte as follows:

Bit	Meaning
1...	Running in disabled interrupt exit (IOSDIE is not zero)
.1..	ERP work area address exists (IOSERP is not zero)
..1.	Exception condition (IOSEX flag is on)
...1	Error routine in control (IOSERR flag is on)

Byte (hex)	Contents
15	I/O completion code: IOSCOD
16 – 17	Sense data in IOSSNS if this was a sense channel program; otherwise, zero
18 – 1F	Channel status word from IOSB.

VM Interrupt (Part 1)

Entry:	VM INT
VIT Option:	CIO
Event:	Channel Interrupt (Part 1)
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCLC
Applies to:	VM

This trace record is written when a channel program interrupt occurs.

Every INT entry has a previously issued SIO entry, although it may have been issued some time ago. Use the CUA field to correlate SIO and INT entries. The SIO entry provides additional information about this completing channel program.

See *VTAM Data Areas* for a description of the NCB fields. See the diagnostic manuals for your operating system for a description of the IOS fields.

00 01 02 03	04	05	06	07 08	09 0A 0B	0C 0D 0E 0F	10 11 12 13	14	15	16 17	18 19 1A 1B 1C 1D 1E 1F
INTB or INTH	ID	LFSM	0		CUA	NCB ADDRESS	FLAG BYTES	0	CODE	SENSE	CSW
INTI	ID	LNKST	0		CUA	NCB ADDRESS	FLAG BYTES	0	CODE	SENSE	CSW
INTL	ID	LNKST	0		CUA	NCB ADDRESS	FLAG BYTES	0	CODE	0	CSW
INTX	ID	STATE	EOPCD	0	CUA	NCB ADDRESS	FLAG BYTES	FLAG	CODE	SENSE or 0	CSW

Byte (hex) **Contents**

00 – 03 Record ID:

- C'INTB' for BSCTYPE
- C'INTH' for HALTYPE
- C'INTI' for ICNCB
- C'INTL' for LDNCB
- C'INTX' for XCNCB

04 Machine and task ID

05 For INTB and INTH, link finite state machine (LFSM)
For INTI and INTL, link station state (NCBLNKST)
For INTX, station state (XCNSSFSM)

06 Ending operation code or zero

07–08 Zero

09–0B Channel device name in EBCDIC (either a CUA or device number)

0C–0F NCB address

10–13 Flag bytes (NCBFLAGS)

14 Flag byte as follows, or zero

Bit	Meaning
1...	Running in disabled interrupt exit (IOSDIE is not zero)
.1..	ERP work area address exists (IOSERP is not zero)
..1.	Exception condition (IOSEX flag is on)
...1	Error routine in control (IOSERR flag is on)

15 I/O completion code: IOSCOD

16–17 NCB sense data in NCBSENSE if this was a sense channel program; otherwise, zero

18–1F Channel status word from NCB.

VM Interrupt (Part 2)

Entry:	**VM INT2**
VIT Option:	CIO
Event:	Channel Interrupt (Part 2)
VIT Processing Module:	ISTRACCI
Applies to:	VM

This trace record is a continuation of the INT entry.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">INT2</td><td colspan="4">0</td><td colspan="12">DEVICE-DEPENDENT INFORMATION</td><td colspan="12">0</td></tr>
</table>

Byte (hex)	Contents
00–03	Record ID: C'INT2'
04–07	Zero
08–13	Device-dependent information shown below
14–1F	Zero

For SDLC links (NCBTYPE=09):

08	Return code
09	Buffer offset (SPLOFFST)
0A	SDLC station address (SPLADDR)
0B	Control flag (SPLCTLFG)
0C	Current SDLC number of last I-frame sent (SPLNSCUR)
0D	Last acknowledged I-frame (SPLNSACK)
0E	Next I-frame to be received (SPLNRACC)
0F	Command-in field (SCXCMDIN)
10	Count of frames sent and acknowledged (SCXCFRS)
11	SCXCRBUF
12–13	Exception flags and exception code (SCXEXFCD)

For BSC links (NCBTYPE=0A):

08	First selection character (BSCSEL1)
09	First device address (BSCDEV1)
0A	Index of BPL entry for which input data was received (BSCPESV)
0B	Device address from which input data was received (BSCDEVSV)
0C	Index for last station for which there was output data (BSCISOD)
0D	First response byte (BSCRSPA1)
0E	Second response byte (BSCRSPA2)
0F	Last block type received (BSCLBR)
10	Current acknowledgment (BSCALTAK)
11	Error retry count (BSCRCNT)
12	Connection manager flags (BSCCMIO)
13	BSC flags (BSCFLAGS)

VM VSE Interrupt (V3R1.2 and Later)

Entry:	VM VSE **INT**
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRAC6T
Control is Returned to:	ISTTSC6E
Applies to:	VM (V3R1.2 and later), VSE (V3R2)

This trace record is created when a channel program interrupt occurs.

If the INT entry signals the completion of a channel program, it has a previously issued SIO entry, although it may have been issued some time ago. Use the CUA field to correlate SIO and INT entries. The SIO entry provides additional information about this completing channel program. If the CUA field contains the address specified on the PORT statement, this INT entry may not be related to an SIO entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
INTP				ID	CPFSM	0			CUA			NCB ADDRESS				PDATA		0		STIM	CODE	SENSE	0	ENDING CSW							

Byte (hex)	Contents
00 – 03	Record ID: C'INTP'
04	Machine and task ID
05	Channel program FSM
	00 Reset
	01 Sense ID
	02 Interrupt port
	03 Control port
	04 Running
	05 Sense
	06 Orderly halt (HLTC)
	07 Termination
	08 Shutdown
06 – 08	Zero
09 – 0B	Channel device name in EBCDIC (either a CUA or device number)
0C – 0F	NCB address
10	Port state (PDATA)
	00 Reset
	01 Halt
	02 Halt complete
	03 Sense ID
	04 Sense ID complete
	05 Interrupt pending
	06 Active
	07 Sense

11	Port type (PDATA)	
	00	Interrupt
	01	Control
	02	Inbound
	03	Outbound
12–13	Zero	
14	I/O stimulus for FSM manager	
	06	Normal completion
	07	Attention
	08	Attention and unit exception
	09	Unit check
	0A	Error
15	Channel-command code for the last CCW	
	04	Sense (3088 compatibility)
	08	TIC
	C1	Data buffer write
	C2	Data buffer read
	C4	Sense control unit state
	C5	Control block write
	C6	Control block read
	C7	Data synchronization
	C9	Data parameters write
	CB	Set CETI mode on
	CD	Interrupt parameters write
	E4	Sense ID
16	First byte of sense data	
	80	Command reject
	02	Control block reject
17	Zero	
18–1F	Channel status word.	

VSE Interrupt (Part 1)

Entry:	VSE **INT**
VIT Option:	CIO
Event:	Channel Interrupt (Part 1)
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCLC
Applies to:	VSE

This trace record is written when a channel program interrupt occurs.

Every INT entry has a previously issued SIO entry, although it may have been issued some time ago. Use the CUA field to correlate SIO and INT entries. The SIO entry provides additional information about this completing channel program.

See *VTAM Data Areas* for a description of the NCB fields. See the diagnostic manuals for your operating system for a description of the IOS fields.

<table>
<tr><th>0
0</th><th>0
1</th><th>0
2</th><th>0
3</th><th>0
4</th><th>0
5</th><th>0
6</th><th>0
7</th><th>0
8</th><th>0
9</th><th>0
A</th><th>0
B</th><th>0
C</th><th>0
D</th><th>0
E</th><th>0
F</th><th>1
0</th><th>1
1</th><th>1
2</th><th>1
3</th><th>1
4</th><th>1
5</th><th>1
6</th><th>1
7</th><th>1
8</th><th>1
9</th><th>1
A</th><th>1
B</th><th>1
C</th><th>1
D</th><th>1
E</th><th>1
F</th></tr>
<tr><td colspan="4">INTB
or
INTH</td><td>ID</td><td>LFSM</td><td colspan="3">0</td><td colspan="3">CUA</td><td colspan="4">NCB
ADDRESS</td><td colspan="4">FLAG
BYTES</td><td>0</td><td>CODE</td><td colspan="2">SENSE</td><td colspan="8">CSW</td></tr>
<tr><td colspan="4">INTI</td><td>ID</td><td>LNKST</td><td>ERACT</td><td colspan="2">0</td><td colspan="3">CUA</td><td colspan="4">NCB
ADDRESS</td><td colspan="4">FLAG
BYTES</td><td>ERCOD</td><td>CODE</td><td colspan="2">SENSE</td><td colspan="8">CSW</td></tr>
<tr><td colspan="4">INTL</td><td>ID</td><td>LNKST</td><td>ERACT</td><td colspan="2">0</td><td colspan="3">CUA</td><td colspan="4">NCB
ADDRESS</td><td colspan="4">FLAG
BYTES</td><td>0</td><td>CODE</td><td colspan="2">0</td><td colspan="8">CSW</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'INTB' for BSCTYPE C'INTH' for HALTYPE C'INTI' for ICNCB C'INTL' for LDNCB
04	Task ID
05	For INTB and INTH, link finite state machine (LFSM) For INTI and INTL, link station state (NCBLNKST)
06	ERACT or zero
07 – 08	Zero
09 – 0B	Channel device name in EBCDIC (either a CUA or device number)
0C – 0F	NCB address
10 – 13	Flag bytes (NCBFLAGS)

14	ERCOD or zero
15	I/O completion code: IOSCOD
16—17	NCB sense data in NCBSENSE if this was a sense channel program; otherwise, zero
18—1F	Channel status word from NCB.

VSE Interrupt (Part 2)

Entry:	VSE **INT2**
VIT Option:	CIO
Event:	Channel Interrupt (Part 2)
VIT Processing Module:	ISTRACCI
Applies to:	VSE

This trace record is a continuation of the INT entry.

00 01 02 03	04 05 06 07	08 09 0A 0B 0C 0D 0E 0F 10 11 12 13	14 15 16 17 18 19 1A 1B 1C 1D 1E 1F
INT2	0	DEVICE-DEPENDENT INFORMATION	0

Byte (hex)	**Contents**
00 – 03	Record ID: C'INT2'
04 – 07	Zero
08 – 13	Device-dependent information shown below
14 – 1F	Zero

For SDLC links (NCBTYPE = 09):

08	Return code
09	Buffer offset (SPLOFFST)
0A	SDLC station address (SPLADDR)
0B	Control flag (SPLCTLFG)
0C	Current SDLC number of last I-frame sent (SPLNSCUR)
0D	Last acknowledged I-frame (SPLNSACK)
0E	Next I-frame to be received (SPLNRACC)
0F	Command-in field (SCXCMDIN)
10	Count of frames sent and acknowledged (SCXCFRS)
11	SCXCRBUF
12 – 13	Exception flags and exception code (SCXEXFCD)

For BSC links (NCBTYPE = 0A):

08	First selection character (BSCSEL1)
09	First device address (BSCDEV1)
0A	Index of BPL entry for which input data was received (BSCPESV)
0B	Device address from which input data was received (BSCDEVSV)
0C	Index for last station for which there was output data (BSCISOD)
0D	First response byte (BSCRSPA1)
0E	Second response byte (BSCRSPA2)
0F	Last block type received (BSCLBR)
10	Current acknowledgment (BSCALTAK)
11	Error retry count (BSCRCNT)
12	Connection manager flags (BSCCMIO)
13	BSC flags (BSCFLAGS)

VM VSE Interrupt (V3R1.2 and Later)

Entry:	VM VSE **INT**
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTAPCES
Applies to:	VM VSE (V3R1.2 and later)

This trace record provides information about interruptions at the telecommunications subsystem or communication adapter interface. An entry is built for each I/O interrupt for an X.25 port.

The telecommunications subsystem or communication adapter uses two channel programs: a control link channel program and a buffer pool channel program.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
INT			TYPE	ID	CUA			CSW		SENSE_STATUS CCW																		NCB ADDR			

Byte (hex)	Contents
00–02	Record ID: C'INT'
03	Type: 0C = Control link channel program 0D = Buffer pool channel program
04	Machine and task ID
05–07	Channel device name in EBCDIC (either a CUA or device number)
08–09	Channel status word from NCB
0A–1B	NCB sense data in NCBSENSE
1C–1F	NCB address

TPIO Request (Part 1)

Entry:	IO1
VIT Option:	API
Event:	TPIO Request, Part 1
VIT Processing Module:	ISTRACAP
Control is Returned to:	ISTAPCRS or ISTOCCSM
Applies to:	MVS, VM, VSE

This trace record shows API requests. It also provides the RPL information supplied by a VTAM application program when it issues an RPL-based macroinstruction (such as SEND or OPNDST). The API routine ISTAICIR issues the TPIO SVC. ISTAPCTI gets control from the SVC, does validity checking, and queues the RPL to the PST request PAB. After the PST request PAB is dispatched, ISTAPCRS gets control and writes the entry, or calls ISTOCCSM to write the entry.

See *VTAM Data Areas* for an explanation of the RPL fields.

The process that is started with this event will be completed by a post (UP), or an exit (RE or UE). To correlate the two entries, match the RPL addresses.

00 01 02	03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10	11	12	13	14	15 16 17	18 19 1A 1B	1C 1D 1E 1F
I01	REQTY	ID	EXTDS	0	OPT1	RPL ADDRESS	NIB ADDRESS or CID	RH3	SRTYP	VTFL1	VTFL2	CHN	RU CNTL CODE	OPTC2	OPTC3

Byte (hex) **Contents**

00–02 Record ID: C'IO1'

03 RPL request type (RPLREQ)

X'15'	SETLOGON
X'16'	SIMLOGON
X'17'	OPNDST
X'1A'	INQUIRE
X'1B'	INTRPRET
X'1F'	CLSDST
X'22'	SEND
X'23'	RECEIVE
X'24'	RESETSR
X'25'	SESSIONC
X'27'	SENDCMD
X'28'	RCVCMD
X'29'	REQSESS
X'2A'	OPNSEC
X'2C'	TERMSESS

04 ID

- For MVS, address space ID
  (or zero if address space ID is greater than X'FF')
- For VM, machine and task ID
- For VSE, task ID

05	Exit definition (RPLEXTDS)
06	Zero
07	Option code byte 1 (RPLOPT1)
08–0B	RPL address
0C–0F	NIB address or CID
10	Third byte of RH (RPLRH3)
11	Send/receive type (RPLSRTYP)
12	VTAM flags (RPLVTFL1)
13	Post/respond flags (RPLVTFL2)
14	RU chain position (RPLCHN)
15–17	RU control codes (RPLCNTRL)
18–1B	VTAM options (RPLOPTC2)
1C–1F	VTAM options (RPLOPTC3)

MVS VSE TPIO Request (Part 2)

Entry:	MVS VSE IO2
VIT Option:	API
Event:	TPIO Request, Part 2
VIT Processing Module:	ISTRACAP
Applies to:	MVS, VSE

This trace record is a continuation of the IO1 entry. It shows additional information about the PLU and SLU in a session established or terminated by a SIMLOGON, OPNDST, CLSDST, REQSESS, OPNSEC, or TERMSESS macroinstruction. This information includes some of the parameters exchanged and the names of the PLU and SLU.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
I02			0	RETURN ADDRESS				LOGON MODE NAME or 0								PLU NAME (if available)								SLU NAME (if available)							

Byte (hex)	Contents
00–02	Record ID: C'IO2'
03	Zero
04–07	Address of the issuer of the macro
08–0F	For SIMLOGON, OPNDST, CLSDST, or REQSESS: Logon mode name (NIBLMODE) or zero if the NIB is not available For OPNSEC or TERMSESS: Zero
10–17	PLU name or zero
18–1F	SLU name or zero

MVS VSE TPIO Request (Part 3)

Entry:	MVS VSE I03
VIT Option:	API
Event:	TPIO Request, Part 3
VIT Processing Module:	ISTRACAP
Applies to:	MVS, VSE

This trace record is a continuation of the I02 entry. It shows additional information in a session established or terminated by a SIMLOGON, OPNDST, CLSDST, REQSESS, or OPNSEC macroinstruction.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
I03			0	ADDRESS OF USER DATA, BIND, or 0				0																							

Byte (hex)	Contents
00 – 02	Record ID: C'I03'
03	Zero
04 – 07	For SIMLOGON, CLSDST, or REQSESS: Address of user data (RPLAREA) For OPNDST or OPNSEC: Address of BIND (NIBNDAR) or zero if the NIB is not available
08 – 1F	Zero

VM TPIO Request (Part 2)

Entry:	VM I02
VIT Option:	API
Event:	TPIO Request, Part 2
VIT Processing Module:	ISTRACAP
Applies to:	VM

This trace record is a continuation of the IO1 entry. It shows information about the PLU and SLU, such as some of the parameters exchanged and the names of the PLU and SLU.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
I02			0	RETURN ADDRESS				LOGON MODE NAME or 0								PLU NAME (if available)								SLU NAME (if available)							

Byte (hex)	Contents
00–02	Record ID: C'I02'
03	Zero
04–07	Address of the issuer of the macro
08–0F	Logon mode name (NIBLMODE) or zero if the NIB is not available
10–17	PLU name or zero
18–1F	SLU name or zero

VM TPIO Request (Part 3)

Entry:	VM I03
VIT Option:	API
Event:	TPIO Request, Part 3
VIT Processing Module:	ISTRACAP
Applies to:	VM

This trace record is a continuation of the IO2 entry.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
I03			0	ADDRESS OF BIND or 0				0																							

Byte (hex)	Contents
00 – 02	Record ID: C'IO3'
03	Zero
04 – 07	Address of BIND (NIBNDAR) or zero if the NIB is not available
08 – 1F	Zero

MVS IRB Dispatch

Entry:	MVS **IRBD**
VIT Option:	PSS
Event:	IRB Dispatch
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCIE
Applies to:	MVS

This trace record provides information when an IRB is dispatched. See the *Debugging* handbook for your operating system for a map of the TCB.

Note: In addition to specifying the PSS option, you must also set the RACON bits to B'1' for this trace record to be created.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">IRBD</td><td>ID</td><td colspan="3">0</td><td colspan="4">PST ADDRESS</td><td colspan="4">PST SYNC TPPOSTD QUEUE</td><td colspan="4">PST SYNC NORMAL QUEUE</td><td colspan="4">IRB ADDRESS</td><td colspan="4">TCB ADDRESS</td><td colspan="2">FLG1</td><td colspan="2">0</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'IRBD'
04	Address space ID (or zero if address space ID is greater than X'FF')
05 – 07	Zero
08 – 0B	PST address
0C – 0F	PST synchronous TPPOSTed queue anchor (PSTSTPP)
10 – 13	PST synchronous normal queue anchor (PSTSNRM)
14 – 17	IRB address (TCBRBP)
18 – 1B	TCB address (PSTTCBA)
1C – 1D	PST flags (PSTFLG1)
1E – 1F	Zero

MVS IRB Exit

Entry:	MVS **IRBX**
VIT Option:	PSS
Event:	IRB EXIT
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCSD
Applies to:	MVS

This trace record provides information about an IRB exit.

Note: In addition to specifying the PSS option, you must also set the RACON bits to B'1' for this trace record to be created.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
IRBX				ID	0			PST ADDRESS				NEW TCB ADDRESS				OLD TCB ADDRESS				NEW ASCB ADDRESS				OLD ASCB ADDRESS				IRB ADDRESS			

Byte (hex)	Contents
00 – 03	Record ID: C'IRBX'
04	Address space ID (or zero if address space ID is greater than X'FF')
05 – 07	Zero
08 – 0B	PST address
0C – 0F	New TCB address
10 – 13	Old TCB address
14 – 17	New ASCB address
18 – 1B	Old ASCB address
1C – 1F	IRB address

VM VSE LAN Interface Control (V3R1.2 and Later)

Entry:	VM VSE LIC
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRAC6T
Control is Returned to:	ISTTSC6R for LICI ISTTSC6W for LICO ISTTSC6K or ISTTSC6R for LICS
Applies to:	VM (V3R1.2 and later), VSE (V3R2)

One of these trace records is created when a command is sent to or received from the Token-Ring Subsystem.

An LICI record is written when a command is received from the subsystem. This entry always corresponds to a previous ADPA or ADPI entry. Use the TSCB address to correlate these entries.

An LICO record is written when a command is ready to be sent to the subsystem. If the command is transferring a PIU onto the ring, it corresponds to a previous PIU entry. The TSCB continuation address of the LICO entry correlates to the TSCB address of the PIU entry.

An LICS record is written when a special condition related to controlling the subsystem is detected.

00 01 02 03	04	05 06 07	08 09 0A 0B	0C 0D 0E 0F	10	11 12 13	14 15	16 17 18	19 1A 1B	1C 1D 1E 1F
LICI LICO	ID	CUA	TSCB ADDRESS	NCB ADDRESS	CBID	CMD	TFSMS	VFSMS	FLAGS	TSCB CONT ADDRESS
LICS	ID	CUA	TSCB ADDRESS	NCB ADDRESS	CBID	CMD	TFSMS	REASON		

Byte (hex)	Contents
00 – 03	Record ID: C'LICI' for an inbound record C'LICO' for an outbound record C'LICS' for a special record.
04	Machine and task ID
05 – 07	Channel device name in EBCDIC (either a CUA or device number)
08 – 0B	TSCB address
0C – 0F	NCB address
10	Control block ID of NCB 0E NCB for local station (VTAM) 0B NCB for remote station

11 – 13 Command

See "VM VSE Command Values Reported in Trace Entries" on page 297 for more information about this field.

14 LLC Port FSM (TFSMS)
01 Logon pending
02 Logon completion
03 Logoff pending
04 Enable pending
05 Disable pending

15 Channel Program FSM (TFSMS)
00 Reset
01 Sense ID
02 Interrupt port
03 Control port
04 Running
05 Sense
06 Orderly halt (HLTC)
07 Termination
08 Shutdown

For LICI or LICO:

16 LLC Link FSM if CBID=X'0B' (VFSMS)
00 Reset
10 ACTLINK pending
20 Active
30 DACTLINK pending

17 LLC Connection FSM if CBID=X'0B' (VFSMS)
00 Reset
10 Connect in
20 Connect out
30 Connected
40 XID pending

18 LLC Station FSM if CBID=X'0B' (VFSMS)
00 Reset
10 XID zero pending
20 Set mode pending
30 Set mode wait
40 Active state
50 Request discontact
60 Disconnect pending

19 – 1B Remote station flags if CBID=X'0B'

1C – 1F TSCB continuation address

For LICS:

16 – 1F Special reason text:

C'CLEAN UP' VTAM has completed shutdown of the Token-Ring Subsystem. This entry is preceded by a LICS C'FINAL TERM' entry.

C'DISCARD' Subsystem input has been discarded.

C'FINAL TERM' Channel programs to the subsystem have been stopped.

C'HALT RQST' VTAM is initiating shutdown of the subsystem.

TPLOCK Exclusive

Entry:	**LKEX**
VIT Option:	LOCK
Event:	TPLOCK Exclusive
VIT Processing Module:	ISTRACLK
Control is Returned to:	ISTAPC36
Applies to:	MVS, VM, VSE

This trace record identifies a request for exclusive control of a VTAM lock and the current status of a lock. Exclusive locks can be held by only one VTAM process. When a TPWAIT entry immediately follows this entry, it *could* be because a wait-for-lock condition exists. Look at the lockword in the trace output to see if the lock is already held.

Locks obtained by LKEX are eventually released by UNLK or ULKA. See Figure 9 on page 42 for a list of VTAM locks.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">LKEX</td><td>ID</td><td>0</td><td>LOCK</td><td>LEVEL</td><td colspan="4">LOCK ADDRESS</td><td colspan="4">CRA LKACT</td><td colspan="4">RETURN ADDRESS</td><td colspan="8">LOCKWORD</td><td colspan="4">RPH ADDRESS</td></tr>
</table>

Byte (hex)	Contents
00–03	Record ID: C'LKEX'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Zero
06–07	Level of lock to be obtained
08–0B	Address of lock to be obtained
0C–0F	CRA lock account word (CRALKACT; see step 11 on page 41)
10–13	Address of the issuer of the TPLOCK macro
14–1B	Lockword for above level of lock
1C–1F	RPH address

TPLOCK Shared

Entry:	**LKSH**
VIT Option:	LOCK
Event:	TPLOCK Shared
VIT Processing Module:	ISTRACLK
Control is Returned to:	ISTAPC35
Applies to:	MVS, VM, VSE

This trace record identifies a request for a shared VTAM lock and the current status of a lock. Shared locks can be held by more than one VTAM process at a time. When a TPWAIT entry immediately follows this entry, it *could* be because a wait-for-lock condition exists. Look at the lockword in the trace output to see if the lock is already held.

Locks obtained by LKEX are eventually released by UNLK or ULKA. See Figure 9 on page 42 for a list of VTAM locks.

00 01 02 03	04	05	06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17 18 19 1A 1B	1C 1D 1E 1F
LKSH	ID	0	LOCK	LEVEL	LOCK ADDRESS	CRA LKACT	RETURN ADDRESS	LOCKWORD	RPH ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'LKSH'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Zero
06–07	Level of lock to be obtained
08–0B	Address of lock to be obtained
0C–0F	CRA lock account word (CRALKACT; see step 11 on page 41)
10–13	Address of the issuer of the TPLOCK macro
14–1B	Lockword
1C–1F	RPH address

Lost Trace Record

Entry:	**LOST**
VIT Option:	N/A
Event:	Lost Trace Record
VIT Processing Module:	ISTRACRR
Control is Returned to:	The location shown in register 14
Applies to:	MVS, VM, VSE

This trace record indicates that the internal trace table is incomplete. Information from events that were supposed to be traced was lost because there were not enough save areas available for the tracing routine to record the information.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
LOST				ID	0			REG 14				LOST TRACE RECORD																			

Byte (hex)	**Contents**
00 – 03	Record ID: C'LOST'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05 – 07	Zero
08 – 0B	Register 14 (return address)
0C – 1F	C'LOST TRACE RECORD'

TPMSG or CPMSG (Part 1)

Entry:	**MSG**
VIT Option:	MSG
Event:	TPMSG Macro or CPMSG Macro (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTCFCTM
Applies to:	**MVS, VM, VSE**

This trace record is written each time a message is issued by a VTAM module. It can be used to find the originator of a message and to see whether it was solicited or unsolicited.

MSG trace entries are generated for *all* messages, even those that are suppressed by message-flooding prevention (described in *VTAM Messages and Codes*) or by the MODIFY SUPP command (described in *VTAM Operation*). MSG trace entries thus provide information that may be missing from the operator's console.

TSO/VTAM messages (beginning with IKT) are not traced in the VIT. See Chapter 4, "**MVS** Procedures for TSO/VTAM Problems" for TSO/VTAM information.

00 01 02 03	04	05 06	07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B 1C 1D 1E 1F
MSG	ID	0	S or U	SAVE AREA ADDRESS	MSG ID or 'USS '	RETURN ADDRESS	MODULE ID	DESTINATION

Byte (hex)	**Contents**
00–03	Record ID: C'MSG'
04	ID - For MVS, address space ID (or zero if address space ID is greater than X'FF') - For VM, machine and task ID - For VSE, task ID
05–06	Zero
07	'S' or 'U'. If this field is 'S', the message was solicited (in direct response to a VTAM command). If this field is 'U', the message was unsolicited.
08–0B	Address of the issuer of the macro
0C–0F	Message ID or 'USS'. If USS, the message ID appears in the variable text field.
10–13	Save area address
14–17	Issuing module ID (4th, 5th, 7th, and 8th characters of the issuing module name)

18–1F Destination. If byte 07 is 'S', then this field contains one of the following:

- 'SYSTEMxx', where xx is the ID of the system console. This is the name of the program operator application that requested the message.
- 'NLDM', when the message is destined for NLDM.
- 'URC', when the message is to be sent to the system console using routing codes, instead of to one console in particular.

If byte 07 is 'U', this field is zero.

Message Sent Record (Part 1)

Entry:	**MSGS**
VIT Option:	MSG
Event:	Message Sent Record (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTCFCWM or ISTLUCRC
Applies to:	MVS, VM, VSE

This trace record is written each time an attempt is made to send a message to its destination. It can be used to correlate the console sheet with the internal trace. The destination may indicate that the message was sent to a system console (via a WTO macro) or to a program operator application, such as the NetView program (via a RCVCMD macro).

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
MSGS				I D	0			DESTINATION								MESSAGE HEADER				FIRST 12 CHARACTERS OF MESSAGE TEXT											

Byte (hex)	Contents
00 – 03	Record ID: C'MSGS'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05 – 07	Zero
08 – 0F	Destination. If 'SYSTEMxx', the message was sent to system console 'xx', via a WTO macro. If an application name, the message was sent to a program operator application, via a RCVCMD macro.
10 – 13	Message header (in hex). If destination is 'SYSTEMxx', this will be the WPL header. If destination is a program operator application, this will be a program operator header.
14 – 1F	First 12 characters of the message text. If the message text is longer than 12 characters, it will continue in MSG2.

TPMSG or CPMSG or Message Sent Record (Part 2)

Entry:	**MSG2**
VIT Option:	MSG
Event:	TPMSG Macro or CPMSG Macro, or Message Sent Record (Part 2)
VIT Processing Module:	ISTRACOT
Applies to:	**MVS, VM, VSE**

This trace record contains variable data for the MSG entry, or more message text for the MSGS entry. This is a continuation of the MSG or MSGS trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
MSG2				VARIABLE DATA or MORE MESSAGE TEXT																											

Byte (hex)	**Contents**
00 – 03	Record ID: C'MSG2'
04 – 1F	Variable data (if this is a continuation of a MSG entry), or more message text (if this is a continuation of a MSGS entry).

LU 6.2 Message Unit (Part 1)

Entry:	**MU1**
VIT Option:	APPC
Event:	LU 6.2 message unit (Part 1)
VIT Processing Module:	ISTRACAC
Control is Returned to:	Modules invoking the INTRACE macro causing the record to be produced
Applies to:	**MVS** (V3R2), **VM** (V3R2), **VSE** (V3R2)

This trace record shows information about an LU 6.2 message unit. It is generated whenever a message unit is queued to another component.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
MU1				ID	MU ID			MU ADDRESS				HS ID				CORR ADDRESS				RETCD	0			AMU FLAGS				SENSE DATA or 0			

Byte (hex) **Contents**

00 – 03 Record ID: C'MU1'

04 ID

- For MVS, address space ID
  (or zero if address space ID is greater than X'FF')
- For VM, machine and task ID
- For VSE, task ID

05 – 07 Message unit ID

X'010101' (END_CONVERSATION)
X'010201' (PS_COPR_FMH5_RCVD)
X'010302' (CONFIRMED)
X'010303' (REQUEST_TO_SEND)
X'010304' (SEND_DATA_RECORD)
X'010305' (SEND_ERROR)
X'010306' (SEND_PACING_RSP)
X'010501' (ALLOCATE_RCB)
X'010502' (DEALLOCATE_RCB)
X'010503' (GET_SESSION)
X'010504' (UNBIND_PROTOCOL_ERROR)
X'020101' (CNOS_ABORT)
X'020201' (CNOS_CLEANUP)
X'020501' (CNOS_COMPLETE)
X'020502' (CHANGE_SESSIONS)
X'030102' (CONFIRMED)
X'030103' (REQUEST_TO_SEND)
X'030104' (RECEIVE_DATA)
X'030105' (RECEIVE_ERROR)
X'030106' (PACING_RSP_RCVD)
X'030107' (RSP_TO_REQUEST_TO_SEND)
X'030108' (INITIAL_PACING_COUNT)

	X'030109' (DEALLOCATE_ABEND_REJECTED)
	X'030401' (ABORT_HS)
	X'030501' (ATTACH_HEADER)
	X'030502' (FREE_SESSION)
	X'030503' (BID)
	X'030504' (BID_RSP)
	X'030505' (BIS_RQ)
	X'030506' (BIS_REPLY)
	X'030507' (RTR_RQ)
	X'030508' (RTR_RSP)
	X'040501' (SESSION_ACTIVATED)
	X'040502' (SESSION_DEACTIVATED)
	X'040503' (ACTIVATE_SESSION_RSP)
	X'040504' (CTERM_DEACTIVATE_SESSION)
	X'040505' (LNS_LRM_FREE_AMU)
	X'050101' (RCB_ALLOCATED)
	X'050103' (SESSION_ALLOCATED)
	X'050104' (ATTACH_RECEIVED)
	X'050105' (CONVERSATION_FAILURE)
	X'050301' (HS_PS_CONNECTED)
	X'050302' (YIELD_SESSION)
	X'050303' (BID_WITHOUT_ATTACH)
	X'050304' (BID_RSP)
	X'050305' (BIS_RQ)
	X'050306' (BIS_REPLY)
	X'050307' (RTR_RQ)
	X'050308' (RTR_RSP)
	X'050401' (ACTIVATE_SESSION)
	X'050402' (DEACTIVATE_SESSION)
08—0B	Message unit address
0C—0F	Half session ID
10—13	Address of correlator
14	LU 6.2 message unit return code (AMURETCD)
15—17	Zero
18—1B	LU 6.2 message unit flags (AMUFLAGS)
1C—1F	LU 6.2 message unit sense data (AMUSENSE) or 0

LU 6.2 Message Unit (Part 2)

Entry:	**MU2**
VIT Option:	APPC
Event:	LU 6.2 message unit (Part 2)
VIT Processing Module:	ISTRACAC
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record is a continuation of the MU1 trace record.

00 01 02 03	04	05 06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15	16 17	18 19 1A 1B	1C 1D 1E 1F
MU2	0	MU ID	DATA ADDRESS or 0	DATA LENGTH or 0	RAB ADDRESS or 0	SES CNT or 0	SES LIM or 0	CONV ID or 0	VARIABLE OVERLAY FIELD

Byte (hex)	Contents
00 – 03	Record ID: C'MU2'
04	Zero
05 – 07	Message unit ID (see "LU 6.2 Message Unit (Part 1)" on page 435)
08 – 0B	Data address or 0
0C – 0F	Data length or 0
10 – 13	Resource allocation block address or 0
14 – 15	For MU ID X'020502': Session count For all other MU IDs: Zero
16 – 17	For MU ID X'020502': Session limit For all other MU IDs: Zero
18 – 1B	For MU IDs associated with an active conversation (X'010201', X'030108', X'050301'): Conversation ID For all other MU IDs: Zero
1C – 1F	Variable overlay field
	For MU IDs X'010201', X'010501', and X'050101': Task ID
	For MU ID X'020201': COPR control block address
	For MU ID X'020502': 1C – 1D CNOS delta 1E – 1F Zero
	For MU IDs X'030106' and X'030108': Pacing count increment
	For MU IDs X'030501', X'030503', X'030504', and X'030507': RPH address for inbound half session
	For all other MU IDs: Zero

LU 6.2 Message Unit (Part 3)

Entry:	**MU3**
VIT Option:	APPC
Event:	LU 6.2 message unit (Part 3)
VIT Processing Module:	ISTRACAC
Applies to:	**MVS** (V3R2), **VM** (V3R2), **VSE** (V3R2)

This trace record is a continuation of the MU2 trace record. It is generated for only those message units which contain MU IDs X'010201', X'010501', X'020502', X'040501', and X'050401'.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
MU3				0	MU ID			MODE NAME or 0								LOCAL LU NAME								REMOTE LU NAME							

Byte (hex)	Contents
00 – 03	Record ID: C'MU3'
04	Zero
05 – 07	Message unit ID (see "LU 6.2 Message Unit (Part 1)" on page 435)
08 – 0F	For MU IDs X'010501', X'020502', X'040501', and X'050401': Mode name For all other MU IDs: Zero
10 – 17	Local logical unit name
18 – 1F	Remote logical unit name

Negative Response to a PIU Request (Part 1)

Entry:	**NRSP**
VIT Option:	PIU
Event:	Negative response to a PIU request (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTTSCGR
Applies to:	VSE (V3R2), MVS , VM

This trace record is written when VTAM'S TSC response generating routine, ISTTSCGR, generates a negative response to a PIU request. Since this entry is associated with an event failure, it is treated as an exception condition and is always traced, regardless of the VIT options specified, if the VIT is active.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">NRSP</td><td>ID</td><td>CBID</td><td>FLAGS</td><td>0</td><td colspan="4">TSCB ADDRESS</td><td colspan="20">20 BYTES OF PIU</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'NRSP'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID
06	Flag field (TSCFLAG1)
07	Zero
08 – 0B	Address of TSCB
0C – 1F	First 20 characters of the PIU.

Negative Response to a PIU Request (Part 2)

Entry:	**NRS2**
VIT Option:	PIU
Event:	Negative response to a PIU request (Part 2)
VIT Processing Module:	ISTRACOT
Applies to:	VSE (V3R2), MVS, VM

This trace record is a continuation of the NRSP entry. It contains 24 more characters of the PIU.

<table>
<tr><td>0
0</td><td>0
1</td><td>0
2</td><td>0
3</td><td>0
4</td><td>0
5</td><td>0
6</td><td>0
7</td><td>0
8</td><td>0
9</td><td>0
A</td><td>0
B</td><td>0
C</td><td>0
D</td><td>0
E</td><td>0
F</td><td>1
0</td><td>1
1</td><td>1
2</td><td>1
3</td><td>1
4</td><td>1
5</td><td>1
6</td><td>1
7</td><td>1
8</td><td>1
9</td><td>1
A</td><td>1
B</td><td>1
C</td><td>1
D</td><td>1
E</td><td>1
F</td></tr>
<tr><td colspan="4">NRS2</td><td colspan="24">24 MORE CHARACTERS OF THE PIU</td><td colspan="4">RETURN ADDRESS</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'NRS2'
04 – 1B	24 more characters of the PIU
1C – 1F	Return address of the issuer

Operator Command (Part 1)

Entry:	**OPER**
VIT Option:	MSG
Event:	Operator Command
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTLUCMD MVS ISTCFF3D, ISTINCF9 VM ISTINV00
Applies to:	VSE (V3R2), MVS, VM

This trace record is written each time a VTAM command is issued at the operator console. This can help you match the console log to a surge of activity shown in the VIT. This trace record is also useful when the console log is unavailable. Additionally, it provides a record of POA commands that have been issued. These do not appear on the console log.

The variable data in bytes 08 – 1F is the command text, with minor modifications. If this data is longer than 24 characters, it will continue in OPE2. The command always appears in abbreviated form:

D	DISPLAY command
F	MODIFY command
V	VARY command
Z	HALT command
?	Unknown command passed to VTAM

NET and procname are excluded from the command text. Since the VTAM internal trace is not active when VTAM is started, the START command is not traced.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
OPER				ID	FLAG	0		FIRST 24 CHARACTERS OF OPERATOR COMMAND																							

Byte (hex)	Contents
00 – 03	Record ID: C'OPER'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Flag byte • If bit 0 = 1, a POA command was issued. • If bit 0 = 0, it was not a POA command.
06 – 07	Zero
08 – 1F	First 24 characters of the operator command

Operator Command (Part 2)

Entry:	**OPE2**
VIT Option:	MSG
Event:	Operator Command (Part 2)
VIT Processing Module:	ISTRACOT
Applies to:	**VSE** (V3R2), **MVS**, **VM**

This trace record is a continuation of the OPER entry. It contains up to 28 more characters of the operator command.

<table>
<tr><th>00</th><th>01</th><th>02</th><th>03</th><th>04</th><th>05</th><th>06</th><th>07</th><th>08</th><th>09</th><th>0A</th><th>0B</th><th>0C</th><th>0D</th><th>0E</th><th>0F</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th><th>18</th><th>19</th><th>1A</th><th>1B</th><th>1C</th><th>1D</th><th>1E</th><th>1F</th></tr>
<tr><td colspan="4">OPE2</td><td colspan="28">UP TO 28 MORE CHARACTERS OF THE OPERATOR COMMAND</td></tr>
</table>

Byte (hex)	**Contents**
00 – 03	Record ID: C'OPE2'
04 – 1F	Up to 28 more characters of the operator command

VM VSE Packet Interface (V3R1.2 and Later)

Entry:	VM VSE PIC
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTAPCES
Applies to:	VM VSE (V3R1.2 and later)

This trace record provides information about internal control flows of the telecommunications subsystem or communication adapter.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
PIC			INTERNAL TRACE DATA																												

Byte (hex)	Contents
00–02	Record ID: C'PIC'
03–1F	Internal trace data

PIU Record (Part 1)

Entry:	**PIU**
VIT Option:	PIU
Event:	PIU Record (Part 1)
VIT Processing Module:	ISTRACOT
Control is Returned to:	ISTTSCPR, ISTTSCLS, ISTTSCRI, ISTTSCSC, ISTTSCSR, ISTTSCWS, ISTTSCPD, ISTTSCLE, ISTTSCXS (or ISTINCF1 and INTINCS1 for RUPE)
Applies to:	MVS, VM, VSE

This trace record provides information about external and internal FID4 PIUS.

For most PIUS, only the first 48 bytes are traced (a 26-byte TH, a 3-byte RH, and up to the first 19 bytes of RU). However, certain classes of PIUS, such as session control RUS and VTAM RUS, are traced in their entirety.

If the control block is a RUPE, the TH is reconstructed by the internal trace using what information is available. For TSCB entries, the PIU is recorded as is.

If the PIU is a response with sense data, this entry will be generated whether the PIU option is in effect or not. It is treated as an exception condition, and consequently, is traced, if the VIT is active.

<table>
<tr><th>00</th><th>01</th><th>02</th><th>03</th><th>04</th><th>05</th><th>06</th><th>07</th><th>08</th><th>09</th><th>0A</th><th>0B</th><th>0C</th><th>0D</th><th>0E</th><th>0F</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th><th>18</th><th>19</th><th>1A</th><th>1B</th><th>1C</th><th>1D</th><th>1E</th><th>1F</th></tr>
<tr><td colspan="4">PIU</td><td>ID</td><td>CBID</td><td>FLAGS</td><td>0</td><td colspan="4">TSCB or RUPE ADDRESS</td><td colspan="20">20 BYTES OF PIU</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'PIU'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID: X'54' = RUPE, X'99' = TSCB
06	Flags (TSCFLAG1, or zero if control block is a RUPE)
07	Zero
08 – 0B	TSCB or RUPE address (see byte 5)
0C – 1F	20 bytes of PIU

PIU Record (Part 2)

Entry:	**PIU2**
VIT Option:	PIU
Event:	PIU Record (Part 2)
VIT Processing Module:	ISTRACOT
Applies to:	MVS, VM, VSE

This trace record contains 28 more bytes of an FID4 PIU. It is a continuation of PIU Record (Part 1).

VIT will generate as many as 32 PIU2 records for PIUs which are formatted and contain session control RUs or RUs to or from the SSCP or PUs. The number of records generated depends on the data count field in the TH.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
PIU2				28 MORE BYTES OF PIU																											

Byte (hex)	**Contents**
00–03	Record ID: C'PIU2' (an extension to a PIU record)
04–1F	Up to 28 more bytes of the FID4 PIU (padded at right with zeros)

VM VSE Packet (V3R1.2 and Later)

Entry:	VM VSE **PKT**
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTAPCES
Applies to:	VM VSE (V3R1.2 and later)

This trace entry provides information about the packets received at (PKTI) or sent from (PKTO) the X.25 port.

A PKTI trace record is written whenever the buffer control channel program completes with packets received.

A PKTO trace record is written whenever a packet is transferred to the telecommunications subsystem or communication adapter by inserting it into the buffer control channel program.

Note: If the packet is longer than 20 bytes (including packet header), up to 27 bytes of data are traced in an extension identified by the PKTX trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
PKTI PKTO				ID	CUA			PFSM	INDX	LNGTH		FIRST 20 BYTES OF THE PACKET																			

Byte (hex)	Contents
00 – 03	Record ID: C'PKTI' for inbound packet C'PKTO' for outbound packet
04	Machine and task ID
05 – 07	Channel device name in EBCDIC (either a CUA or device number)
08	Packet finite state machine (PFSM)
09	Index to slot that packet arrived on
0A – 0B	Packet length
0C – 1F	First 20 bytes of the packet

VM VSE Packet Extension (V3R1.2 and Later)

Entry:	VM VSE **PKTX**
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTAPCES
Applies to:	VM VSE (V3R1.2 and later)

This trace record is a continuation of the PKTI or PKTO trace record.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
PKTX				ID	ADDITIONAL BYTES OF PACKET (if needed)																										

Byte (hex)	Contents
00–03	Record ID: C'PKTX'
04	Machine and task ID
05–1F	Up to 27 additional bytes of the packet.

TPPOST

Entry:	**POST**
VIT Option:	PSS
Event:	TPPOST Macro
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTP
Applies to:	MVS, VM, VSE

This trace record identifies the RPH that is being posted for restart after a TPWAIT macro. Usually the POST entry is followed sometime later by a RESM entry. However, when a TPPOST precedes the TPWAIT, the TPWAIT returns to the caller immediately without waiting. Then POST is not followed by RESM.

00 01 02 03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
POST	ID	0	PAB0F	PST ADDRESS	PAB ADDRESS	RETURN ADDRESS	WORK ELEMENT ADDRESS	DVT ADDRESS	RPH ADDRESS

Byte (hex)	**Contents**
00 – 03	Record ID: C'POST'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Zero
06 – 07	PAB offset
08 – 0B	PST address
0C – 0F	PAB address
10 – 13	Address of the issuer of the TPPOST macro
14 – 17	Work element address (from RPHWEA)
18 – 1B	PAB DVT address
1C – 1F	Address of the RPH being posted

MVS VM Queued REQSTORE

Entry:	MVS VM **QREQ**
VIT Option:	SMS
Event:	Queued REQSTORE
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTORMBQ
Applies to:	MVS, VM

This trace record identifies the REQSTORE requests that were waiting for a buffer, and which are now satisfied.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">QREQ</td><td>ID</td><td>CBID</td><td colspan="2">0</td><td colspan="4">PST ADDRESS</td><td colspan="4">BUFFER ADDRESS</td><td colspan="4">0</td><td colspan="2">NUM BUF REQ</td><td colspan="6">0</td><td colspan="4">RPH ADDRESS</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'QREQ'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID
05	Control block ID index value
06 – 07	Zero
08 – 0B	PST address
0C – 0F	Address of buffer obtained
10 – 13	Zero
14 – 15	Number of buffers requested
16 – 1B	Zero
1C – 1F	RPH address

TPQUE (Part 1, for V3R1, V3R1.1, and V3R1.2)

Entry:	**QUE**
VIT Option:	PSS
Event:	TPQUE Macro (Part 1)
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTQ
Applies to:	**MVS** (V3R1.1 and V3R1.2), **VM** (V3R1.1 and V3R1.2), **VSE** (V3R1 and V3R1.2)

This trace record shows a work element queued to a PAB to allow another VTAM routine to do further processing with the work element. This entry may be followed shortly by a DISP entry, representing the PSS dispatch of this PAB. However, if the PAB is already running, it might not be dispatched again. In this case, no DISP entry follows the TPQUE.

00 01 02 03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
QUE	ID	CBID	PABOF	PST ADDRESS	PAB ADDRESS	WORK ELEMENT ADDRESS	PAB FIELD	MODULE NAME or DVT ADDRESS	RPH ADDRESS or 0

Byte (hex)	**Contents**
00 – 03	Record ID: C'QUE'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID of work element (Control block IDs are shown in Appendix E, "Storage and Control Block ID Codes" on page 657.)
06 – 07	PAB offset
08 – 0B	Address of PST associated with PAB
0C – 0F	PAB address
10 – 13	Address of work element to be queued
14 – 17	PAB chain field
18 – 1B	Module name abbreviation (bytes 4, 5, 7, and 8 of the module name), or PAB DVT address (high order bit of X'18' = 0). The module name might be unavailable if the PAB being scheduled is associated with a different address space than the current one.
1C – 1F	RPH address or zero

TPQUE (Part 2, for V3R1, V3R1.1, and V3R1.2)

Entry:	**QUE2**
VIT Option:	PSS
Event:	TPQUE Macro (Part 2)
VIT Processing Module:	ISTRACPS
Applies to:	MVS (V3R1.1 and V3R1.2), VM (V3R1.1 and V3R1.2), VSE (V3R1 and V3R1.2)

This trace record is a continuation of the QUE trace entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
QUE2				RETURN ADDRESS				FLAGS	APNOP	0						PAB EXTENSION or 0															

Byte (hex)	Contents
00–03	Record ID: C'QUE2'
04–07	Address of the issuer of the TPQUE macro
08	PAB scheduling flags (PABFLAGS)
09	TPQUE options (See field APNFLGS in control block ISTAPNOP, *VTAM Data Areas*)
0A–0F	Zero
10–1F	PAB extension (PABXTNSN) if this is an extended PAB.

TPQUE (V3R2)

Entry:	**QUE**
VIT Option:	PSS
Event:	TPQUE Macro
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTQ
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record shows a work element queued to a PAB to allow another VTAM routine to do further processing with the work element. This entry may be followed shortly by a DISP entry, representing the PSS dispatch of this PAB. However, if the PAB is already running, it might not be dispatched again. In this case, no DISP entry follows the TPQUE.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
QUE			CBID	ID	FLAG	NOP	GATE	PST ADDRESS				PAB ADDRESS				RETURN ADDRESS				WORK ELEMENT ADDRESS				MODULE NAME or DVT ADDRESS				RPH ADDRESS or 0			

Byte (hex) **Contents**

00–02 Record ID: C'QUE'

03 Control block ID of work element (Control block IDs are shown in Appendix E, "Storage and Control Block ID Codes" on page 657.)

04 ID

- For MVS, address space ID (or zero if address space ID is greater than X'FF')
- For VM, machine and task ID
- For VSE, task ID

05 Flag field (PABFLAGS)

06 TPQUE options (See field APNFLGS in control block ISTAPNOP in *VTAM Data Areas.*)

07 Flag field

- Bit 0 is the PAB work element queue gate bit (PABWEQG).
- Bits 1, 2, and 3 are reserved by IBM.
- Bit 4 is the PAB change bit (PABCHNG). PABCHNG=0 if the PAB *was not* previously scheduled to run. PABCHNG=1 if the PAB *was* previously scheduled to run.
- Bits 5, 6, and 7 are reserved by IBM.

08–0B PST address

0C–0F PAB address

10–13 Address of the issuer of the macro

14–17 Address of work element to be queued

18–1B Module name abbreviation (bytes 4, 5, 7, and 8 of the module name), or PAB DVT address (high order bit of X'18' = 0).

The module name might be unavailable if the PAB being scheduled is associated with a different address space than the current one.

1C–1F RPH address or zero

TPQUE NONE

Entry:	**QUEN**
VIT Option:	PSS
Event:	TPQUE NONE Macro
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTQ
Applies to:	MVS, VM, VSE

This trace record shows a work element queued to any control block (not just a PAB) to allow another VTAM routine to further process the work element.

00 01 02 03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
QUEN	ID	CBID	0	PST ADDRESS or 0	QUEUE ADDRESS	RETURN ADDRESS	WORK ELEMENT ADDRESS	0	RPH ADDRESS or 0

Byte (hex)	**Contents**
00 – 03	Record ID: C'QUEN'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID of work element (Control block IDs are shown in Appendix E, "Storage and Control Block ID Codes" on page 657.)
06 – 07	Zero
08 – 0B	Address of PST or zero
0C – 0F	Address of the control block field to which the work element is queued
10 – 13	Address of the issuer of the TPQUE NONE macro
14 – 17	Address of work element to be queued
18 – 1B	Zero
1C – 1F	RPH address or zero

RCE (V3R2)

Entry:	**RCE**
VIT Option:	NRM
Event:	RCE Macro
VIT Processing Module:	ISTRACNR
Control is Returned to:	ISTRACSC
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record contains information about an RCEADD, RCECHG, RCEDEL, or RCEFIND macroinstruction issued by a VTAM module.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
RCEA RCEC RCED RCEF				ID	RC	TYPE	MODE	KEY VALUE								RETURN ADDRESS				OUT1 or NEW TABLE NAME				OUT2 or NEW TABLE NAME				DATA ADDRESS			

Byte (hex)	Contents
00 – 03	Record ID: C'RCEA', C'RCEC', C'RCED', C'RCEF'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Return code
06	Type of RCE entry
07	Mode of RCE entry (in hex) 01 = Base entity mode 02 = Reference entity mode
08 – 0F	Key value (base entity name, base index, or reference index, depending on mode)
10 – 13	Address of the issuer of the macro
14 – 17	For RCEADD, RCEDEL, and RCEFIND: Value in RCEPOUT1 For RCECHG: First half of new table name
18 – 1B	For RCEADD, RCEDEL, and RCEFIND: Value in RCEPOUT2 For RCECHG: Second half of new table name
1C – 1F	Data address

RPL Exit

Entry:	**RE**
VIT Option:	API
Event:	RPL Exit
VIT Processing Module:	ISTRACAP
Control is Returned to:	ISTAPCSX or ISTAPCUE
Applies to:	MVS, VM, VSE

This trace record contains RPL information passed from VTAM to the application program. If the return code or feedback is non-zero, this entry will be generated whether the API option is in effect or not. It is usually treated as an exception condition, and consequently, is traced, if the VIT is active.

See *VTAM Programming* for a description of RPL return codes.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
RE		REQTY	EXTDS	ID	RTNCD	FDB2	FDB3	RPL ADDRESS				EXIT ADDRESS				RPL AREA				RPL RLEN				CID or 0				RPL FDBK2			

Byte (hex)	Contents
00 – 01	Record ID: C'RE'
02	RPL request type (RPLREQ)
	X'15' SETLOGON
	X'16' SIMLOGON
	X'17' OPNDST
	X'1A' INQUIRE
	X'1B' INTRPRET
	X'1F' CLSDST
	X'22' SEND
	X'23' RECEIVE
	X'24' RESETSR
	X'25' SESSIONC
	X'27' SENDCMD
	X'28' RCVCMD
	X'29' REQSESS
	X'2A' OPNSEC
	X'2C' TERMSESS
03	Exit definition (RPLEXTDS)
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	RPLRTNCD (See *VTAM Programming*)
06	RPLFDB2 (See *VTAM Programming*)
07	RPLFDB3 (See *VTAM Programming*, under INQUIRE macro with OPTCD = APPSTAT)

08 – 0B	RPL address
0C – 0F	Exit address
10 – 13	RPL data area pointer (RPLAREA)
14 – 17	Record length (RPLRLEN)
18 – 1B	CID from NIB or from RPL (RPLARG) or zero
1C – 1F	Second RPL feedback area - Sense code (RPLFDBK2)

RELSTORE

Entry:	**RELS**
VIT Option:	SMS
Event:	RELSTORE Macro
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTORMBD
Applies to:	**MVS, VM, VSE**

This trace record provides the status of each RELSTORE request issued by VTAM components to release fixed-length buffers to one of the predefined buffer pools.

RELSTORE is the complement of REQSTORE. Storage obtained by REQSTORE must be released by RELSTORE. However, since more than one buffer may be obtained and released at a time, there may not be a RELS for every REQS, and there may not be a REQS for each RELS. Use the "next buffer address" in RELS and the "number of buffers" in REQS as clues.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
RELS				ID	CBID	POOL	ID	PST ADDRESS				BUFFER ADDRESS				RETURN ADDRESS				NEXT BUFFER ADDRESS or 0				REG 1				CALLER OF UTILITY or 0			

Byte (hex)	Contents
00 – 03	Record ID: C'RELS'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	For MVS and VM, control block ID index value For VSE, zero
06 – 07	For MVS and VM, zero For VSE, buffer pool ID
08 – 0B	PST address
0C – 0F	Address of buffer to be released
10 – 13	Address of the issuer of the RELSTORE macro
14 – 17	Pointer to the next buffer to be released if more than one buffer to be RELSTOREd; otherwise, zero
18 – 1B	Register 1 (normally RPH address)
1C – 1F	Caller of utility routine, or zero • For MVS, address of the utility's caller, if the RELSTORE macro was issued from a utility routine. Otherwise, zero if the macro was issued directly by the caller without a utility routine. (See bytes 10 – 13 for the address of the issuer of the RELSTORE macro.) • For VM and VSE, zero.

REQSTORE

Entry:	**REQS**
VIT Option:	SMS
Event:	REQSTORE Macro
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTORMBA
Applies to:	**MVS, VM, VSE**

This trace record provides the status of each REQSTORE request issued by VTAM components to obtain fixed-length buffers from one of the predefined buffer pools. The storage obtained by REQSTORE is released by RELSTORE.

If this entry is associated with an event failure, that is, the return code is non-zero, this entry will be generated whether the SMS option is in effect or not. It is treated as an exception condition, and consequently, is traced, if the VIT is active.

<table>
<tr><td>0
0</td><td>0
1</td><td>0
2</td><td>0
3</td><td>0
4</td><td>0
5</td><td>0
6</td><td>0
7</td><td>0
8</td><td>0
9</td><td>0
A</td><td>0
B</td><td>0
C</td><td>0
D</td><td>0
E</td><td>0
F</td><td>1
0</td><td>1
1</td><td>1
2</td><td>1
3</td><td>1
4</td><td>1
5</td><td>1
6</td><td>1
7</td><td>1
8</td><td>1
9</td><td>1
A</td><td>1
B</td><td>1
C</td><td>1
D</td><td>1
E</td><td>1
F</td></tr>
<tr><td colspan="4">REQS</td><td>I
D</td><td>C
B
I
D</td><td colspan="2">0</td><td colspan="4">PST
ADDRESS
or 0</td><td colspan="4">BUFFER
ADDRESS
or 0</td><td colspan="4">RETURN
ADDRESS</td><td colspan="2">NUM
BUF
REQ</td><td colspan="2">0</td><td colspan="4">REG 1</td><td colspan="4">RETURN
CODE</td></tr>
</table>

Byte (hex)	**Contents**
00 – 03	Record ID: C'REQS'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	For MVS and VM, control block ID index value For VSE, zero
06 – 07	For MVS and VM, zero For VSE, buffer pool ID
08 – 0B	Pointer to the PST, or zero
0C – 0F	Address of buffer obtained, or zero if buffer not obtained
10 – 13	Address of the issuer of the REQSTORE macro
14 – 15	Number of buffers requested
16 – 17	Zero
18 – 1B	Register 1 (normally RPH address)
1C – 1F	Return code

Resume from TPWAIT

Entry:	**RESM**
VIT Option:	PSS
Event:	Resume from TPWAIT
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCPD
Applies to:	**MVS, VM, VSE**

This trace record gives information about which VTAM process is about to be redispatched after a TPWAIT. A RESM entry follows a POST, although not every POST has a RESM. Usually the POST entry is followed sometime later by a RESM entry. However, when a TPPOST precedes the TPWAIT, the TPWAIT returns to the caller immediately without waiting. Then POST is not followed by RESM. Use the RPH address to find the original DISP entry, which may have occurred quite some time earlier.

00 01 02 03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
RESM	ID	CBID	PABOF	PST ADDRESS	PAB ADDRESS	WORK ELEMENT ADDRESS	WORK ELEMENT QUEUE	DVT ADDRESS	RPH ADDRESS

Byte (hex)	Contents
00–03	Record ID: C'RESM'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Control block ID of work element (Control block IDs are shown in Appendix E, "Storage and Control Block ID Codes" on page 657.)
06–07	PAB offset
08–0B	PST address
0C–0F	PAB address
10–13	Work element address
14–17	PAB work element queue
18–1B	PAB DVT address
1C–1F	RPH address

TPSCHED (Part 1, for V3R1, V3R1.1, and V3R1.2)

Entry:	**SCHD**
VIT Option:	PSS
Event:	TPSCHED Macro (Part 1)
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTS
Applies to:	MVS (V3R1.1 and V3R1.2), VM (V3R1.1 and V3R1.2), VSE (V3R1 and V3R1.2)

This trace record shows the schedule of a PAB to be dispatched and the last work element, if any, that was queued to the PAB. This entry should be followed shortly by a DISP entry, representing the PSS dispatch of this PAB.

00 01 02 03	04	05	06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
SCHD	ID	CBID	PAB0F	PST ADDRESS	PAB ADDRESS	WORK ELEMENT QUEUE	PAB CHAIN FIELD	MODULE NAME or DVT ADDRESS	RPH ADDRESS

Byte (hex)	Contents
00 – 03	Record ID: C'SCHD'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	For MVS and VSE, zero For VM, control block ID. (Control block IDs are shown in Appendix E, "Storage and Control Block ID Codes" on page 657.)
06 – 07	PAB offset
08 – 0B	PST address associated with PAB
0C – 0F	PAB address
10 – 13	PAB work element queue
14 – 17	PAB chain field
18 – 1B	Module name abbreviation (bytes 4, 5, 7, and 8 of the module name), or PAB DVT address (high order bit of X'18' = 0). The module name might be unavailable if the PAB being scheduled is associated with a different address space than the current one.
1C – 1F	RPH address or zero

TPSCHED (Part 2, for V3R1, V3R1.1, and V3R1.2)

Entry:	**SCH2**
VIT Option:	PSS
Event:	TPSCHED Macro (Part 2)
VIT Processing Module:	ISTRACPS
Applies to:	MVS (V3R1.1 and V3R1.2), VM (V3R1.1 and V3R1.2), VSE (V3R1 and V3R1.2)

This trace record is a continuation of the SCHD entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SCH2				RETURN ADDRESS				0	APNOP	0						PAB EXTENSION or 0															

Byte (hex)	Contents
00–03	Record ID: C'SCH2'
04–07	Address of the issuer of the TPQUE macro
08	Zero
09	TPSCHED options (See field APNFLGS in control block ISTAPNOP in *VTAM Data Areas*)
0A–0F	Zero
10–1F	PAB extension (PABXTNSN) if this is an extended PAB.

TPSCHED (V3R2)

Entry:	**SCHD**
VIT Option:	PSS
Event:	TPSCHED Macro
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTS
Applies to:	MVS (V3R2), VM (V3R2), VSE (V3R2)

This trace record shows the schedule of a PAB to be dispatched and the last work element, if any, that was queued to the PAB. This entry should be followed shortly by a DISP entry, representing the PSS dispatch of this PAB.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">SCHD</td><td>ID</td><td>FLAG</td><td>NOP</td><td>GATE</td><td colspan="4">PST ADDRESS</td><td colspan="4">PAB ADDRESS</td><td colspan="4">RETURN ADDRESS</td><td colspan="4">WORK ELEMENT QUEUE</td><td colspan="4">MODULE NAME or DVT ADDRESS</td><td colspan="4">RPH ADDRESS</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'SCHD'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Flag field (PABFLAGS)
06	TPSCHED options (See field APNFLGS in control block ISTAPNOP in *VTAM Data Areas.*)
07	Flag field • Bit 0 is the PAB work element queue gate bit (PABWEQG). • Bits 1, 2, and 3 are reserved by IBM. • Bit 4 is the PAB change bit (PABCHNG). PABCHNG=0 if the PAB *was not* previously scheduled to run. PABCHNG=1 if the PAB *was* previously scheduled to run. • Bits 5, 6, and 7 are reserved by IBM.
08 – 0B	PST address associated with PAB
0C – 0F	PAB address
10 – 13	Address of the issuer of the TPSCHED macro
14 – 17	PAB work element queue
18 – 1B	Module name abbreviation (bytes 4, 5, 7, and 8 of the module name), or PAB DVT address (high order bit of X'18' = 0). The module name might be unavailable if the PAB being scheduled is associated with a different address space than the current one.
1C – 1F	RPH address

MVS Start I/O (Part 1)

Entry:	**MVS SIO**
VIT Option:	CIO
Event:	START I/O
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCIO or ISTTSCLC
Applies to:	MVS

- The SIOI trace record is written when a START I/O SVC is issued for a channel-attached communication controller or SNA cluster controller.
- The SIOL trace record is written when a START I/O SVC is issued for a channel-attached non-SNA device.
- The SIOX trace record is written when a START I/O SVC is issued for a channel-to-channel attachment.

The SIO trace record will eventually be followed by an INT entry. Use the CUA field to match the correct entries. See *VTAM Data Areas* for a description of the NCB fields.

00 01 02 03	04	05	06 07	08	09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B 1C 1D 1E 1F
SIOI or SIOL	ID	STATE	MODID	WRITE	CUA	NCB ADDRESS	FLAG BYTES	CAW	CCW or DEVICE-DEPENDENT INFORMATION
SIOX	ID	STATE	MODID	0	CUA	NCB ADDRESS	FLAG BYTES	CAW	CCW or OUTPUT CONTROL AREA (STATUS INFORMATION)

Byte (hex)	Contents
00 – 03	Record ID: C'SIOI' for ICNCB C'SIOL' for LDNCB C'SIOX' for XCNCB
04	Address space ID (or zero if address space ID is greater than X'FF')
05	For SIOI and SIOL, link station state (NCBLNKST) For SIOX, station state (XCNSSFSM)
06 – 07	Module ID (last two characters of TSC module name)
08	For SIOI and SIOL, Write Start command For SIOX, zero.
09 – 0B	Channel device name in EBCDIC (either a CUA or device number)
0C – 0F	NCB address
10 – 13	Flag bytes (NCBFLAGS)
14 – 17	Virtual channel address word (NCBCAW)

18—1F If link is active:

- For SIOI and SIOL, device-dependent information:

Byte (hex)	Contents
18—1B	Virtual buffer address of the first Write CCW or zero
1C	Read Start command code or zero
1D—1F	Zero

- For SIOX, output control area (status information).

If link is not active, first channel command word (CCW).

MVS Start I/O (Part 2)

Entry:	MVS **SIO2**
VIT Option:	CIO
Event:	START I/O
VIT Processing Module:	ISTRACCI
Applies to:	MVS

This trace record is a continuation of the SIO entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SIO2				VIRTUAL BUFFER ADDRESS				0																							

Byte (hex)	Contents
00 – 03	Record ID: C'SIO2'
04 – 07	Virtual buffer address of the first Read CCW, or zero.
08 – 1F	Zero, or station state (XCNSSFSM) for SIOX.

VM Start I/O (Part 1)

Entry:	VM SIO
VIT Option:	CIO
Event:	START I/O
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCIO or ISTTSCLC
Applies to:	VM

This trace record is written when a START I/O SVC is issued for a channel-attached communication controller or SNA cluster controller. It will eventually be followed by an INT entry. Use the CUA field to match the correct entries. See *VTAM Data Areas* for a description of the NCB fields.

00 01 02 03	04	05	06 07	08	09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B 1C 1D 1E 1F
SIOB or SIOH	ID	LFSM	MODID	0	CUA	NCB ADDRESS	FLAG BYTES	CAW	0
SIOI or SIOL	ID	LNKST	MODID	0	CUA	NCB ADDRESS	FLAG BYTES	CAW	0
SIOX	ID	STATE	MODID	0	CUA	NCB ADDRESS	FLAG BYTES	CAW	CCW or OUTPUT CONTROL AREA (STATUS INFORMATION)

Byte (hex)	Contents
00–03	Record ID: C'SIOB' for BSCLB C'SIOH' for HALCB C'SIOI' for ICNCB C'SIOL' for LDNCB C'SIOX' for XCNCB
04	Machine and task ID
05	For SIOB or SIOH, link finite state machine (LFSM) For SIOI or SIOL, link station state (NCBLNKST) For SIOX, station state (XCNSSFSM)
06–07	Module ID (last two characters of TSC module name)
08	Zero
09–0B	Channel device name in EBCDIC (either a CUA or device number)
0C–0F	NCB address
10–13	Flag bytes (NCBFLAGS)
14–17	Virtual channel address word (NCBCAW)

18–1F For SIOB, SIOH, SIOI, SIOL: Zero
For SIOX:

- If link is active, output control area (status information)
- If link is not active, first channel command word (CCW)

VM VSE Start I/O (V3R1.2 and Later)

Entry:	VM VSE **SIO**
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRAC6T
Control is Returned to:	ISTTSC6F
Applies to:	VM (V3R1.2 and later), VSE (V3R2)

This trace record is created when a Start I/O SVC is issued for the Token-Ring Subsystem. Channel programs started for the subsystem do not always complete. If the channel program completes, it is eventually followed by an INT entry. Use the CUA field to match the correct entries.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SIOP				ID	CPFSM	MODID		0	CUA			NCB ADDRESS				PDATA		0		CAW				CODE	0						

Byte (hex)	Contents
00–03	Record ID: C'SIOP'
04	Machine and task ID
05	Channel program FSM
	00 Reset
	01 Sense ID
	02 Interrupt port
	03 Control port
	04 Running
	05 Sense
	06 Orderly halt (HLTC)
	07 Termination
	08 Shutdown
06–07	Module ID (last 2 characters of TSC module name)
08	Zero
09–0B	Channel device name in EBCDIC (either a CUA or device number)
0C–0F	NCB address
10	Port state (PDATA)
	00 Reset
	01 Halt
	02 Halt complete
	03 Sense ID
	04 Sense ID complete
	05 Interrupt pending
	06 Active
	07 Sense
11	Port type (PDATA)
	00 Interrupt
	01 Control
	02 Inbound
	03 Outbound

12—13	Zero	
14—17	Channel address word	
18	Channel-command code for the first CCW	
	C4	Sense control unit state
	CB	Set CETI mode on
	E4	Sense ID
19—1F	Zero	

VM Start I/O (Part 2)

Entry:	**VM SIO2**
VIT Option:	CIO
Event:	START I/O
VIT Processing Module:	ISTRACCI
Applies to:	VM

This trace record is a continuation of the SIO entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SIO2				ERACT	TECT	ERCOD	RCODE	DEVICE-DEPENDENT INFORMATION or 0																0							

Byte (hex)	Contents
00 – 03	Record ID: C'SIO2'
04	ERACT
05	Temporary error counter (TECT)
06	ERCOD
07	Return code
08 – 17	Device-dependent information shown below, or zero
18 – 1F	Zero

For SDLC links (NCBTYPE = 09):

08	Index to current PUT
09	Station flags
0A	Station finite state machine
0B	Connection finite state machine
0C-17	Release-dependent information:

For VTAM V3R1.1:

0C	Connection manager flags
0D – 17	First 11 bytes of CCW command code

For VTAM V3R1.2 and later:

0C-0D	Connection manager flags
0E – 17	First 10 bytes of CCW command code

For BSC links (NCBTYPE = 0A):

08	Current channel program finite state machine (BSCCCFSM)
09	Original channel program finite state machine (BSCOCFSM)
0A	Previous channel program finite state machine (BSCPCFSM)
0B	Original failing channel program finite state machine (BSCFCFSM)
0C	First selection character (BSCSEL1)
0D	First device address (BSCDEV1)
0E	Index of BPL entry for which input data was received (BSCPESV)
0F	Device address from which input data was received (BSCDEVSV)
10	Index for last station for which there was output data (BSCISOD)
11	First response byte (BSCRSPA1)

12	Second response byte (BSCRSPA2)
13	Last block type received (BSCLBR)
14	Current acknowledgment (BSCALTAK)
15	Error retry count (BSCRCNT)
16	Connection manager flags (BSCCMIO)
17	BSC flags (BSCFLAGS)

VSE Start I/O (Part 1)

Entry:	VSE SIO
VIT Option:	CIO
Event:	START I/O
VIT Processing Module:	ISTRACCI
Control is Returned to:	ISTTSCIO or ISTTSCLC
Applies to:	VSE

This trace record is written when a START I/O SVC is issued for a channel-attached communication controller or SNA cluster controller. It will eventually be followed by an INT entry. Use the CUA field to match the correct entries. See *VTAM Data Areas* for a description of the NCB fields.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">SIOB
or
SIOH</td><td>ID</td><td>LFSM</td><td colspan="2">MODID</td><td>0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="4">FLAG BYTES</td><td colspan="4">CAW</td><td colspan="8">0</td></tr>
<tr><td colspan="4">SIOI
or
SIOL</td><td>ID</td><td>LNKST</td><td colspan="2">MODID</td><td>0</td><td colspan="3">CUA</td><td colspan="4">NCB ADDRESS</td><td colspan="4">FLAG BYTES</td><td colspan="4">CAW</td><td colspan="8">0</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'SIOB' for BSCLB C'SIOH' for HALCB C'SIOI' for ICNCB C'SIOL' for LDNCB
04	Task ID
05	Link finite state machine (LFSM), or link station state (NCBLNKST)
06 – 07	Module ID (last two characters of TSC module name)
08	Zero
09 – 0B	Channel device name in EBCDIC (either a CUA or device number)
0C – 0F	NCB address
10 – 13	Flag bytes (NCBFLAGS)
14 – 17	Virtual channel address word (NCBCAW)
18 – 1F	Zero

VSE Start I/O (Part 2)

Entry:	VSE **SIO2**
VIT Option:	CIO
Event:	START I/O
VIT Processing Module:	ISTRACCI
Applies to:	VSE

This trace record is a continuation of the SIO entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SI02				ERACT	TECT	ERCOD	RCODE	DEVICE-DEPENDENT INFORMATION or 0																0							

Byte (hex)	**Contents**
00–03	Record ID: C'SIO2'
04	ERACT
05	Temporary error counter (TECT)
06	ERCOD
07	Return code
08–17	Device-dependent information shown below, or zero
18–1F	Zero

For SDLC links (NCBTYPE=09):

08	Index to current PUT
09	Station flags
0A	Station finite state machine
0B	Connection finite state machine
0C-17	Release-dependent information:

For VTAM V3R1:

0C	Connection manager flags
0D–17	First 11 bytes of CCW command code

For VTAM V3R1.2 and later:

0C-0D	Connection manager flags
0E–17	First 10 bytes of CCW command code

For BSC links (NCBTYPE=0A):

08	Current channel program finite state machine (BSCCCFSM)
09	Original channel program finite state machine (BSCOCFSM)
0A	Previous channel program finite state machine (BSCPCFSM)
0B	Original failing channel program finite state machine (BSCFCFSM)
0C	First selection character (BSCSEL1)
0D	First device address (BSCDEV1)
0E	Index of BPL entry for which input data was received (BSCPESV)
0F	Device address from which input data was received (BSCDEVSV)
10	Index for last station for which there was output data (BSCISOD)
11	First response byte (BSCRSPA1)

12	Second response byte (BSCRSPA2)
13	Last block type received (BSCLBR)
14	Current acknowledgment (BSCALTAK)
15	Error retry count (BSCRCNT)
16	Connection manager flags (BSCCMIO)
17	BSC flags (BSCFLAGS)

VM VSE Start I/O (V3R1.2 and Later)

Entry:	**VM VSE SIO**
VIT Option:	CIO
Event:	Start I/O
VIT Processing Module:	ISTRACTR (SNAP trace recording routine)
Control is Returned to:	ISTAPCES
Applies to:	**VM VSE** (V3R1.2 and later)

This trace record provides information about Start I/O at the channel interface for an X.25 port.

The telecommunications subsystem or communication adapter uses two channel programs: a control link channel program and a buffer pool channel program. An SIO entry is built for the control channel program whenever it is started, and also when the buffer pool channel program is started. This channel program is an endless channel program and terminates only in case of error or when the adapter stops.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
SIO			TYPE	ID	CUA			PFSM	0	DUMP OF WRITE CONTROL AND READ CONTROL DATA AREAS																					

Byte (hex)	Contents
00 – 02	Record ID: C'SIO'
03	Type: 0C = Control link channel program 0D = Buffer pool channel program
04	Machine and task ID
05 – 07	Channel device name in EBCDIC (either a CUA or device number)
08	Packet finite state machine (PFSM)
09	Zero
0A – 1F	Dump of Write Control and Read Control data areas

MVS SRB Dispatch

Entry:	MVS **SRBD**
VIT Option:	PSS
Event:	SRB Dispatch
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCAD
Applies to:	MVS

This trace record provides information when PSS is dispatched under an SRB.

Note: In addition to specifying the PSS option, you must also set the RACON bits to B'1' for this trace record to be created.

00 01 02 03	04	05 06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D	1E 1F
SRBD	I D	0	PST ADDRESS	PST ASYNC DISP QUEUE (ADSP)	PST ASYNC FLAG FIELD (IAF)	PST ASYNC N-DISP QUEUE (ANDSP)	TCB ADDRESS	F L G 1	0

Byte (hex)	Contents
00 – 03	Record ID: C'SRBD'
04	Address space ID (or zero if address space ID is greater than X'FF')
05 – 07	Zero
08 – 0B	PST address
0C – 0F	PST asynchronous dispatchable queue address (PSTADSP)
10 – 13	PST asynchronous flag field (PSTIAF)
14 – 17	PST asynchronous nondispatchable queue address (PSTANDSP)
18 – 1B	TCB address (PSTTCBA)
1C – 1D	PST flags (PSTFLG1)
1E – 1F	Zero

MVS SRB Exit

Entry:	MVS **SRBX**
VIT Option:	PSS
Event:	SRB Exit
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCSX
Applies to:	MVS

This trace record provides information when PSS exits an SRB.

Note: In addition to specifying the PSS option, you must also set the RACON bits to B'1' for this trace record to be created.

00 01 02 03	04	05 06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
SRBX	ID	0	PST ADDRESS	PST ASYNC DISP QUEUE (ADSP)	PST ASYNC FLAG FIELD (IAF)	NEW ASCB ADDRESS	OLD ASCB ADDRESS	0

Byte (hex)	**Contents**
00 – 03	Record ID: C'SRBX'
04	Address space ID (or zero if address space ID is greater than X'FF')
05 – 07	Zero
08 – 0B	PST address
0C – 0F	PST asynchronous dispatchable queue address (PSTADSP)
10 – 13	PST asynchronous flag field (PSTIAF)
14 – 17	New ASCB address
18 – 1B	Old ASCB address
1C – 1F	Zero

SRTCHG (V3R2 only), SRTADD, SRTDEL, SRTFIND

Entry:	**SRT**
VIT Option:	NRM
Event:	SRTADD, SRTCHG, SRTDEL, SRTFIND Macros
VIT Processing Module:	ISTRACNR
Control is Returned to:	ISTNRCSA for SRTADD; ISTNRCSD for SRTCHG, SRTDEL, and SRTFIND
Applies to:	MVS, VM, VSE

This trace record contains information about a SRTADD, SRTCHG, SRTDEL, or SRTFIND macroinstruction issued by a VTAM module.

Note: SRTCHG is new for VTAM V3R2.

00 01 02 03	04	05	06	07	08 09 0A 0B 0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B 1C 1D 1E 1F
SRTA SRTC SRTD SRTF	ID	RC	0	TYPE	HASH NAME (NAME or NETWORK ADDRESS)	RETURN ADDRESS	SRT ENTRY ADDRESS	NETWORK ID

Byte (hex)	**Contents**
00 – 03	Record ID: C'SRTA', C'SRTC', C'SRTD', or C'SRTF'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Return code
06	Zero
07	Type of SRT entry
08 – 0F	Network name or network address (padded on left with zeros) passed to the hashing algorithm
10 – 13	Address of the issuer of the SRTADD, SRTCHG, SRTDEL, or SRTFIND macroinstruction
14 – 17	Address of SRT entry for the resource
18 – 1F	Network ID of the resource

USER Exit (DFASY or RESP)

Entry:	**UE**
VIT Option:	API
Event:	User Exit
VIT Processing Module:	ISTRACAP
Control is Returned to:	ISTAPCUE
Applies to:	MVS, VM, VSE

This trace record identifies the exit dispatched and data passed back to the user application program.

If this entry is associated with an event failure, that is, the return code is non-zero, this entry will be generated whether the API option is in effect or not. It is treated as an exception condition, and consequently, is traced, if the VIT is active.

For DFASY and RESP exits (type codes = 07 and 08), the user exit trace record has the following format:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
UE		CODE	EXTDS	ID	RTNCD	FDB2	FDB3	RPL ADDRESS				EXIT ADDRESS				RPL AREA				RPL RLEN				CID or 0				RPL FDBK2			

Byte (hex)	**Contents**
00–01	Record ID: C'UE'
02	Exit type code (X'7' = DFASY, X'8' = RESP)
03	Exit definition (RPLEXTDS)
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	RPLRTNCD (See *VTAM Programming*)
06	RPLFDB2 (See *VTAM Programming*)
07	RPLFDB3 (See *VTAM Programming*)
08–0B	RPL address
0C–0F	Exit address
10–13	RPL data area pointer (RPLAREA) or zero
14–17	Record length (RPLRLEN)
18–1B	CID from NIB or RPL (RPLARG) or zero
1C–1F	Second RPL feedback area — Sense code (RPLFDBK2)

USER Exit (Part 1, not DFASY or RESP)

The user exit trace record (Not DFASY or RESP, Part 1) has the following format:

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">UE1</td><td>ID</td><td>CODE</td><td colspan="2">0</td><td colspan="8">EXIT-TYPE DEPENDENT INFORMATION</td><td colspan="8">PRIMARY (APPL) LU NAME</td><td colspan="8">SECONDARY LU NAME or 0</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'UE1'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Exit type code (see Figure 69 on page 481)
06 – 07	Zero
08 – 0F	Exit-type-dependent information (see Figure 69 on page 481)
10 – 17	Primary (application program) logical unit name
18 – 1F	Secondary logical unit name, or zero if this is a TPEND exit (see Figure 69 on page 481).

Exit Type Code	Exit Type	Byte (hex)	Contents
05	SCIP (BIND)	08–09	RPL control flags (RPLCNTDC, RPLCNTSC)
		0A–0F	Session parameters
05	SCIP (STSN)	08–09	RPL control flags (RPLCNTDC, RPLCNTSC)
		0A–0B	RPLOBSQV
		0C–0D	RPLIBSQV
		0E	RPLOBSQ
		0F	RPLIBSQ
05	SCIP (other)	08–09	RPL control flags (RPLCNTDC)
		0C–0F	CID
06	LOGON	08–0B	Logon data length
		0C–0F	CID
09	LOSTERM	08–0B	Reason code
		0C–0F	CID
0A	RELREQ	08–0F	Unused
0D	TPEND	08–0B	Reason code
0E	NSEXIT	08–0F	NS RU bytes 0–7

Figure 69. Exit-Type–Dependent Information for a UE1 Entry

USER Exit (Part 2, not DFASY or RESP)

The user exit trace record (Not DFASY or RESP, Part 2) has the following format:

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">UE2</td><td colspan="4">EXIT ADDRESS</td><td colspan="24">0</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'UE2'
04 – 07	Exit address
08 – 1F	Zero

TPUNLOCK All

Entry:	**ULKA**
VIT Option:	LOCK
Event:	TPUNLOCK All Macro
VIT Processing Module:	ISTRACLK
Control is Returned to:	ISTAPC39
Applies to:	MVS, VM, VSE

This trace record contains information about a routine releasing all locks it currently holds.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ULKA				ID	0			PST ADDRESS				CRA LKACT				RETURN ADDRESS				0								RPH ADDRESS			

Byte (hex)	**Contents**
00 – 03	Record ID: C'ULKA'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05 – 07	Zero
08 – 0B	PST address
0C – 0F	CRA lock account word (CRALKACT; see step 11 on page 41)
10 – 13	Address of the issuer of the TPUNLOCK macro
14 – 1B	Zero
1C – 1F	RPH address

TPUNLOCK

Entry:	**UNLK**
VIT Option:	LOCK
Event:	TPUNLOCK Macro
VIT Processing Module:	ISTRACLK
Control is Returned to:	ISTAPC33
Applies to:	MVS, VM, VSE

This trace record contains information for a routine releasing a lock. It is the complement of a LKEX or LKSH entry.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
UNLK				ID	0	LOCK	LEVEL	LOCK ADDRESS				CRA LKACT				RETURN ADDRESS				LOCKWORD								RPH ADDRESS			

Byte (hex)	Contents
00 – 03	Record ID: C'UNLK'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	Zero
06 – 07	Level of lock to be released
08 – 0B	Address of lock to be released
0C – 0F	CRA lock account word (CRALKACT; see step 11 on page 41)
10 – 13	Address of the issuer of the TPUNLOCK macro
14 – 1B	Lockword
1C – 1F	RPH address

User Post

Entry:	**UP**
VIT Option:	API
Event:	User Post
VIT Processing Module:	ISTRACAP
Control is Returned to:	ISTAICPT or ISTAPCSX
Applies to:	MVS, VM, VSE

This trace record provides the RPL information that VTAM passes to the application program. The RPL information and the API TPIO request record (AIn or IOn) show VTAM activity for a given API request.

See *VTAM Programming* for a description of the RPL return codes.

If this entry is associated with an event failure, that is, the return code is non-zero, this entry will be generated whether the API option is in effect or not. It is treated as an exception condition, and consequently, is traced, if the VIT is active.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
UP		REQTY	EXTDS	ID	RTNCD	FDB2	FDB3	RPL ADDRESS				EXIT or ECB ADDRESS				RPL AREA				RPL RLEN				CID or 0				RPL FDBK2			

Byte (hex)	Contents
00 – 01	Record ID: C'UP'
02	RPL request type (RPLREQ) X'15' SETLOGON X'16' SIMLOGON X'17' OPNDST X'1A' INQUIRE X'1B' INTRPRET X'1F' CLSDST X'22' SEND X'23' RECEIVE X'24' RESETSR X'25' SESSIONC X'27' SENDCMD X'28' RCVCMD X'29' REQSESS X'2A' OPNSEC X'2C' TERMSESS
03	Exit definition (RPLEXTDS)
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05	RPLRTNCD (See *VTAM Programming*)

06	RPLFDB2 (See *VTAM Programming*)
07	RPLFDB3 (See *VTAM Programming*, under INQUIRE macro with OPTCD=APPSTAT)
08–0B	RPL address
0C–0F	Exit address or ECB address
10–13	RPL data area pointer (RPLAREA)
14–17	Record length (RPLRLEN)
18–1B	CID from NIB or from RPL (RPLARG), or zero
1C–1F	Second RPL feedback area — Sense code (RPLFDBK2)

VTALLOC

Entry:	**VTAL**
VIT Option:	SMS
Event:	VTALLOC Macro
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTORMAF
Applies to:	, ,

This trace record provides the status of each VTALLOC request issued by VTAM components.

If this entry is associated with an event failure, that is, the return code is non-zero, this entry will be generated whether the SMS option is in effect or not. It is treated as an exception condition, and consequently, is traced, if the VIT is active.

0 0 0 0 0 1 2 3	0 4	0 0 0 5 6 7	0 0 0 0 8 9 A B	0 0 0 0 C D E F	1 1 1 1 0 1 2 3	1 1 1 1 4 5 6 7	1 1 1 1 8 9 A B	1 1 1 1 C D E F
VTAL	I D	0	STORAGE ADDRESS	SUBPOOL NUMBER or POOL ID	RETURN ADDRESS	LENGTH OF STORAGE REQUEST	CALLER OF UTILITY or 0	RETURN CODE

Byte (hex)	Contents
00–03	Record ID: C'VTAL'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05–07	Zero
08–0B	Address of area allocated
0C–0F	For MVS, subpool number of the area allocated For VM, buffer pool ID For VSE, buffer pool ID
10–13	Address of the issuer of the VTALLOC macro
14–17	Length of area allocated
18–1B	Address of issuer of utility routine or 0
1C–1F	Return code

VTFREE

Entry:	**VTFR**
VIT Option:	SMS
Event:	VTFREE Macro
VIT Processing Module:	ISTRACSM
Control is Returned to:	ISTORMAF
Applies to:	MVS, VM, VSE

This trace record provides the status of each VTFREE request issued by VTAM components.

00 01 02 03	04	05 06 07	08 09 0A 0B	0C 0D 0E 0F	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
VTFR	I D	0	STORAGE ADDRESS	SUBPOOL NUMBER or POOL ID	RETURN ADDRESS	LENGTH OF STORAGE FREED	CALLER OF UTILITY or 0	0

Byte (hex)	**Contents**
00 – 03	Record ID: C'VTFR'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05 – 07	Zero
08 – 0B	Address of area to be freed
0C – 0F	For MVS, subpool number of the area allocated For VM, buffer pool ID For VSE, buffer pool ID
10 – 13	Address of the issuer of the VTFREE macro
14 – 17	Length of area to be freed
18 – 1B	Address of issuer of utility routine or zero
1C – 1F	Zero

TPWAIT

Entry:	**WAIT**
VIT Option:	PSS
Event:	TPWAIT Macro
VIT Processing Module:	ISTRACPS
Control is Returned to:	ISTAPCTW
Applies to:	MVS , VM , VSE

This trace record identifies an RPH that has been suspended to wait for some VTAM resource. This entry might be used with the entry immediately preceding it in the table to determine whether the wait is for storage or for a lock. If the preceding entry is for a REQSTORE, a waiting-for-storage condition may exist. If the preceding entry is for a lock request, the wait *could* be caused by that request. However, VTAM frequently uses a TPWAIT/TPPOST to synchronize intercomponent functions. Therefore, the TPWAIT entry does not necessarily identify an unusual condition.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>0A</td><td>0B</td><td>0C</td><td>0D</td><td>0E</td><td>0F</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>1A</td><td>1B</td><td>1C</td><td>1D</td><td>1E</td><td>1F</td></tr>
<tr><td colspan="4">WAIT</td><td>ID</td><td colspan="3">0</td><td colspan="4">PST ADDRESS</td><td colspan="4">PAB ADDRESS</td><td colspan="4">RETURN ADDRESS</td><td colspan="4">WORK ELEMENT QUEUE</td><td colspan="4">MODULE NAME or DVT ADDRESS</td><td colspan="4">RPH ADDRESS</td></tr>
</table>

Byte (hex)	Contents
00 – 03	Record ID: C'WAIT'
04	ID • For MVS, address space ID (or zero if address space ID is greater than X'FF') • For VM, machine and task ID • For VSE, task ID
05 – 07	Zero
08 – 0B	PST address
0C – 0F	PAB address
10 – 13	Address of the issuer of the TPWAIT macro
14 – 17	PAB work element queue
18 – 1B	Module name abbreviation (bytes 4, 5, 7, and 8 of the name of the next module to get control), or PAB DVT address (high order bit of X'18' = 0)
18 – 1B	PAB DVT address
1C – 1F	RPH address

Appendixes

Appendix A. Finding VTAM Buffer Pools

MVS VM General Procedure for Finding VTAM Buffer Pools

Use the following steps to find VTAM buffer pools and to determine whether any have used all their buffers. Figure 70 on page 494 shows the control blocks involved in these steps.

MVS Dumps processed with PRDMP contain all the data necessary to use this procedure if the operand VTAMMAP is coded in the PRDMP control statements. (See "**MVS VM** Formatting and Printing Dump Output" on page 279 for more information about dumps processed using PRDMP.)

VM Dumps processed with PRTDUMP contain all the data necessary to use this procedure if VTAM control block formatting was requested. (See the *VM/SP Interactive Problem Control System Guide* for more information about dumps processed using PRTDUMP.)

1. Find the **ATCVT**. Its address is at low-storage location MVS X'408' or VM X'200'.

   MVS If this low-address location is not available in a dump, use the pointer in the CVT extension (CVTATCVT) to find the AVT. Location X'00' in the AVT points to the ATCVT.

   The ATCVT is identified by release level at offset X'00':

   - For VTAM V3R1.1, the identifier is V311 (X'E5F3F1F1').
   - For VTAM V3R1.2, the identifier is V312 (X'E5F3F1F2').
   - For VTAM V3R2, the identifier is VT32 (X'E5E3F3F2').

2. From the ATCVT, find the address of the buffer pool directory (BPDTY) in field ATCBPDA.

3. BPDTY + X'170' is the first buffer pool entry (BPENT). There are 12 BPENTs, one for each buffer pool; each is X'10' bytes long and contains the name of a buffer pool at offset X'00', and the address of the buffer pool control block (BPCB) at offset X'08'.

4. Find each BPCB. Each BPCB represents a buffer pool.

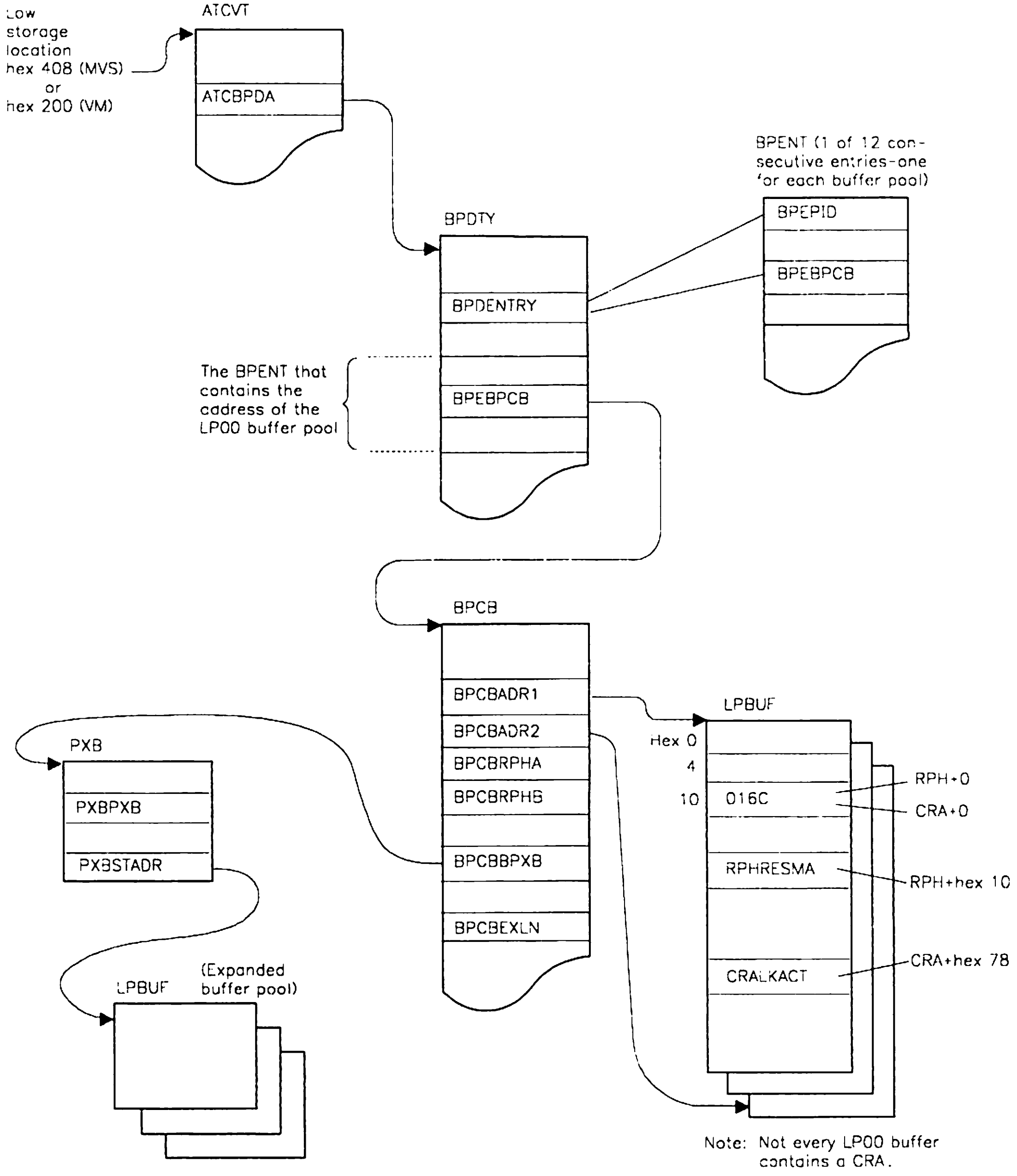

Low storage location hex 408 (MVS) or hex 200 (VM)
ATCVT
ATCBPDA
BPENT (1 of 12 consecutive entries-one for each buffer pool)
BPEPID
BPEBPCB
BPDTY
BPDENTRY
The BPENT that contains the address of the LP00 buffer pool
BPEBPCB
BPCB
BPCBADR1
BPCBADR2
BPCBRPHA
BPCBRPHB
BPCBBPXB
BPCBEXLN
PXB
PXBPXB
PXBSTADR
LPBUF
(Expanded buffer pool)
LPBUF
Hex 0
4
10
016C
RPHRESMA
CRALKACT
RPH+0
CRA+0
RPH+hex 10
CRA+hex 78
Note: Not every LP00 buffer contains a CRA.

Figure 70. Buffer Pool Control Block Relationships

MVS VM Finding LPBUF

Use the following steps to find the large pageable buffer pool (LPBUF) in MVS and VM. LPBUF is the buffer pool that contains the CRA/RPHS. Figure 70 on page 494 shows the control block relationships involved in these steps.

1. Find the ATCVT. Its address is at low-storage location MVS X'408' or VM X'200'.

   MVS If these locations are not available in a dump, use the pointer in the CVT extension (CVTATCVT) to find the AVT. Location X'00' in the AVT points to the ATCVT.

   The ATCVT is identified by release level at offset X'00':

   - For VTAM V3R1.1, the identifier is V311 (X'E5F3F1F1').
   - For VTAM V3R1.2, the identifier is V312 (X'E5F3F1F2').
   - For VTAM V3R2, the identifier is VT32 (X'E5E3F3F2').

2. From the ATCVT, find the address of the buffer pool directory (BPDTY) in field ATCBPDA.

3. From the BPDTY, find the buffer pool control block (BPCB) for the large pageable buffer pool (LP00). The BPCB is pointed to by BPDTY + X'1A8'.

4. The beginning address of the static portion of the LP00 buffer pool is at BPCB + X'10'; the end of the static portion is pointed to by BPCB + X'14'.

5. The first buffer begins at the address found in the BPCB + X'10'. To verify the correct buffer pool, offset X'08' in each buffer should have the address of the BPCB found in step 3. A buffer is allocated when bit BFPALLOC is ON.

6. To find the second and third buffers, add X'550' to the address of the previous buffer. To find the fourth buffer, skip to the next page boundary (its address is a multiple of X'1000'), or add X'560' to the address of the third buffer. Three buffers are on each 4K page of storage, followed by 16 unused bytes.

   When you reach the address in BPCB + X'14', you have found the end of the LP00 buffer pool.

7. To find LP00 buffer pool expansion areas, use the address of the pool extension block (PXB) at BPCB + X'90'. If the address is zero, there are no expansions — the buffer pool has not been expanded since VTAM was started. In this case, skip steps 8 and 9. If there is an address, the buffer pool has been expanded dynamically and the expansions need to be examined; continue with the next step.

8. Find the PXB, pointed to by BPCB + X'90'. The PXB offset X'0C' (PXBSTADR) is the beginning of the pool expansion area. If PXBSTADR=0, go to step 9. PXBS are not freed by VTAM; only the area containing buffers is freed. The length of the expansion area (BPCBEXLN) is found at BPCB + X'D8. Add the length (BPCBEXLN) to the starting address (PXBSTADR) for the end of the expansion area.

9. Find the next PXB, pointed to by PXB + X'04' (PXBPXB). If PXBPXB≠0, proceed as in step 8. If PXBPXB=0, there are no more expansion areas; the search is complete.

VSE General Procedure for Finding VTAM Buffer Pools

Use the following steps to find the buffer pools and determine whether any have used all of their available buffers. Figure 71 on page 497 shows the control blocks involved in these steps.

1. Find the **ATCVT**. Its address is at low-storage location X'10'.

   If this low-address location is not available in a dump, use the pointer at SYSCOM + X'60' to find the AVT. Location X'00' in the AVT points to the ATCVT.

   The ATCVT is identified by release level at offset X'00':

   - For VTAM V3R1, the identifier is VT31 (X'E5E3F3F1').
   - For VTAM V3R1.2, the identifier is V312 (X'E5F3F1F2').
   - For VTAM V3R2, identifier is VT32 (X'E5E3F3F2').

2. The ATCVT is immediately preceded in storage by the buffer pool directory (BPDIR) and the fixed-length pool PCBS, which can be identified by the pool IDs in EBCDIC. ("VSE Relative Locations of VTAM Buffer Pools" shows the relative locations of the buffer pools.)

   ATCVT + X'3D4' also points to the BPDIR.

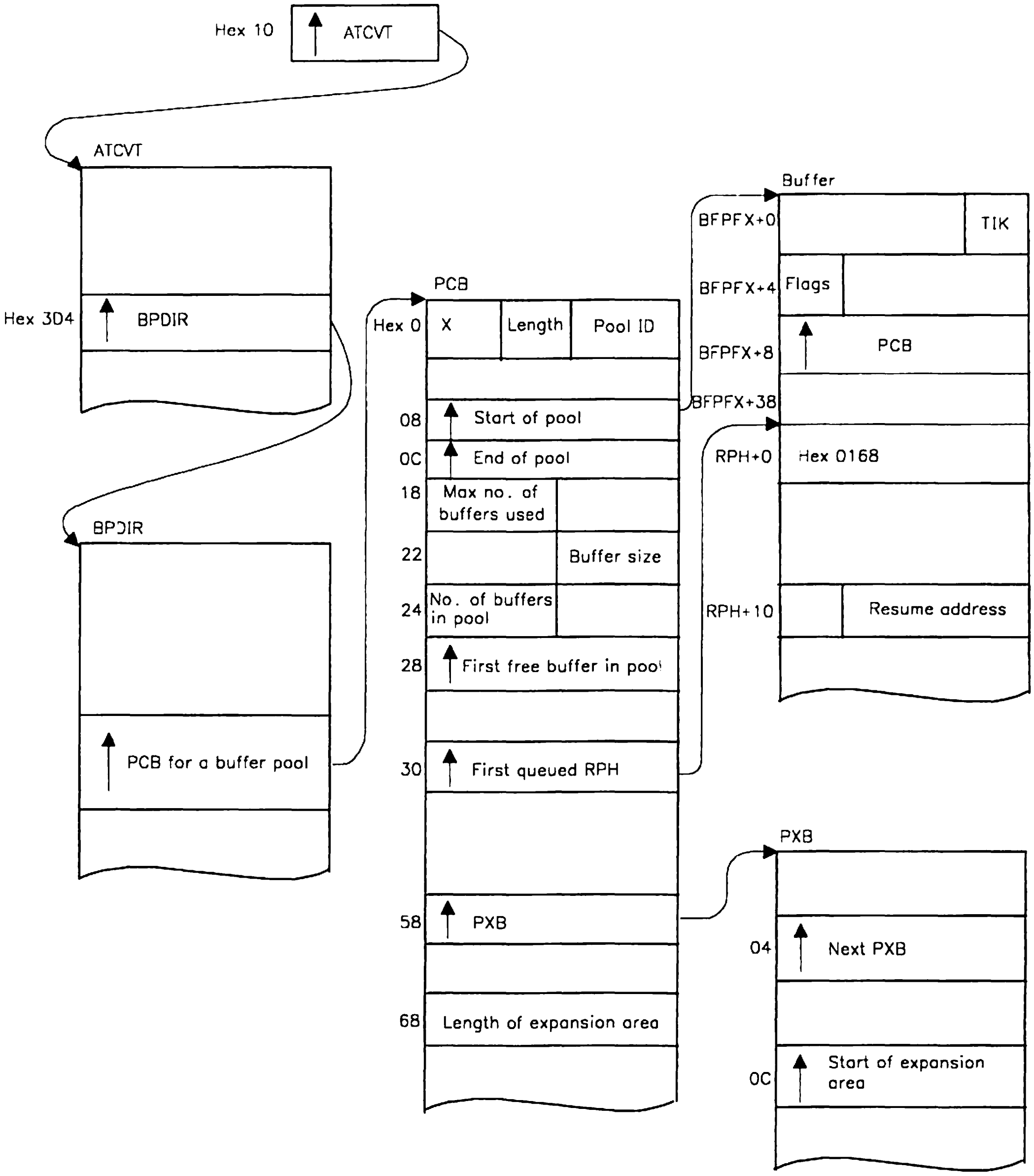


Figure 71. VSE Buffer Pool Control Block Relationships

VSE Finding LPBUF

Use the following steps to find the large pageable buffer pool (LPBUF) in VSE. LPBUF is the buffer pool that contains the CRA/RPHS. Figure 71 on page 497 shows the control block relationships involved in these steps. ("VSE Relative Locations of VTAM Buffer Pools" has more information on the location of VTAM buffer pools in VSE.)

1. Find the ATCVT. Its address is at low-storage location X'10'. If this location is not available in a dump, use the pointer at SYSCOM + X'60' to find the AVT. Location X'00' in the AVT points to the ATCVT.

   The ATCVT is identified by release level at offset X'00':

   - For VTAM V3R1, the identifier is VT31 (X'E5E3F3F1').
   - For VTAM V3R1.2, the identifier is V312 (X'E5F3F1F2').
   - For VTAM V3R2, the identifier is VT32 (X'E5E3F3F2').

2. From the ATCVT, find the address of the buffer pool directory (BPDIR) in field ATCBPDA (ATCVT + X'3D4').

3. From the pointer at BPDIR + X'14', find the buffer pool control block (PCB) for the large pageable buffer pool (LP). At offset X'08' in the PCB is the address of the start of the LP pool.

4. To find the next buffer, add X'550' to the address of the buffer prefix (BFPFX) of the previous buffer. Continue with this process until the end of the LP buffer pool (at the address in PCB + X'0C').

   Each buffer (including the prefix) is 1360 (X'550') bytes, and the buffers can cross page boundaries. The buffer size is also given in PCBBUFSZ (PCB + X'22').

   The total number of buffers is given in PCBTOTNO (PCB + X'24').

5. To find the LP buffer pool expansion areas, use the address of the pool extension block (PXB) at PCB + X'58'. If the address is zero, there are no expansions — the buffer pool has not been expanded since VTAM was started. In this case, skip steps 6 and 7. If there is an address, the buffer pool has been expanded dynamically and the expansions need to be examined; continue with the next step.

6. Find the PXB, pointed to by PCB + X'58'. The PXB offset X'0C (PXBSTADR) is the beginning of the pool expansion area. If PXBSTADR=0, go to step 7. PXBs are not freed by VTAM; only the area containing buffers is freed. The length of the expansion area (PCBEXLN) is found at PCB + X'68'. Add the length (PCBEXLN) to the starting address (PXBSTADR) for the end of the expansion area.

7. Find the next PXB, pointed to by PXB + X'04' (PXBPXB). If PXBPXB≠0, proceed as in step 6 If PXBPXB=0, there are no more expansion areas; the search is complete.

VSE Relative Locations of VTAM Buffer Pools

To find a particular VTAM buffer pool in a VSE system, see the partition map in Figure 72 on page 500 and follow these steps:

1. Find the ATCVT from location X'10', if available, or from offset X'00' in the AVT.
2. Find the buffer pool directory (BPDIR) from ATCVT + X'3D4', or by subtracting X'30' from the ATCVT address.
3. Find the pool control blocks (PCBS) from the pointers in BPDIR, or by finding the EBCDIC pool ID preceding the BPDIR in the storage dump (for fixed-length pools only).

For all buffer pools, the beginning and ending addresses of the initial pool extent are contained in the PCB at offsets X'08' and X'0C', respectively.

For fixed-length pools, the storage area may consist of one or more extents. The initial extent allocated when VTAM is started contains the number of buffers indicated as a start parameter. Additional extents may be allocated if dynamic expansion is allowed for the pool. See *Network Program Products Planning* and *VTAM Installation and Resource Definition* for more information.

To find the buffers in a fixed-length pool, start with the buffer prefix (BFPFX) pointed to by PCB + X'08'. The buffer prefix (BFPFX) is the 60-byte area, the third word of which points to the PCB. To find other buffers, add to or subtract from the address of the current prefix the length of the buffer (contained in the PCB + X'22') plus 60 (X'3C') for the prefix. If the new location does not contain a BFPFX pointing to the same PCB, you have reached the end of the pool.

Use the count of buffers at PCB + X'24' to determine whether there are also expansion extents. Information about them is found in the PXB, which is chained from PCB + X'58'. If that location is not equal to zero, and if the total buffer count at PCB + X'24' is greater than the number of buffers found in the initial extent, there are additional buffers pointed to by PXBS. Each PXB may point to a storage area. Examine location X'0C' in the PXB. If that location is not equal to zero, it is the beginning address of the extent. The number of buffers is contained at PXB + X'16'. Examine the PXB chain pointer at X'04' to find the next PXB, if any.

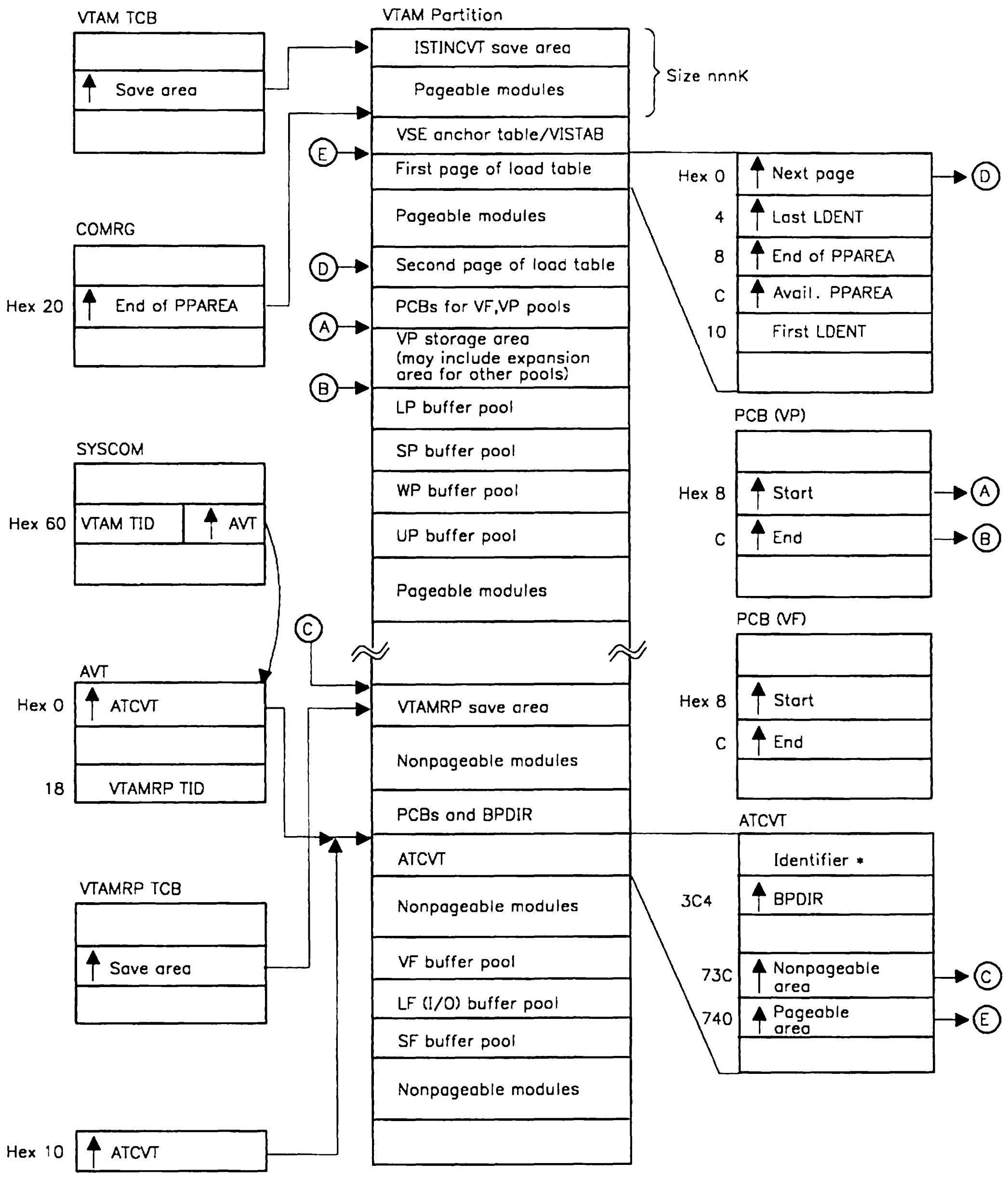


Figure 72. **VSE** Map of the VTAM Partition.

* The identifier in ATCVT is VT31 (X'E5E3F3F1') for VTAM V3R1, V312 (X'E5F3F1F2') for VTAM V3R1.2, or VT32 (X'E5E3F3F2') for VTAM V3R2.

Appendix B. Channel Programs

If you determine that the problem is in an I/O sequence, you will need to know the normal sequence of I/O CCWs within the channel programs for the communication controller and channel-attached devices. With a dump, the VIT trace with the CIO option, or a CCWTRACE (if available), you can compare the sequence that happened with the expected sequence. When there is a deviation, you can then look at status and sense bytes returned from the communication controller or the cluster controller for information that can help you determine the location of the error condition.

Channel Programs for Channel-Attached Type 2 and 4 Physical Units

The ICNCB represents type 2 and 4 physical units, and contains addresses and CCWS needed for channel programs. Figure 73 shows the ICNCB and the location in storage of various CCWS.

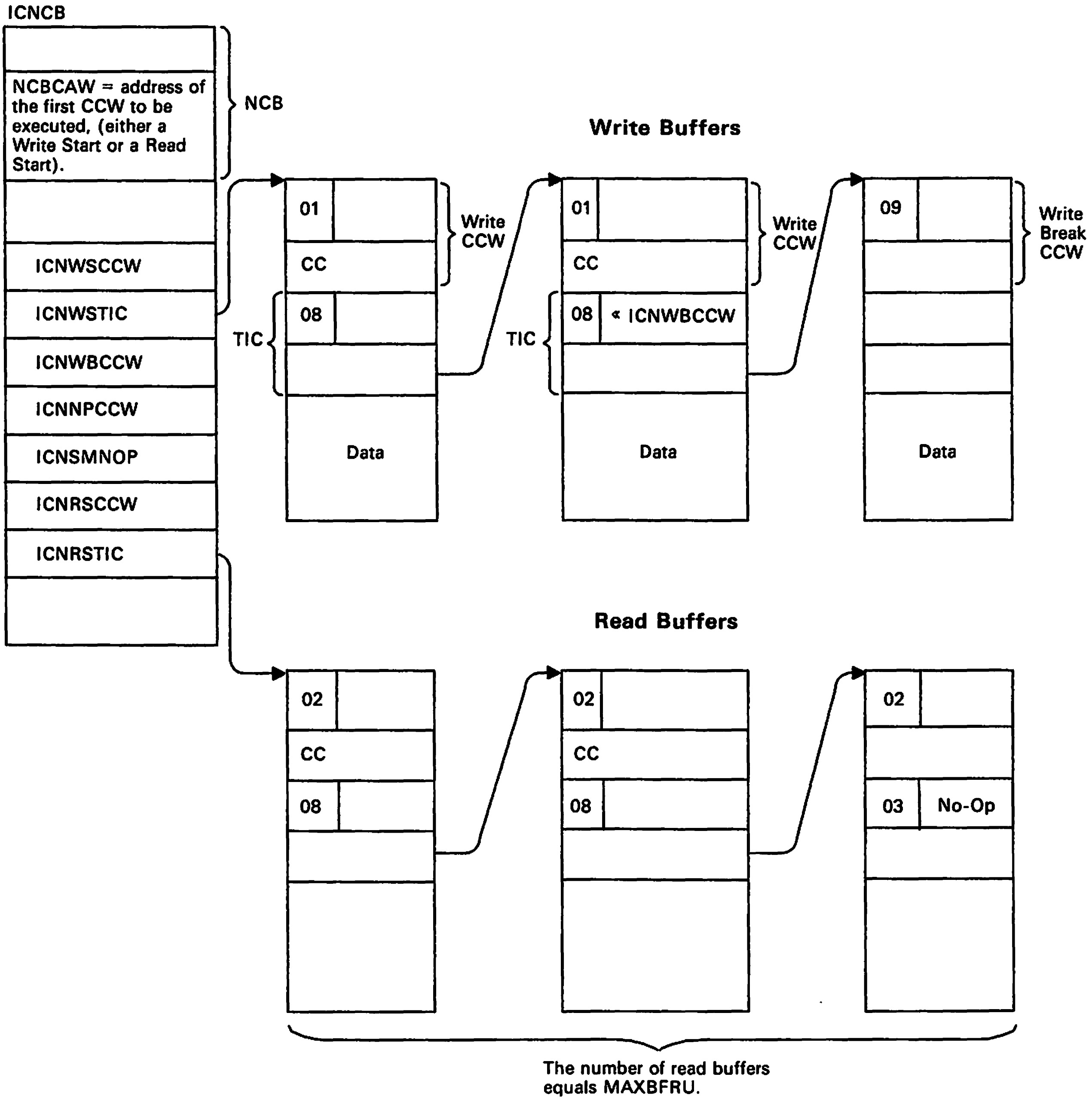


Figure 73. Data Areas Used by Channel Programs for Physical Unit Types 2/4

Channel Commands

The following channel commands are used by VTAM to write data to and read data from channel-attached type 2 and 4 physical units.

Command Code (hex)	Command	Description	Contents of Address Field
01	Write	Transfers data from storage in the host processor to the I/O device.	Output area
02	Read	Transfers data from the I/O device to storage in the host processor.	Input area
03	No-op	Causes the channel to respond with a channel end and device end. It is the last CCW in a read or write chain.	Zero
04	Sense	Transfers 1 or 2 bytes of sense data to storage in the host processor.	Address of sense data area
08	TIC (Transfer in Channel)	Causes the channel to fetch an instruction that is not the next sequential instruction within the channel program sequence.	Address of next CCW to be executed
09	Write Break	Transfers data from storage in the host processor to the I/O device and indicates that it is the last or only Write command in a chain of Write CCWs.	Output area
31	Write Start 0	Begins a write sequence. Alternates with Write Start 1.	Zero
32	Read Start 0	Begins a read sequence. Alternates with Read Start 1.	Zero
51	Write Start 1	Begins a write sequence. Alternates with Write Start 0.	Zero
52	Read Start 1	Begins a read sequence. Alternates with Read Start 0.	Zero
61	Write XID	The host sends the Write XID command to signal the NCP that a channel contact sequence is beginning and to prepare to receive the host's XID.	Zero
62	Read XID	The host sends the Read XID command to signal the NCP that the host expects to read the NCP's XID.	Zero

Figure 74 (Part 1 of 2). VTAM Channel Commands for Type 2 and 4 Physical Units

Command Code (hex)	Command	Description	Contents of Address Field
A3	Discontact	Indicates that the channel is no longer contacted and the attachment to the transmission group should be broken. Releases the PIUS on the channel hold and intermediate queues.	Zero
C3	Contact	Establishes contact between the host and the NCP. Tells the NCP to use XID information for operations with the host.	Zero
93	Restart	Causes the controller to reset its switches to indicate that the last Write Start and Read Start commands were Write Start 1 and Read Start 1 commands.	Zero

Note: Data transfer does not occur on Read Start or Write Start commands.

Figure 74 (Part 2 of 2). VTAM Channel Commands for Type 2 and 4 Physical Units

Format of TIC CCWs

MVS/370 VM VSE The Format 0 TIC CCW is formatted as follows:

Byte (hex)	Contents
00	X'08' (TIC identifier)
01 – 03	Real address
04 – 07	Reserved

MVS/XA The Format 1 TIC CCW is formatted as follows:

Byte (hex)	Contents
00	X'08' (TIC identifier)
01 – 03	Zero
04 – 07	Real address

A doubleword TIC extension immediately follows both the Format 0 TIC and the Format 1 TIC. VTAM uses the last 4 bytes of the TIC extension to contain the virtual address of the next buffer in the chain.

The TIC extension is formatted as follows:

Byte (hex)	Contents
08 – 0B	Reserved
0C – 0F	Virtual address

For Write buffers, the next to the last physical buffer is handled specially. The real address of the TIC points to the last Write Break CCW (ICNWBCCW), but the virtual address points to the last Write buffer that contains data (see Figure 75 on page 505.) This last buffer is formatted with a Write Break CCW which is not used, but is copied into ICNWBCCW. If only one buffer exists in the channel program, the Write Start TIC is formatted so that the real address points to ICNWBCCW, but the virtual address points to the only Write buffer.

Note that the last Write buffer looks unusual because the Write Break command is chained but the next CCW is zero. This Write Break is never physically executed by the channel, but the copied version of the CCW (in the ICNCB) is executed.

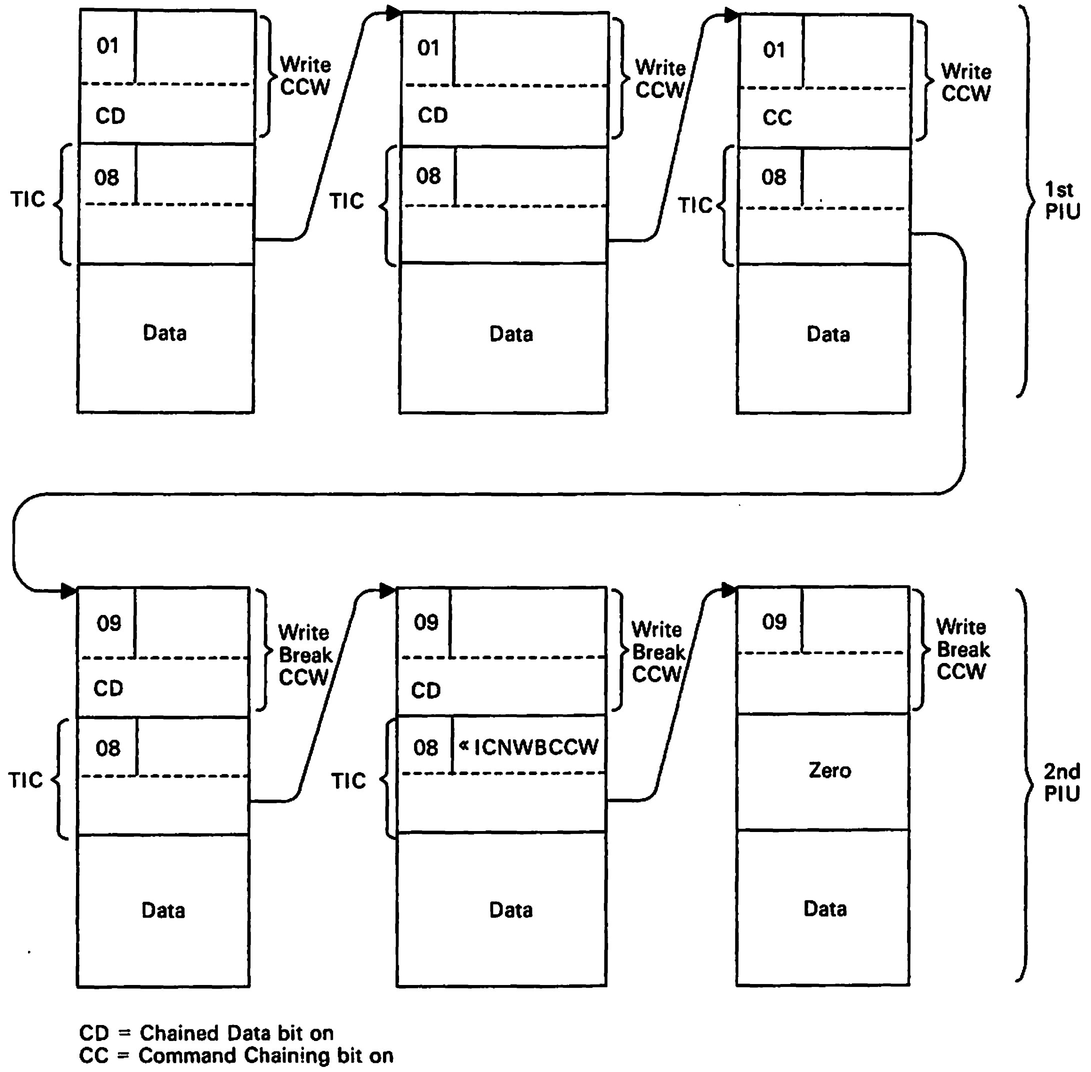


Figure 75. Write CCWs with Chained Data

Channel Program Sequences

The following describes the CCWS, in the order that they are executed, in a typical Write channel program.

1. *ICNWSCCW - Write Start:* Each time a Write sequence (the Write Start and all associated Write buffers) has completed successfully, the Write Start CCW is alternated: The first Write Start operation code is X'31' (Write Start 0), the second Write Start code is X'51' (Write Start 1), the third Write Start code is X'31', and so on. There is no data transfer associated with this CCW; it is used to inform the controller that the previous Write has successfully completed.

2. *ICNWSTIC - Write Start TIC:* This CCW is used to chain to the first Write CCW. When the Write channel program consists of a single Write buffer, the real address points to ICNWBCCW (the last Write Break). The virtual address of this CCW always points to the first Write buffer on a chain or is zero when no Write channel program is pending or active for the device.

3. *Write or Write Break CCWs Alternating with TICs:* Figure 73 on page 502 shows the write buffers required for writing three PIUS, each of which is contained in a single buffer. Figure 75 on page 505 shows the Write buffers required for writing two PIUS, when each spans three buffers.

4. *ICNWBCCW CCW(3) - Last Write Break:* This is the last Write CCW in a channel program that is physically executed by the device. The data address points to the last Write buffer that contains data. If a read is requested, when this operation completes the controller will signal a status modifier (in addition to channel end and device end). This causes ICNNPCCW to be skipped by the channel.

5. *ICNNPCCW - No-Op:* If a Write completes and no Read is requested, this is the ending CCW in a channel program. If a Read was requested, this CCW is skipped by the channel. The command chain flag will be on in this CCW if a Read was previously requested but was not previously initiated, and Read buffers are available.

6. *ICNSMNOP - Status Modifier No-Op:* This CCW receives control after the last Write Break, if a Read is signaled. This CCW normally command chains to the Read Start so that writing and reading both occur without an intervening interruption. If no buffers are available, however, the channel program ends here and VTAM recognizes that a Read is required.

7. *ICNRSCCW - Read Start:* Each time a Read sequence (the Read Start and at least one Read) has completed successfully, the Read Start CCW is alternated.

   The first Read Start generation code is X'32' (Read Start 0), the second Read Start generation code is X'52' (Read Start 1), the third Read Start generation code is X'32', and so on.

   As with the Write Start CCW, there is no data transfer. The CCW alternation is used to inform the controller that the previous Read has successfully completed.

   When a Write is not required but a Read has been requested by an attention status, this is the first CCW in the channel program.

8. *ICNRSTIC - Read Start TIC:* This CCW is used to chain to the first Read CCW. The virtual address of this CCW points to the first Read buffer on the chain, except:

   - When deblocking PIUs, in which case the virtual address is changed by the channel end appendage.
   - When there are not enough read buffers available, in which case it is zero.

9. *Read CCWs Alternating with TICs:* Figure 73 on page 502 shows the buffers required for reading three PIUs.

10. *Read No-Op:* This CCW should never be executed. If it is, it indicates that the controller and VTAM do not agree on how many Read buffers are required.

    The normal ending status for a Read is channel end/device end/attention/unit exception. The unit exception indication is presented on the Read CCW that has completed data transfer, and terminates the command chaining. The attention is the same as a unit exception but it also indicates that a Read is requested.

    Unit exception may also be presented to the Read Start CCW. This is used by the controller to release input buffers. If the same Read Start (as opposed to the alternate Read Start) is given to the controller, the data buffers must be re-sent.

PUNS-Related Channel Programs

During activation of an NCP in a channel-attached communication controller, the SSCP sends a Contact RU to PUNS. PUNS responds by giving control to ISTTSCP4 to schedule one of five channel programs. These channel programs follow (A – E).

	CCW	Code (hex)	Flags	Notes
A.	Sense	04	SLI	Determines if the device needs to be loaded.
B.	Write XID No-Op	61 03	SLI,CC SLI	This channel program follows A if the device does not need to be loaded. It is followed by either C or D.
C.	Write Break Read XID Read No-Op	09 62 02 03	SLI,CC SLI,CC SLI,CC SLI	This channel program follows B if there is no command reject.
D.	Restart/Reset No-Op	93 03	SLI,CC SLI	Executed only if the Write XID in B caused a command reject (implies the NCP is NCP Release 2 or earlier).
E.	Contact Restart/Reset No-Op	C3 93 03	SLI,CC SLI,CC SLI	Restart/Reset is executed only if a Contacted (error) response is sent to PUNS.

One of the following channel programs is executed when the SSCP sends a Discontact RU to PUNS.

	CCW	Code (hex)	Flags	Notes
F.	Discontact No-Op	A3 03	SLI,CC SLI	This channel program is executed only if B did not cause a command reject.
G.	No-Op	03	SLI	This channel program is executed if B caused a command reject.

These CCWs are contained in I/O buffers that are allocated from the IOBUF buffer pool when doing PUNS I/O.

The data area pointed to by the address portion of a Read XID or Write XID CCW is described in *VTAM Data Areas* (see ISTXID).

The following channel programs are written by adjacent physical unit network services (APUNS) for pseudo-dial support of a type 2 physical unit.

	Channel Program Name	CCW	Code (hex)	Flags	Notes
A.	Dial Out	Sense No-Op	04 03	SLI,CC SLI	Executed when the SSCP sends a Connect Out request to PUNS. Success means the physical connection is complete. Failure means the device is not operational.
B.	Dial Enable Answer	Sense No-Op	04 03	SLI,CC SLI	Executed when the SSCP sends an Activate Connect In. Success means that the physical connection exists. (The type 2 physical unit is dialed in.) Failure means that the physical connection does not exist. (The type 2 physical unit has not dialed in yet.)
C.	Dial Contact (Connect)	IPL (Connect) Restart/Reset No-Op	05 93 03	SLI,CC SLI,CC SLI	Executed during activation of a type 2 physical unit when the SSCP sends a Contact request to PUNS. The IPL CCW transfers initialization parameters to the device. The Restart/Reset CCW resets the Read/Write indicators so that VTAM and the device are synchronized.
D.	Abandon Connection	IPL (Disconnect) No-Op	05 03	SLI,CC SLI	Executed when the SSCP sends an Abandon Connection RU to PUNS to disconnect the type 2 physical unit.

These CCWs are contained in pageable working storage allocated by APUNS (ISTINCYO).

Figure 76 on page 510 shows the data area pointed to by an IPL (Connect) CCW.

Figure 77 on page 510 shows the data area pointed to by an IPL (Disconnect) CCW.

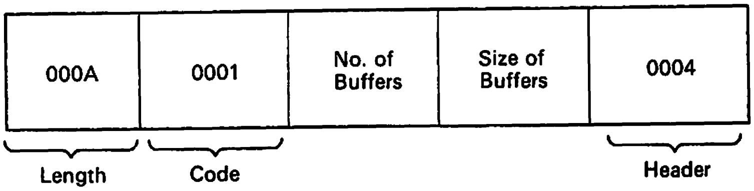


Figure 76. Data Area Pointed to by an IPL (Connect) CCW

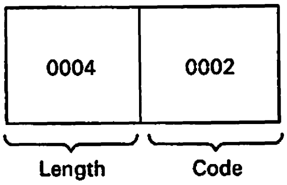


Figure 77. Data Area Pointed to by an IPL (Disconnect) CCW

Channel Programs for Channel-to-Channel Adapters

Channel Programs for Channel-to-Channel Adapters for Pre-V3R2

The following applies to VTAM V3R1, V3R1.1, and V3R1.2 in which the "CTC enhancement" PTF has *not* been applied. If the "CTC enhancement" PTF has been applied to your VTAM V3R1, V3R1.1, or V3R1.2 system, or if you are using VTAM V3R2, please see "Channel Programs for Channel-to-Channel Adapters for V3R2" on page 514 for the description of channel programs for channel-to-channel adapters.

Each side of a channel-to-channel adapter (CTCA) is represented by a cross-channel node control block (XCNCB). In addition, each side has a physical unit service I/O (PIO) control block and a station control block (SCB). The PIO is used for exchange ID (XID) channel programs (the physical unit services I/O that occurs before the link is active). The PIO is mapped by ISTPIO. The SCB is a station work area where CCWS for normal data transfer are built. It is not mapped, and is *not* described in *VTAM Data Areas.*

Normal data transfer channel programs can be either a sequence of write CCWS followed by read CCWS (this is on the X-side) or a sequence of read CCWS followed by write CCWS (on the Y-side). VTAM determines which side will be the X-side and which side will be the Y-side during the initial XID channel program (the X-side/Y-side usage does not depend on the hardware attachment). From that point on, the X-side of the adapter always builds a channel program with the write CCWS first, followed by read CCWS. TheY-side of the adapter always builds a corresponding channel program with read CCWS first, followed by write CCWS.

Figure 78 on page 511 shows how VTAM determines which side will be the X-side and which side will be the Y-side. In this example, the operator in subarea 4 is the first to activate the link. Because subarea 2 is not active, the XID exchange does not complete. Later, when the operator in subarea 2 activates the link, the XID exchange completes.

During the XID exchange, each host informs the other host of the size and number of read buffers that will be used in the read portion of its channel programs. Each host then tailors the write portion of its channel programs to exactly match the read portion of the other host's channel programs. Figure 79 on page 512 and Figure 80 on page 513 show how a write channel program might have to be tailored to match a read channel program.

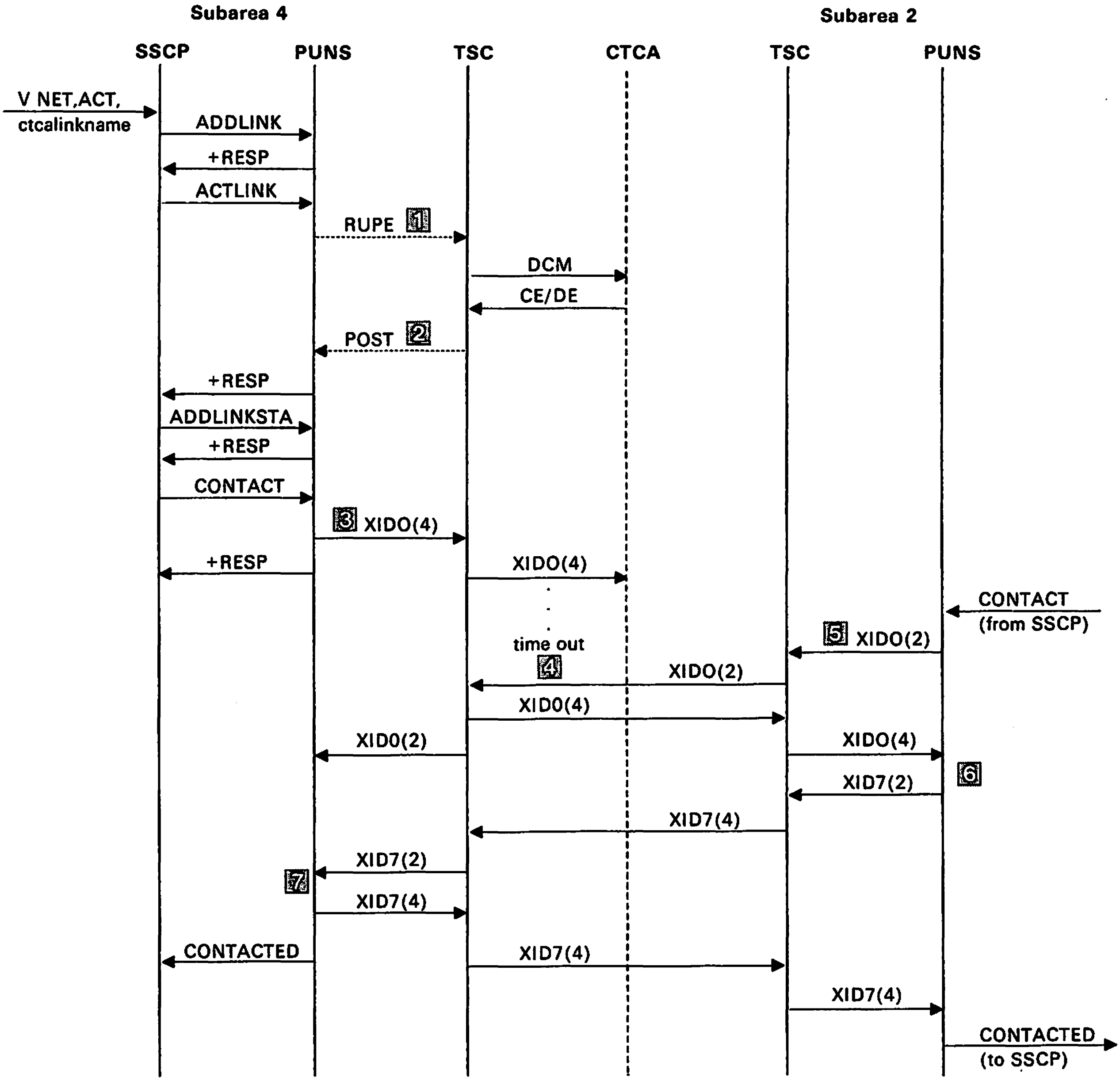


Figure 78. Example of an XID Exchange

1 PUNS queues a RUPE to NCBPUPAB, causing TSC to get control.

2 TSC posts PUNS.

3 Subarea 4 assumes X-side protocols because it is initiating the XID exchange.

4 The XID exchange "times out" because the link in subarea 2 has not been activated. Subarea 4 then assumes Y-side protocols and waits for the other side to initiate its contact procedure.

5 Assume the operator activates the link in subarea 2. Subarea 2 assumes X-side protocols.

6 The low subarea responds first to the XID0 exchange by sending an XID7. (Subarea 2 has a lower subarea number than subarea 4.)

7 Subarea 4 checks the XID7 from subarea 2 and responds with its XID7. It also sends a CONTACTED RU to the SSCP.

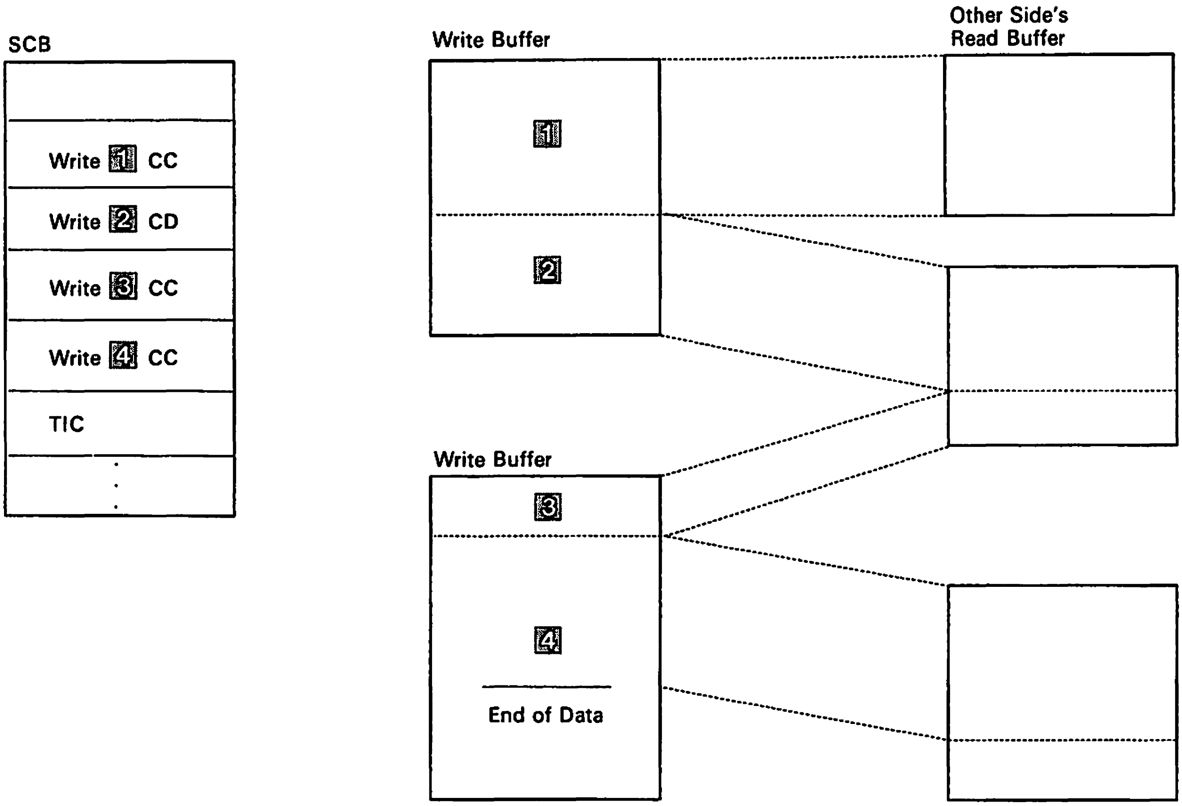


Figure 79. Writing from Large Buffers into Small Buffers. In this example, the write buffers are larger than the read buffers. Since the second and third Write CCWs are writing into the same read buffer, the second Write CCW has the Chained Data (CD) bit set on, and the second and third CCWs cause a single write operation.

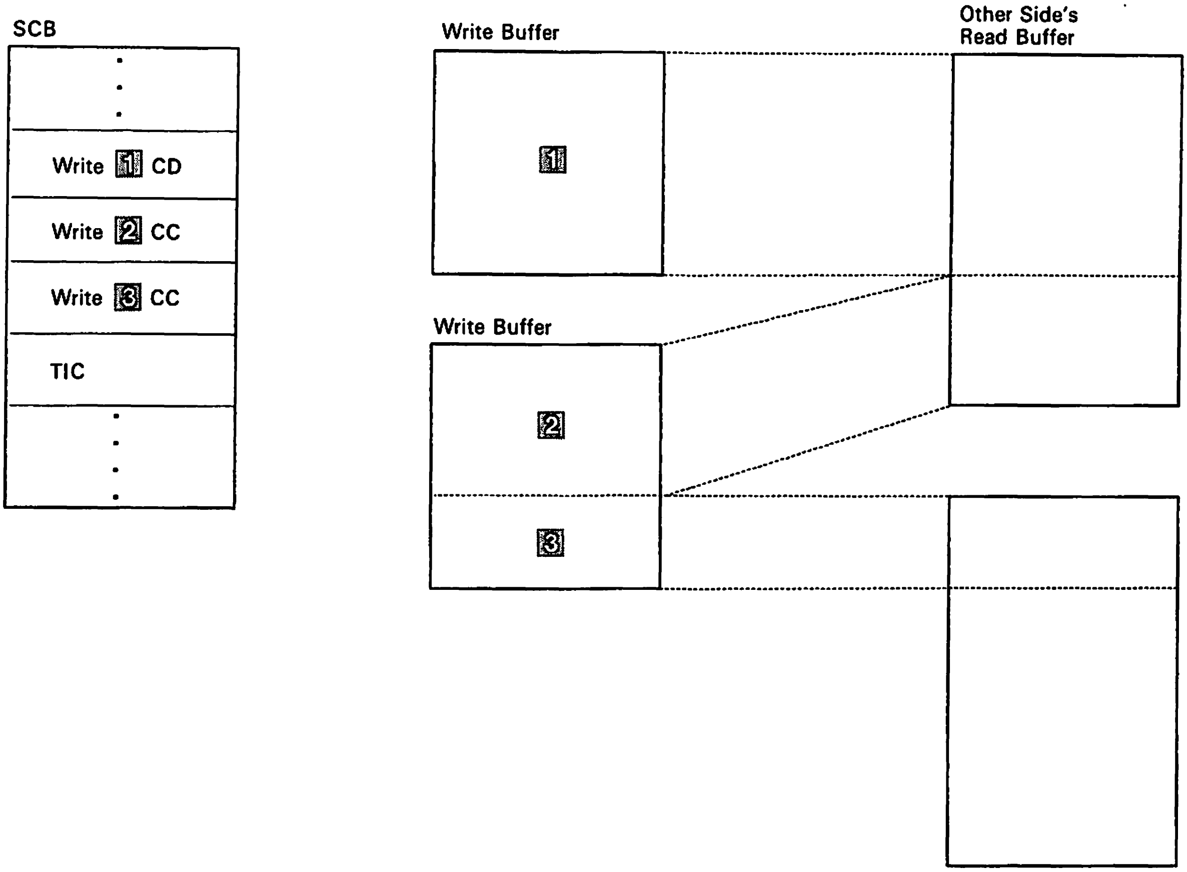


Figure 80. Writing from Small Buffers into Large Buffers. In this example, the read buffers are larger than the write buffers. Therefore, two consecutive write operations are required to fill one read buffer. The first Write CCW has the Chained Data (CD) bit set on, since the first and second Write CCWs are writing into the same read buffer.

Channel Programs for Channel-to-Channel Adapters for V3R2

The following applies to VTAM V3R2, and to VTAM V3R1, V3R1.1, and V3R1.2 *with* the "CTC enhancement" PTF applied. References to PRE-V3R2 are for VTAM V3R1, V3R1.1, and V3R1.2 in which the "CTC enhancement" PTF has *not* been applied.

During XID processing, VTAM indicates the support level of CTC communications. If the other host in the connection is back-level, the support used will be the VTAM CTC support as described in "Channel Programs for Channel-to-Channel Adapters for Pre-V3R2" on page 510.

VTAM V3R2 reduces the number of CCWS exchanged during XIO processing.

Prior to V3R2, there were MAXBFRU READS and WRITES, plus a constant seven environmental CCWS.

With V3R2, MAXBFRU defines a *single* READ buffer. The opposing VTAM WRITE buffer is exactly the same size. PIUS are blocked for transfer, and are written by a single WRITE, and read by a single READ.

With the new blocking protocol, there are just three CCWS. V3R2 uses the WRITE and READ CCWS, as well as a CCW which is either a WCTL or SCB. For each buffer write, only 3 CCWS are exchanged in V3R2. Control information is included with the first buffer transferred in the same CCW as the PIUS in the block. This eliminates the use of WRITE CONTROL, READ CONTROL, validity check, and NO_OP CCWS.

Each side of a channel-to-channel adapter (CTCA) is represented by a cross-channel node control block (XCNCB/YCNCB). In addition, each side has a physical unit service I/O (PIO) control block. The PIO is used for exchange ID (XID) channel programs (the physical unit services I/O that occurs before the link is active). The PIO is mapped by ISTPIO.

Normal data transfer for channel programs is a write CCW followed by a read CCW (this is on the X-side) or a read CCW followed by a write CCW (on the Y-side). VTAM determines which side will be the X-side and which side will be the Y-side as described in "Channel Programs for Channel-to-Channel Adapters for Pre-V3R2" on page 510.

During the XID exchange, each host informs the other host of the size in pages of the read buffer that will be used in the read portion of its channel programs. Each host then allocates a WRITE buffer to exactly match the READ buffer of the other host.

Channel Commands

The following channel commands are used for channel programs between two hosts connected by a channel-to-channel adapter.

Command Code (hex)	Command	Description
01	Write	Transfers data from storage to this host processor to the CTCA. For CTCAs, this CCW is used to transfer status information from XCNOCTL, XID information from PIOPOTXT, validity checking information from XCNOVTXT, and data from buffers. In VTAM V3R2, status information, mapped by XCNOCTL, is transferred in the first eight bytes of the write buffer, and only one write buffer is used. Figure 81 on page 516 illustrates the buffer usage in V3R2.
02	Read	Transfers data from the CTCA to this host processor. For CTCAs, this CCW is used to read status information into XCNICTL, XID information into PIOPITXT, validity checking information into XCNIVTXT, and data into buffers. In VTAM V3R2, status information, mapped by XCNICTL, is transferred in the first eight bytes of the read buffer, and only one read buffer is used. Figure 81 on page 516 illustrates the buffer usage in V3R2.
03	No-Op	Causes the channel to respond with a channel end or device end. It is the last CCW in a read or write chain.
08	TIC	Causes the channel program to execute an instruction that is not the next sequential instruction within the channel program sequence.
14	Sense Command Byte (SCB)	The SCB is normally issued in response to an attention generated when the adapter processes a WCTL from the other side. The SCB clears WCTL from the adapter, allowing the WCTL CCW to complete.
17	Write Control (WCTL)	Causes an attention interruption on the other side of the channel-to-channel adapter. The WCTL is issued to alert the other side that a channel program is active at the adapter.

Command Code (hex)	Command	Description
43	Enable Compatibility Mode (ECM)	Prepares the adapter to operate in System/360 (compatibility) mode. The ECM is issued when VTAM gives up control of the adapter.
C3	Disable Compatibility Mode (DCM)	Prepares the adapter to operate in System/370 (Extended) mode. The DCM is issued when VTAM acquires control of the adapter.

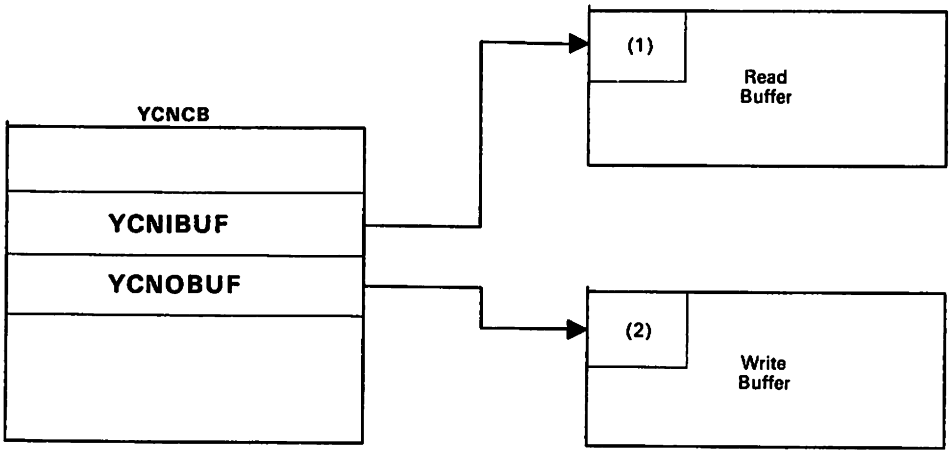


Figure 81. Buffers Used for Normal Data Transfer in V3R2

1. The first eight bytes of the READ buffer is the control information mapped by XCNICTL.

2. The first eight bytes of the WRITE buffer is the control information mapped by XCNOCTL.

Channel Programs

XID Channel Program (X-Side)

Figure 82 on page 518 shows the data areas associated with the following XID channel program.

	CCW	Command Code (hex)	Flags	Address	Count
1	Write Control	17	CC,SLI	Zero	1
2	TIC (Pre-V3R2)	08	—	Write CCW	—
3	Write	01	CC,SLI	XCNOCTL	8
4	Write	01	CC,SLI	PIOPOTXT	Length of XID
5	TIC (Pre-V3R2)	08	—	Read CCW	—
6	Read	02	CC,SLI	XCNICTL	8
7	Read	02	CC,SLI	PIOPITXT	Length of XID
8	TIC (Pre-V3R2)	08	—	Read CCW	—
9	Read	02	CC,SLI	XCNOVTXT	4

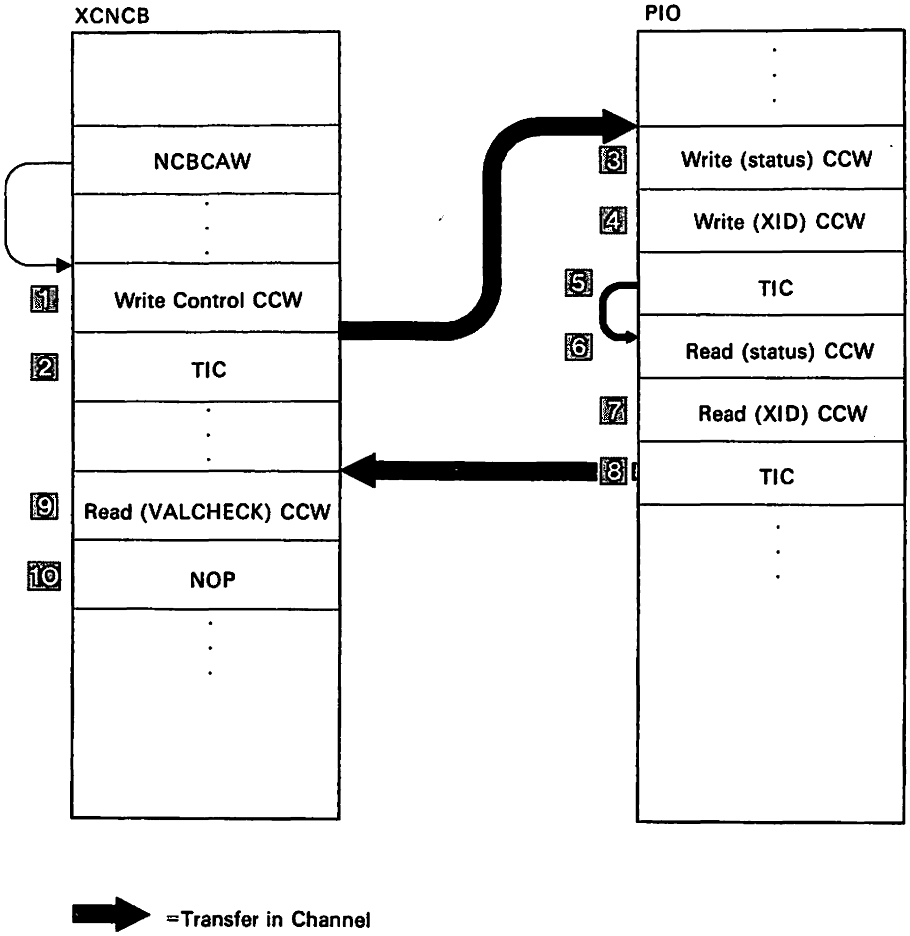


Figure 82. Data Areas Used for XID Channel Programs (X-Side)

XID Channel Program (Y-Side)

Figure 83 shows the data areas associated with the following XID channel program.

	CCW	Command Code (hex)	Flags	Address	Count
1	Sense Command Byte	14	CC,SLI	Zero	1
2	TIC (Pre-V3R2)	08	—	Read CCW	—
3	Read	02	CC,SLI	XCNICTL	8
4	Read	02	CC,SLI	PIOPITXT	Length of XID
5	TIC (Pre-V3R2)	08	—	Write CCW	—
6	Write	01	CC,SLI	XCNOCTL	8
7	Write	01	CC,SLI	PIOPOTXT	Length of XID
8	TIC (Pre-V3R2)	08	—	Write CCW	—
9	Write	01	CC,SLI	SCNOVTXT	4

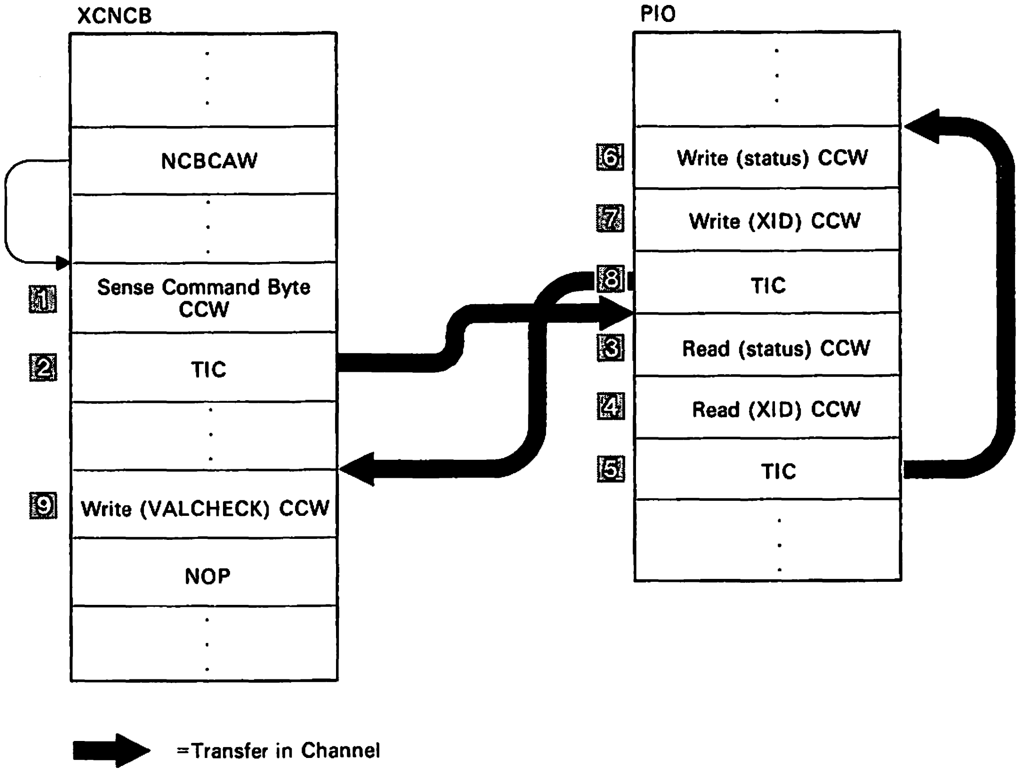


Figure 83. Data Areas Used for XID Channel Programs (Y-Side)

Normal Data Transfer (X-Side) in Pre-V3R2

Figure 84 on page 521 shows the data areas associated with the X-side of a normal data transfer channel program in PRE-V3R2 processing.

	CCW	Command Code (hex)	Flags	Address	Count
1	WCTL[3] or SCB	17/ 14	CC,SLI	Zero	1
2	TIC	08	—	Write CCW	—
3	Write	01	CC,SLI	XCNOCTL	8
4	TIC	08	—	Write CCW	—
	Write	01	CC,SLI (CD)	Buffer	Length
	:				
	:				
5	Write	01	CC,SLI, (CD)	Buffer	Length
	:				
	:				
	Write	01	CC,SLI	Buffer	Length
6	TIC	08	—	Read CCW	—
7	Read	02	CC,SLI	XCNICTL	8
8	TIC	08	—	Read CCW	—
	Read	02	CC,SLI	Address of Data	Length
9	TIC	08	—	Read CCW	—
	:				
	:				
	Read	02	CC,SLI	Address of Data	Length
	TIC	08	—	Write CCW	—
10	Write	01	CC,SLI	XCNOVTXT	4

[3] WCTL is used when this host is initiating a write operation. SCB is used when this host is responding to an attention (because the other host has data that it wants this host to read).

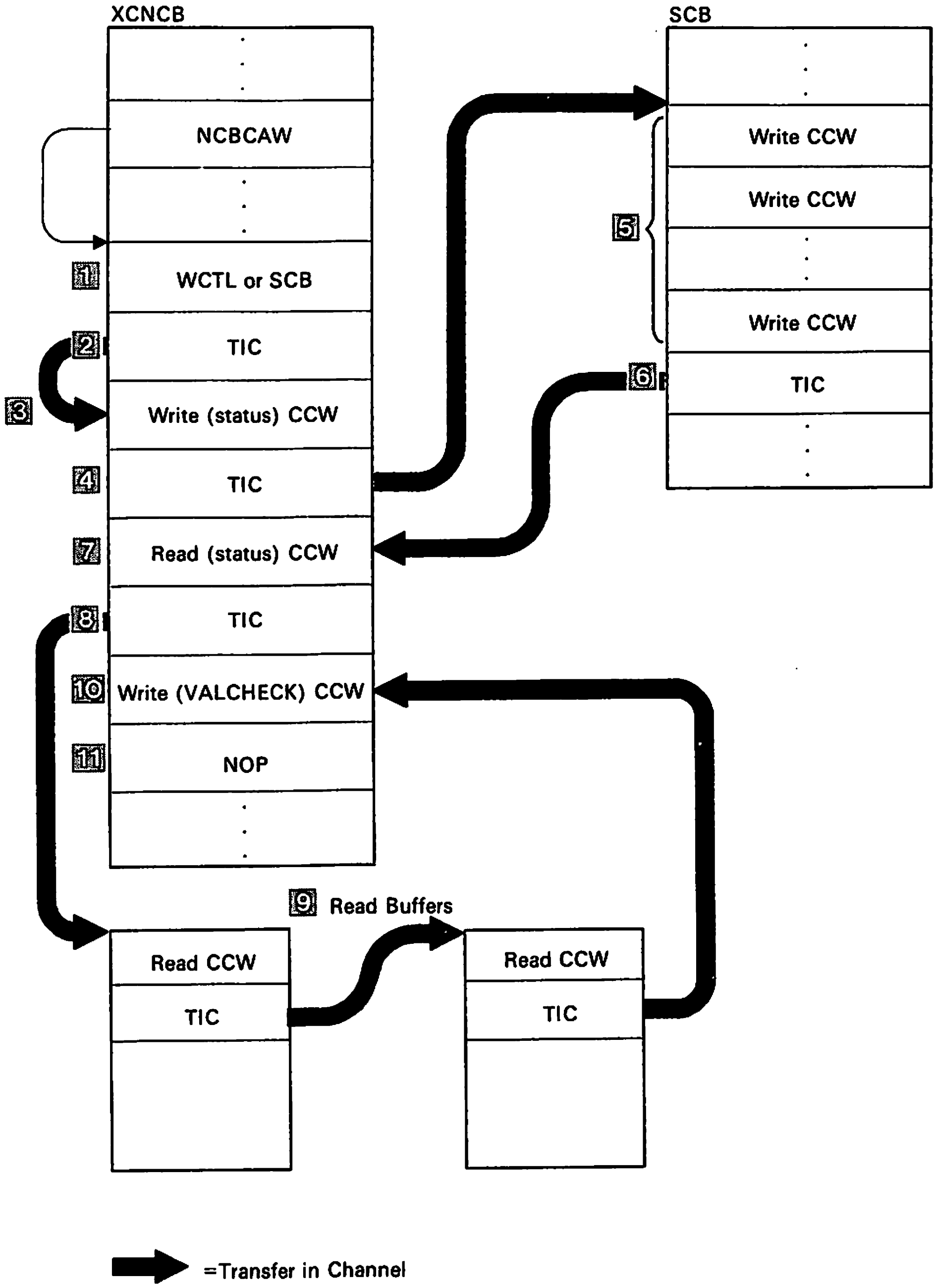


Figure 84. Data Areas Used for Normal Data Transfer (X-Side) in Pre-V3R2

Normal Data Transfer (Y-Side) in Pre-V3R2

Figure 85 on page 523 shows the data areas associated with the following normal data transfer channel program.

	CCW	Command Code (hex)	Flags	Address	Count
1	WCTL[4] or SCB	14/ 17	CC,SLI	Zero	1
2	TIC	08	—	Read CCW	—
3	Read	02	CC,SLI	XCNICTL	8
4	TIC	08	—	Read CCW	—
	Read	02	CC,SLI	Address of Data	Length
	TIC	08	—	Read CCW	—
5	: :				
	Read	02	CC,SLI	Address of Data	Length
	TIC	08	—	Write CCW	—
6	Write	01	CC,SLI	XCNOCTL	8
7	TIC	08	—	Write CCW	—
	Write : :	01	CC,SLI, (CD)	Buffer	Length
8	Write : :	01	CC,SLI, (CD)	Buffer	Length
	Write	01	CC,SLI	Buffer	Length
9	TIC	08	—	Read CCW	—
10	Read	02	CC,SLI	XCNIVTXT	4

[4] WCTL is used when this host is initiating a write operation. SCB is used when this host is responding to an attention (because the other host has data that it wants this host to read).

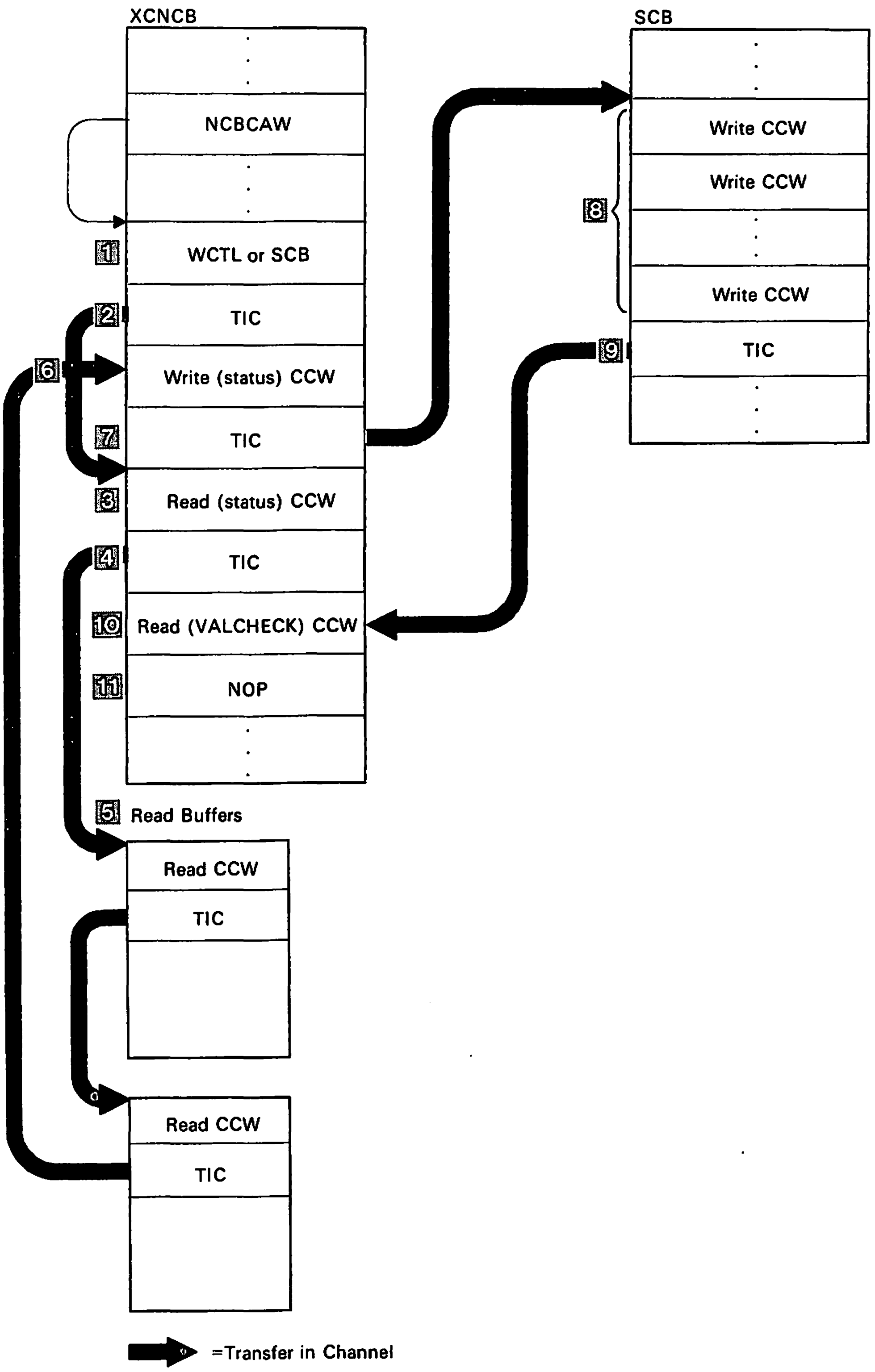


Figure 85. Data Areas Used for Normal Data Transfer (Y-Side) in Pre-V3R2

Normal Data Transfer (X-Side) for V3R2

Figure 86 shows the data areas associated with the X-side of a normal data transfer channel program in V3R2 processing.

	CCW	Command Code (hex)	Flags	Address	Count
1	WCTL[5] or SCB	17/ 14	CC,SLI	Zero	1
2	Write	01	CC,SLI	YCNWRIDA	Number of bytes to transfer
3	Read	02	SLI	YCNRDIDA	Total length of Read Buffer

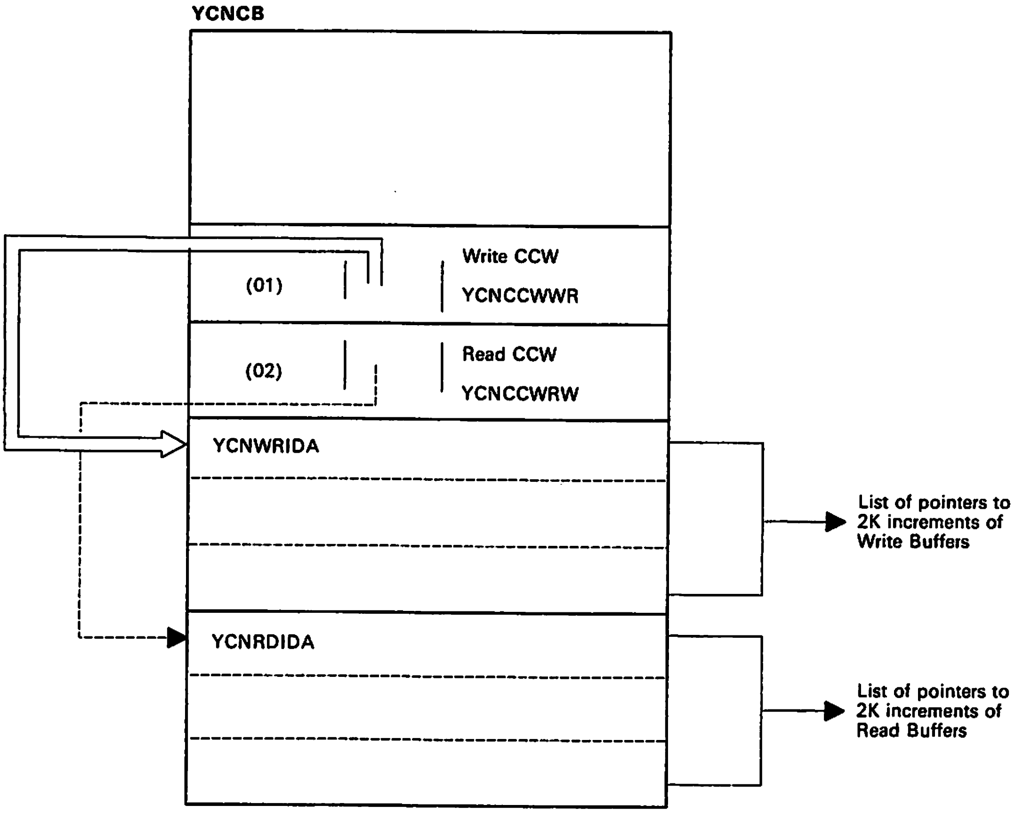

YCNCB
(01)
Write CCW
YCNCCWWR
(02)
Read CCW
YCNCCWRW
YCNWRIDA
List of pointers to 2K increments of Write Buffers
YCNRDIDA
List of pointers to 2K increments of Read Buffers

Figure 86. Data Areas Used for Normal Data Transfer (X-Side) in V3R2

[5] WCTL is used when this host is initiating a write operation. SCB is used when this host is responding to an attention (because the other host has data that it wants this host to read).

Normal Data Transfer (Y-Side) in V3R2

Figure 87 shows the data areas associated with the following normal data transfer channel program in V3R2

	CCW	Command Code (hex)	Flags	Address	Count
1	WCTL[6] or SCB	17/ 14	CC,SLI	Zero	1
2	Read	02	CC,SLI	YCNRDIDA	Total length of Read Buffer
3	Write	01	SLI	YCNWRIDA	Number of bytes to transfer

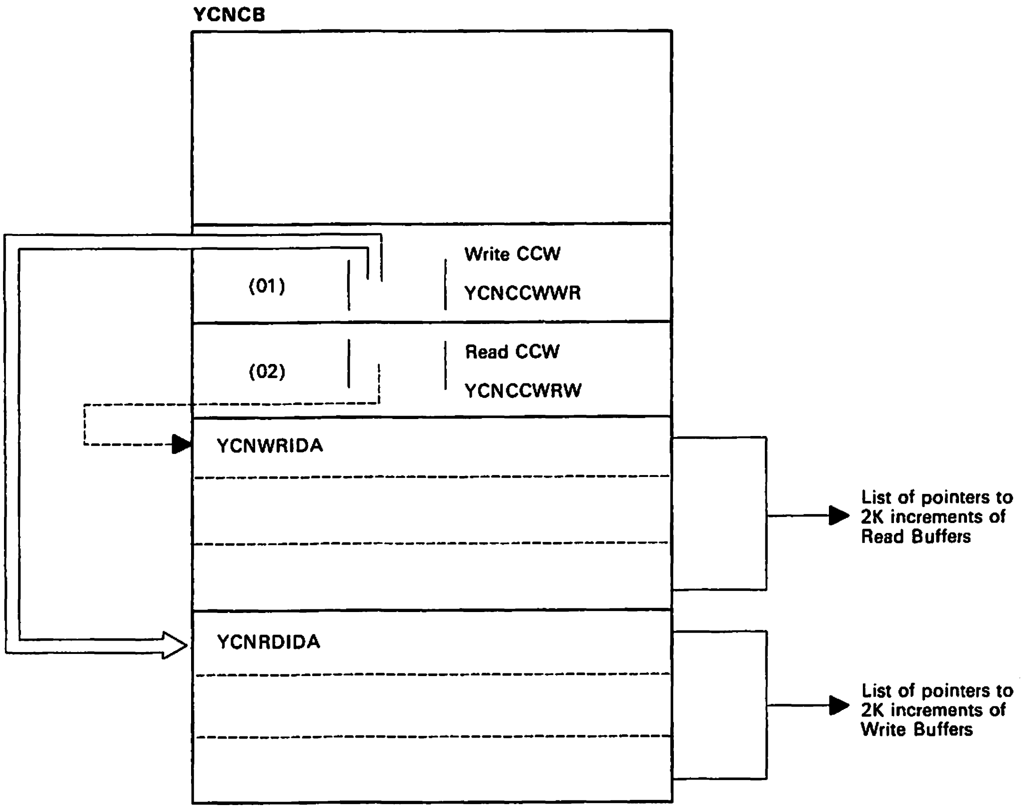

YCNCB
(01)
Write CCW
YCNCCWWR
(02)
Read CCW
YCNCCWRW
YCNWRIDA
List of pointers to 2K increments of Read Buffers
YCNRDIDA
List of pointers to 2K increments of Write Buffers

Figure 87. Data Areas Used for Normal Data Transfer (Y-Side) in V3R2

[6] WCTL is used when this host is initiating a write operation. SCB is used when this host is responding to an attention (because the other host has data that it wants this host to read).

Channel Programs for Channel-Attached Non-SNA 3270 Devices

The publications for the non-SNA 3270 devices contain diagnostic procedures. For more information, see the *3270 Component Description.*

The LDNCB represents local devices and contains addresses and CCWS needed for channel programs. Figure 88 shows the LDNCB and the location in storage of various CCWS.

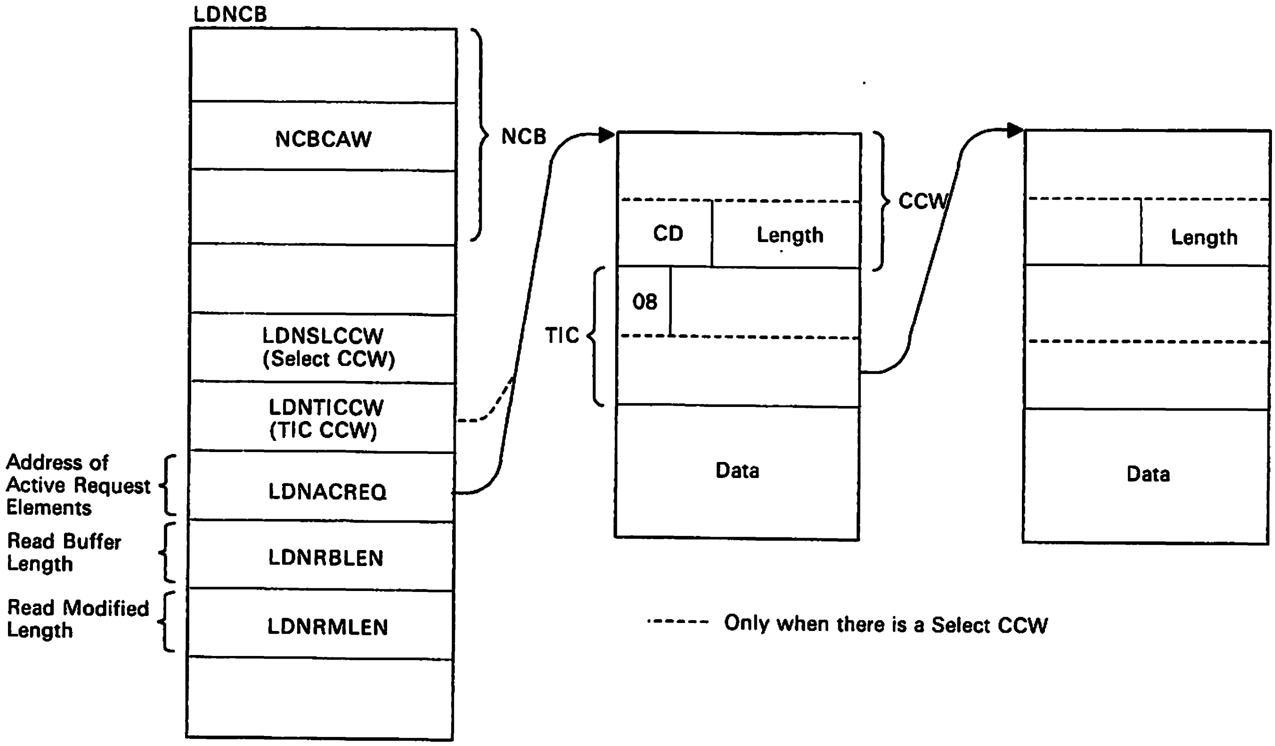


Figure 88. Data Areas Used by a Channel Program for Channel-Attached Non-SNA Devices

Channel Command Words

The following channel commands are used by VTAM to send data to and to receive data from channel-attached non-SNA 3270 terminals.

Command Code (hex)	Command	Description
01	Write	Transfers data from storage in the host processor to the I/O device. Modifies existing buffer data.
02	Read or Read Buffer	Transfers the entire buffer contents from the I/O device to storage in the host processor.
05	Erase/Write	Clears the device buffer before starting the Write.
06	Read Modified	Transfers data (that was modified since the last Read) from the I/O device to storage in the host processor.
0B	Select	Transfers data from the I/O device buffer to the controller buffer. Separates the device-to-controller unit buffer transfer from the execution command.
0D	Erase/Write Alternate	Same as Erase/Write, except that it allows for a larger buffer size required for some devices.
1B	Select Read Buffer	Same as Select, except that it is only used for Read Buffer channel programs.
4B	Select Write	Same as Select, except that it is only used for Write channel programs.
0F	Erase All Unprotected	Clears all unprotected buffers
11	Write Structured Field	Writes a structured field

Channel Programs

The following tables show the CCWS used in various channel programs. The order of execution is the same as the order in which they appear.

Write Data Channel Program

	CCW	Command Code (hex)	Flags	Address	Count
	Select	0B or 4B[7]	CC,SLI	Zero	1
	TIC	08	—	Write CCW	—
As many as needed	Write[8]	01	CD,SLI	Output Area	Length

[7] If the UCB indicates that the device will accept the Select Write, code X'4B', it is used until a command reject is received. Then Select, code X'0B', is used.

[8] The first Write CCW is pointed to by LDNACREQ. The last Write CCW is not Command Code X'1B'.

Read Modified Channel Program

	CCW	Command Code (hex)	Flags	Address	Count
	Select	0B	CC,SLI	Zero	1
	TIC	08	—	Read Modified CCW	—
As many as needed	Read Modified[9]	06	CD,SLI	Input Area	Length
	TIC	08	—	Ready Modified CCW	Length
	Read Modified[10]	06	CD,SLI	Input Area	Length
	Read Modified (Skip)	06	SKIP, SLI	Zero	X'7FFF'

[9] The first Read Modified CCW is pointed to by LDNACREQ.

[10] The number of bytes transmitted in the previous Read Modified CCW is stored in the LDNRMLEN field in the LDNCB. For this Read Modified, enough buffers are allocated to hold LDNRMLEN bytes of data. If this is not enough buffers for the device to send all of the pending data, the channel program will end on the Read Skip CCW. The Read Skip CCW reads all of the pending data but does not transmit any of it. The new length is stored in LDNRMLEN. (The length is computed by subtracting the residual byte count in the CSW from X'7FFF'.) Then, the channel program is executed again with enough buffers allocated to read and transmit all of the pending data.

Read Buffer Channel Program

	CCW	Command Code (hex)	Flags	Address	Count
	Select	0B or 1B[11]	CC,SLI	Zero	1
	TIC	08	—	Read Buffer CCW	—
As many as needed	Read Buffer[12]	02	CD,SLI	Input Area	Length
	TIC	08	—	Read Buffer CCW	—
	Read Buffer[13]	02	CD,SLI	Input Area	Length
	Read	02	SKIP, SLI	Zero	X'7FFF'
	Buffer (Skip)				

[11] If the UCB indicates that the device will accept the Select Read, code X'1B', it is used until a command reject is received. Then Select, code X'0B', is used.

[12] The first Read CCW is pointed to by LDNACREG.

[13] The number of bytes transmitted in the previous Read Buffer CCW is stored in the LDNRBLEN field in the LDNCB. For this Read Buffer, enough buffers are allocated to hold LDNRBLEN bytes of data. If this is not enough buffers for the device to send all of the pending data, the channel program will end on the Read Skip CCW. The Read Skip CCW reads all of the pending data but does not transmit any of it. The new length is stored in LDNRBLEN. (The length is computed by subtracting the residual byte count in the CSW from X'7FFF'.) Then the channel program is executed again with enough buffers allocated to read and transmit all of the pending data.

Erase/Write Channel Program

	CCW	Command Code (hex)	Flags	Address	Count
As many as needed	Erase/ Write[14]	05	CD,SLI	Output Area	Length
	TIC	08	—	Erase/Write CCW	—
	Erase/ Write	05	SLI	Output Area	Length

Erase/Write Alternate Channel Program

	CCW	Command Code (hex)	Flags	Address	Count
As many as needed	Erase/ Write[15]	0D	CD,SLI	Output Area	Length
	Alternate				
	TIC	08	—	Erase/Write Alternate CCW	—
	Erase/ Write	0D	SLI	Output Area	Length

Erase All Unprotected Channel Program

	CCW	Command Code (hex)	Flags	Address	Count
	Erase All[16] Unprotected	0F	SLI	Zero	1

[14] LDNACREQ has the address of the first Erase/Write CCW.

[15] LDNACREQ contains the address of the first Erase/Write Alternate CCW.

[16] LDNACREQ contains the address of the Erase All Unprotected CCW.

VM VSE BSC Communication Adapter Line Channel Programs

The BSCLB represents a BSC line attached through a BSC communication adapter and contains addresses and CCWS needed for channel programs. See the *IBM 9370 Information System: Telecommunications Subsystem Description and Reference* or the *IBM 4361 Processor Communication Adapter* for a detailed description of channel programs, the interface between VTAM and the telecommunications subsystem or communication adapter, and the sense codes returned from a telecommunications subsystem or communication adapter.

VM VSE Channel Commands

The following channel commands are used by VTAM to write data to or read data from a BSC terminal attached through a communication adapter line. Of the channel commands supported by the BSC communication adapter, VTAM only uses those required for master-station multipoint operations.

Command Code (hex)	Command	Description
01	Write	Transfers data from storage in the host processor to the BSC device.
02	Read	Transfers data from the control unit to storage in the host processor.
03	No-Op	Used to stop channel program execution (for example, General Poll). It also causes the channel to respond with a channel end and device end.
04	Sense	Transfers two bytes of sense data to VTAM.
08	TIC (Transfer in Channel)	Requests a transfer in the channel program. It causes the channel program to execute a command other than the next sequential command in the channel program.
09	Poll	Starts the autopoll of the poll list at the entry specified.
23	Set Mode	Used to condition a line in an Enable Write channel program. It sets the desired operating environment.
27	Enable	Sets the BSC line to the enabled state. It is the last command in an Enable Write channel program.
2F	Disable	Sets the line to the disabled state. It is a part of the Enable Write and Disable channel programs.

VM VSE I/O Scheduling Overview of BSC Lines

During activation of a BSC line, a polling list is constructed containing entries for all of the BSC cluster controllers that are defined for the line. Initially, all the poll list entries are inactive and no I/O is started.

At the time a Contact RU is processed for the cluster controller, activation processing activates the appropriate polling list entry and, if it is the first cluster on the line to be activated, causes the General Poll channel program to be started. The General Poll channel program is designed to run continuously, without interruption , as long as negative responses to poll (EOTS) are being received. When an interruption occurs, VTAM analyzes the non-negative response to determine the next action.

Normally, it would be a Continue Read channel program to acknowledge the previous response and to solicit the next transmission. The Continue Read channel program is issued repeatedly until an EOT is received that indicates that the cluster controller has completed its data transmission.

Upon receipt of an EOT, the station pending traffic queues are checked for pending output to any active terminal on the line. If any is found, that transmission is scheduled prior to restarting the General Poll. Any pending activation or deactivation of polling list entries is done at this time as well.

If output I/O, activation, or deactivation occurs while the General Poll is active, but prior to an interruption, the General Poll is stopped to allow this processing to proceed. This is accomplished in a controlled manner by changing the TIC (the sixth command) in the General Poll channel programs to a No-op. Changing the TIC to No-op causes the channel program to end when it has polled all entries in the polling list.

Following completion of any scheduled operations, the General Poll is restarted at the first entry in the polling list. If General Poll is restarted after the completion of a General Poll, it is restarted at the next entry from the one that caused the General Poll to be completed. This ensures an even distribution of poll operations among the cluster controllers on a line.

The following channel programs are used in the I/O sequence that sends data to or receives data from a BSC terminal.

Note: Except for the Enable Write and Disable Write channel programs, all of the channel programs (or portions of them) are in the BSCLB (BSC line control block).

VM VSE Channel Programs

The following tables illustrate the CCWS used in various channel programs.

VM VSE Enable Write Channel Program

This channel program is started for ACTLINK processing in order to condition and enable the line.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Disable	2F	CC,SLI	—	1
(2)	Set Mode	23	CC,SLI	Control Information	1
(3)	Enable	27	SLI	—	1

VM VSE Disable Write Channel Program

This program is started for DACTLINK processing in order to disable a line.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Disable	2F	SLI	—	1

VM VSE General Poll (Initial Read) Channel Program

To solicit data from BSC terminals, this program is constructed to allow polling to begin with any entry in the poll list and to continue wrapping through the list until:

- A non-negative response is received from a terminal.
- The channel program is stopped for the scheduling of output.
- The operations terminate.

The program also is updated dynamically to reflect a changed poll list entry, poll list length, or read data area. When this program is initiated, the BSCOCFSM, BSCPCFSM, and BSCCCFSM fields of the BSCLB are set to X'01'.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Write	01	CC,SLI	EOT	1
(2)	Poll	09	CC,SLI	Poll list entry	n + 1
(3)	TIC	08	—	Command (5)	1
(4)	TIC	08	—	Command (7)	1
(5)	Poll	09	CC,SLI	First entry poll list	m + 1
(6)	TIC	08	—	Command (5)	1
(7)	Read	02	SLI	I/O buffer	Length

VM VSE Continue Read Channel Program

This program is used after either an Initial Read channel program or another Continue Read channel program. It sends a positive acknowledgment to previously received data and to read in additional data. When this program is executing, the BSCOCFSM, BSCCPFSM, and BSCCCFSM fields of the BSCLB are set to X'02'.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Write	01	CC,SLI	ACKO or ACK1	2
(2)	Read	02	SLI	I/O buffer	Length

VM VSE Repeat Read Channel Program

During error recovery procedure (ERP) for BSC lines, this channel program is executed to request that the terminal retransmit the previous text block in which a "block check" error was detected. At this point, the BSCOCFSM, BSCPCFSM, and BSCCCFSM fields of this BSCLB are set to X'0B'.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Write	01	CC,SLI	NAK	1
(2)	Read	02	SLI	I/O Buffer	Length

VM VSE Interrupt Write Channel Program

When the inbound data flow must be stopped immediately, the Interrupt Write channel program is used to stop the General Poll (Initial Read) channel program. In order to prevent data loss, the channel program is executed following a text block that ends with ETX. The BSCCCFSM and BSCPCFSM fields of the BSCLB are set to X'0C'.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Write	01	CC,SLI	RVI	2
(2)	Read	02	SLI	Response area	2

VM VSE Reset Write Channel Program

The Reset Write channel program is used to write an EOT to the terminal and to return the line to control mode. When this program is executing, the BSCCCFSM AND BSCPCFSM fields of the BSCLB are set to X'07'.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Write	01	SLI	EOT	1

VM VSE

Status Read Channel Program

The Status Read channel program is executed when a terminal indicates that it has pending BSC sense/status information. The program specifically polls the terminal and reads the BSC sense/status information only. The BSCCCFSM AND BSCPCFSM fields of the related BSCLB are set to X'0A'.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Write	01	CC,SLI	EOT	1
(2)	Write	01	CC,SLI	Specific poll characters	5
(3)	Read	02	SLI	I/O buffer	Length

VM VSE

Inquiry Read Channel Program

The Inquiry Read channel program clears a communication adapter line buffer. Any data read is discarded because of the skip flag in the program's Read CCW. The BSCCCFSM and BSCPCFSM fields of the BSCLB are set to X'0D'.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Read	02	SLI,SKIP	Response area	4095

VM VSE

Sense Channel Program

The Sense channel program is issued when VTAM receives unit-check status from the communication adapter.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Sense	04	SLI	Sense area	2

VM VSE Initial Write (Conversational) Channel Program

A variation of the Initial Write channel program, the Initial Write (Conversational) channel program performs a required conversational function. This function allows a terminal to respond to a text block with another text block, rather than with a response. The channel program must have an appropriate length buffer ready to receive the text block. The BSCOCFSM, BSCCCFSM, and BSCPCFSM fields of the BSCLB are set to X'08' when an Initial Write (Conversational) channel program is executing. They are set to X'04' when an Initial Write channel program is executing.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Write	01	CC,SLI	EOT	1
(2)	Write	01	CC,SLI	Selecting characters	5
(3)	Read	02	SLI	Response area	2
(4a)	Write	01	CD,SLI	STX ESC characters	2
(4b)	TIC	08	—	Command (4c) in I/O buffers	1
(4c)	Write	01	CD,SLI	Data in I/O buffer	Length
(4d)	TIC	08	—	Command (4e)	1
(4e)	Write	01	CC,SLI	ETX character	1
(5a)	Read	02	SLI	Response area	2
(5b)	Read	02	SLI	I/O buffers	Length

VM VSE Inquiry Write (Conversational) Channel Program

The Inquiry Write channel program is issued during ERP (error recovery procedure) in order to request the terminal to retransmit its last response. The BSCCCFSM and BSCPCFSM fields of the BSCLB are set to X'06'.

No.	CCW	Command Code (hex)	Flags	Address	Count
(1)	Write	01	CC,SLI	ENQ	1
(2a)	Read	02	SLI	Response area	2
(2b)	Read	02	SLI	I/O buffer	Length

VM VSE Channel Program Finite State Machines for BSC Lines

The finite state machine (FSM) represents what is happening on the BSC line. The following table shows the finite state machines and what they mean.

FSM	State	Explanation
(1) GP	General Poll	The General Poll channel program is executing.
(2) CR	Continue Read	The Continue Read channel program is executing.
(3) CM	Control Mode	The BSC line is in control mode. No channel program is executing for the line.
(4) IW1	Initial Write 1	The first part of an Initial Write channel program is executing.
(5) IW2	Initial Write 2	The second part of an Initial Write channel program is executing.
(6) INQ	Inquiry Write	An Inquiry Write channel program is executing.
(7) EOT	End of Transmission	An EOT has been sent, or an EOT is expected.
(8) IC1	Initial Write Conversational 1	The first part of an Initial Write Conversational channel program is executing.
(9) IC2	Initial Write Conversational 2	The second part of an Initial Write Conversational channel program is executing.
(10) STR	Status Read	A Status Read channel program is executing.
(11) RR	Repeat Read	A Repeat Read channel program is executing.
(12) IWT	Interrupt Write	An Interrupt Write Channel program is executing.
(13) IQR	Inquiry Read	An Inquiry Read channel program is executing.
(14) IQW	Inquiry Write Conversational	An Inquiry Write Conversational channel program is executing.

Figure 89 on page 540 shows the finite state machines representing activity on a BSC line.

At the top of each column is the state that the line is in. The state is one of the following:

- The channel program that is executing for the line
- The condition of the line
- An end of transmission (EOT).

Above the arrow are the stimuli (flags, responses) that cause a state change. At the end of the arrow is the new state or a number that refers to a new state (the number at the top of a column). Below the arrow is the action that VTAM takes as a result of the state change: the action may be to send a response, to issue an RU such as a station INOP or an exception request, or to set up a procedure to delay processing until a temporary condition has ended.

The following abbreviations are used in the diagrams:

ACK0	Positive acknowledgement 0; alternates with ACK1
ACK1	Positive acknowledgement 1; alternates with ACK0
ENQ	Enquiry; requests a reply
EOT	End of transmission
ETB	End of test block
ETX	End of text
NAK	Negative acknowledgment
RVI	Reverse interrupt
SOH	Start of heading
STX	Start of text
TRM	Test request message

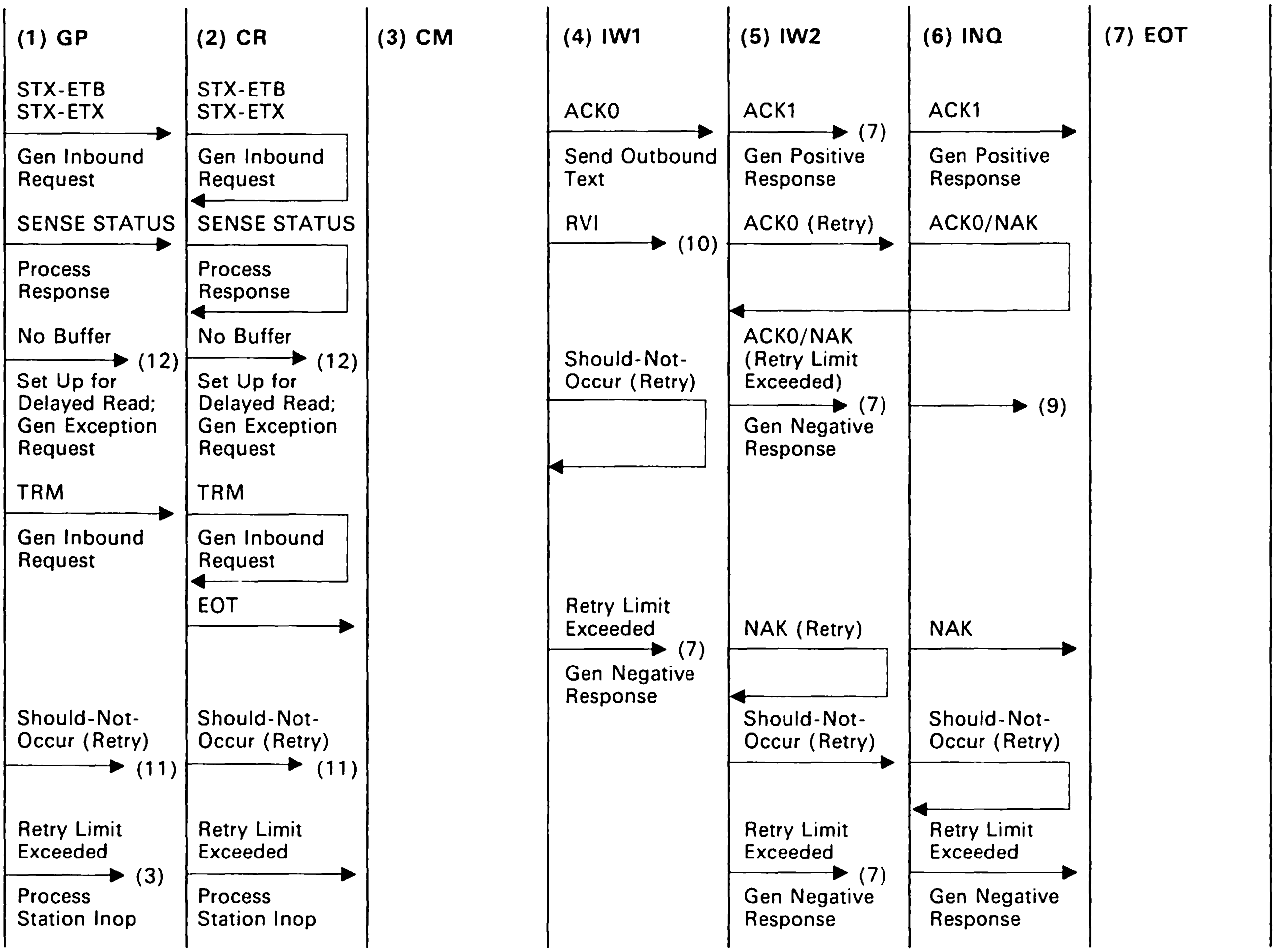
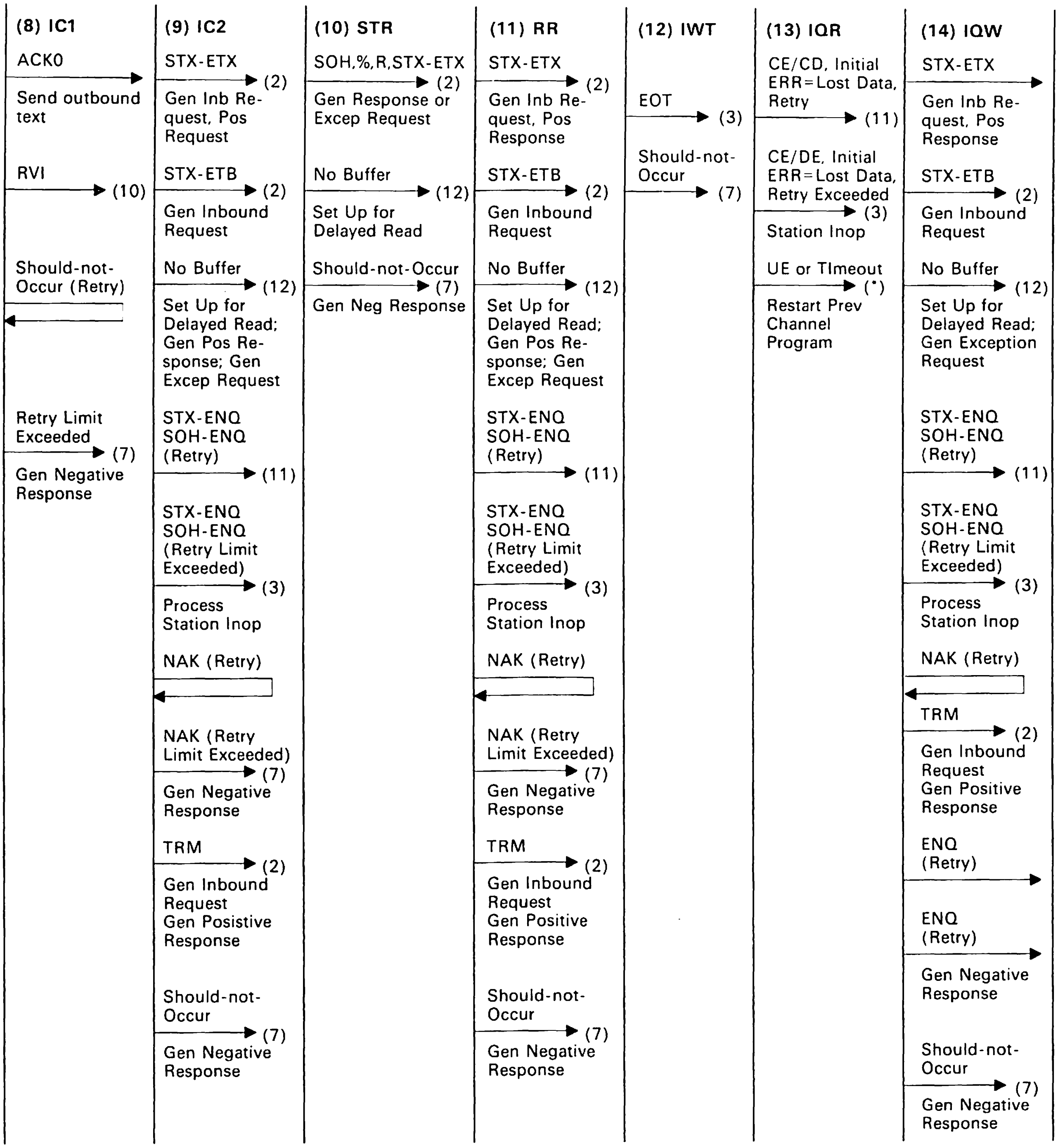


Figure 89 (Part 1 of 2). Finite State Machines for BSC Lines



(8) IC1

ACKO →
Send outbound text

RVI → (10)

Should-not-Occur (Retry) ←

Retry Limit Exceeded → (7)
Gen Negative Response

(9) IC2

STX-ETX → (2)
Gen Inb Request, Pos Request

STX-ETB → (2)
Gen Inbound Request

No Buffer → (12)
Set Up for Delayed Read; Gen Pos Response; Gen Excep Request

STX-ENQ
SOH-ENQ
(Retry) → (11)

STX-ENQ
SOH-ENQ
(Retry Limit Exceeded) → (3)
Process Station Inop

NAK (Retry) ←

NAK (Retry Limit Exceeded) → (7)
Gen Negative Response

TRM → (2)
Gen Inbound Request
Gen Posistive Response

Should-not-Occur → (7)
Gen Negative Response

(10) STR

SOH,%,R,STX-ETX → (2)
Gen Response or Excep Request

No Buffer → (12)
Set Up for Delayed Read

Should-not-Occur → (7)
Gen Neg Response

(11) RR

STX-ETX → (2)
Gen Inb Request, Pos Response

STX-ETB → (2)
Gen Inbound Request

No Buffer → (12)
Set Up for Delayed Read; Gen Pos Response; Gen Excep Request

STX-ENQ
SOH-ENQ
(Retry) → (11)

STX-ENQ
SOH-ENQ
(Retry Limit Exceeded) → (3)
Process Station Inop

NAK (Retry) ←

NAK (Retry Limit Exceeded) → (7)
Gen Negative Response

TRM → (2)
Gen Inbound Request
Gen Positive Response

Should-not-Occur → (7)
Gen Negative Response

(12) IWT

EOT → (3)

Should-not-Occur → (7)

(13) IQR

CE/CD, Initial ERR=Lost Data, Retry → (11)

CE/DE, Initial ERR=Lost Data, Retry Exceeded → (3)
Station Inop

UE or TImeout → (*)
Restart Prev Channel Program

(14) IQW

STX-ETX →
Gen Inb Request, Pos Response

STX-ETB → (2)
Gen Inbound Request

No Buffer → (12)
Set Up for Delayed Read; Gen Exception Request

STX-ENQ
SOH-ENQ
(Retry) → (11)

STX-ENQ
SOH-ENQ
(Retry Limit Exceeded) → (3)
Process Station Inop

NAK (Retry) ←

TRM → (2)
Gen Inbound Request
Gen Positive Response

ENQ
(Retry) →

ENQ
(Retry) →
Gen Negative Response

Should-not-Occur → (7)
Gen Negative Response

Figure 89 (Part 2 of 2). VM VSE Finite State Machines for BSC Lines

VM VSE Channel Programs for an SDLC Communication Adapter Line

The HALCB represents an SDLC line attached through an SDLC communication adapter and contains addresses and CCWS needed for channel programs. See the *IBM 9370 Information System: Telecommunications Subsystem Description and Reference* or the *IBM 4361 Processor Communication Adapter* for a detailed description of channel programs, the interface between VTAM and the telecommunications subsystem or communication adapter, and the sense codes returned from a telecommunications subsystem or communication adapter.

VM VSE Channel Commands

The following channel commands are used in the channel programs to communicate with SNA devices on SDLC lines or with the SDLC lines themselves.

Command Code (hex)	Command	Description
01	Write	Writes an SDLC command located in the TSCODATA field of the TSCB.
02	Read	Reads an SDLC command or control data located at TSCIDATA.
03	No-op	Used to stop channel program execution.
05	Write PIU	Points to TSCILNG, a two-byte prefix to TSCPIU. TSCILNG contains the total length of the data in the PIU. This length includes the TH, RH, RU, and TSCILNG. Data in any continuation buffers for the Write PIU channel program will start at TSCPIU as usual.
06	Read PIU	Points to TSCILNG, a two-byte prefix of TSCPIU. The total length of the data is returned in TSCILNG. Data in any continuation buffers of the Read PIU channel program begins at TSCILNG.
08	TIC (Transfer in Channel)	Requests a transfer in the channel program. It causes the channel program to execute a command other than the next sequential command in the channel program.
09	Autopoll	Points to a polling list of SCBs. The length is the length of the polling list. This command solicits input from active stations, contact polls those stations being activated, and ignores the inactive stations.
0B	Control SCB (Control SPL)	Transfers 12 bytes of SCB information from VTAM to the communication adapter.

Command Code (hex)	Command	Description
0F	Poll	Transfers a final RR or RNR after a chain of Write P commands. When the Poll command is executed, the line is turned to inbound.
14	Sense SCB (Sense SPL)	Retrieves the current SCB from the communication adapter after a line operation completes. The current SCB indicates which station responded.
23	Set Mode	Transfers up to 8 bytes of setmode data to the communication adapter to change the status of a line.
27	Enable	Makes the line operational.
29	Dial	Transfers dialing data to the automatic calling unit (ACU).
2F	Disable	Makes the line non-operational.
E4	Sense I/O	Transfers up to 7 bytes of line configuration information from the communication adapter.

VM VSE I/O Scheduling Overview of SDLC Lines

During activation of an SDLC line, a station polling list (SPL) is constructed containing entries for all the physical units that are defined for the line. Initially, all the station entries are inactive. When any station has to be contacted and it is the first station on the line to be activated, the Autopoll channel program is started to do a Contact Poll. When a station sends in a nonsequenced acknowledgment (NSA), data polling begins.

The Autopoll channel program is designed to run continuously, without interruption, as long as Receive Ready (RR) or Receive Not Ready (RNR) responses flow. The Autopoll channel program is interrupted when SDLC supervisor frames or information frames are received, when data is to be transmitted, or when certain PU services I/O has to be processed. In order to send the Write data, the Autopoll channel program is interrupted by changing the third CCW, which is a No-op, to a TIC CCW pointing to the Write PIU channel program.

The Write buffers to be transmitted are pointed to by a pending queue (PUTPENDQ), or an error hold queue (PUTEHOLD), in the PUT entry. The SDLC station scheduler scans every PUT entry, starting from the current entry indicated in HALCUSCB, and passes the Write buffers to TSC's SDLC channel program build routine.

If there is no Write data to be sent to any station, the Autopoll is restarted. The communication adapter begins polling from the next station in the station polling list unless the polling index is changed by the Set Mode CCW. One pass of the Autopoll channel program is restarted if a Read Error CCW occurs on one of the stations. This is accomplished by changing the TIC CCW after the Autopoll CCW to a No-op.

VM VSE I/O Scheduling for PU Services

All PU services requests are processed by the SDLC connection manager outbound (ISTTSCHO) on the PU services PAB in the HALCB (NCBPUPAB). Before initiating I/O for PU services, a switch is set to wait for current I/O (including the Autopoll channel program) to finish. This is done by setting the state of PU services I/O (HALPUSIO) to pending, which signals the channel end not to restart application program I/O or Autopoll after the completion of the current I/O. The continuous Autopoll channel program is broken by changing the third CCW, which is a No-op, to a TIC CCW pointing to another No-op.

When the current I/O has been completed, the state of PU services I/O is set to "halted," and the connection manager processes the request and schedules I/O. The state of PU services I/O changes to "active." The state changes to "inactive" after the completion of I/O.

The connection manager builds its channel program in a buffer (at TSCPCCW).

VM VSE I/O Scheduling for the Link Level 2 Test

A TESTMODE request is initiated and completed by the SDLC connection manager outbound routine (ISTTSCHO) on the PU services PAB in the HALCB (NCBPUPAB). The link level 2 test is executed by channel end and the link scheduler.

If the TESTMODE request is to start a link level 2 test, the SDLC connection manager outbound routine sets the station to "test mode," initializes fields in the PUT entry, obtains storage for the write and read buffers, moves the specified data into the write buffer, and starts the Write Test channel program. The Write Test channel program is repeated the specified number of times, or until the test is canceled by the operator.

If the TESTMODE request is to cancel a current test, a RECTR request is built and sent to network operator services to that the operator message with the test results can be built.

VM VSE Channel Programs

The following channel programs are used in I/O sequences that send data to or receive data from devices on nonswitched and switched SDLC lines.

Note: The Autopoll channel program (a part of the Write PIU channel program) and the Sense SCB channel program are in the HALCB (HALCHPGM). The other channel programs are formatted in I/O buffers.

VM VSE Autopoll Channel Program

The Autopoll channel program is used to solicit data from active stations (data poll) and to check whether inactive stations are ready to be activated (contact poll). When the Autopoll CCW is executed, the communication adapter steps through the SDLC polling list table (SPL table; SCB table to the communication adapter) and sends either a data poll or a contact poll until one of the following occurs:

- An active station responds with data.
- An inactive station responds with a nonsequenced acknowledgment (NSA).
- An error occurs.

- VTAM changes the TIC address of point to a Write PIU channel program or to a No-op CCW to stop the Autopoll channel program.

If all responses to the polls are negative, the final TIC chains to the second CCW (the No-op) to start another poll pass through the SPL table.

CCW	Command Code (hex)	Flags	Description	Remarks
Set Mode	23	CC,SLI	Sets the desired operating mode.	Optional
No-Op	03	CC,SLI	Required as the last in a chain.	
TIC	08	—	When polling, transfers control to the Autopoll CCW immediately following the TIC. For a Write channel program, either transfers to a string of Write PIUs or to a No-Op CCW to stop Autopoll. (See the Write PIU channel program.)	
Autopoll	09	CC,SLI	Starts polling, points to the SPL table, sends Receive Ready (RR) for data poll or Set Normal Response Mode (SNRM) for contact poll depending upon the flag settings in the SPL.	
TIC	08	—	Transfers back to the second command (No-Op) to continue polling, is changed to No-Op if continuous Autopoll is not required (for example, when an error occurs on a Read CCW), and transfers to a Write Test channel program when LL2 test is required.	
TIC	08	—	Is executed when status modified is given to the channel by the communication adapter and transfers to Read PIU or Sense SCB channel program.	

VM VSE Read PIU Channel Program

The Read PIU channel program reads data sent to VTAM by another SDLC station. Normally, the channel program is started when an Autopoll TIC points to the first Read PIU buffer. The Read PIU CCWs are built in the buffers that will receive the data and are chained together by TICs. The number of buffers in the chain is determined by the values for the INBFRS operand of the line's definition statement.

When VTAM is a secondary link station or when the line is in an inbound direction, the communication adapter starts the Read PIU channel program with the Control SCB CCW that is located in the HALCB.

When the transmission is complete or when the communication adapter detects error information, the communication adapter skips over any remaining Read PIU CCWs to a Sense CCW by which it can report error information.

The Read PIU channel program is pointed to by the Autopoll channel program unless it starts with a control SCB CCW.

CCW	Command Code (hex)	Flags	Description	Remarks
Control SCB	0B	CC,SLI	Used only when VTAM is a secondary link station, the line is in an inbound direction, and the Autopoll channel program is not used to point to the first Read CCW in the HALCB.	Optional
TIC	08	—	Points to the first Read PIU in the HALCB.	Optional
Read PIU	06	CC,SLI	Points to the input area.	
TIC	08	—	Points to the next Read PIU CCW.	
.				
.				
.				
Read PIU	06	CC,SLI	Points to the input area.	
TIC	08	—	Points to the Sense SCB CCW that is in the HALCB.	

VM VSE Write PIU Channel Program

The Write PIU channel program sends data to another SDLC station. Normally, the channel program is started when an Autopoll TIC points to the first Write PIU buffer. The Write PIU CCWs are built in the buffers that contain the outbound data. After the data in the last buffer is sent to the station, the program executes a Poll CCW that sends RR (Receive Ready) so that the station may acknowledge receipt of the data and, optionally, may send any waiting data. The final TIC points to the Read PIU channel program in order to receive possible inbound data and to verify that the outbound data was received.

The Write PIU channel program is pointed to by the Autopoll channel program in the HALCB.

CCW	Command Code (hex)	Flags	Description
Control SCB	0B	CC,SLI	Points to an SPL entry.
TIC	08	—	Points to the first Write PIU in the HALCB.
Write PIU	05	CC	Points to the buffer containing the PIU to be sent. Even when the data is spanned in more than one buffer, the CCWs are command chained.
TIC	08	—	Points to a Write PIU CCW if there are more PIUs, or points to a Poll CCW if the last PIU has been sent.
.			
.			
.			
Poll	0F	CC,SLI	Starts polling at the next entry in the SPL.
TIC	08	—	Points to a Read PIU or Sense SCB CCW.

VM VSE Write Test Channel Program

The Write Test channel program sends and receives test frames. It is used when a link level 2 test is requested. The Write Test channel program is pointed to by the Autopoll channel program or by NCBCAW in the HALCB.

CCW	Command Code (hex)	Flags	Description
Write	01	CC,SLI	Points to data in the Write buffer.
Read	02	SLI	Points to the Read buffer.

VM VSE Sense SCB Channel Program

The Sense SCB channel program is used to retrieve completion status from the communication adapter. The Sense SCB CCW is the last CCW in the chain for channel programs that perform normal data transfer for the Autopoll channel program, the Read PIU channel program, or the Write SDLC Control channel program.

CCW	Command Code (hex)	Flags	Description
Sense CCB	14	SLI	Points to an input area (HALCUSCB).

VM VSE Sense I/O Channel Program

The Sense I/O channel program requests and receives line configuration information from the communication adapter. This information is used to verify information in the line's definition statement. The verified information is included in the line's OBR record for the line. For this channel program, the HALLFSM field of the HALCB is set to X'01'.

CCW	Command Code (hex)	Flags	Description
Sense I/O	E4	—	Points to an input area used to obtain the line configuration.

VM VSE Enable Line Channel Program

The Enable Line channel program is used when a nonswitched line is activated. When this channel program is being executed, the HALLFSM field in the HALCB is set to X'02'.

CCW	Command Code (hex)	Flags	Description
Disable	2F	CC,SLI	Sets the line to disable state and clears certain pending status conditions.
Set Mode	23	CC,SLI	Tells the communication adapter how VTAM expects to use the line and sets operating values such as timeout values and control indices. Desired operating mode values are contained in the HALSMODE field in the HALCB.
Enable	27	SLI	Causes the communication adapter to make the DCE interface ready. For CCITT V.24 or EIA RS-232C, Enable means activating the Data Terminal Ready (DTR) line and verifying that the response is Data Set Ready (DSR).

VM VSE Answer Channel Program

The Answer channel program is used when a switched line is activated. When a call is received, VTAM checks information in the XID response against values in the corresponding switched line's definition statement. Based on that information, VTAM either accepts or rejects the call. For this channel program, the HALLFSM field in the HALCB is set to X'01'.

CCW	Command Code (hex)	Flags	Description
Disable	2F	CC,SLI	Sets the line to disable state, clears certain pending status conditions, and breaks any outstanding switched connections.
Set Mode	23	CC,SLI	Tells the communication adapter how VTAM expects to use the line and sets operating values such as control indices. The desired operating mode is contained in the HALSMODE field in the HALCB.
Enable	27	CC,SLI	Causes the communication adapter to make the DCE interface ready. This CCW is not completed until an incoming call is accepted, so it may be active for a long time before it completes with the Data Terminal Ready (DTR) sequence.
Write	01	CC,SLI	Points to the area containing all station identifiers and XID commands (TSCODATA).
Read	02	SLI	Points to an input area (TSCIDATA).

VM VSE Dial Channel Program

The Dial channel program is identical to the Answer channel program except that a Dial CCW is used to initiate an outbound call (an Enable CCW is used to receive an inbound call). Execution of the Dial CCW transfer the desired telephone number to the communication adapter which forwards the number, one digit at a time, to an autocall adapter. The HALLFSM field in the HALCB is set to X'02' when this program is active.

CCW	Command Code (hex)	Flags	Description
Disable	2F	CC,SLI	Sets the line to disabled state.
Set Mode	23	CC,SLI	Sets the desired operating mode as specified in HALSMODE.
Dial	29	CC,SLI	Points to an area (TSCPDATA) containing the digits to be dialed.
Write	01	CC,SLI	Points to the area containing all station identifiers and XID commands (TSCODATA).
Read	02	SLI	Points to an input area (TSCIDATA).

VM VSE Write SDLC Control Channel Program

The Write SDLC Control channel program is used when TAM is the secondary station (for example, as in a VTAM-NCP connection). This channel program sends a nonsequenced acknowledgment (NSA) in response to a Set Normal Response Mode (SNRM) from a Contact Poll or in response to a Discontact (DISC) sent during shutdown processing.

CCW	Command Code (hex)	Flags	Description
Set Mode	23	CC,SLI	Sets the desired operating mode.
Write	01	CC,SLI	Points to an output area (TSCODATA) and will be changed to a No-Op CCW when closedown recording is required for the physical unit.
TIC	08	—	Transfers to the Sense SCB CCW that is in the HALCB, causing the communication adapter to wait for some transmission from the primary station.

VM VSE Read SDLC Control Channel Program

Like the Write SDLC Control channel program, the Read SDLC Control channel program is used when VTAM is the secondary station. It waits for the Set Normal Response Mode (SNRM) sent by the primary station on a Contact Poll.

The Read SDLC control program can be pointed to by the Enable Line channel program.

CCW	Command Code (hex)	Flags	Description
Set Mode	23	CC,SLI	Sets the desired operating mode in the HALSMODE field in the HALCB. It can be a Write CCW or No-Op CCW, if required.
Read	02	—	Reads an SDLC command.

VM VSE Disable Line Channel Program

The Disable Line channel program sends a Disable CCW to shutdown the modem interface of the communication adapter.

CCW	Command Code (hex)	Flags	Description
Disable	2F	SLI	Sets the line to disabled state.

VM VSE Primary XID Channel Program

The Primary XID channel program is for a communication adapter-attached type 5 physical unit (XID Exchange).

CCW	Command Code (hex)	Flags	Description	Contents of Address Field
Set Mode	23	CC,SLI	Changes the timeout value.	HALSMODE
Write	01	CC,SLI	Transfers the XID(00) from this host to the communication adapter-attached host.	Output Area
Read	02	SLI	Reads an SDLC command.	Input Area

VM VSE Secondary XID Channel Program

The Secondary XID channel program is for a communication-adapter-attached type 4 physical unit (XID Exchange).

CCW	Command Code (hex)	Flags	Description	Contents of Address Field
Disable	2F	CC,SLI	Makes sure that the line direction is outbound.	—
Set Mode	23	CC,SLI	Changes the timeout value.	HALSMODE
Enable	27	CC,SLI	Causes the communication adapter to make the DCE interface ready. For CCITT v.24 or EIA RS-232C, Enable means activating the Data Terminal Ready (DTR) line and verifying that the response is Data Set Ready (DSR).	
Read	02	SLI	Reads an SDLC command.	Input Area

VM VSE Link, Connection, and Physical Unit Finite State Machines

If an error occurs during the activation or deactivation of the SDLC link or of a physical unit, the finite state machines can be used to determine the state of the link or station.

VM VSE Link State Machine Values

00	Reset
01	Sense I/O pending
02	Enable pending
08	Active
10	Halt I/O pending
20	DACTLINK disable pending
30	INOP disable pending
40	ACTLINK disable pending

VM VSE Connection State Machine Values

00	Reset
01	Answer pending
02	Auto dial pending
04	Manual dial pending
08	Connected
10	Abandon Connection 1
11	Abandon Connection 2
21	DACTCONNIN pending
41	Abandon Connect Out pending

VM VSE Physical Unit State Machine Values

00	Reset
01	Contact pending
03	Pending Discontact pending
04	XID pending
10	Active
11	Test Mode
21	Request Discontact (RQD) received
22	Discontact pending
23	Pending reset
31	Secondary Contact pending
32	Secondary Set Normal Response Mode (SNRM) received
39	Primary/Secondary contention
40	Secondary active

51	Secondary Discontact pending
56	OBR pending INOP
58	OBR pending Contacted
60	OBR pending Discontact

VM VSE Hints for Common Errors: Finite States

If an error occurs, the finite state in the PUT (PUTFSM) and in the HALCB (HALFSM) should be checked to find the state at which the error occurred. The finite state machine diagrams below show the actions occurring as a result of various inputs for the finite state machine values.

The SCX in HALCB (HALCUSCB) contains the status for the last physical unit for which I/O was active.

The sense field in HALCB (NCBSENSE) contains the sense data as a result of a unit check on the line.

The state of PU services I/O (HALPUSIO) indicates whether or not the connection manager was doing I/O.

The NCBPOLL flag, if on, indicates that the line is polling. If off, it indicates that either Write I/O or connection manager I/O is in progress.

The NCBREADY flag, if on, indicates that the link is active.

NCBLINOP indicates a permanent link error.

NCBSINOP indicates a permanent station error.

VM VSE SDLC Finite State Machines

The finite state machine diagrams (Figure 90 on page 557 through Figure 93 on page 560) show how VTAM handles PU services requests and responses for its SDLC support. The requests shown are as follows:

PU Services Request	**State Diagram**
ABCONN	Connection (Figure 91 on page 558)
ABCONNOUT	Connection (Figure 91 on page 558)
ACTCONNIN	Connection (Figure 91 on page 558)
ACTLINK	Line (Figure 90 on page 557)
CONNOUT	Connection (Figure 91 on page 558)
CONTACT	Station (Figure 92 on page 559 and Figure 93 on page 560)
DACTCONNIN	Connection (Figure 91 on page 558)
DACTLINK	Line (Figure 90 on page 557)
DISCONTACT	Station (Figure 92 on page 559 and Figure 93 on page 560)
TESTMODE	Station (Figure 92 on page 559)

The diagrams show the following:

- The request or event (above the line).
- The action taken (under the line). For example, a channel program generated or the response given to PU services as a result of the event.
- The transition (direction of the line) to the next state.

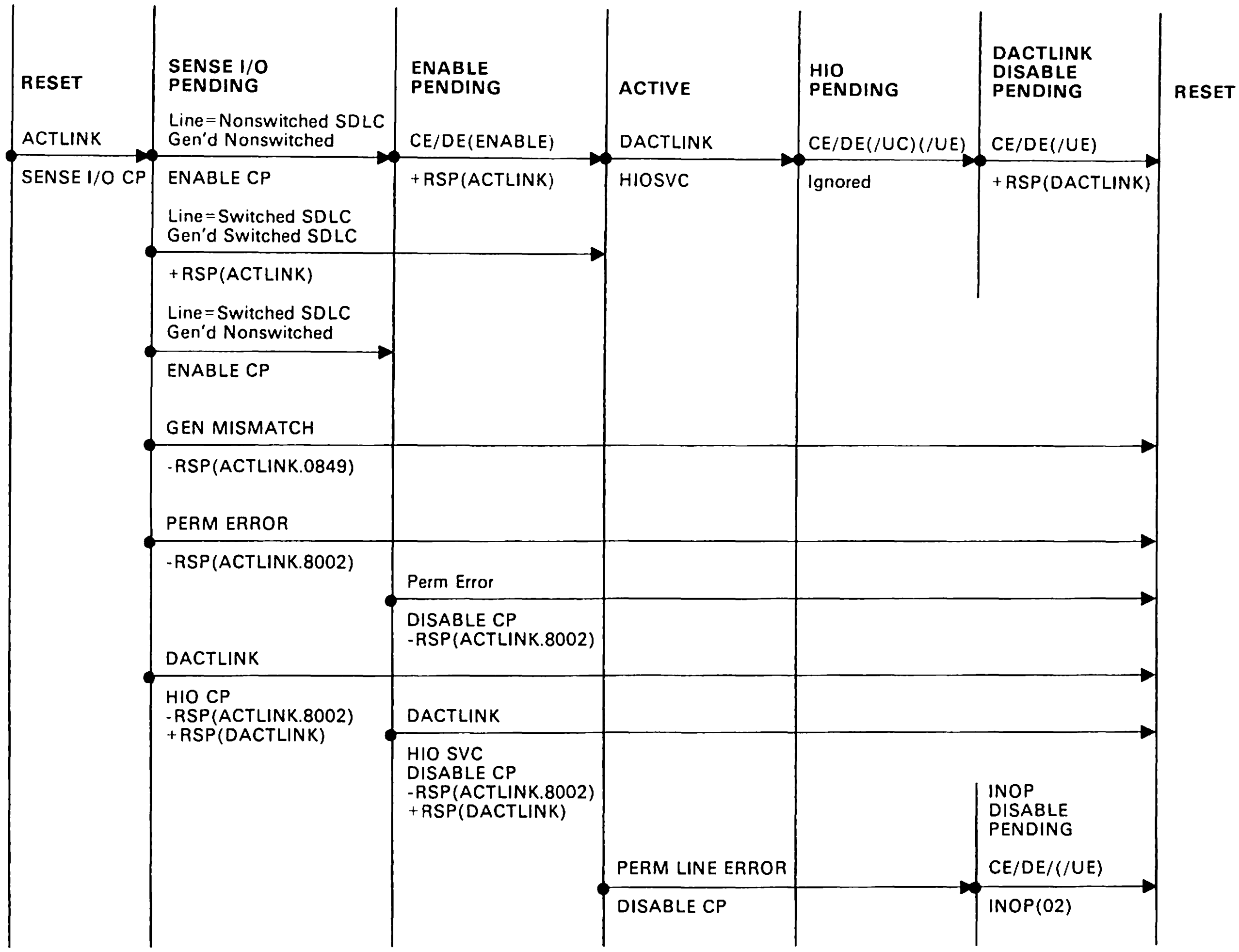


Figure 90. VM VSE Line Finite State Machine

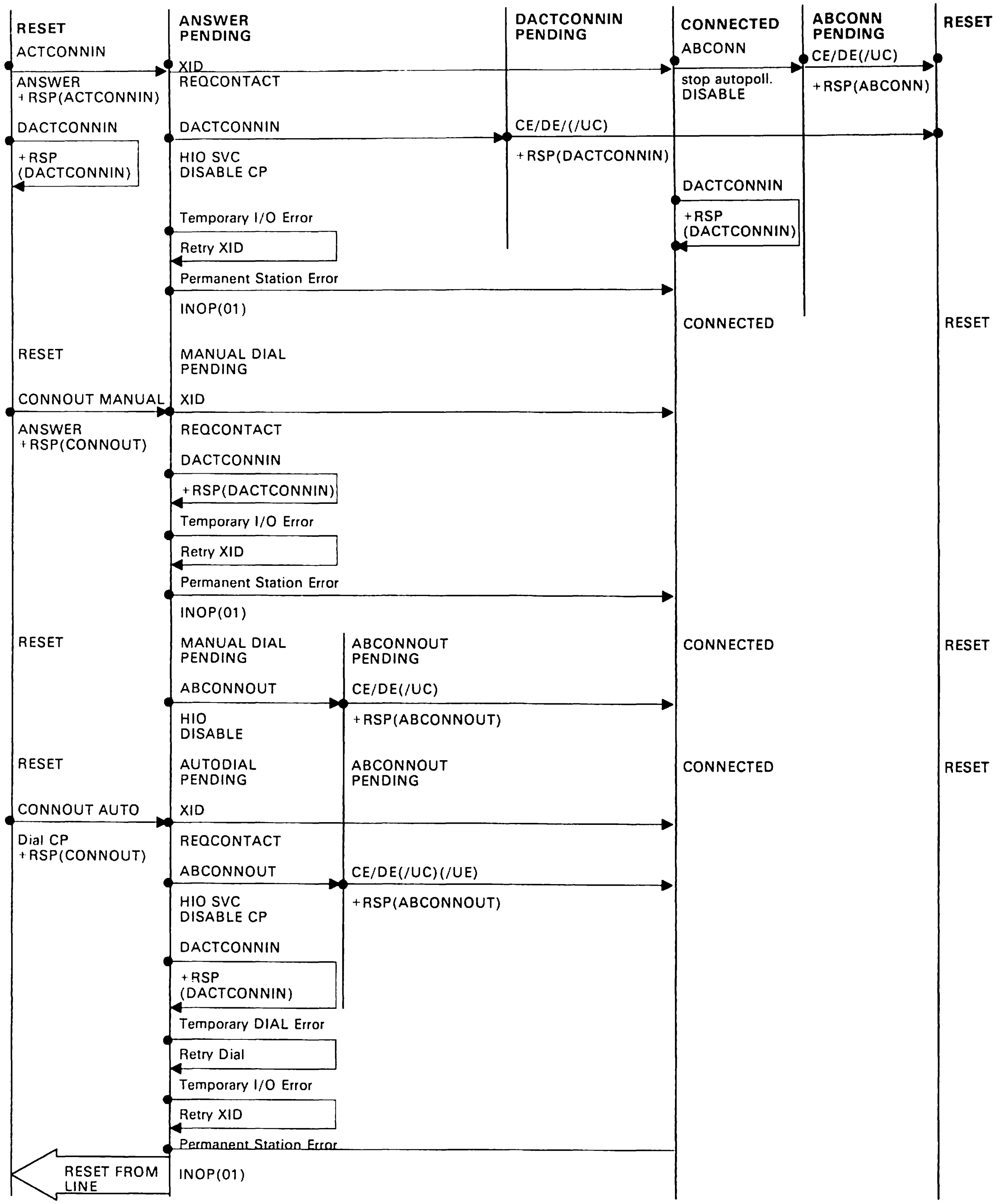


Figure 91. VM VSE Connection Finite State Machine

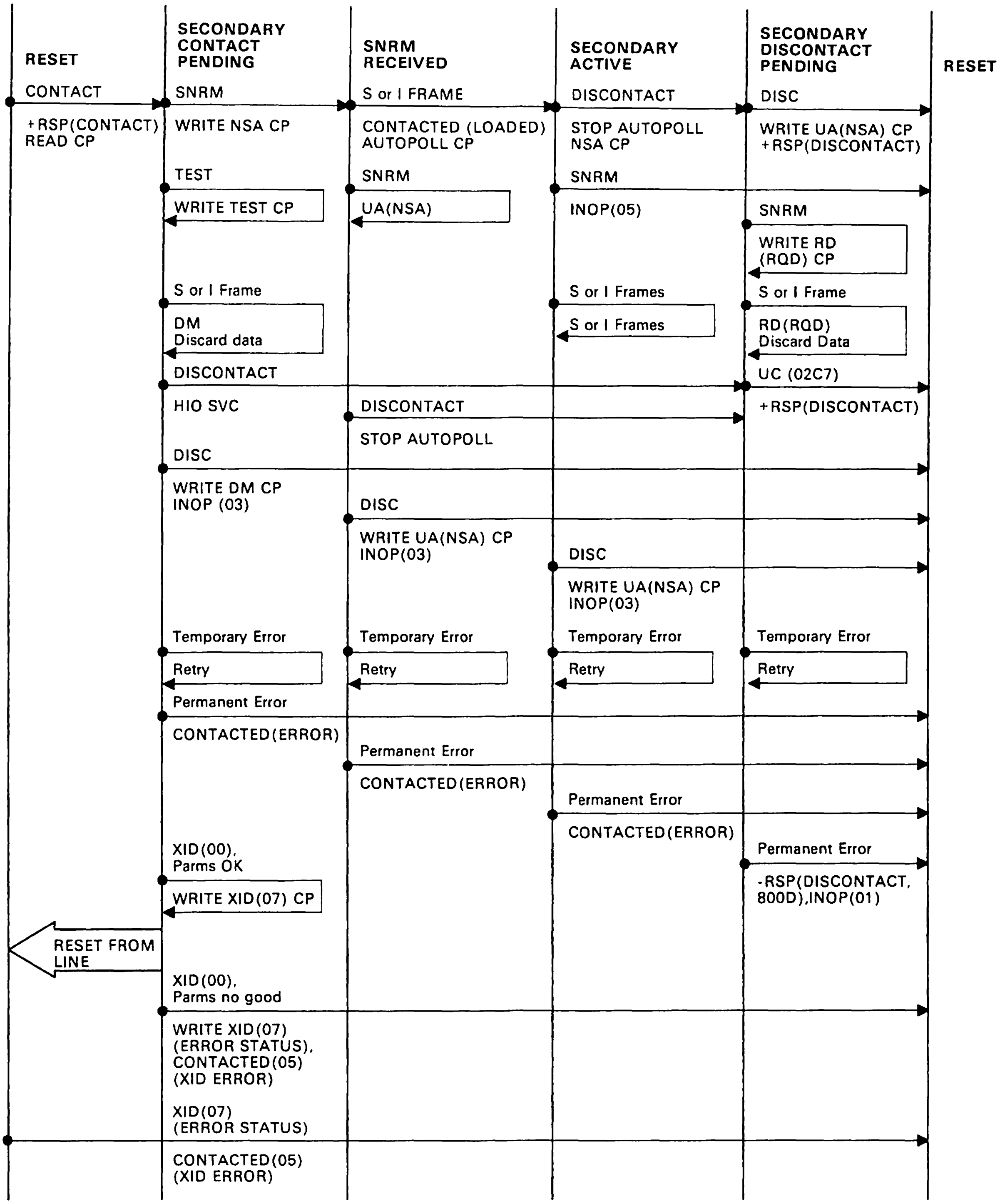


Figure 92. VM VSE Secondary Station Finite State Machine

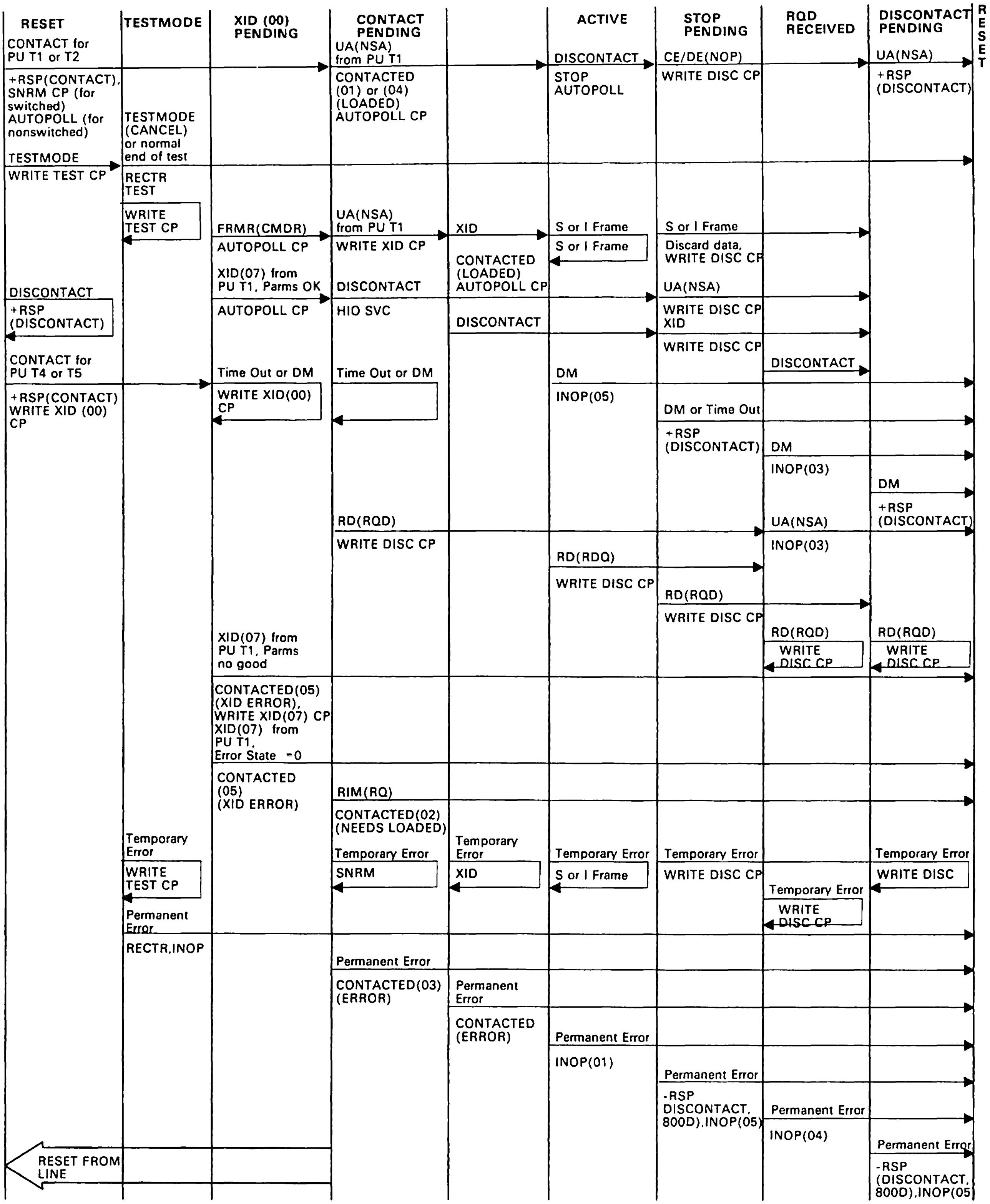


Figure 93. VM VSE Primary Station Finite State Machine

Appendix C. Network Flows

This appendix describes the flow of VTAM RUs and AMRUs between network addressable units, first single network flows and then cross-network flows. The diagrams are divided into the following categories:

- Activation
- Session Establishment
- Deactivation and Session Termination
- Error Recovery Processing (ERP) and Management Services

For certain session establishment RUs (ACTCDRM, ACTPU, ACTLU, and BIND), additional RUs can flow if the explicit route (ER) or virtual route (VR) selected for a session is not active. Because this flow is essentially the same for all four RUs, these RUs are referred to as generic bind (GBIND) AMRUs. To avoid repetition, the flows for these AMRUs are shown once at the beginning of this appendix.

Single Network Flows

Activation Flows

GBIND AMRUs: Figure 94 on page 566 through Figure 97 on page 567 show the flow of requests and responses between the SSCP and logical and physical units when a VR or ER selected for a session is not yet active.

- Sending an ACTPU request for a communication controller or physical unit (Figure 94 on page 566)
- Sending an ACTLU request for a logical unit (LU; Figure 95 on page 566)
- Sending a BIND request to a secondary logical unit (SLU; Figure 96 on page 567)
- Activating a virtual route (Figure 97 on page 567)

Resource Activation: Figure 98 on page 568 through Figure 113 on page 581 show the flow of requests and responses between the SSCP and logical and physical units to activate resources.

- Activating an NCP major node (Figure 98 on page 568 and Figure 99 on page 569)
- Activating the resources controlled by a host or NCP major node (Figure 100 on page 570 through Figure 111 on page 580)
- Takeover of a Type 2.1 peripheral node logical unit (Figure 101 on page 571)
  - Activating a connection to a host or NCP (Figure 100 on page 570 through Figure 104 on page 574)
  - Activating a physical unit (Figure 105 on page 575 and Figure 106 on page 575)
  - Moving the address of a physical unit (Figure 105 on page 575, Figure 107 on page 577, and Figure 108 on page 578)
  - Takeover of a T2.1 Node logical unit upon physical unit activation (Figure 109)
  - Activating a logical unit (Figure 110 on page 580 and Figure 111 on page 580)
  - Activating an application program/Opening an ACB (Figure 111)
- Activating a CDRM (Figure 112 on page 581 and Figure 113 on page 581)

Session Establishment Flows

Figure 114 on page 582 through Figure 128 on page 595 show the flow of requests and responses between the SSCP and logical and physical units to establish sessions.

- Requests initiated by primary logical units (PLUS) (Figure 114 on page 582 through Figure 117 on page 585)
- An independent PLU-initiated request for LU-LU session establishment (Figure 118 on page 586)

- A PLU-initiated request for LU-LU session establishment with an independent SLU (Figure 119 on page 587)
- SLU-initiated requests (Figure 120 on page 588 through Figure 123 on page 590)
- An independent SLU-initiated request for LU-LU session establishment with a PLU (Figure 124 on page 591)
- Third party-initiated requests (Figure 125 on page 592 and Figure 126 on page 593)
- VARY LOGON or LOGAPPL processing (Figure 127 on page 594)
- Notification of PLU Availability (Figure 128 on page 595)

Deactivation and Termination Flows

Figure 129 on page 596 through Figure 138 on page 604 show the flow of requests and responses between the SSCP and logical and physical units to deactivate resources and end sessions.

- Deactivating a logical unit (Figure 129 on page 596 and Figure 130 on page 596)
- Deactivating a logical unit with giveback (Figure 131 on page 597)
- VARY TERM processing (Figure 132 on page 598 and Figure 133 on page 599)
- CLOSE ACB processing (Figure 134 on page 600)
- Deactivating an application program (Figure 135 on page 601)
- Deactivating a CDRM (Figure 136 on page 602 through Figure 138 on page 604)

ERP and Management Services Flows

Figure 139 on page 605 through Figure 141 on page 606 show the flow of requests and responses between the SSCP and logical and physical units to handle error recovery processing (ERP) and route Forward and Deliver RUs.

- ERP processing (Figure 139 on page 605 and Figure 140 on page 606)
- SSCP management services processing (Figure 141 on page 606)

MVS VM Cross-Network Flows

MVS VM Activation Flows

Figure 142 on page 607 through Figure 144 on page 609 show the flow of requests and responses activating a cross-network SSCP-SSCP session.

- Gateway SSCP requests session (Figure 142 on page 607)
- Non-gateway SSCP requests a session (Figure 143 on page 608)
- Back-to-back gateway NCPs request a session (Figure 144 on page 609)

MVS VM Session Establishment Flows

Figure 145 on page 610 through Figure 164 on page 631 show the flow of requests and responses to establish a cross-network LU-LU session.

- SLU-initiated request for single gateway SSCP and single gateway NCP (Figure 145 on page 610)
- PLU-initiated request for single gateway SSCP and single gateway NCP (Figure 147 on page 612)
- Independent PLU-initiated request for single gateway SSCP and single gateway NCP (Figure 149 on page 614)
- PLU-initiated request setup queued for single gateway SSCP and single gateway NCP (Figure 150 on page 615)
- SLU-initiated request for a single gateway connecting three or more networks (Figure 152 on page 617)
- Default partitioning of gateway SSCP responsibility spanning three networks (Figure 153 on page 618)
- SLU-initiated request for predesignated control of gateway NCP by middle host (Figure 154 on page 619)
- Multiple gateway SSCPs and back-to-back gateway NCPs (Figure 156 on page 623)
- Third-party-initiated request spanning three networks (Figure 157 on page 624)
- Initiate with VARY LOGON or Controlling PLU for single gateway SSCP and single gateway NCP (Figure 158 on page 625)
- SLU-initiated session initiation failure (CDINIT rejection) for single gateway SSCP and single gateway NCP (Figure 160 on page 627)
- SLU-initiated session initiation failure (SETCV failure) for single gateway SSCP and single gateway NCP (Figure 162 on page 629)
- SLU-initiated setup procedure failure (CINIT rejection) for single gateway SSCP and single gateway NCP (Figure 164 on page 631)

MVS VM Deactivation and Termination Flows

Figure 166 on page 633 through Figure 178 on page 644 show the flow of requests and responses to deactivate resources and end sessions.

- PLU sends UNBIND for single gateway SSCP and single gateway NCP (Figure 166 on page 633)
- Independent PLU sends UNBIND for independent SLU (Figure 167 on page 633)
- Independent PLU sends CLEANUP for independent SLU (Figure 168 on page 634)
- SLU requests Terminate Self for single gateway SSCP and single gateway NCP (Figure 169 on page 635)
- SLU requests Terminate Self (Cleanup Option) for single gateway SSCP and single gateway NCP (Figure 170 on page 636)
- PLU sends UNBIND for multiple gateway SSCPS and a single gateway NCP (Figure 172 on page 638)
- SLU requests Terminate Self for multiple gateway SSCPS and back-to-back gateway NCPS (Figure 173 on page 639)
- Terminating an SSCP-SSCP session (Figure 174 on page 640)
- Terminating an LU-LU session between two independent LUS (Figure 175 on page 641)
- Route failure causes session termination (Figure 176 on page 642)
- Route failure causes session termination for multiple gateway SSCPS and back-to-back gateway NCPS (Figure 178 on page 644)

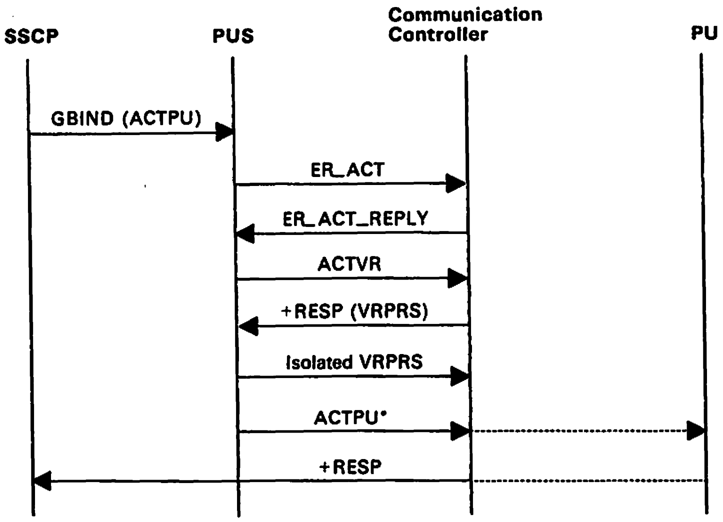


Figure 94. Session Activation: ACTPU.

* The ACTPU may flow either to the communication controller or to a physical unit.

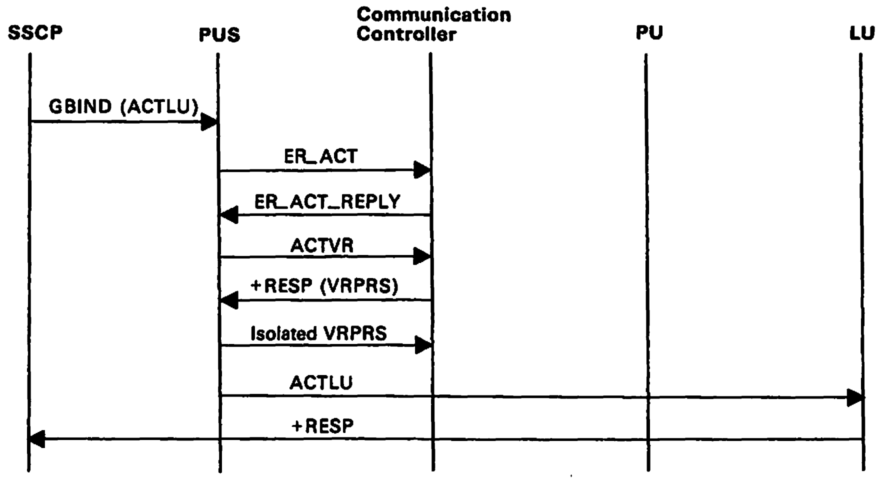


Figure 95. Session Activation: ACTLU

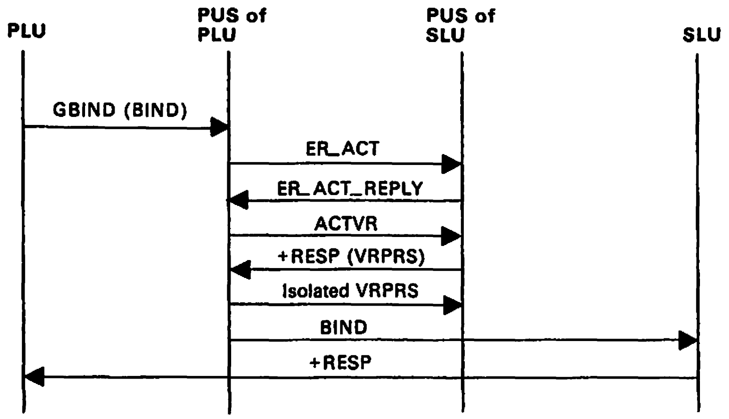


Figure 96. Session Activation: BIND

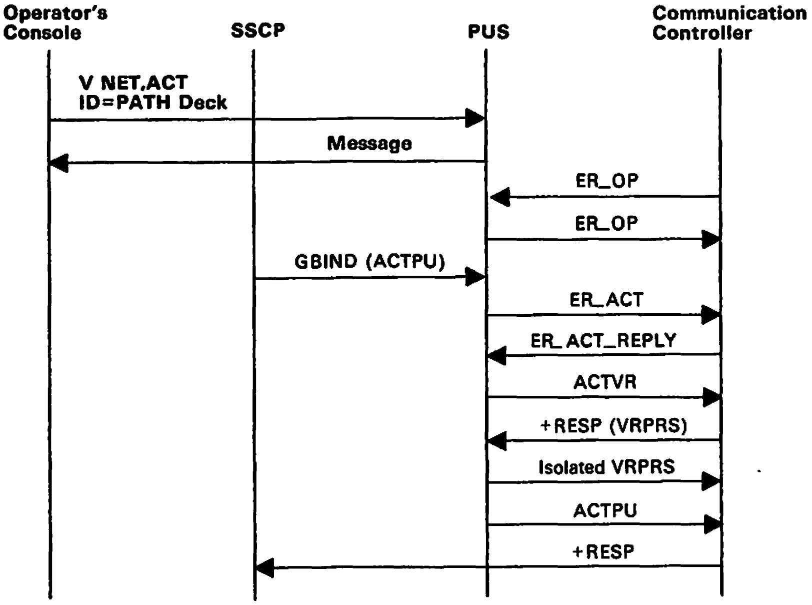


Figure 97. Activating a Virtual Route and the Associated Explicit Route

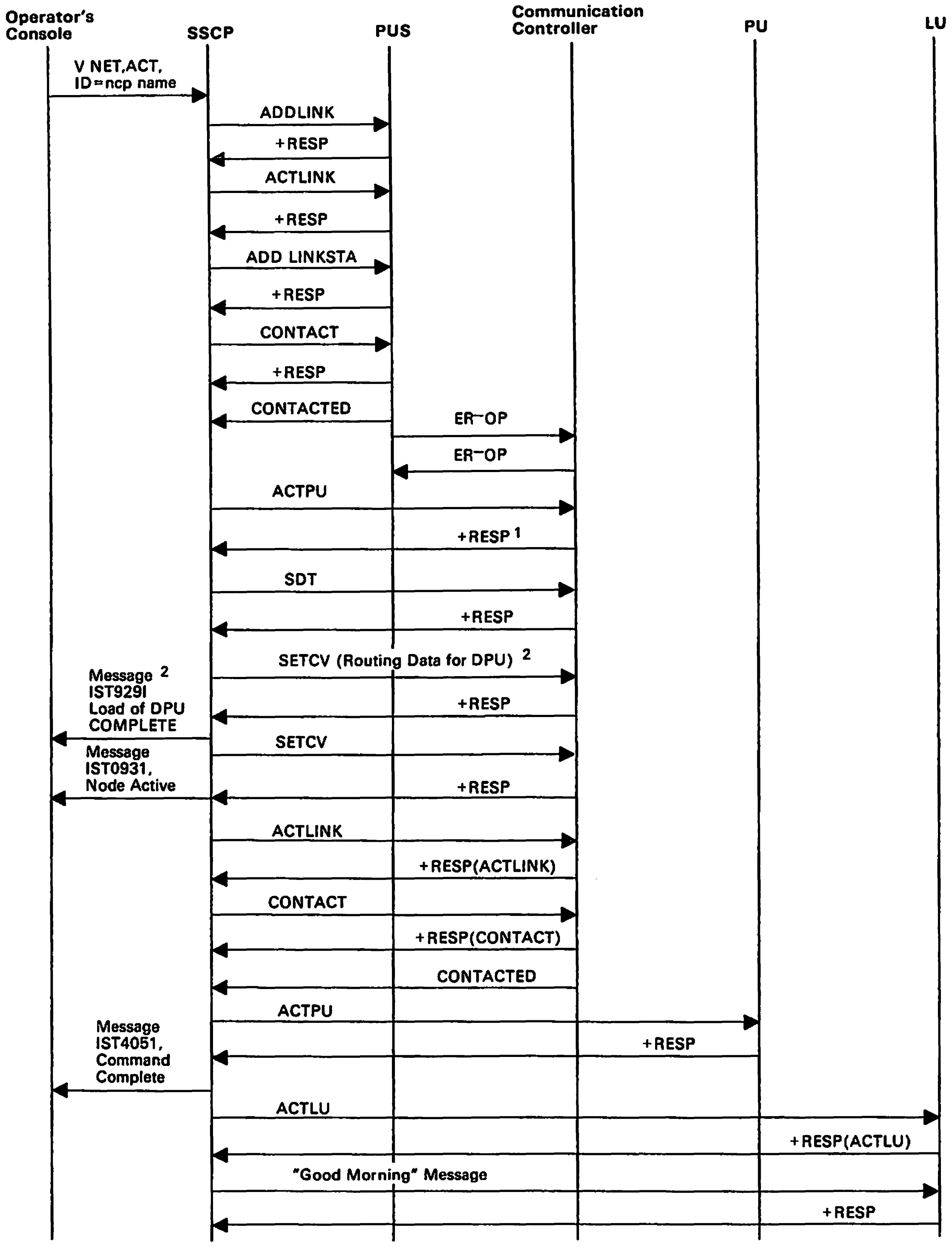


Figure 98. Activating a Channel-Attached Communication Controller.

[1] Includes NCP Dynamic Path Definition capability indicator in V3R2.

[2] Flows only for Dynamic Path Definition in V3R2.
SETCV and IST929I flow for each Dynamic Path Definition Member specified.

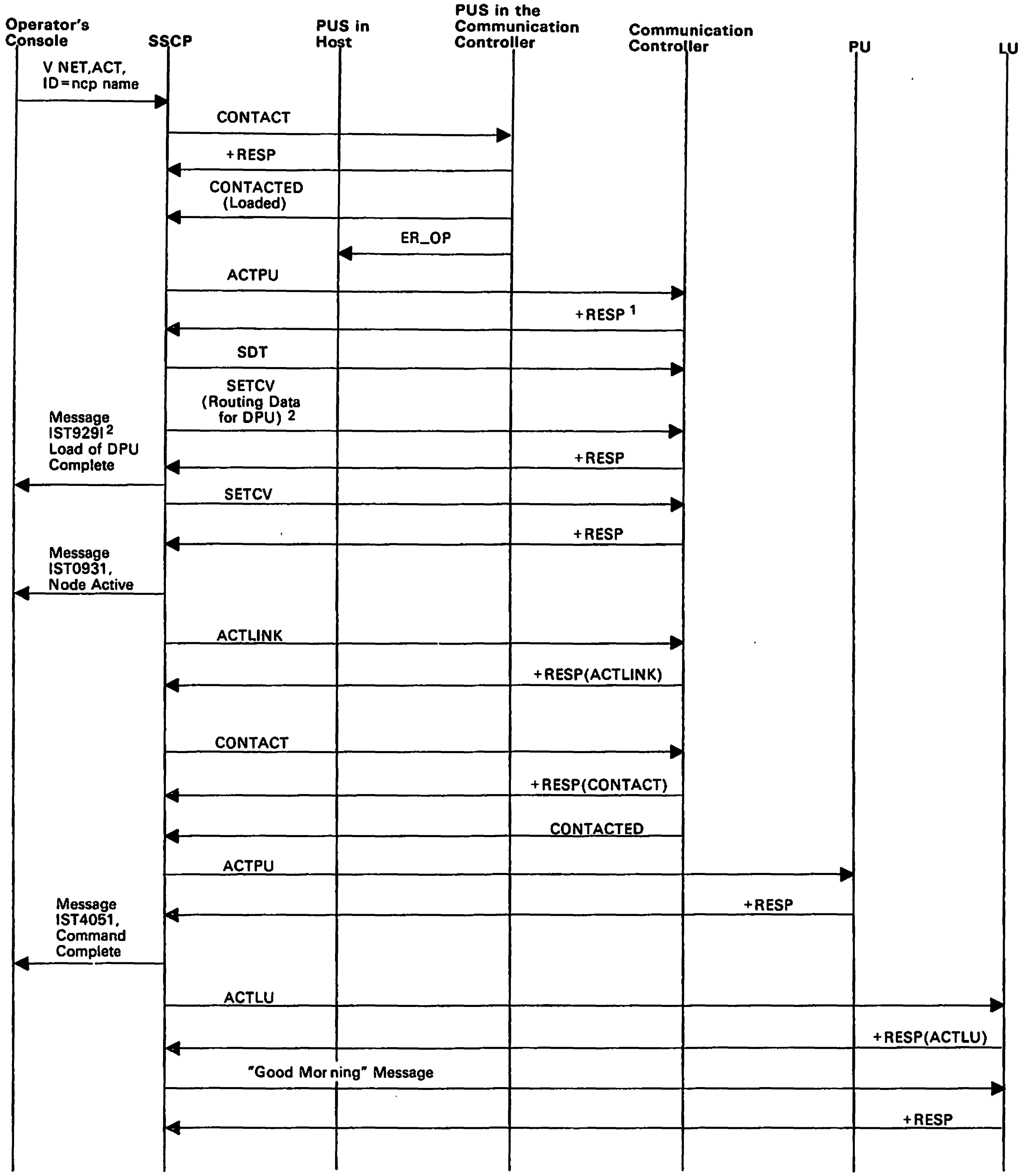


Figure 99. Activating a Link-Attached Communication Controller.

[1] Includes NCP Dynamic Path Definition capability indicator in V3R2.

[2] Flows only for Dynamic Path Definition in V3R2.
SETCV and IST929I flow for each Dynamic Path Definition Member specified.

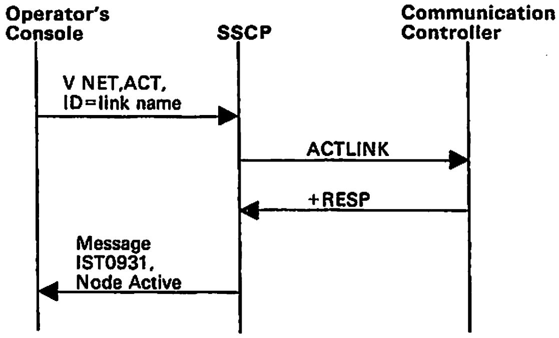


Figure 100. Activating a Link (ACTLINK)

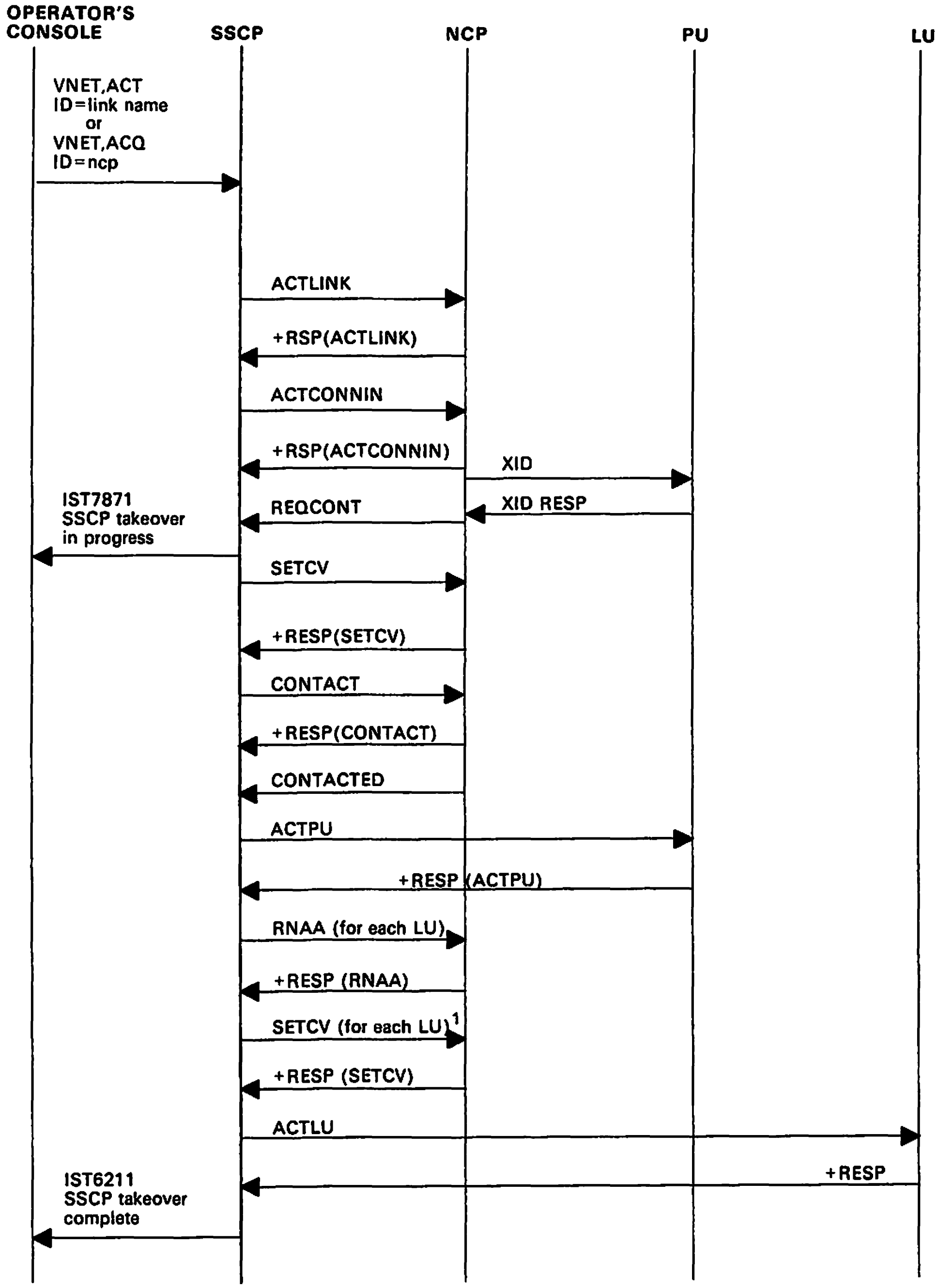


Figure 101. Activating a Switched Link with Takeover.

[1] SETCV does not flow for NCPs that support peripheral nodes.

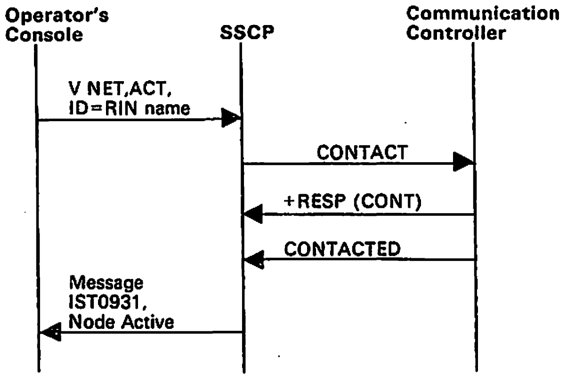


Figure 102. Activating a Cross-Subarea Link Station

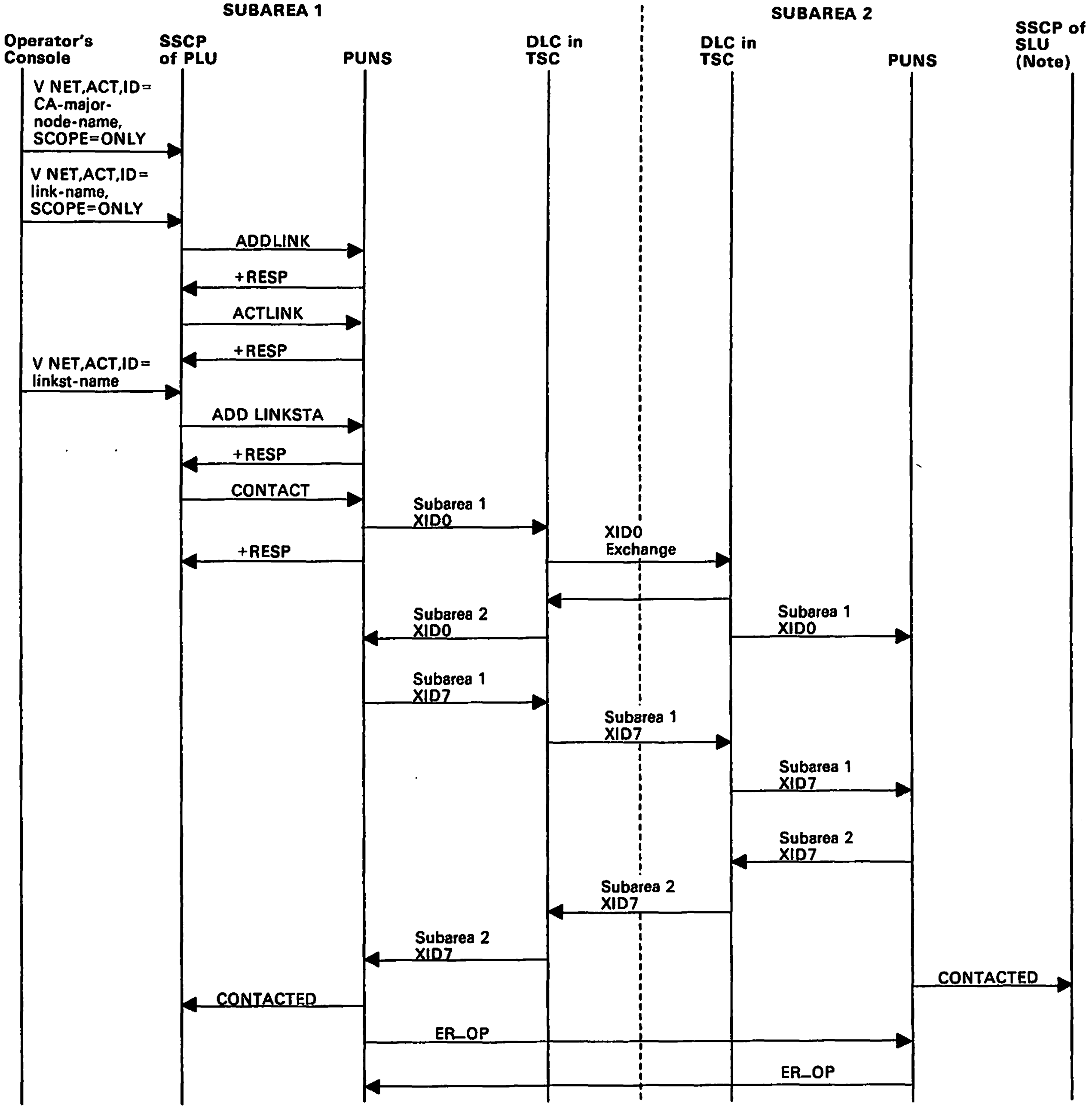


Figure 103. Establishing Contact with a Channel-Attached Host

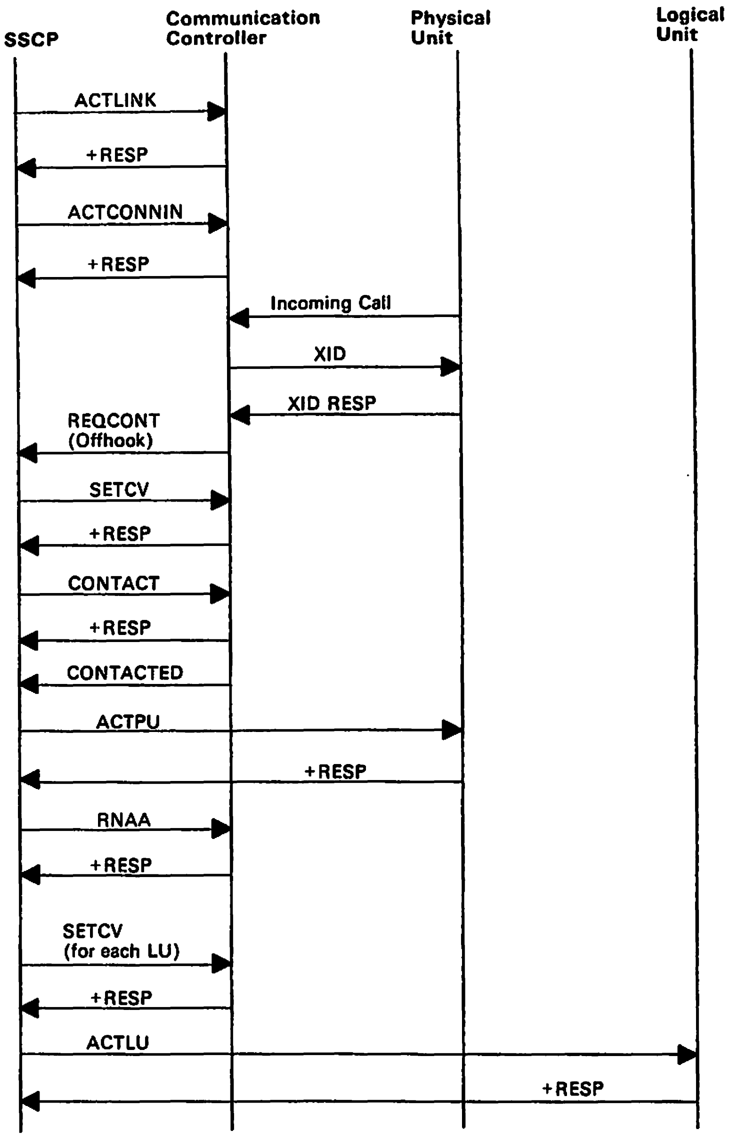


Figure 104. Establishing a Switched Connection. To establish a switched connection, the SSCP sends an Active Link request to indicate that the link is active. An Active Connect In request is sent to enable the communication controller to answer incoming calls. (Instead of Activate Connect In, Dial could be sent to initiate an outbound call.) When a call comes in, the communication controller sends an XID (exchange identification) and the physical unit responds with its ID (station address). The communication controller sends a Request Contact (Offhook) request to the SSCP. The SSCP sends a Set Control Vector request containing address and pacing information to the communication controller. The standard activation sequence then occurs (see Figure 99 on page 569).

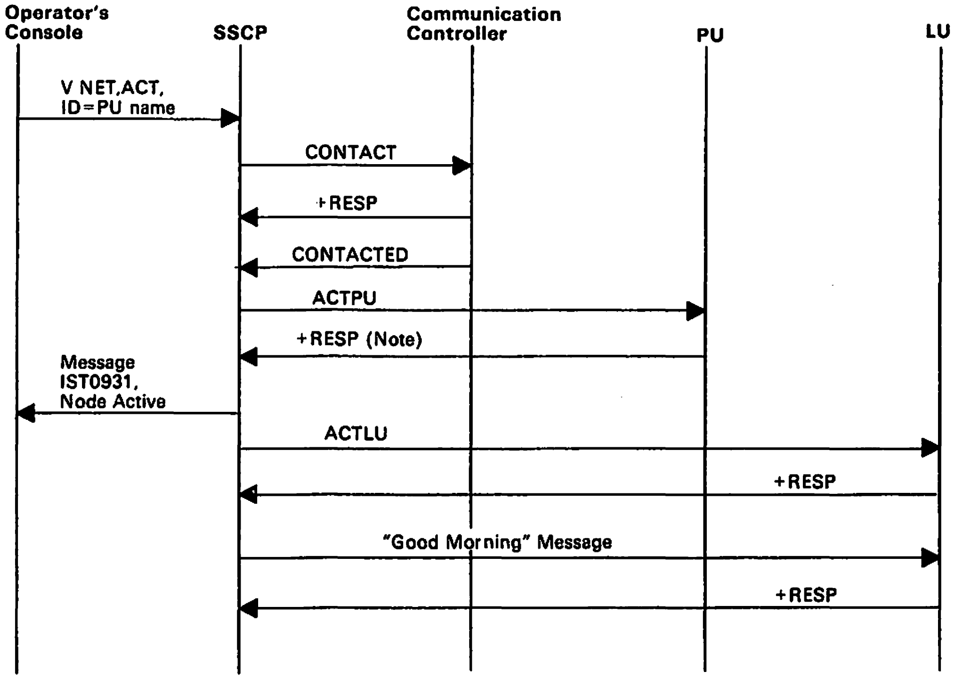


Figure 105. Activating a Type 2 Physical Unit

Note: If the physical unit must be loaded, additional RUs will flow. See Figure 106 for more information.

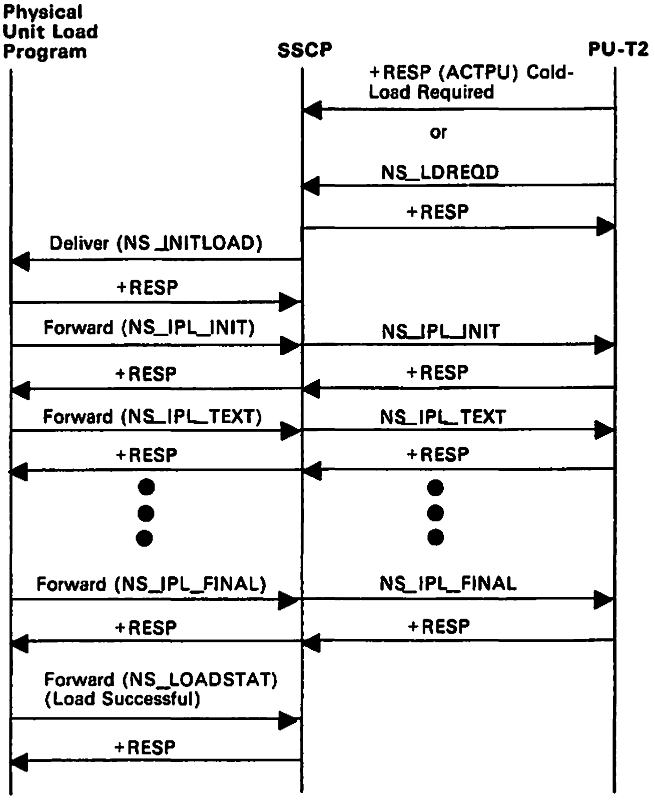


Figure 106. Activating a Type 2 Physical Unit with Load Required

This figure shows only the RUs that flow when a type 2 physical unit requires loading. For the RUs that flow before and after those in this figure, see Figure 105.

Some devices, such as the IBM 8775, may require loading. During activation of the physical unit, the request for load is indicated in the ACTPU response. (Following activation, the physical unit may request loading with an NS_LDREQD RU.)

The SSCP formats the load request into a network services (NS) RU to initiate the load. The management services subcomponent of the SSCP then sends the embedded request to the physical unit load program of the Downstream Load Utility.

If the physical unit load program *is not* available, it sends a negative response to the SSCP's Deliver RU. The SSCP then sends an NS_IPL_ABORT RU to the physical unit for deactivation processing. (If the load was requested with an NS_LDREQD RU, the physical unit is not deactivated; in fact, it may try the load request again.)

If the physical unit load program *is* available, it sends a positive response to the SSCP's Deliver RU. The load program then sends a forward RU, containing an NS_LOADSTAT RU, to relay the status of the load operation.

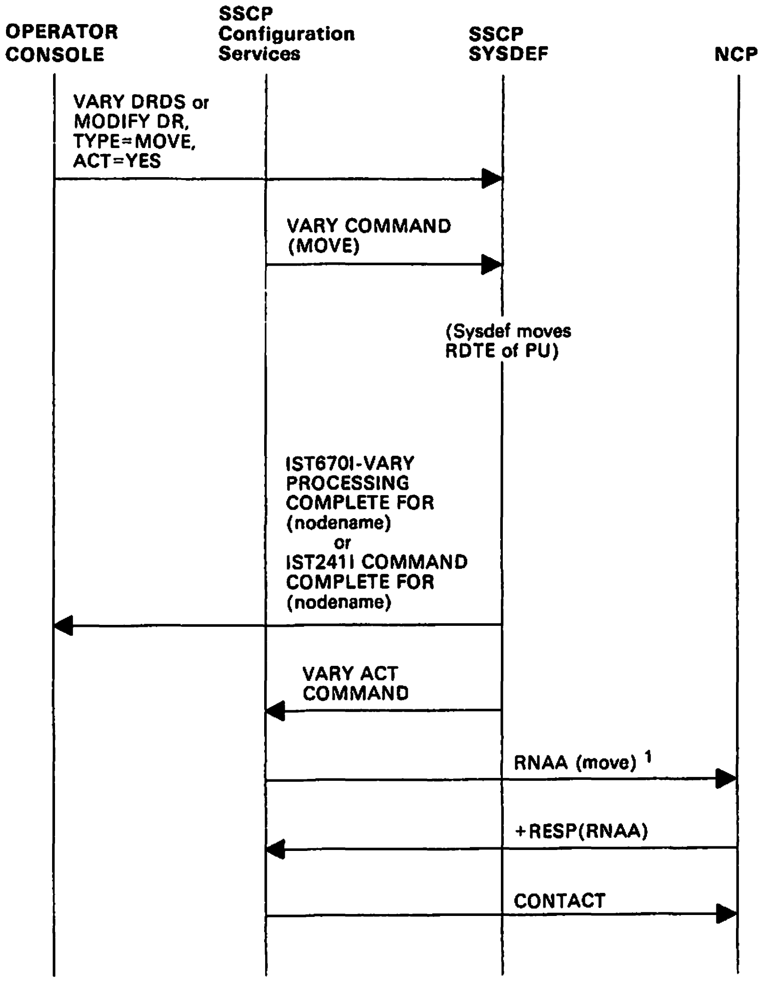

OPERATOR CONSOLE
SSCP Configuration Services
SSCP SYSDEF
NCP
VARY DRDS or MODIFY DR, TYPE=MOVE, ACT=YES
VARY COMMAND (MOVE)
(Sysdef moves RDTE of PU)
IST670I-VARY PROCESSING COMPLETE FOR (nodename) or IST241I COMMAND COMPLETE FOR (nodename)
VARY ACT COMMAND
RNAA (move) 1
+RESP(RNAA)
CONTACT

Figure 107. Moving a SYSGENned Physical Unit.

[1] RNAA flow is as normal. RNAA does not flow for MODIFY DR, TYPE=MOVE, ACT=NO.

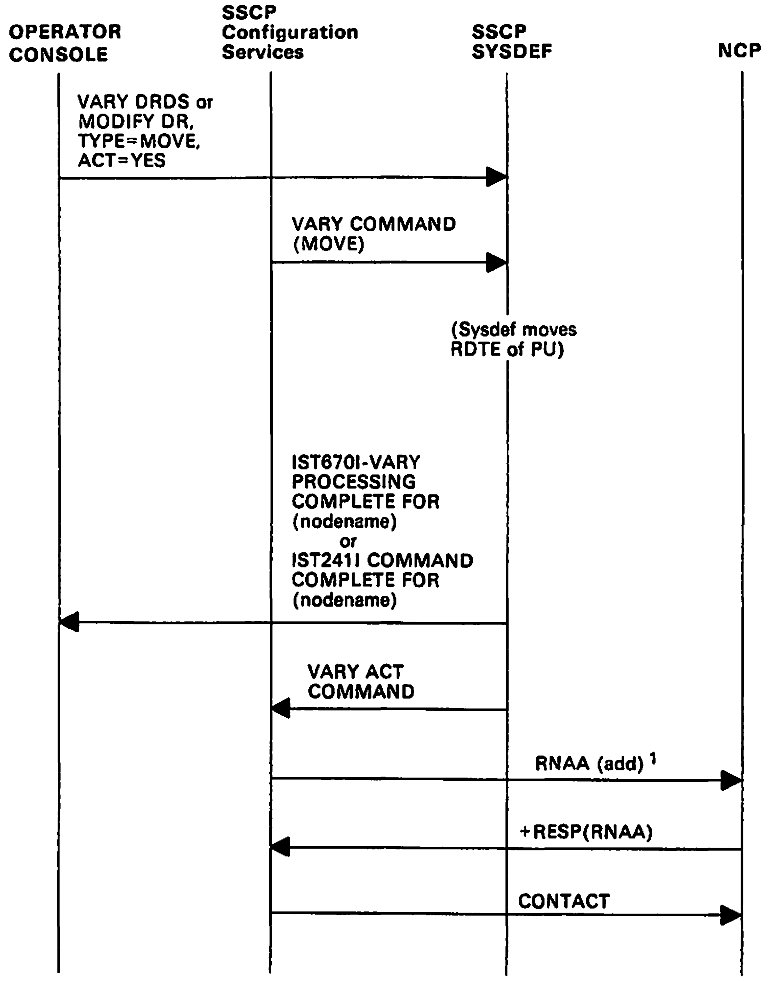

OPERATOR CONSOLE
SSCP Configuration Services
SSCP SYSDEF
NCP
VARY DRDS or MODIFY DR, TYPE=MOVE, ACT=YES
VARY COMMAND (MOVE)
(Sysdef moves RDTE of PU)
IST670I-VARY PROCESSING COMPLETE FOR (nodename) or IST241I COMMAND COMPLETE FOR (nodename)
VARY ACT COMMAND
RNAA (add) 1
+RESP(RNAA)
CONTACT

Figure 108. Moving a Dynamically Added Physical Unit.

[1] RNAA flow is as normal. RNAA does not flow for MODIFY DR, TYPE=MOVE, ACT=NO.

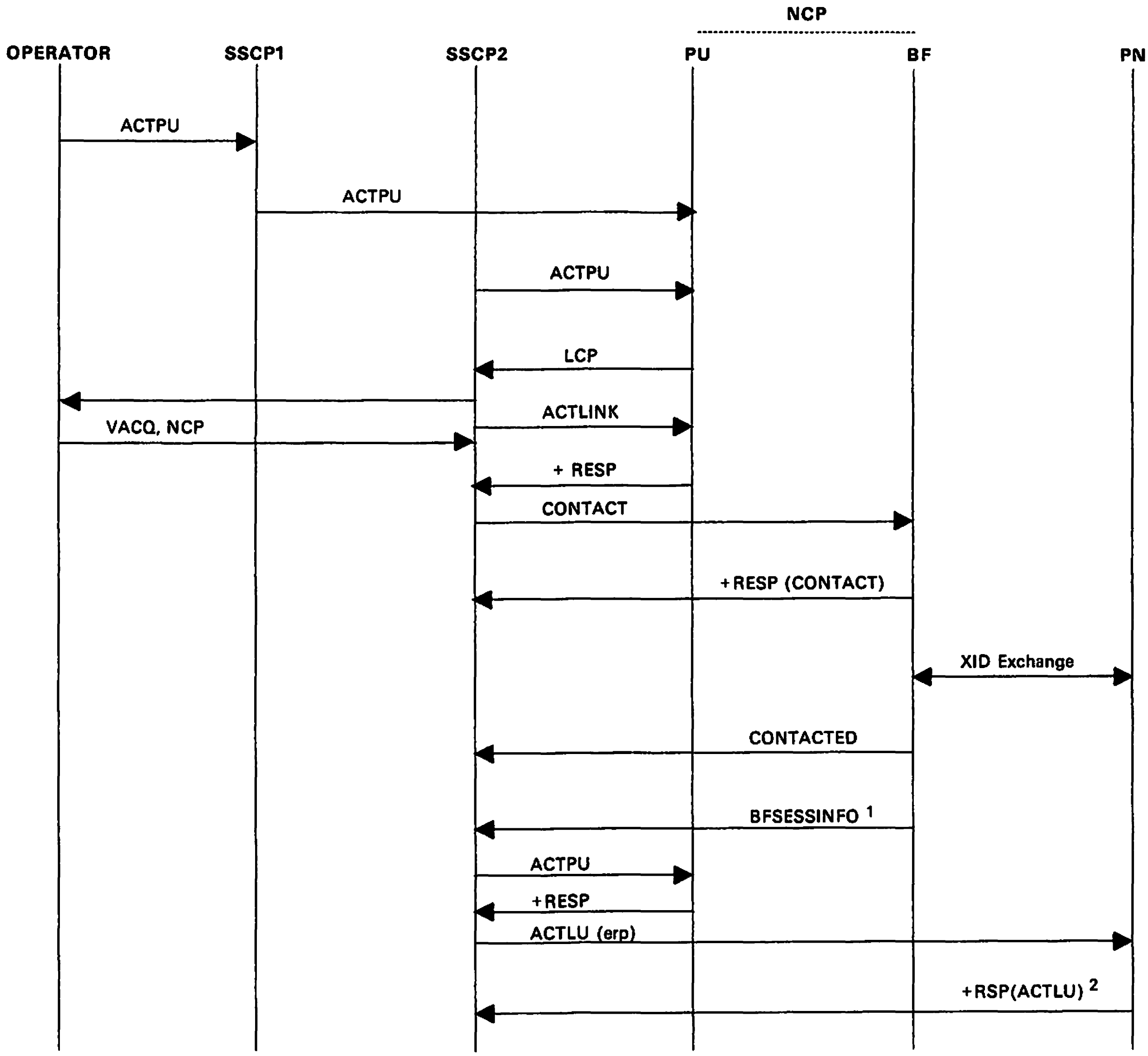


Figure 109. SSCP Takeover of Peripheral Node Logical Units.

[1] Independent logical unit only.

[2] Dependent logical unit only.

Note: The following conditions are assumed for this example:

- The physical units being taken over are defined with ANS=CONTINUE, specifying that any LU-LU sessions that are active at SSCP-takeover time will continue.
- There are some LU-LU sessions active at failure time under the physical unit being taken over.

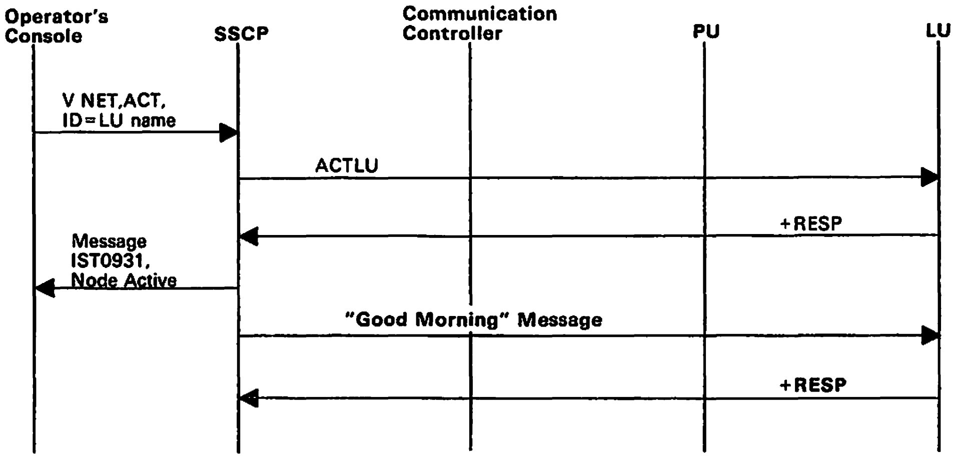


Figure 110. Activating a Logical Unit

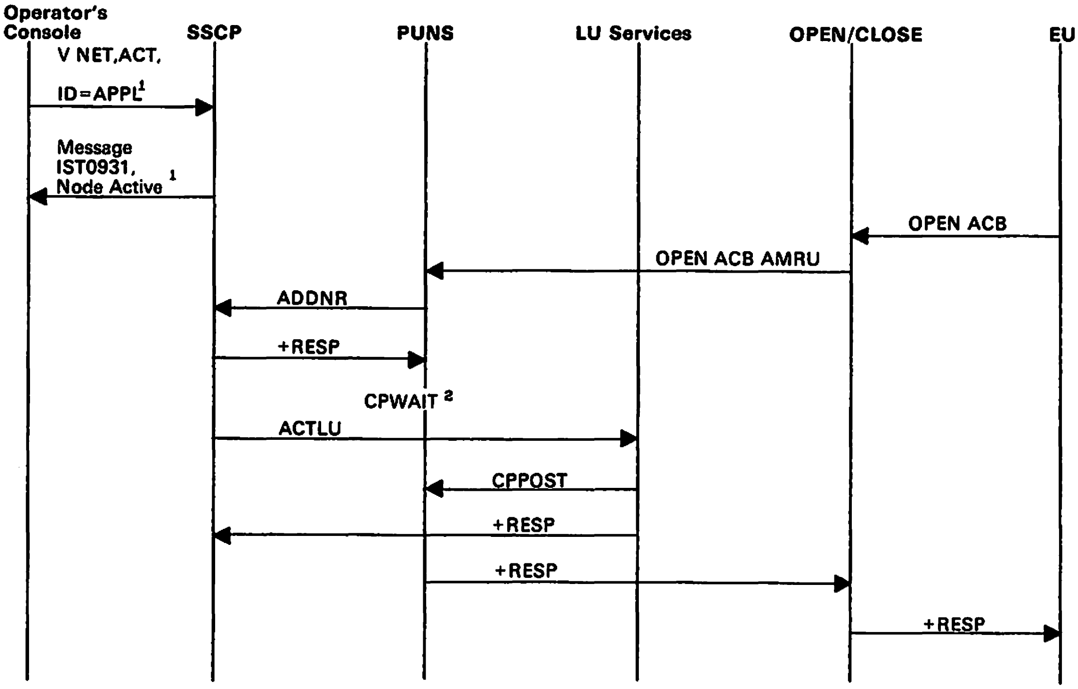


Figure 111. Activating an Application Program/Processing an OPEN ACB Request.

[1] These do not flow for OPEN ACB processing.

[2] PUNS cannot send a response to the OPEN ACB request until LUS receives an ACTLU request for the application program. Therefore, PUNS issues CPWAIT and waits for LUS to post it. Once LUS has received the ACTLU, it posts PUNS, which then sends a response to the OPEN ACB request.

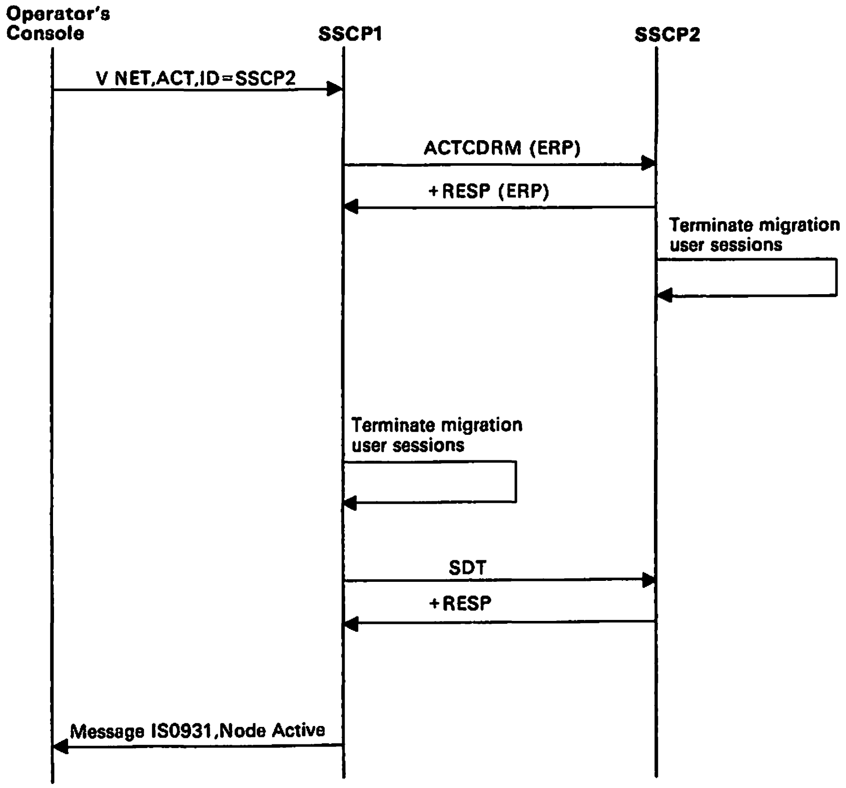

Operator's Console
SSCP1
SSCP2
V NET,ACT,ID=SSCP2
ACTCDRM (ERP)
+RESP (ERP)
Terminate migration user sessions
Terminate migration user sessions
SDT
+RESP
Message IS0931,Node Active

Figure 112. Activate CDRM with ERP Response

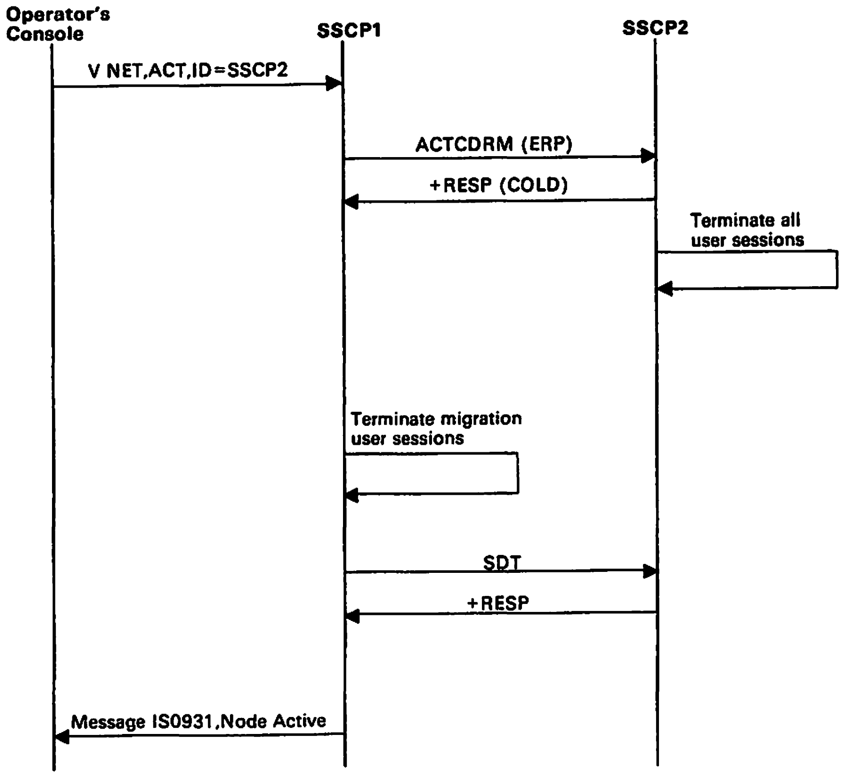

Operator's Console
SSCP1
SSCP2
V NET,ACT,ID=SSCP2
ACTCDRM (ERP)
+RESP (COLD)
Terminate all user sessions
Terminate migration user sessions
SDT
+RESP
Message IS0931,Node Active

Figure 113. Activate CDRM with COLD Response

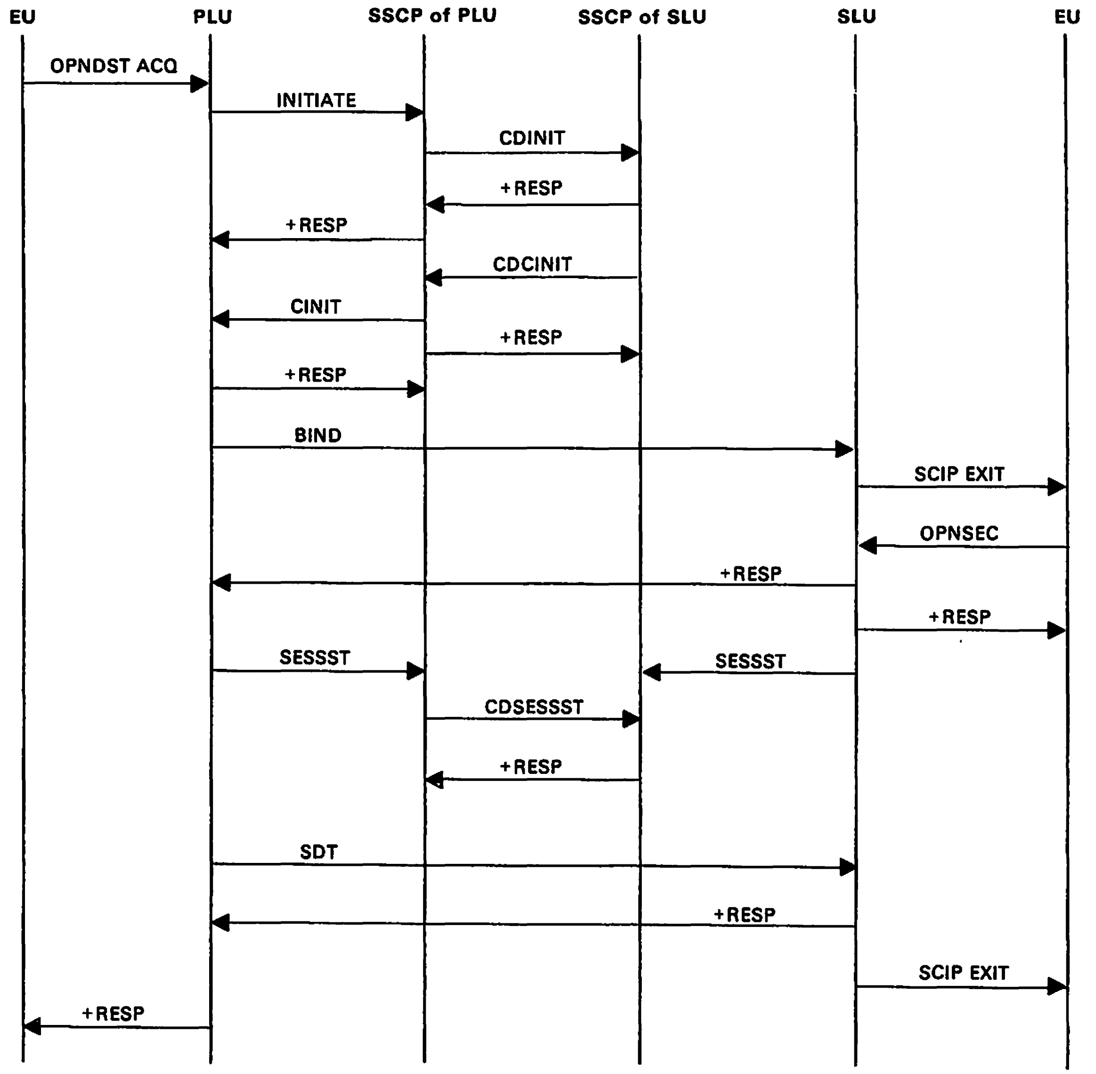


Figure 114. Primary Logical Unit Initiate (OPNDST Acquire)

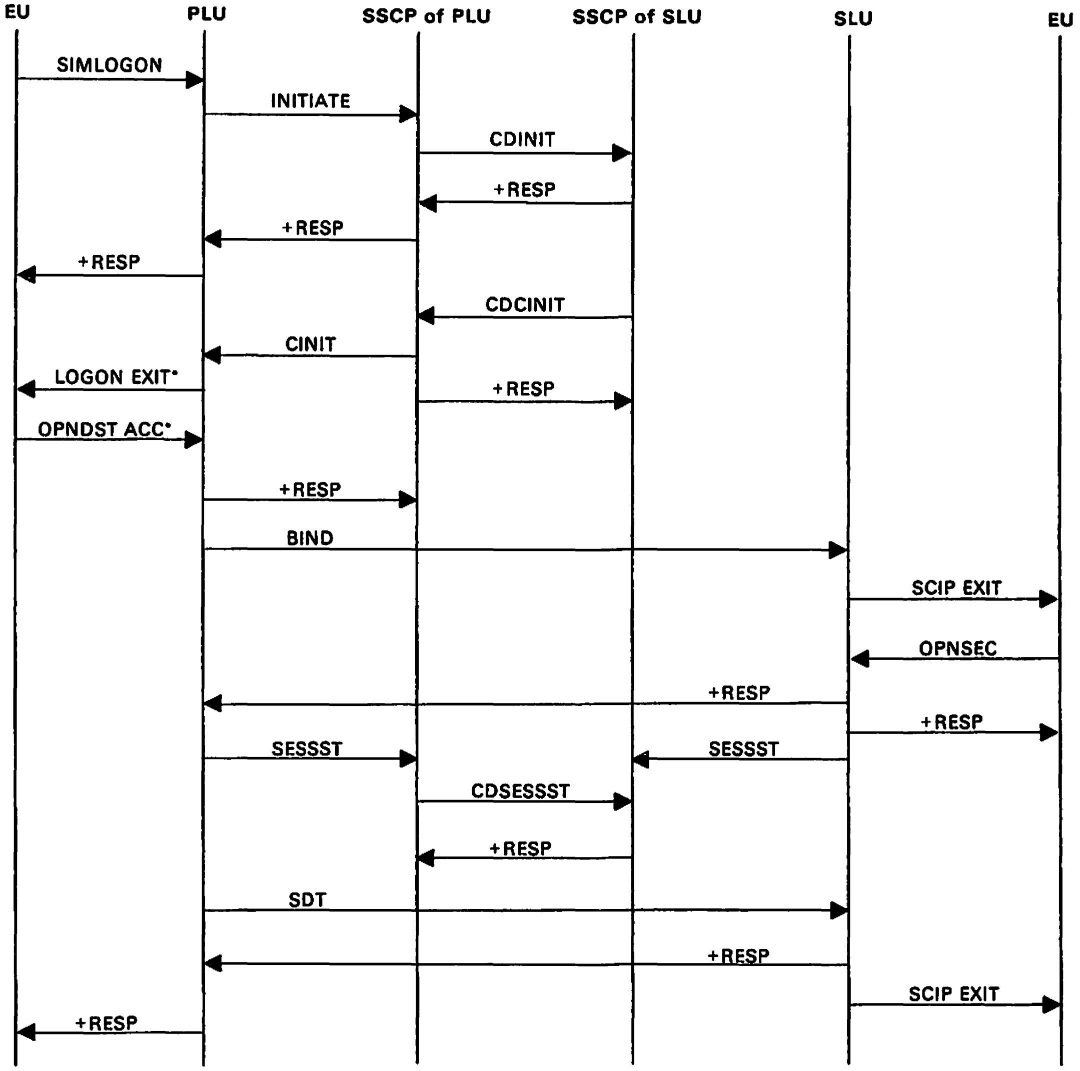


Figure 115. Primary Logical Unit Initiate (SIMLOGON).

* LOGON EXIT and OPNDST ACC flow only when the PLU is associated with an application program. It does not appear in the flow if the PLU is a device-type logical unit. This is true for many following flows with LOGON EXIT and OPNDST ACC.

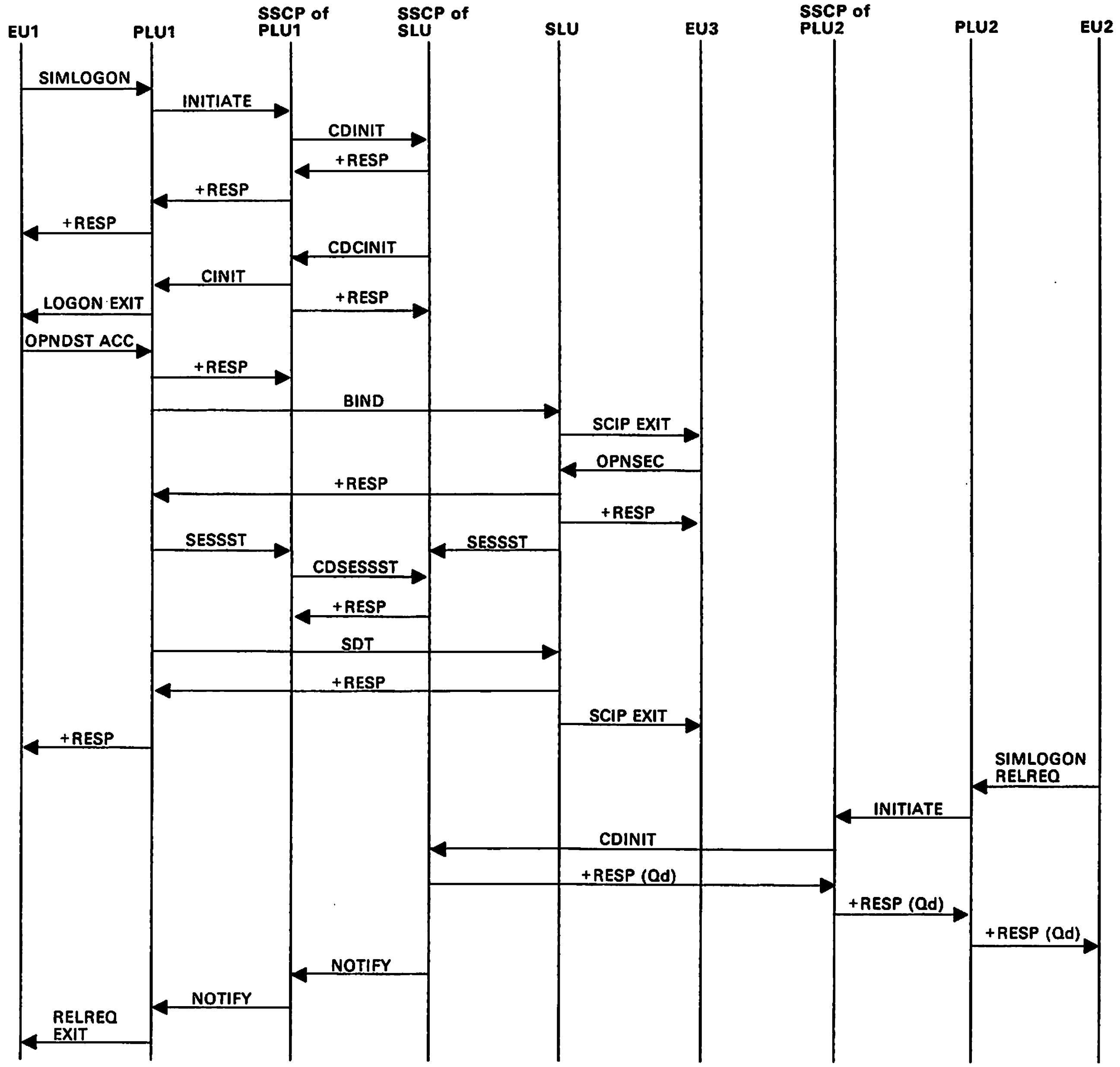


Figure 116. Primary Logical Unit Initiate (SIMLOGON(RELREQ))

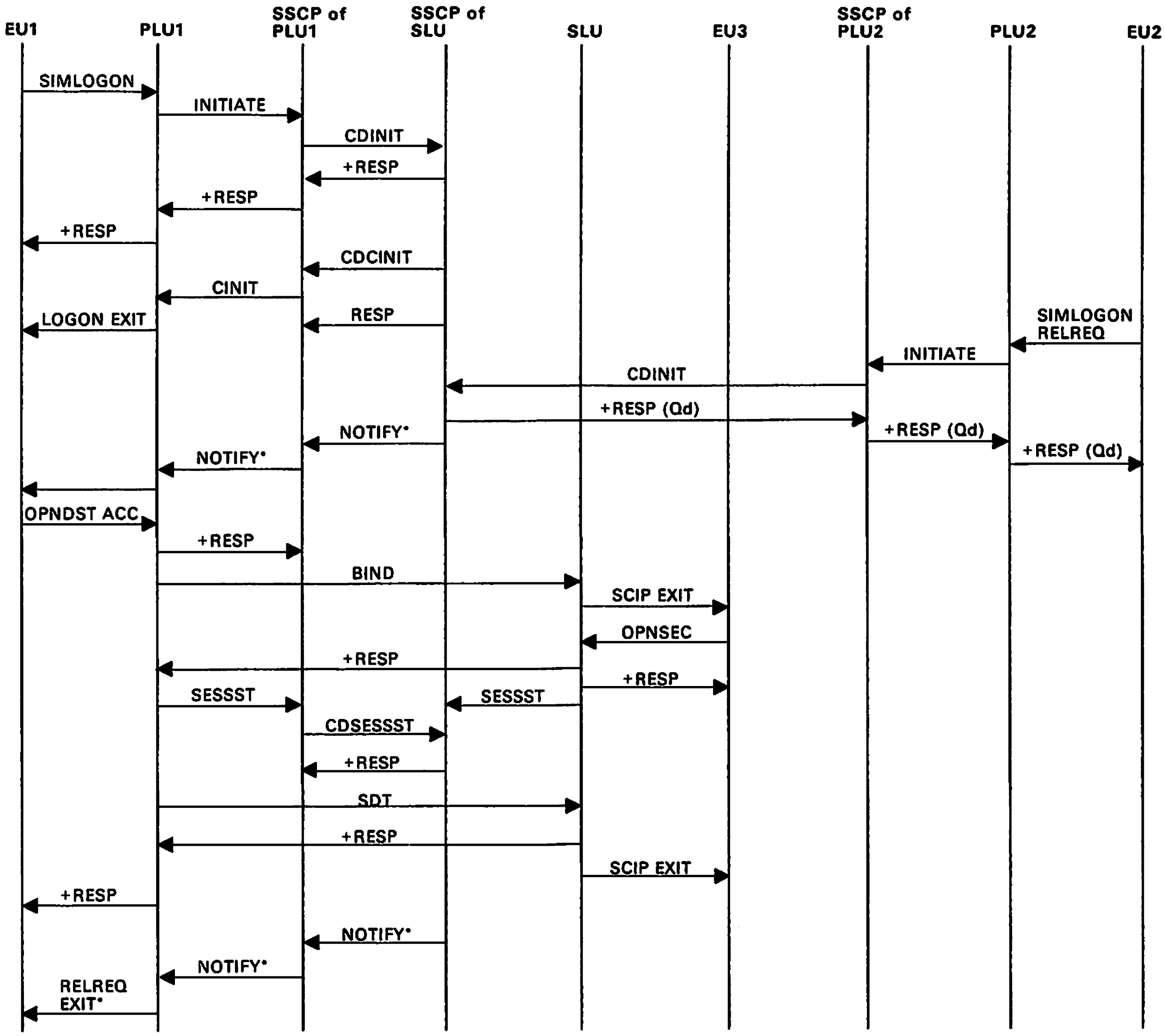


Figure 117. Primary Logical Unit Initiate (SIMLOGON(RELREQ))(Session is Pending Active or Already in Progress).

* If the session is pending active, the NOTIFY request and the RELREQ EXIT request flow after the response to the CDINIT request from PLU2's SSCP. If the session is already in progress, these requests flow after the response to the OPNDST ACC.

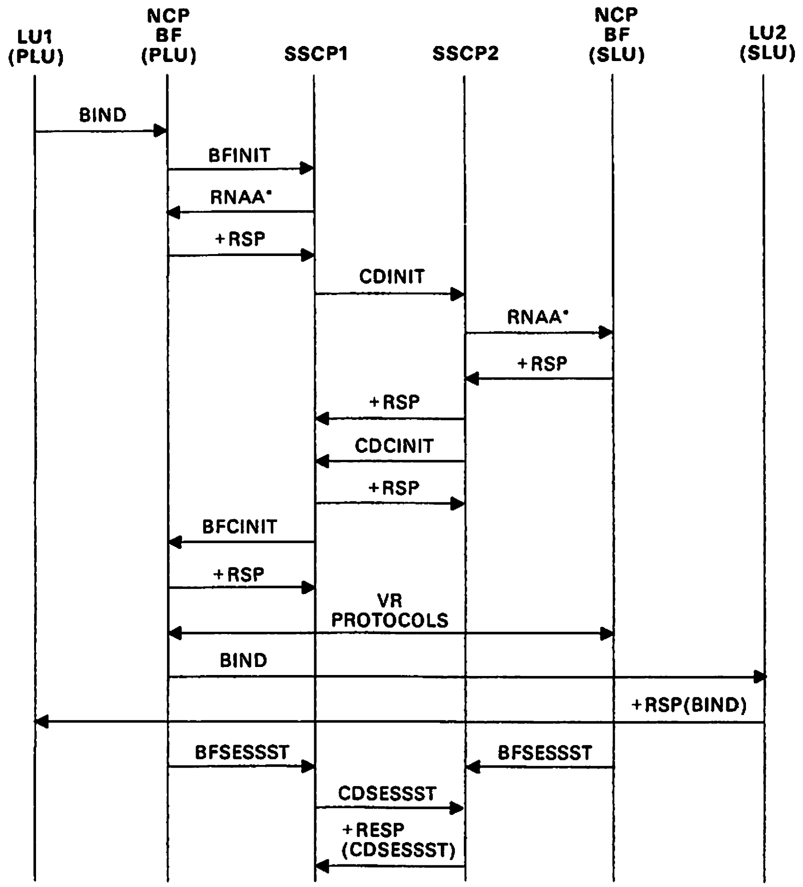


Figure 118. Independent PLU-Initiated Cross-Domain Session Initiation with Independent SLU.

* RNAA flows only if the network address is needed.

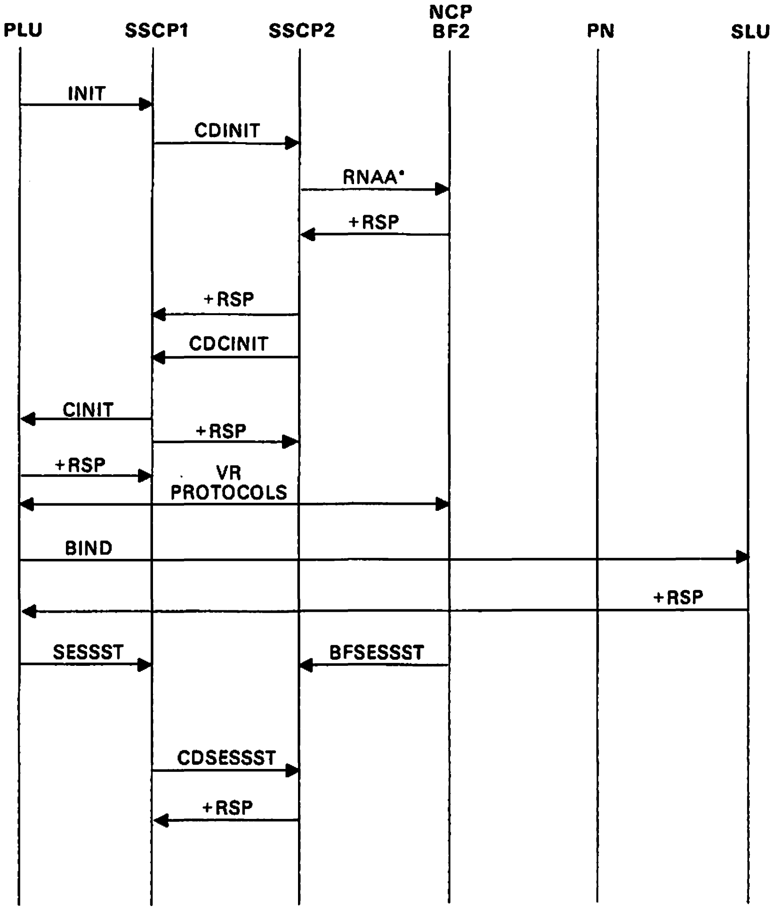


Figure 119. PLU-Initiated Cross-Domain Session Initiation with Independent SLU.

* RNAA flows only if the network address is needed.

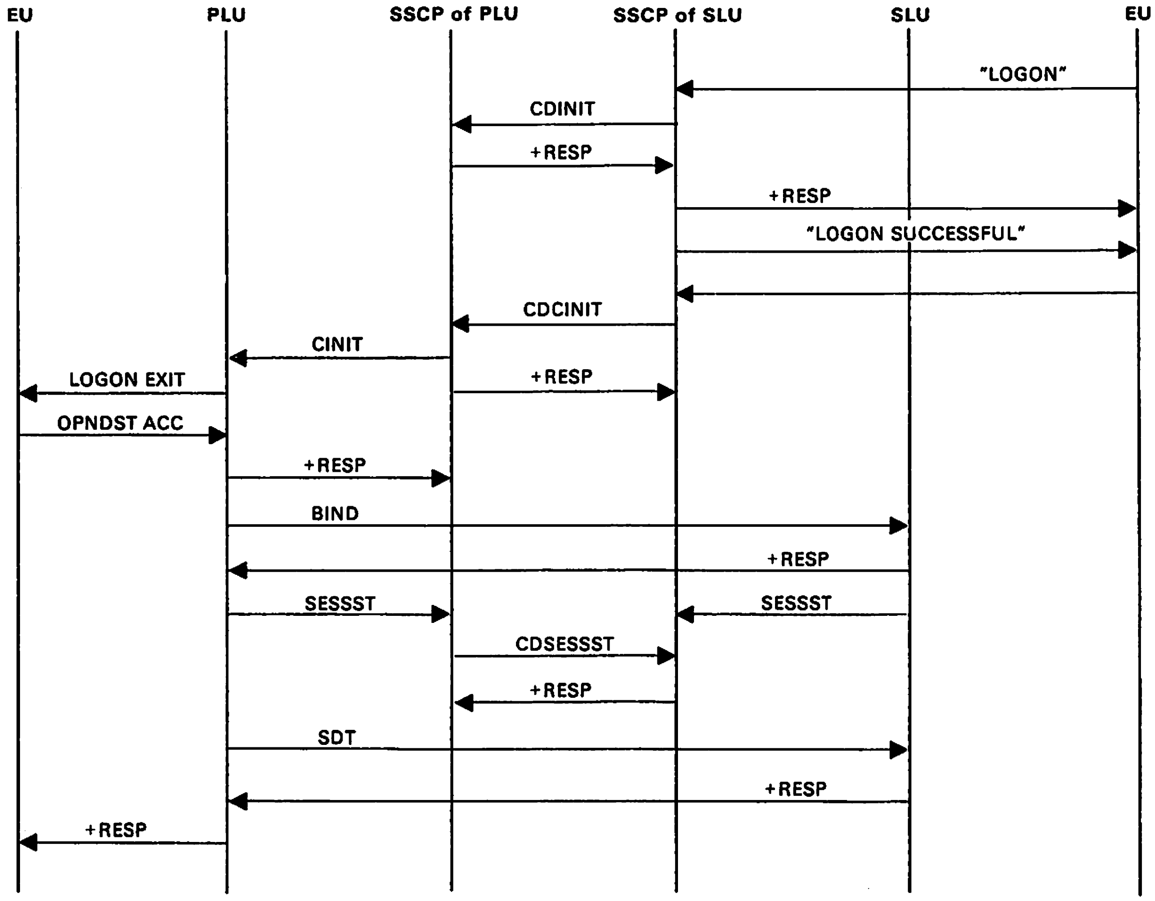


Figure 120. Secondary Logical Unit Initiate (LOGON)

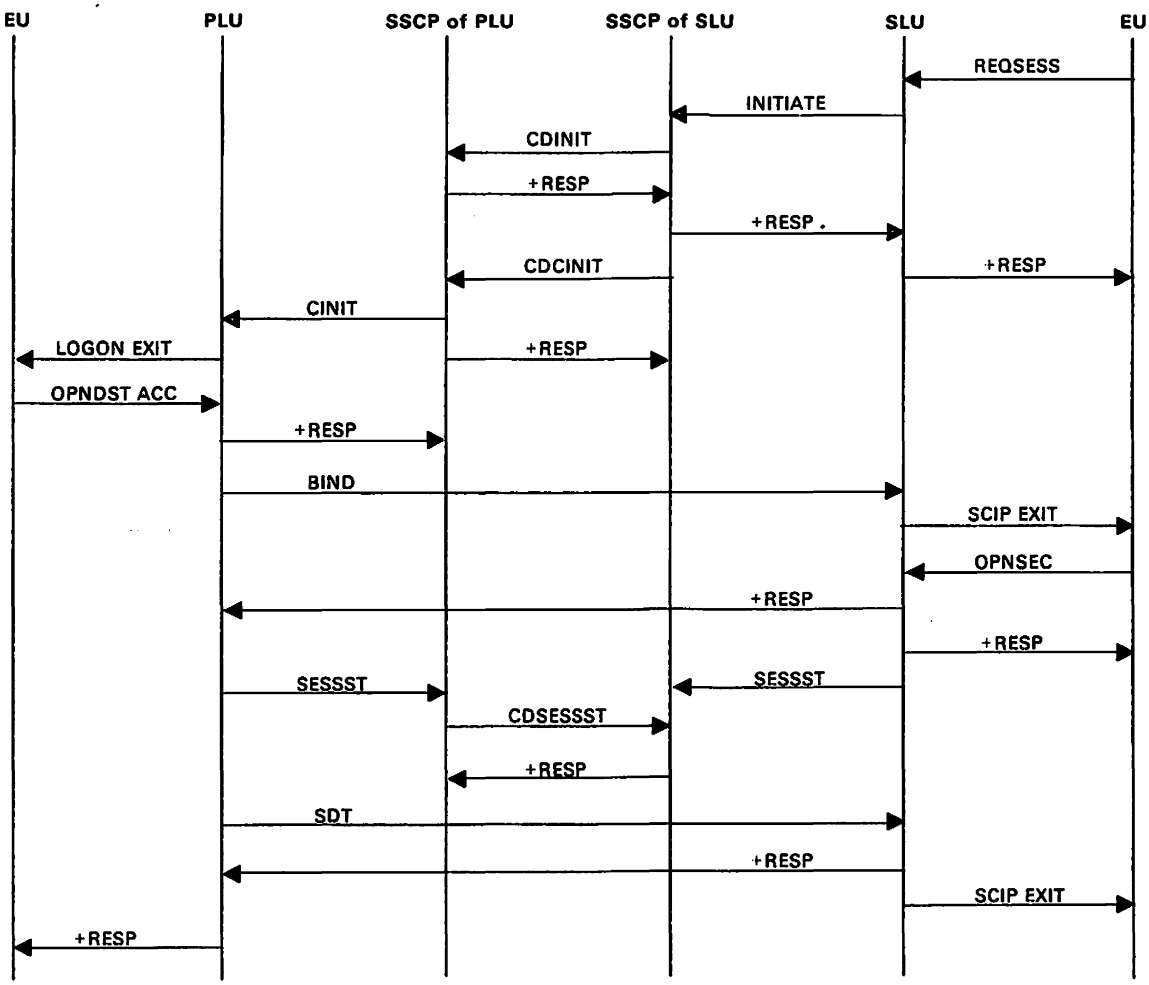


Figure 121. Secondary Logical Unit Initiate (REQSESS)

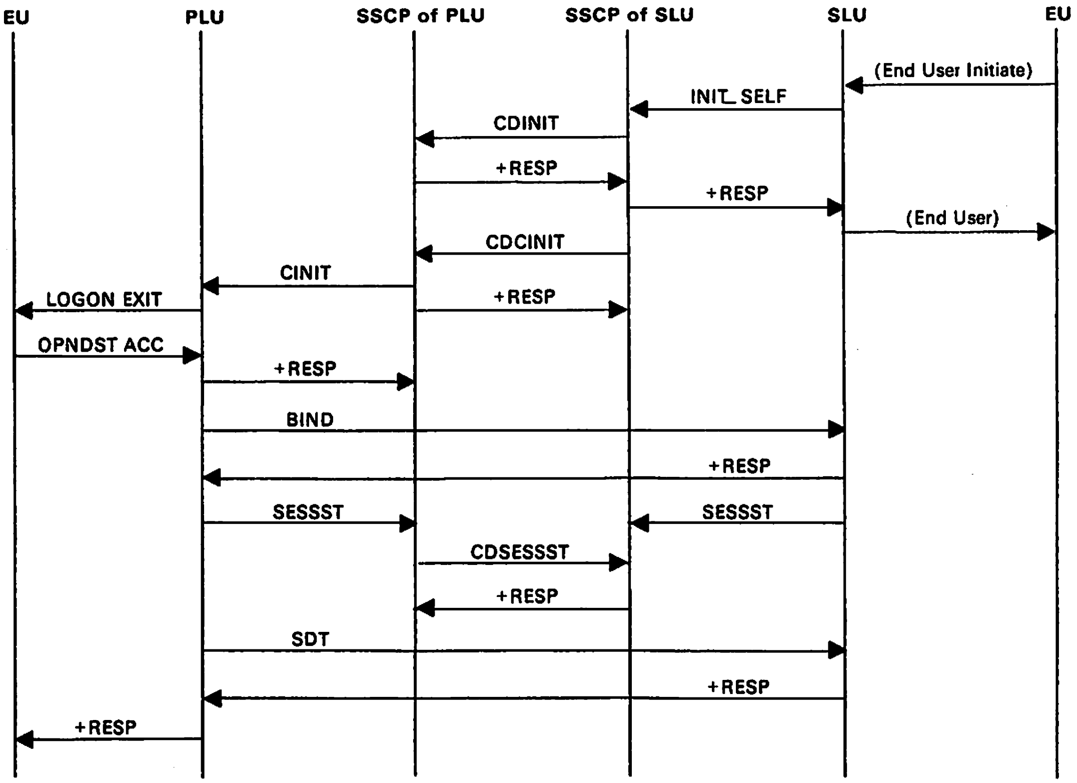


Figure 122. Secondary Logical Unit Initiate (INIT SELF)

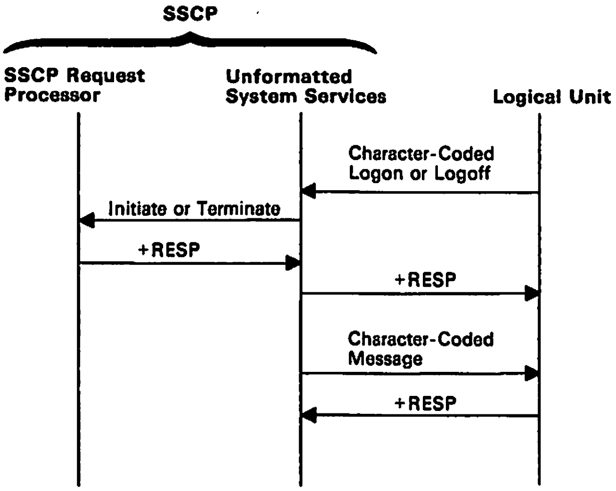


Figure 123. Sending an Unformatted Request to the SSCP

In this example, the logical unit sends a character-coded logon or logoff to the SSCP. The unformatted system services portion of SSCP converts the logon into a field-formatted Initiate Self or Terminate Self request. The request is then passed to the SSCP request processor.

If the return code indicates an unsuccessful transmission, the unformatted system services portion of SSCP converts the request into a form that can be understood by the terminal logical unit.

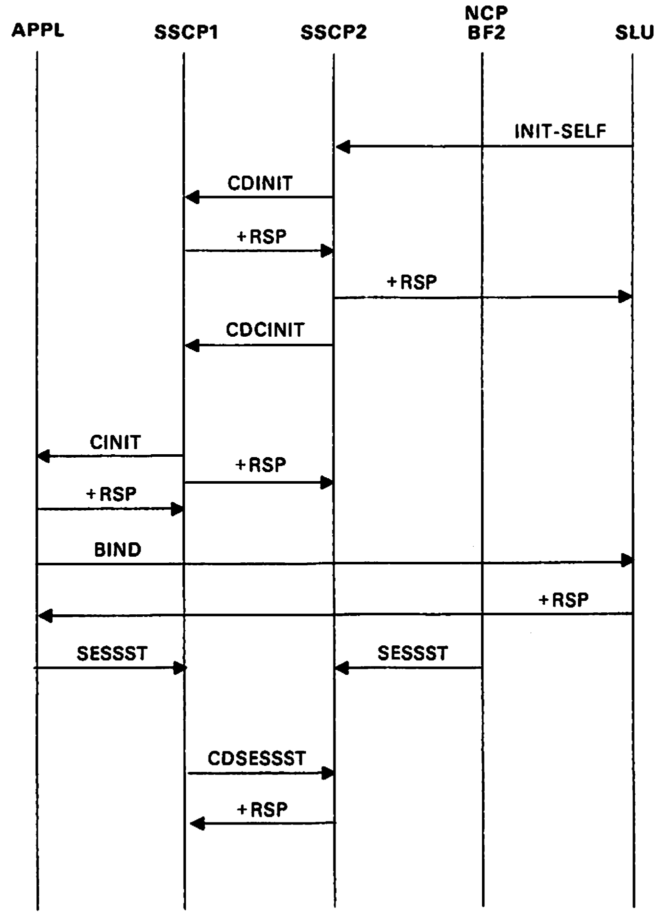


Figure 124. Dependent SLU-Initiated Cross-Domain Session Initiation with Application LU

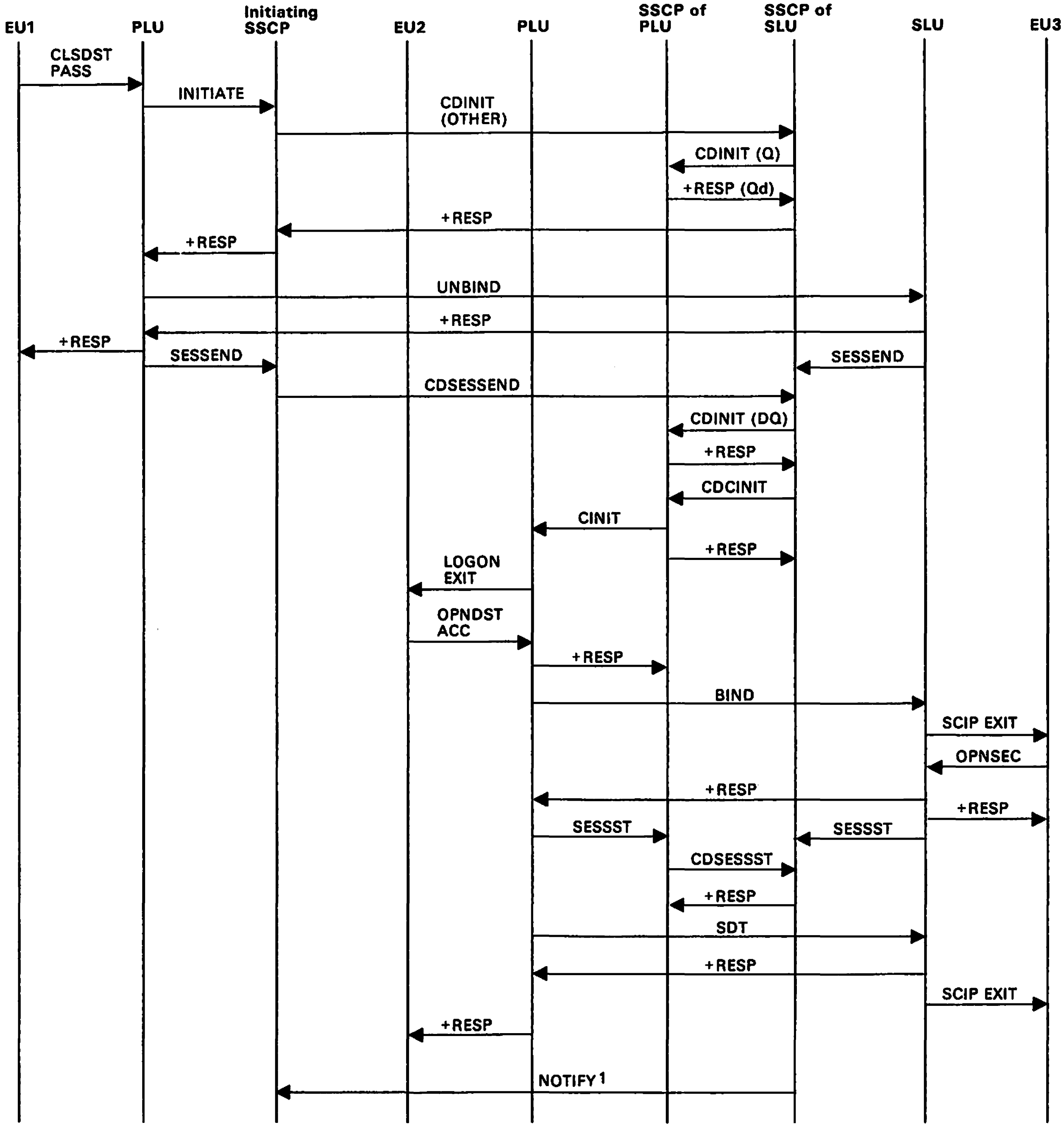


Figure 125. Third Party Initiate (CLSDST PASS).

[1] V3R2 only

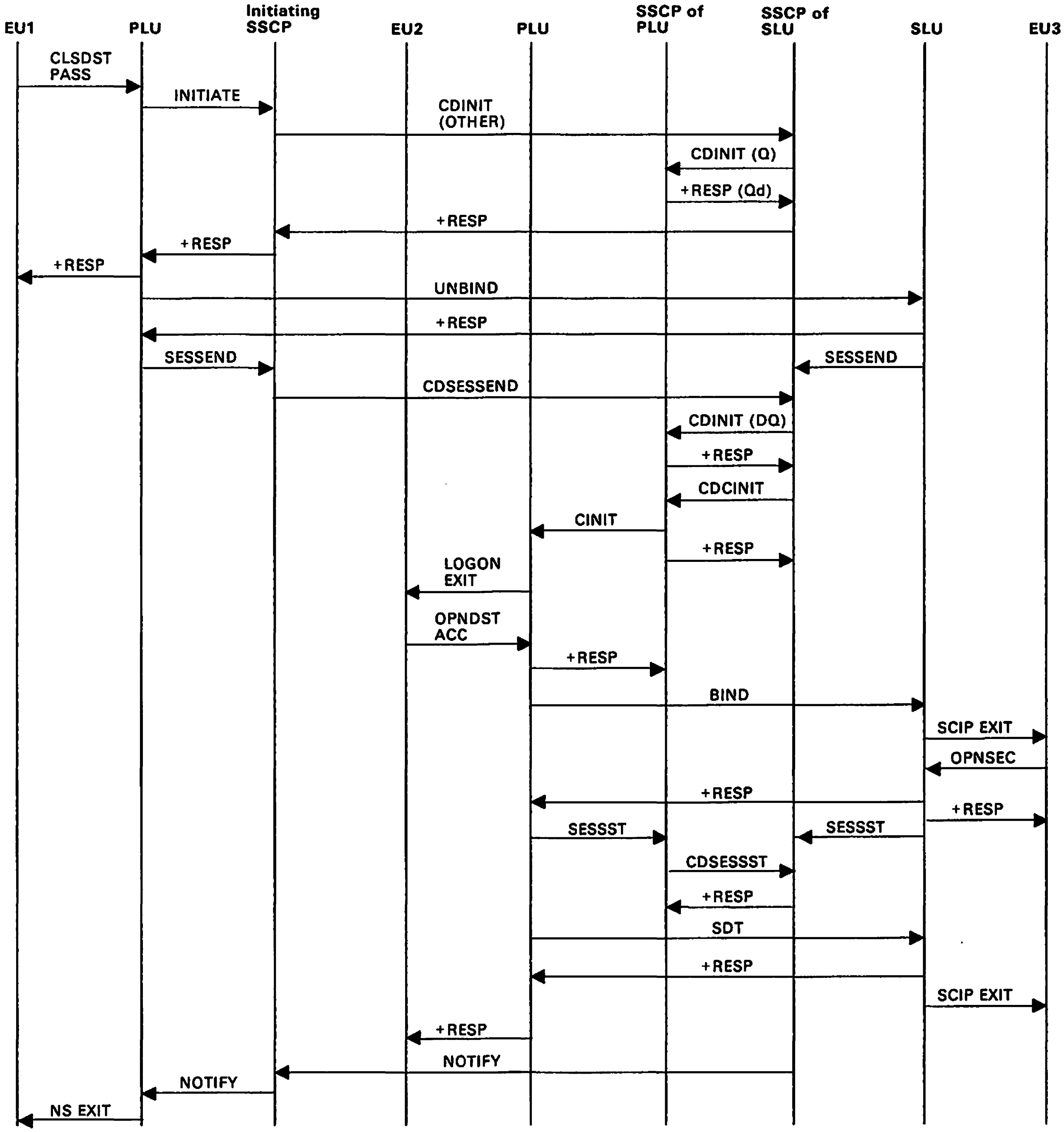


Figure 126. Third Party Initiate (CLSDST PASS with NOTIFY)

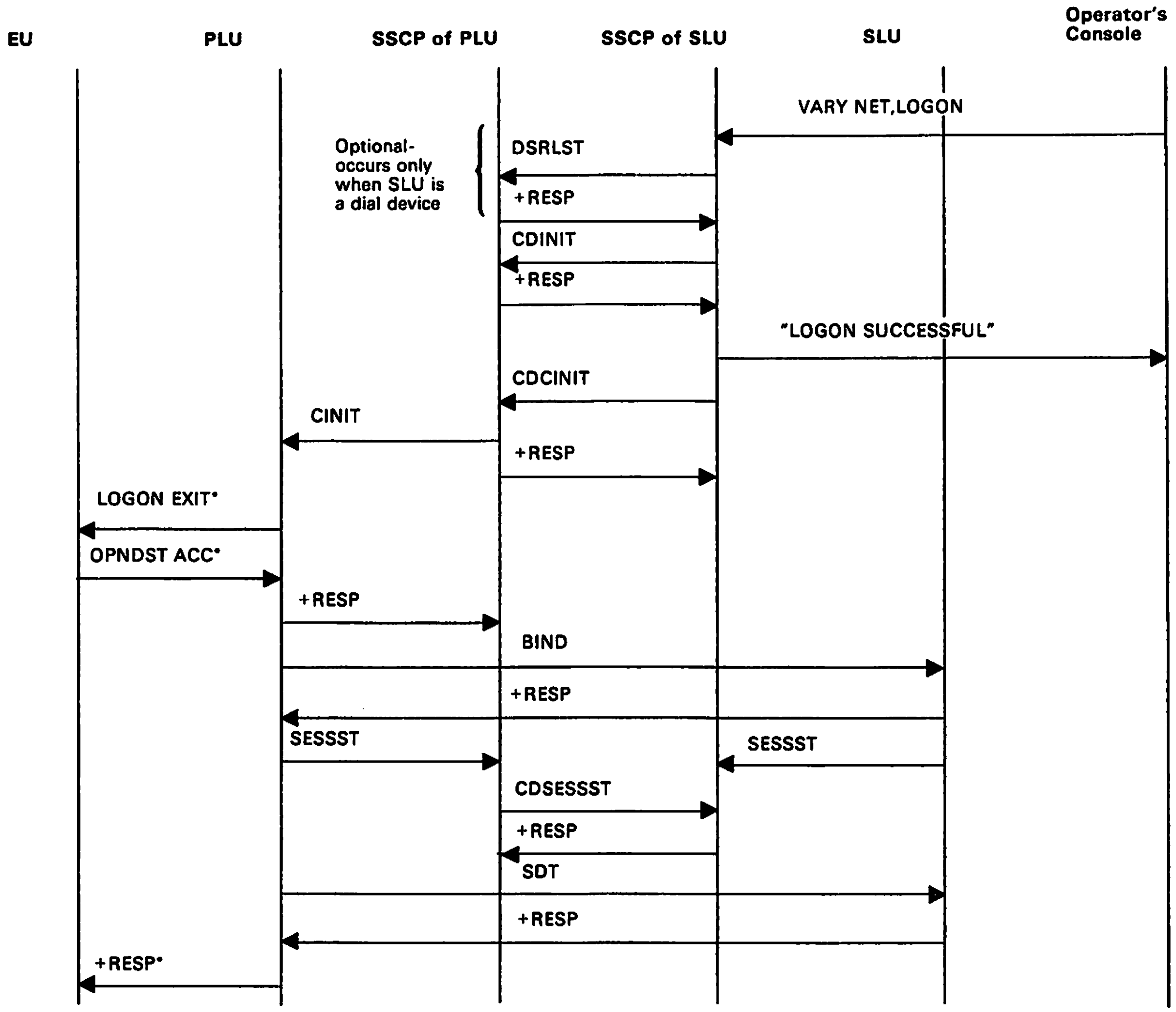


Figure 127. VARY NET,LOGON or LOGAPPL.

* This applies only when the PLU is associated with an application program. It does not appear in the flow if the PLU is a device-type logical unit.

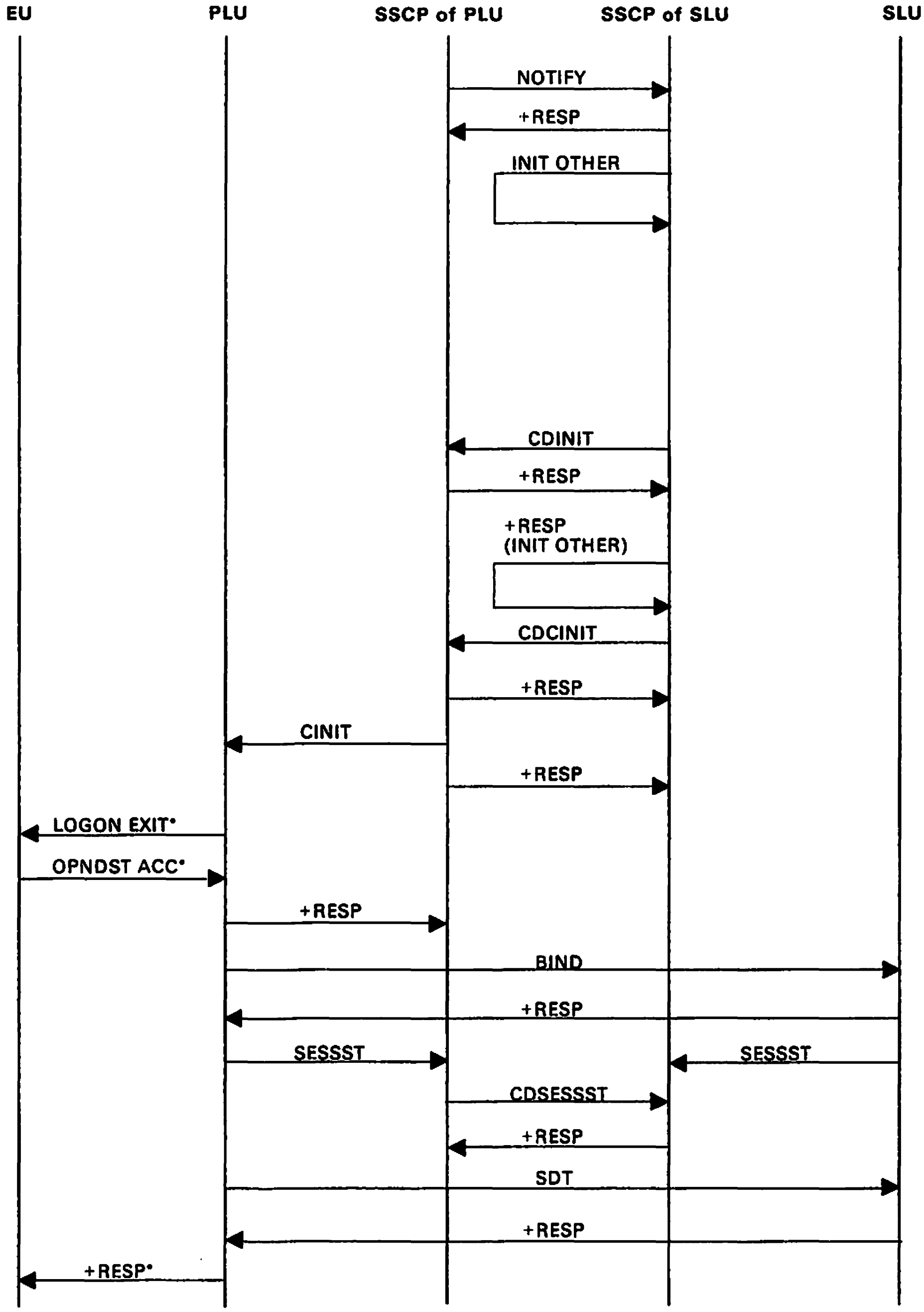


Figure 128. Notification of PLU Availability.

* This applies only when the PLU is associated with an application program. It does not appear in the flow if the PLU is a device-type logical unit.

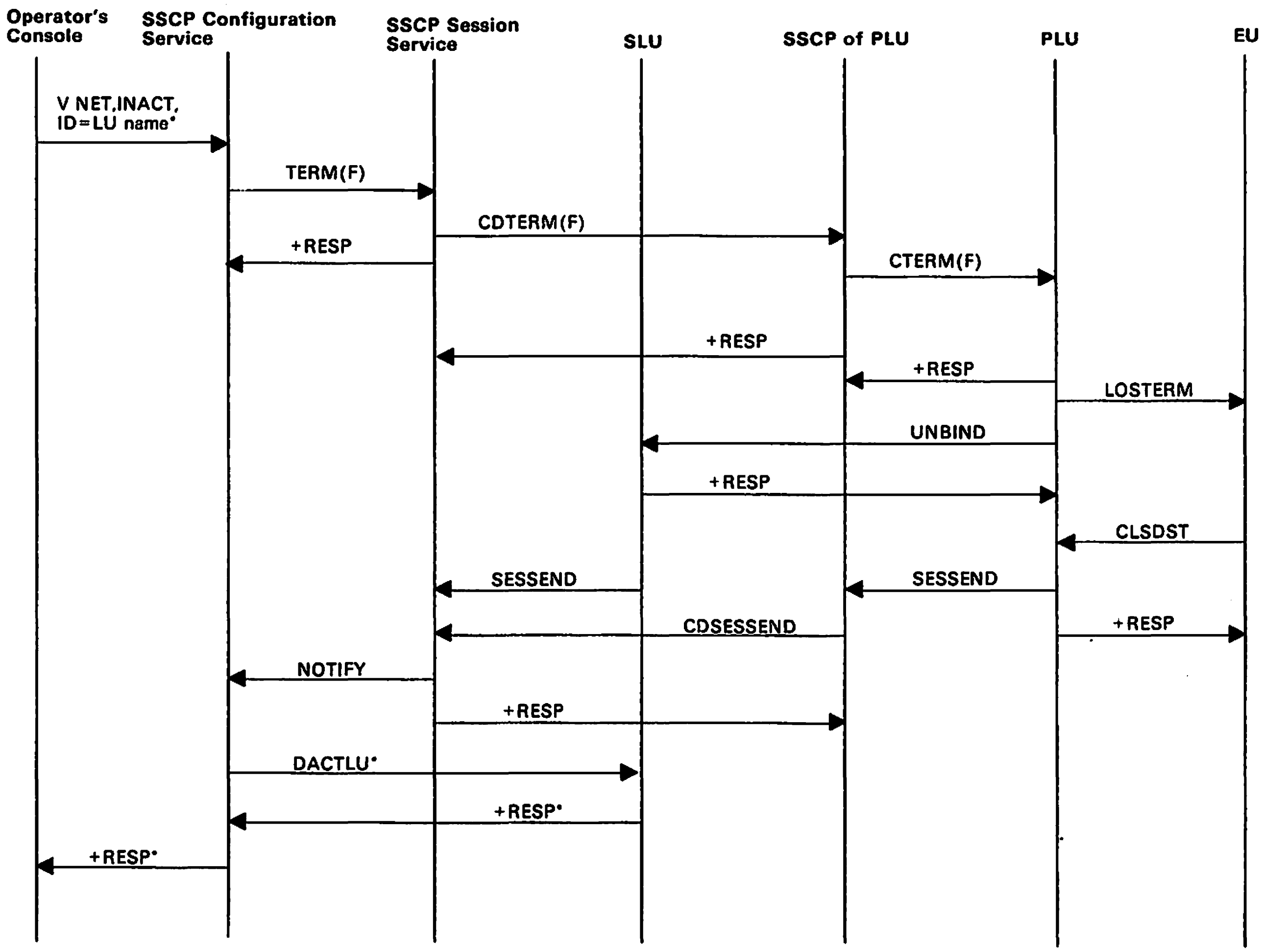


Figure 129. Deactivating a Logical Unit (Immediate).

* These flow only when the operator activates a specific logical unit. For example, they do not flow during immediate deactivation of a CDRM.

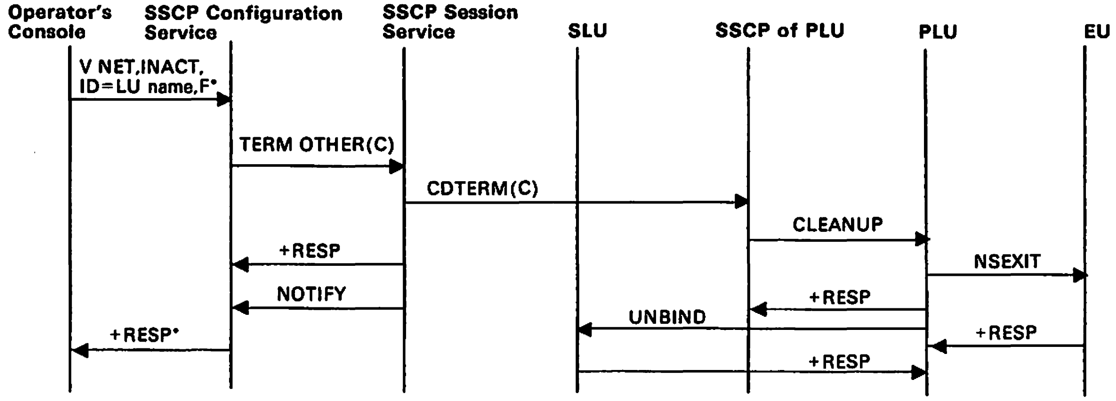


Figure 130. Deactivating a Logical Unit (Forced).

* These flow only when the operator deactivates a specific logical unit. For example, they do not flow during forced deactivation of a CDRM.

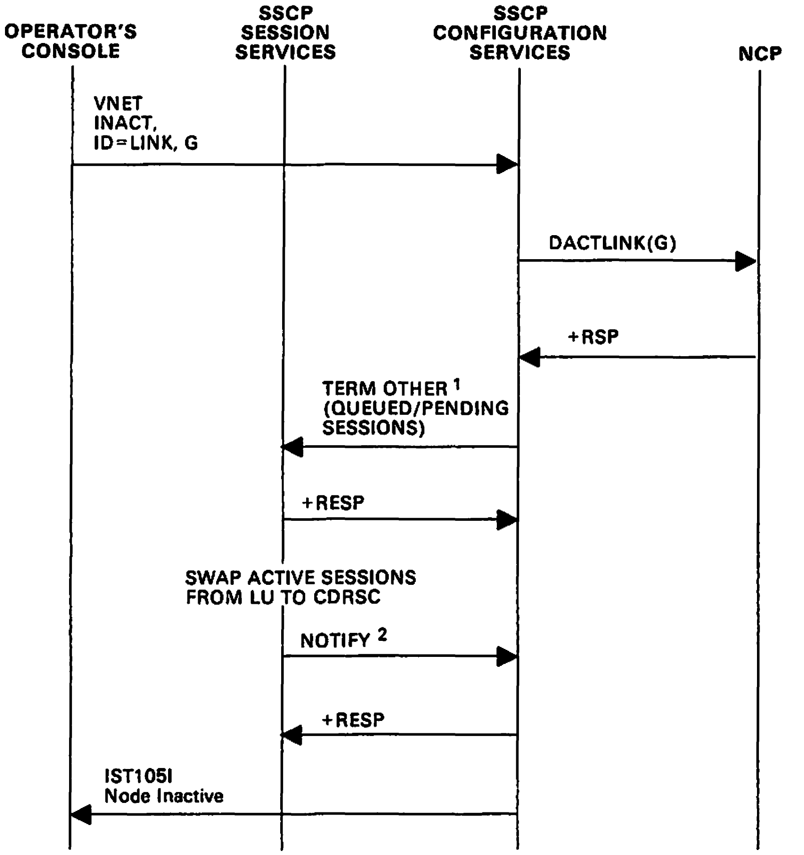
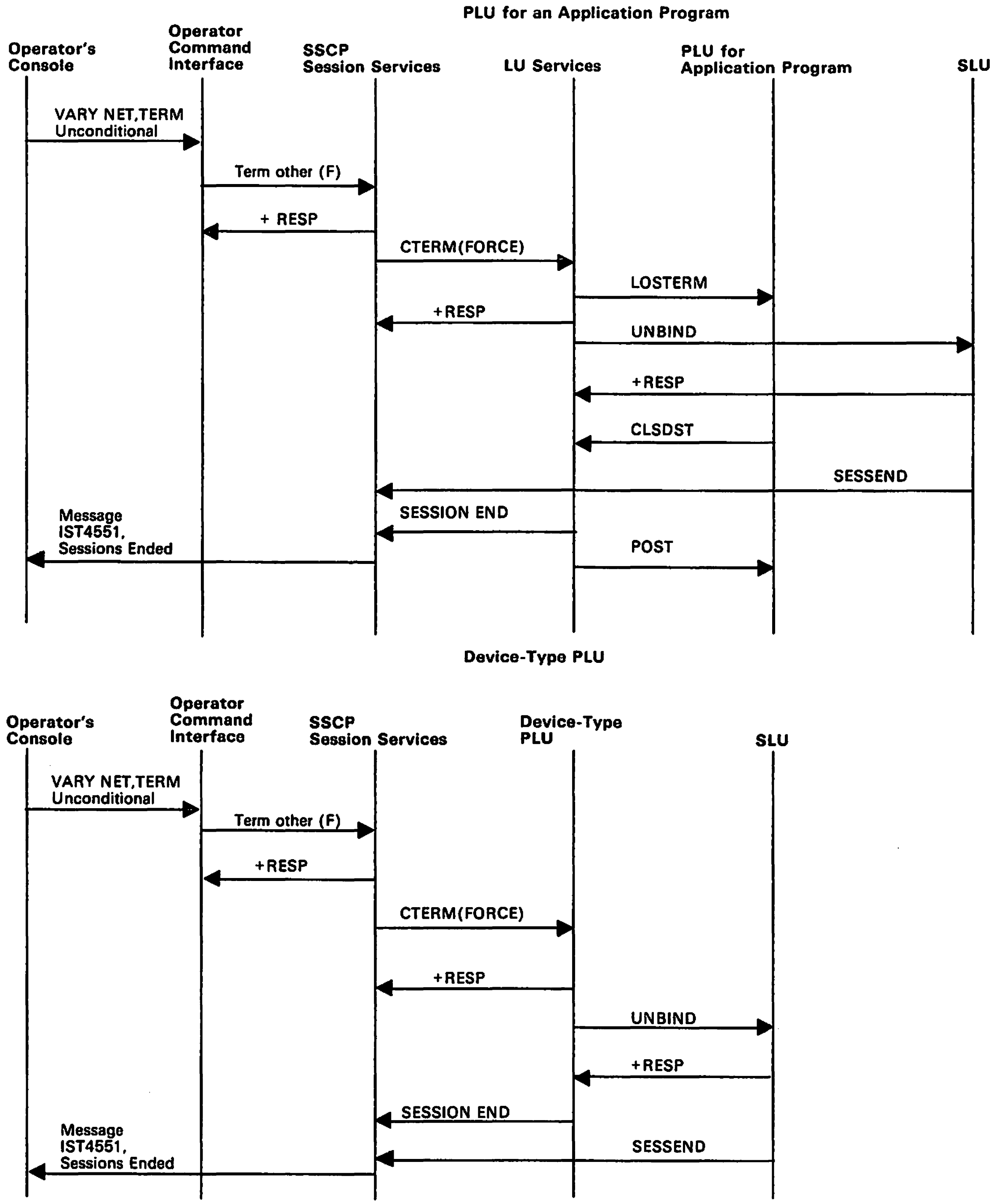


Figure 131. Deactivating a Logical Unit with Giveback.

[1] The DACTLINK X'02' disconnects only queued and pending LU-LU sessions. Active LU-LU sessions remain active.

[2] After session services transfers SIBs of ACTIVE logical units to the CDRSC, configuration services SRT ADDs the CDRSCs as real resources, and the logical units are ADDed as shadow resources. If a CDRSC for a particular logical unit does not exist, a dynamic CDRSC is allocated for the logical unit.



PLU for an Application Program

Device-Type PLU

Figure 132. VARY NET,TERM Unconditional

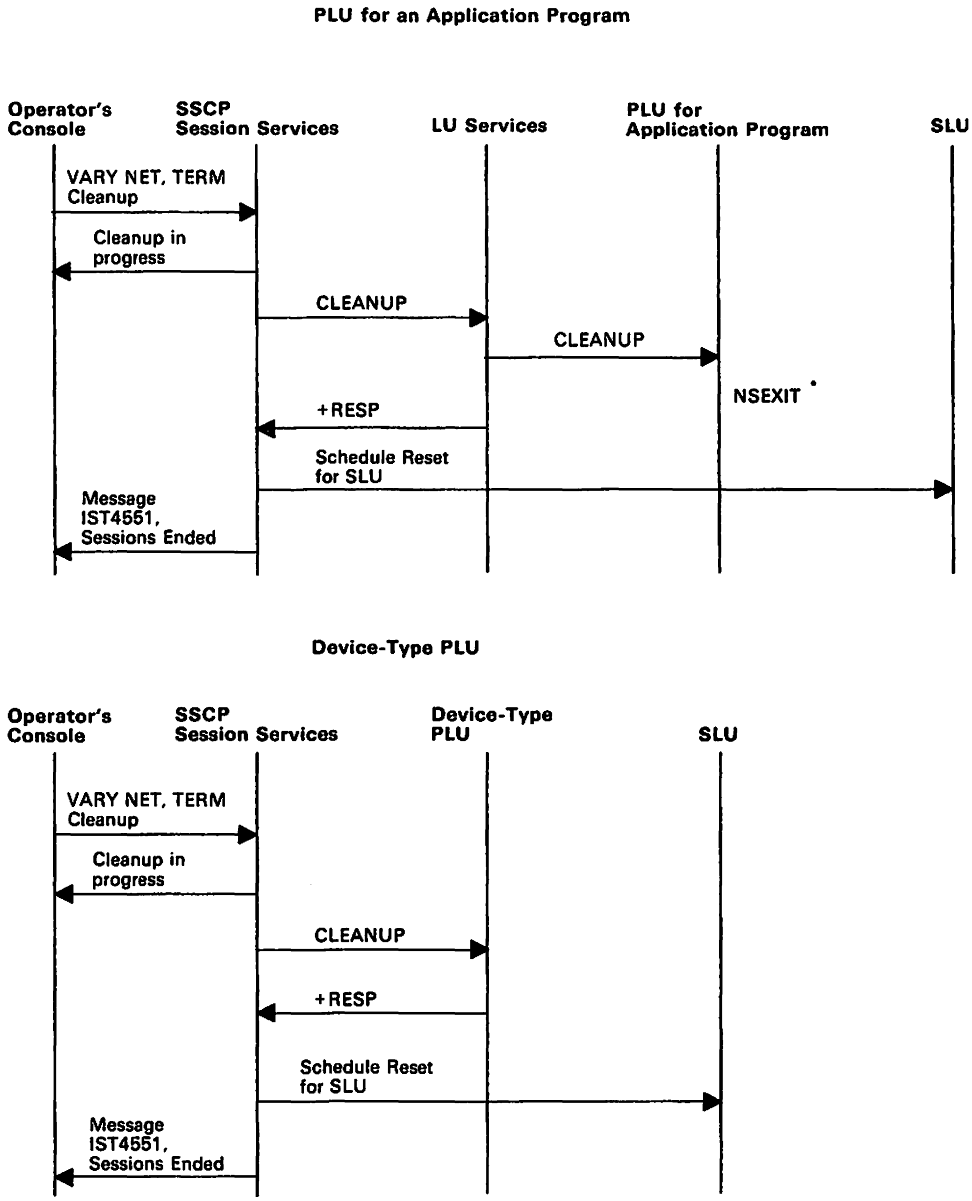


Figure 133. VARY NET,TERM Cleanup

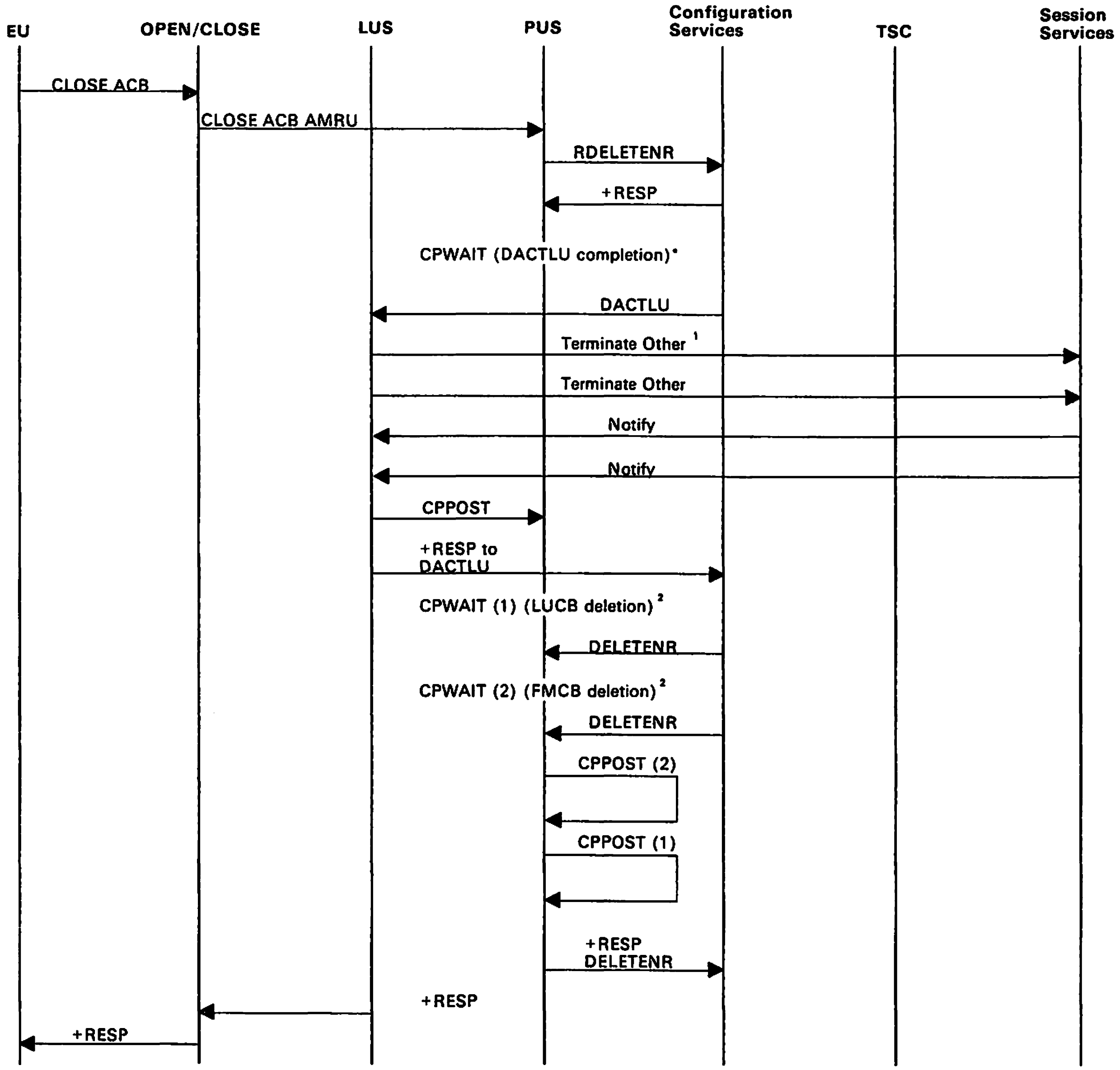


Figure 134. CLOSE ACB Processing.

[1] The repetition of flows shown here indicates multiple sessions.

[2] PUNS cannot send a response to the CLOSE ACB AMRU until DACTLU processing is complete and the LUCB and FMCB for the application program have been deleted. Therefore, after requesting that configuration services deactivate the logical unit, PUNS issues CPWAIT and waits for LUS to post it when the logical unit has been deactivated. Once it is posted, PUNS waits to be notified that there are no more active sessions for the application program. It issues CPWAIT (1) and waits for configuration services to send a request to delete the FMCB. PUNS posts itself when it has processed each of these requests and sends a response to configuration services to notify it that the LUCB has been deleted. After sending this response, PUNS sends a response to the CLOSE ACB AMRU.

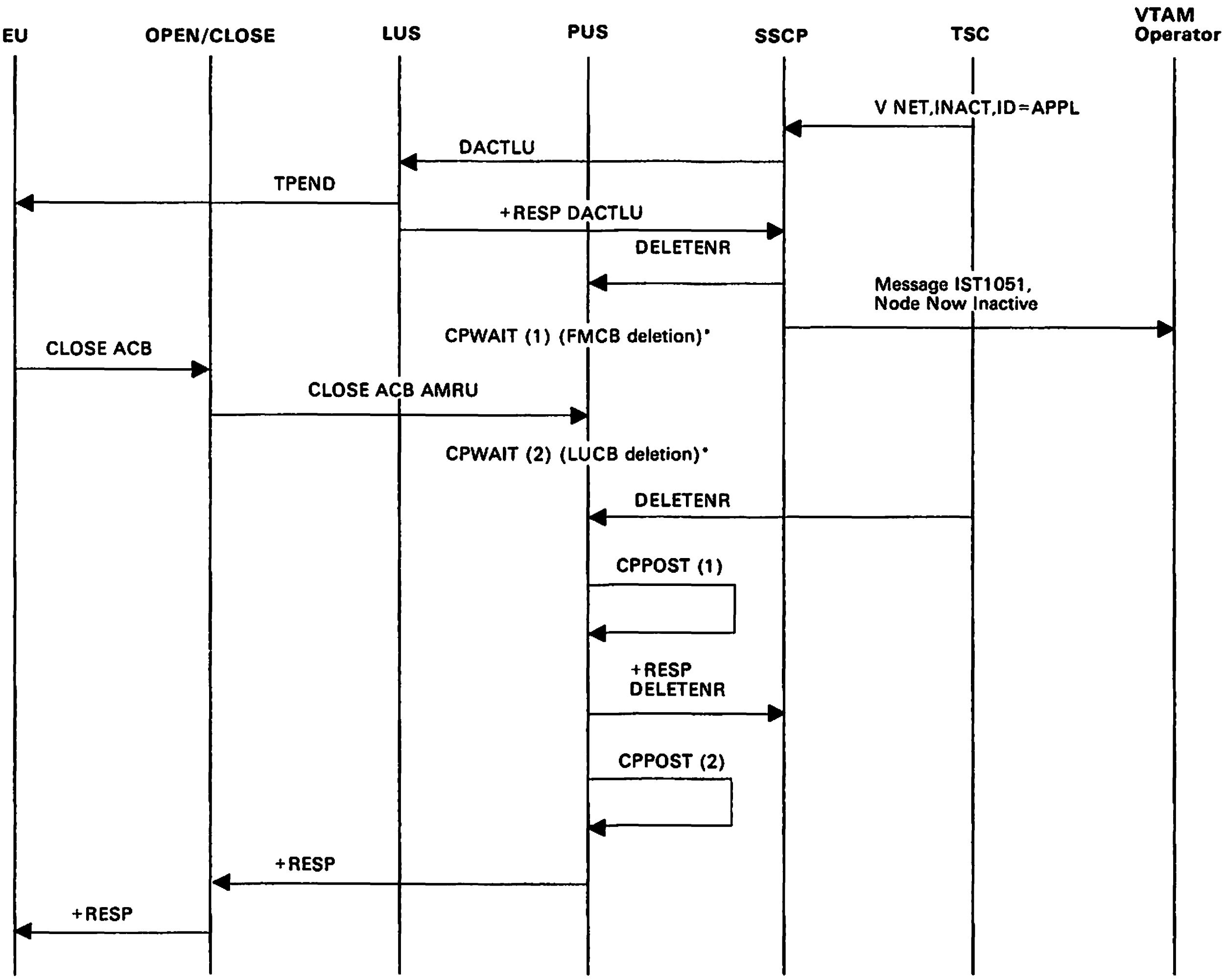


Figure 135. Deactivating an Application Program.

* PUNS cannot send a response to the CLOSE ACB AMRU until there are no more active sessions for the application program. Therefore, after the SSCP sends a request to delete the FMCB, PUNS issues CPWAIT (1) and waits for OPEN/CLOSE to send it a CLOSE ACB AMRU. Once it has received this request, PUNS issues CPWAIT (2) and waits for TSC to send a request to delete the LUCB. PUNS posts itself when it has deleted the FMCB and sends a response to configuration services, then posts itself when it has deleted the LUCB. After posting itself the second time, PUNS sends a response to the CLOSE ACB AMRU.

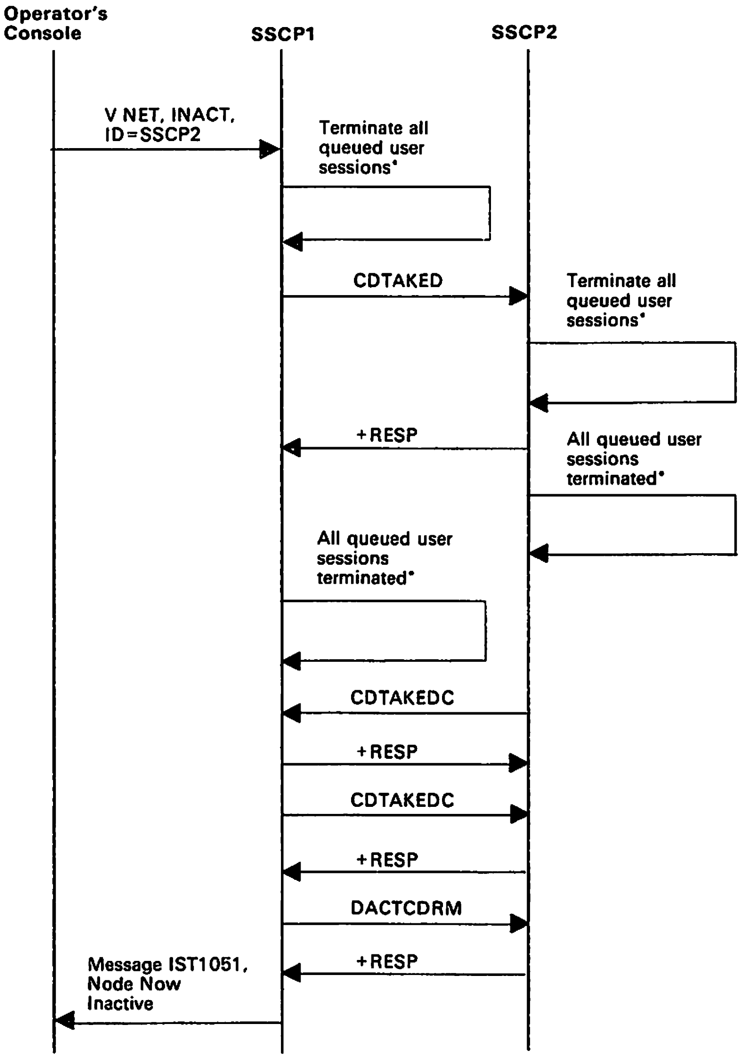

Operator's Console
SSCP1
SSCP2
V NET, INACT, ID=SSCP2
Terminate all queued user sessions*
CDTAKED
Terminate all queued user sessions*
+RESP
All queued user sessions terminated*
All queued user sessions terminated*
CDTAKEDC
+RESP
CDTAKEDC
+RESP
DACTCDRM
Message IST1051, Node Now Inactive
+RESP

Figure 136. Deactivating a CDRM - Normal.

* Session services purges the queues of user session requests. Since this processing is internal to session services, no RUs flow between components.

Note: When the CDRM is actually deactivated, immediate processing takes place. See Figure 137 on page 603 for the RUs that flow for immediate deactivation of a CDRM.

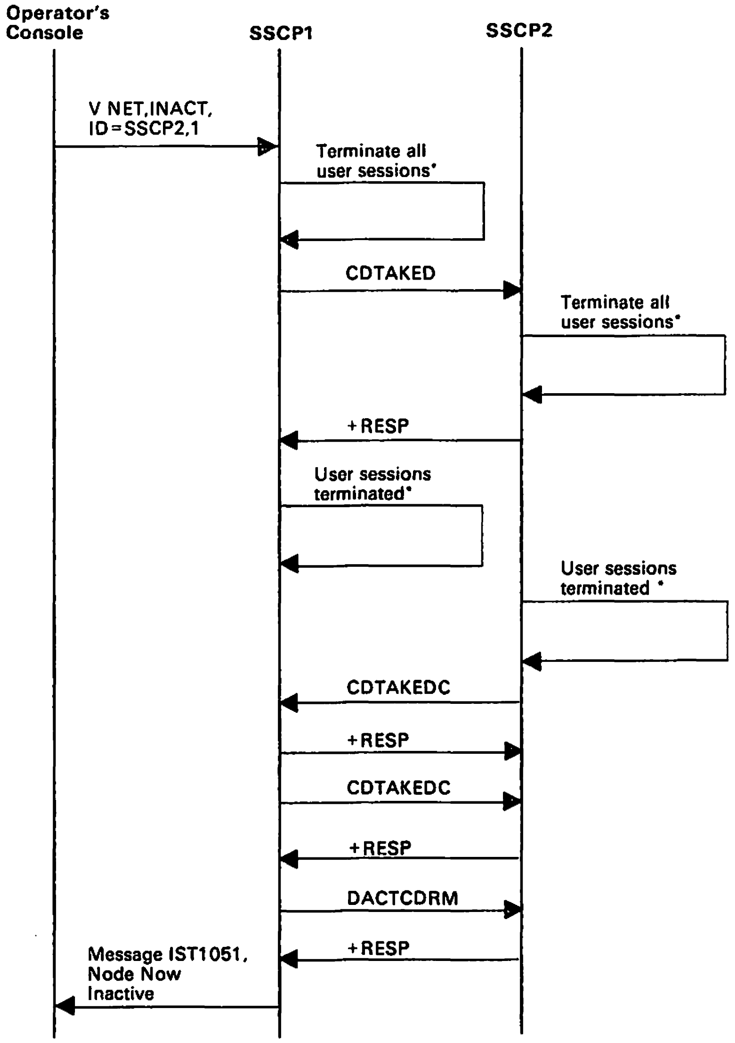

Operator's Console
SSCP1
SSCP2
V NET,INACT, ID=SSCP2,1
Terminate all user sessions*
CDTAKED
Terminate all user sessions*
+RESP
User sessions terminated*
User sessions terminated *
CDTAKEDC
+RESP
CDTAKEDC
+RESP
DACTCDRM
+RESP
Message IST1051, Node Now Inactive

Figure 137. Deactivating a CDRM - Immediate.

* See Figure 129 on page 596 for deactivate immediate processing for a logical unit.

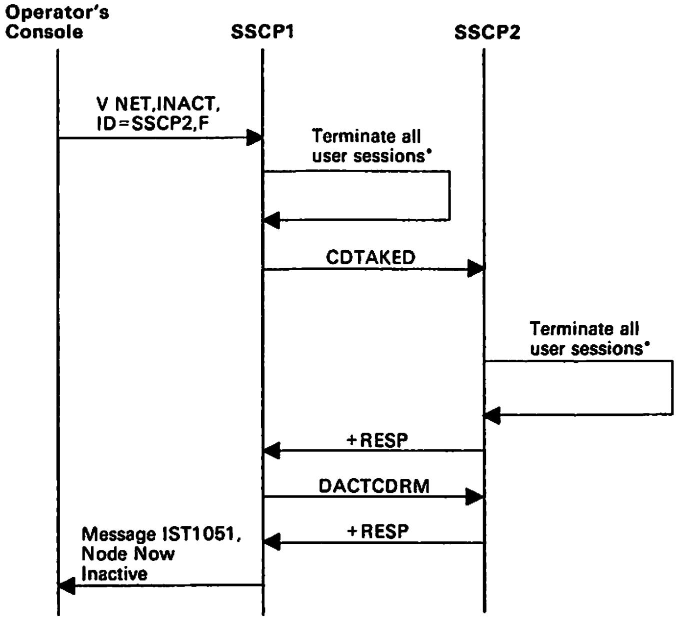

Operator's Console
SSCP1
SSCP2
V NET,INACT, ID=SSCP2,F
Terminate all user sessions*
CDTAKED
Terminate all user sessions*
+RESP
DACTCDRM
+RESP
Message IST1051, Node Now Inactive

Figure 138. Deactivating a CDRM - Forced.

* See Figure 130 on page 596 for deactivate (force) processing for a logical unit.

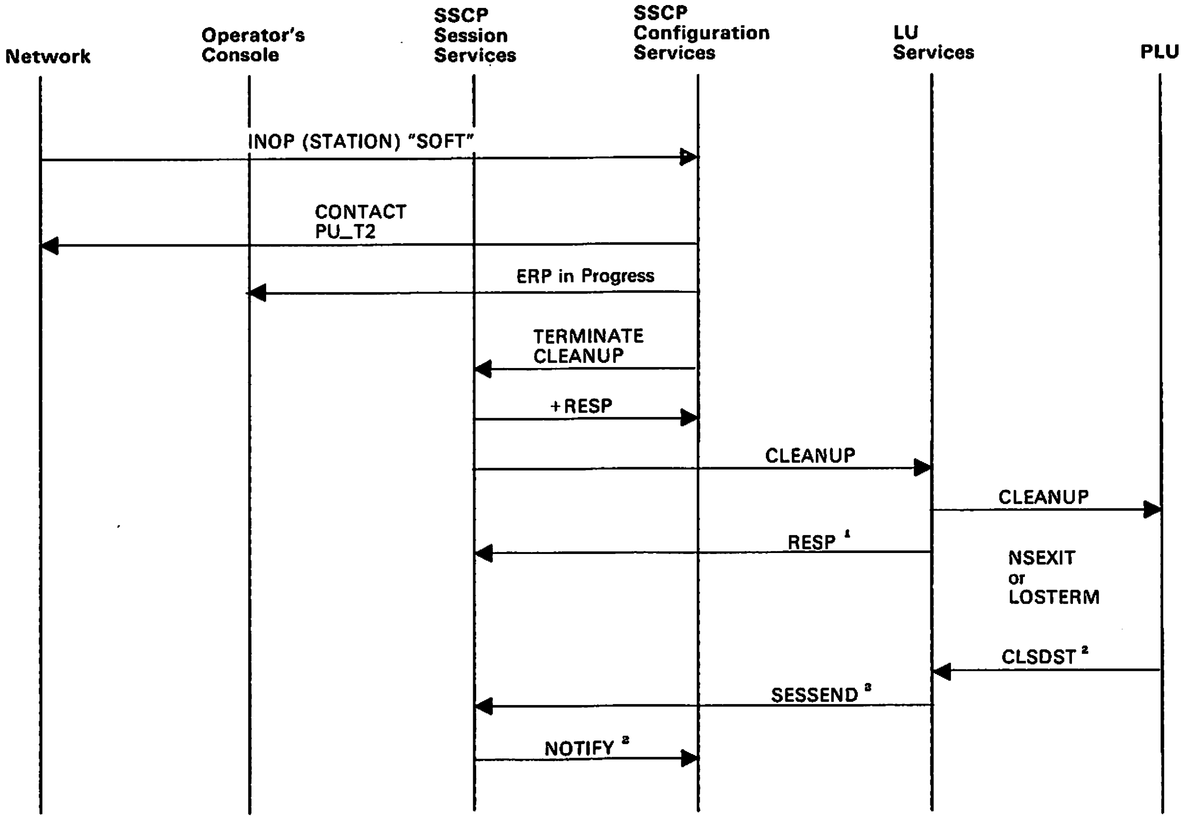


Figure 139. Error Recovery Processing - Soft INOP.

[1] If the NSEXIT exit routine is scheduled, LUS cleans up the session and sends a positive response to the cleanup request. If the LOSTERM exit routine is scheduled, LUS *does not* clean up the session, and it sends a negative response to the cleanup request.

[2] These flow only if the LOSTERM exit routine is scheduled.

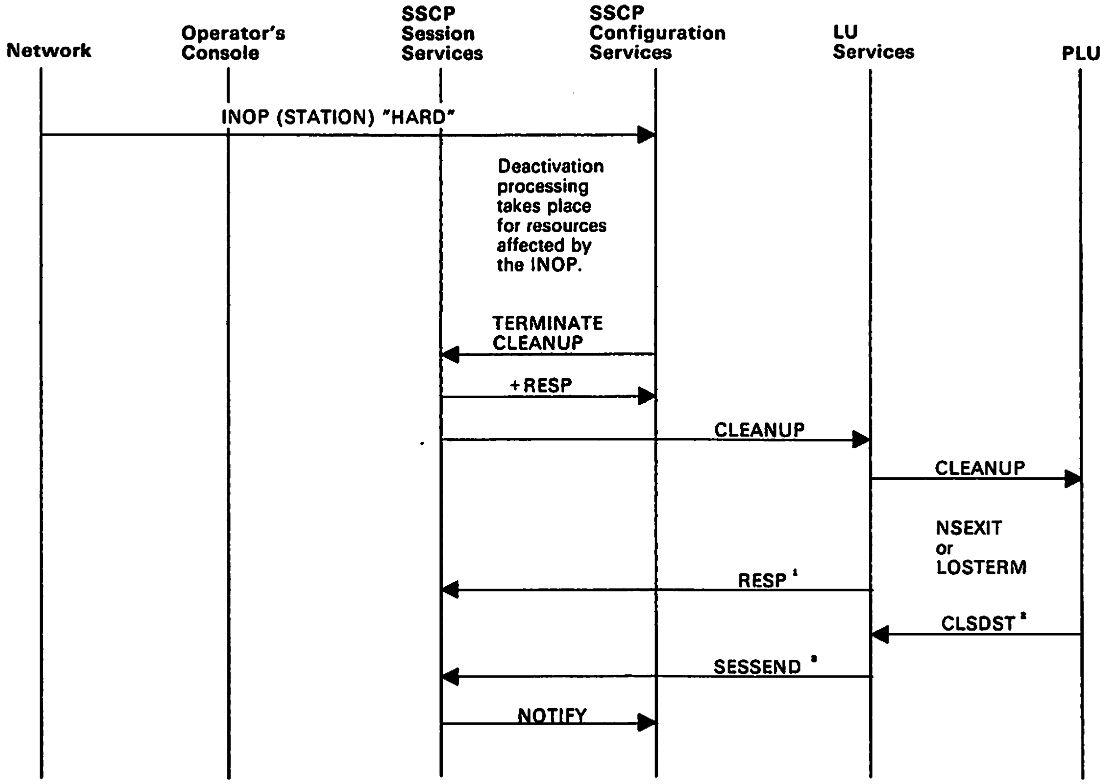


Figure 140. Error Recovery Processing - Hard INOP.

[1] If the NSEXIT exit routine is scheduled, LUS cleans up the session and sends a positive response to the cleanup request. If the LOSTERM exit routine is scheduled, LUS *does not* clean up the session, and it sends a negative response to the cleanup request.

[2] These flow only if the LOSTERM exit routine is scheduled.

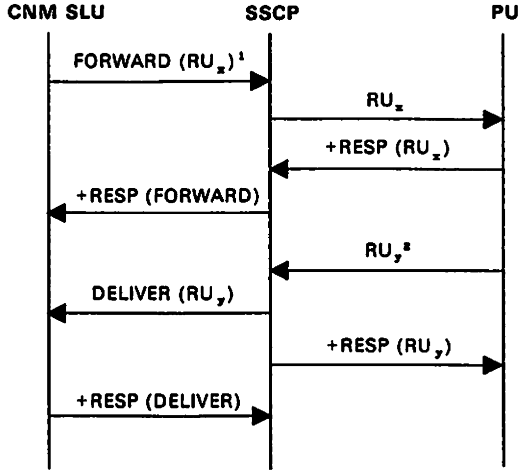


Figure 141. FORWARD and DELIVER Routing.

[1] RUx is a maintenance service RU.

[2] Either RUy contains data in reply to RUx, or it is an unsolicited RU.

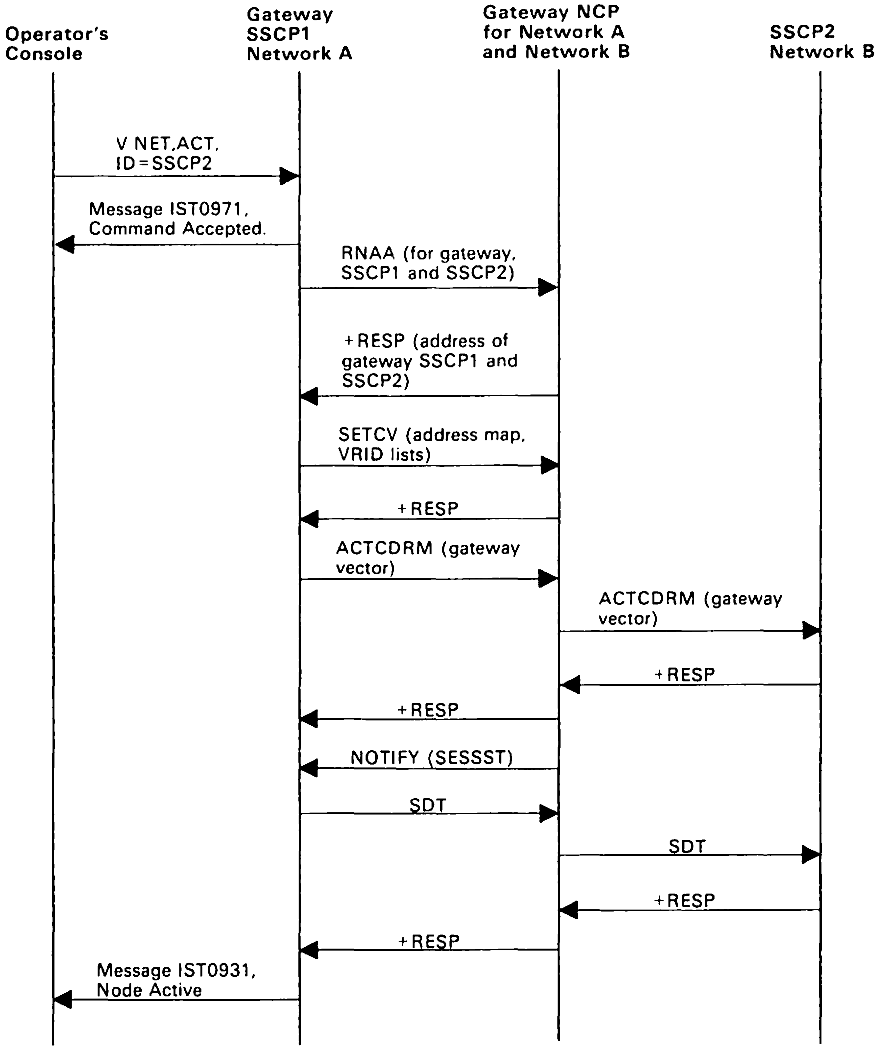

Operator's Console
Gateway SSCP1 Network A
Gateway NCP for Network A and Network B
SSCP2 Network B
V NET,ACT, ID=SSCP2
Message IST0971, Command Accepted.
RNAA (for gateway, SSCP1 and SSCP2)
+RESP (address of gateway SSCP1 and SSCP2)
SETCV (address map, VRID lists)
+RESP
ACTCDRM (gateway vector)
ACTCDRM (gateway vector)
+RESP
+RESP
NOTIFY (SESSST)
SDT
SDT
+RESP
+RESP
Message IST0931, Node Active

Figure 142. MVS VM Activating a Cross-Network SSCP-SSCP Session: GWSSCP Requests Session

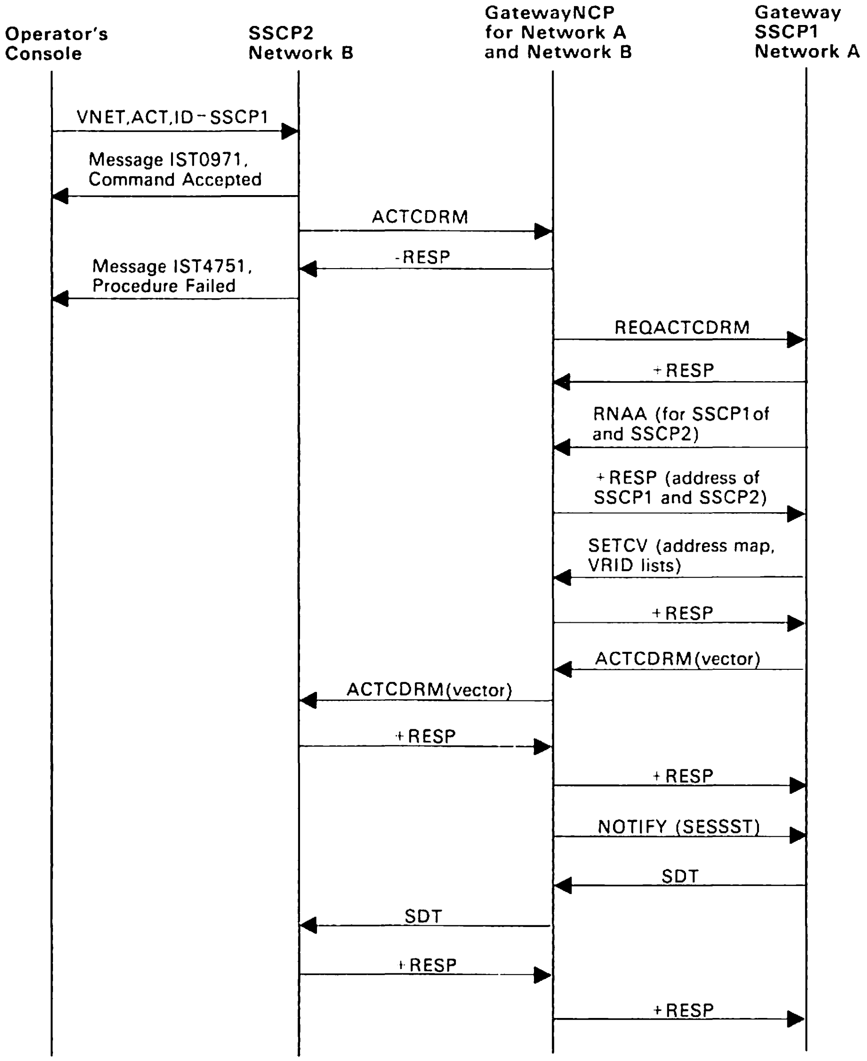

Operator's Console
SSCP2 Network B
GatewayNCP for Network A and Network B
Gateway SSCP1 Network A
VNET,ACT,ID-SSCP1
Message IST0971, Command Accepted
ACTCDRM
-RESP
Message IST4751, Procedure Failed
REQACTCDRM
+RESP
RNAA (for SSCP1of and SSCP2)
+RESP (address of SSCP1 and SSCP2)
SETCV (address map, VRID lists)
+RESP
ACTCDRM(vector)
ACTCDRM(vector)
+RESP
+RESP
NOTIFY (SESSST)
SDT
SDT
+RESP
+RESP

Figure 143. MVS VM Activating a Cross-Network SSCP-SSCP Session: Non-GWSSCP Requests Session

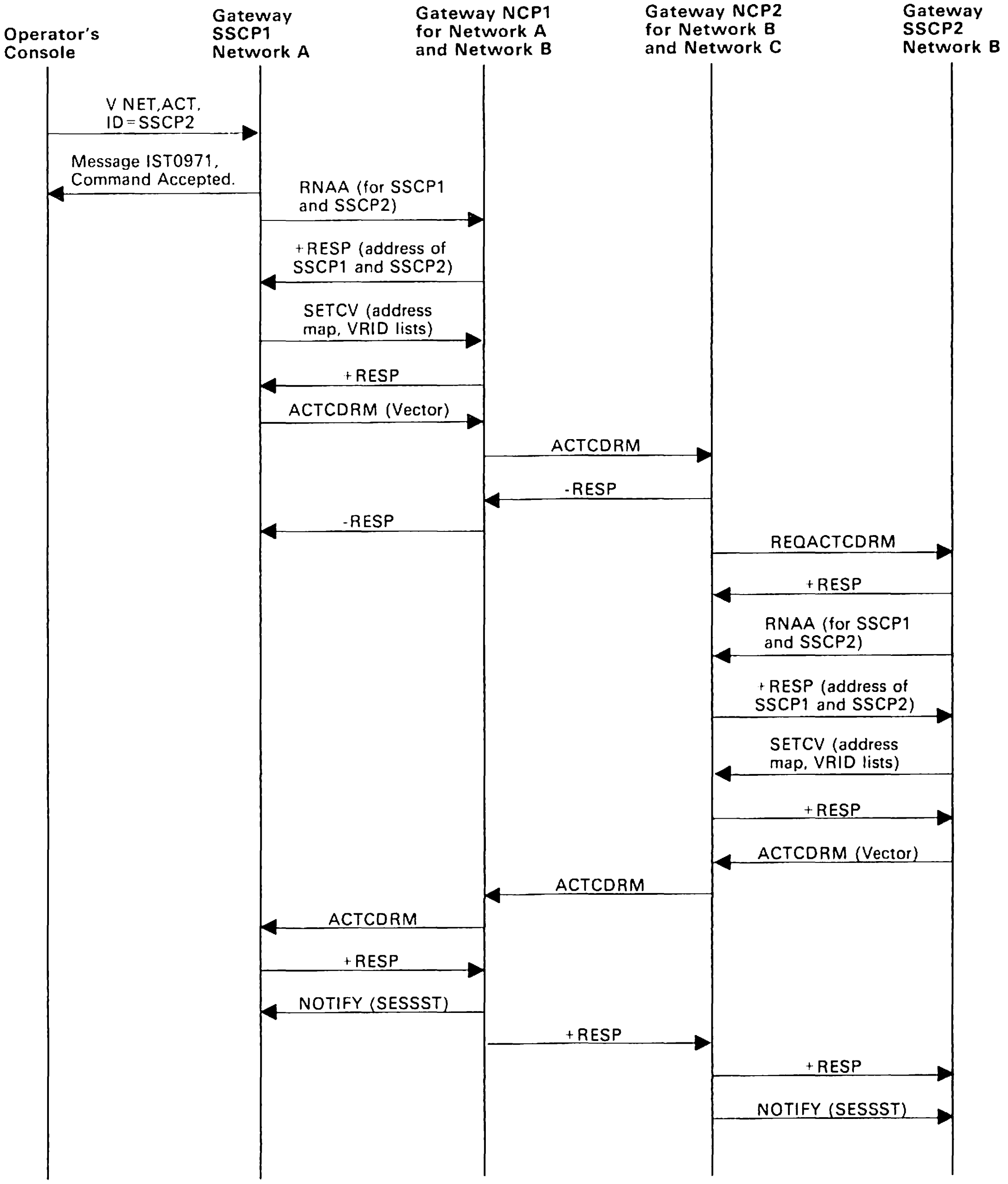

Operator's Console
Gateway SSCP1 Network A
Gateway NCP1 for Network A and Network B
Gateway NCP2 for Network B and Network C
Gateway SSCP2 Network B
V NET,ACT, ID=SSCP2
Message IST0971, Command Accepted.
RNAA (for SSCP1 and SSCP2)
+RESP (address of SSCP1 and SSCP2)
SETCV (address map, VRID lists)
+RESP
ACTCDRM (Vector)
ACTCDRM
-RESP
-RESP
REQACTCDRM
+RESP
RNAA (for SSCP1 and SSCP2)
+RESP (address of SSCP1 and SSCP2)
SETCV (address map, VRID lists)
+RESP
ACTCDRM (Vector)
ACTCDRM
ACTCDRM
+RESP
NOTIFY (SESSST)
+RESP
+RESP
NOTIFY (SESSST)

Figure 144. MVS VM Activating a Cross-Network SSCP-SSCP Session: Back-to-Back GWNCPs Request Sessions

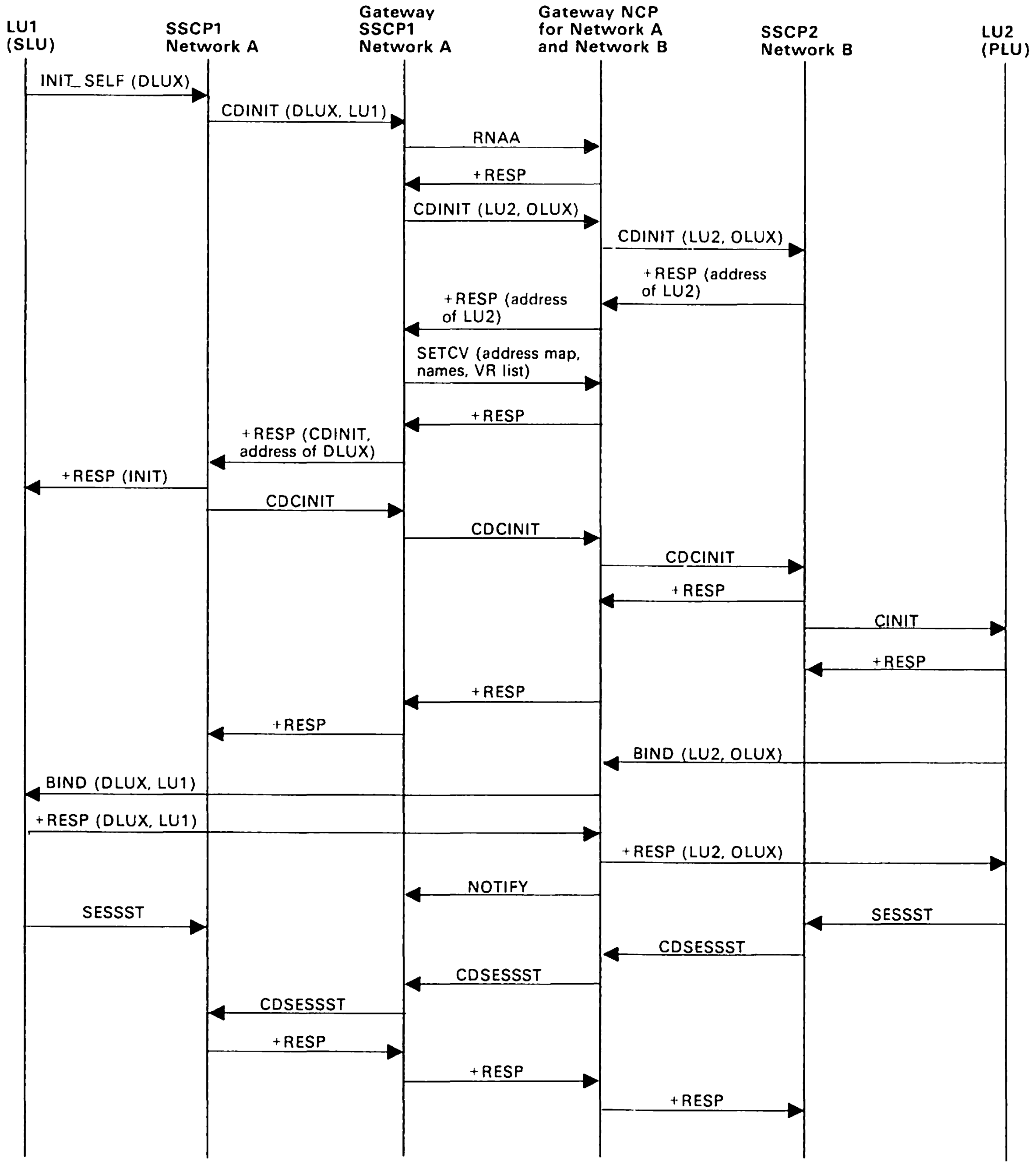


Figure 145. MVS VM Establishing a Cross-Network LU-LU Session: SLU-Initiated Request for Single GWSSCP and Single GWNCP (Pre-V3R2)

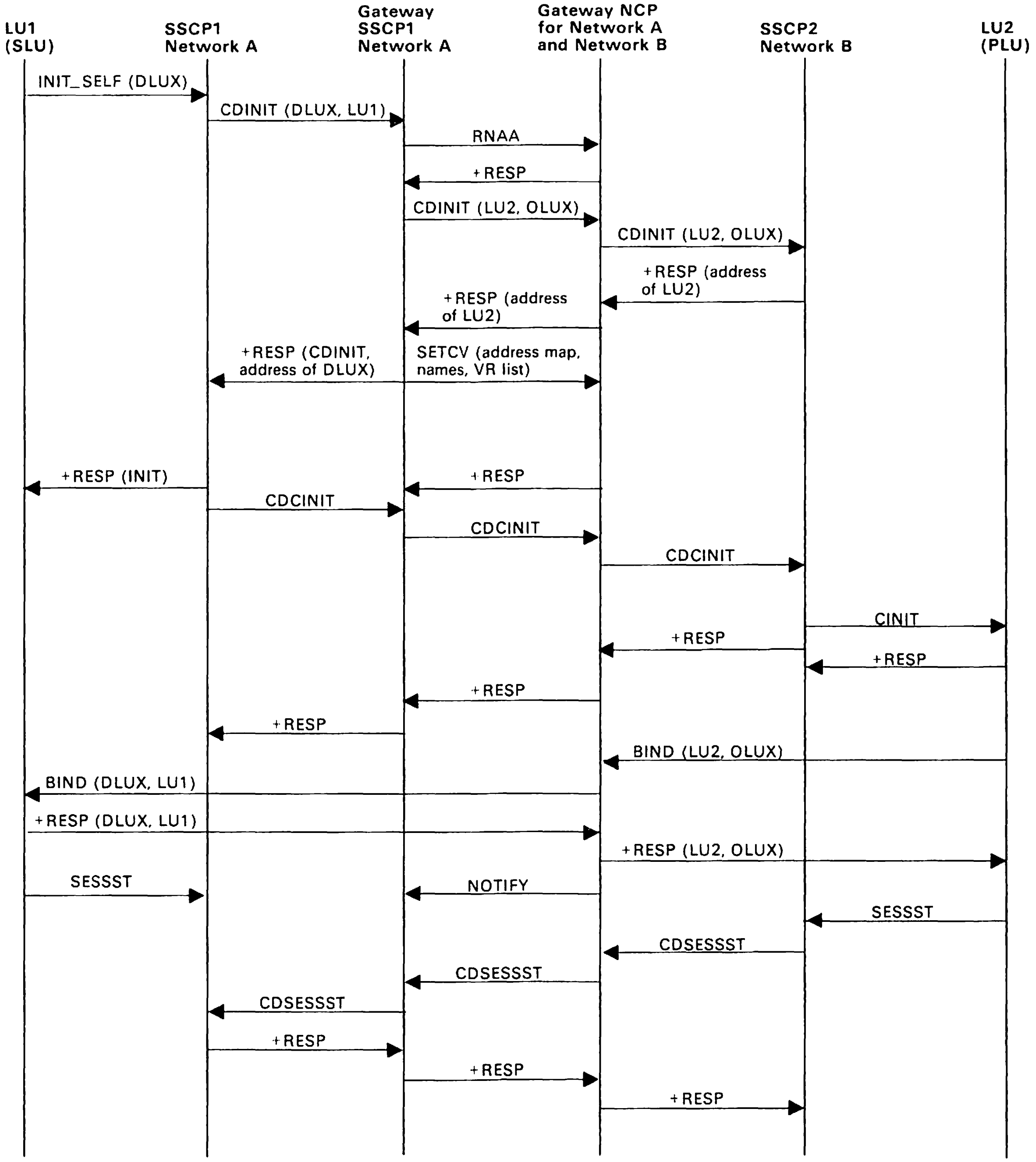


Figure 146. **MVS VM** Establishing a Cross-Network LU-LU Session: SLU-Initiated Request for Single GWSSCP and Single GWNCP (V3R2)

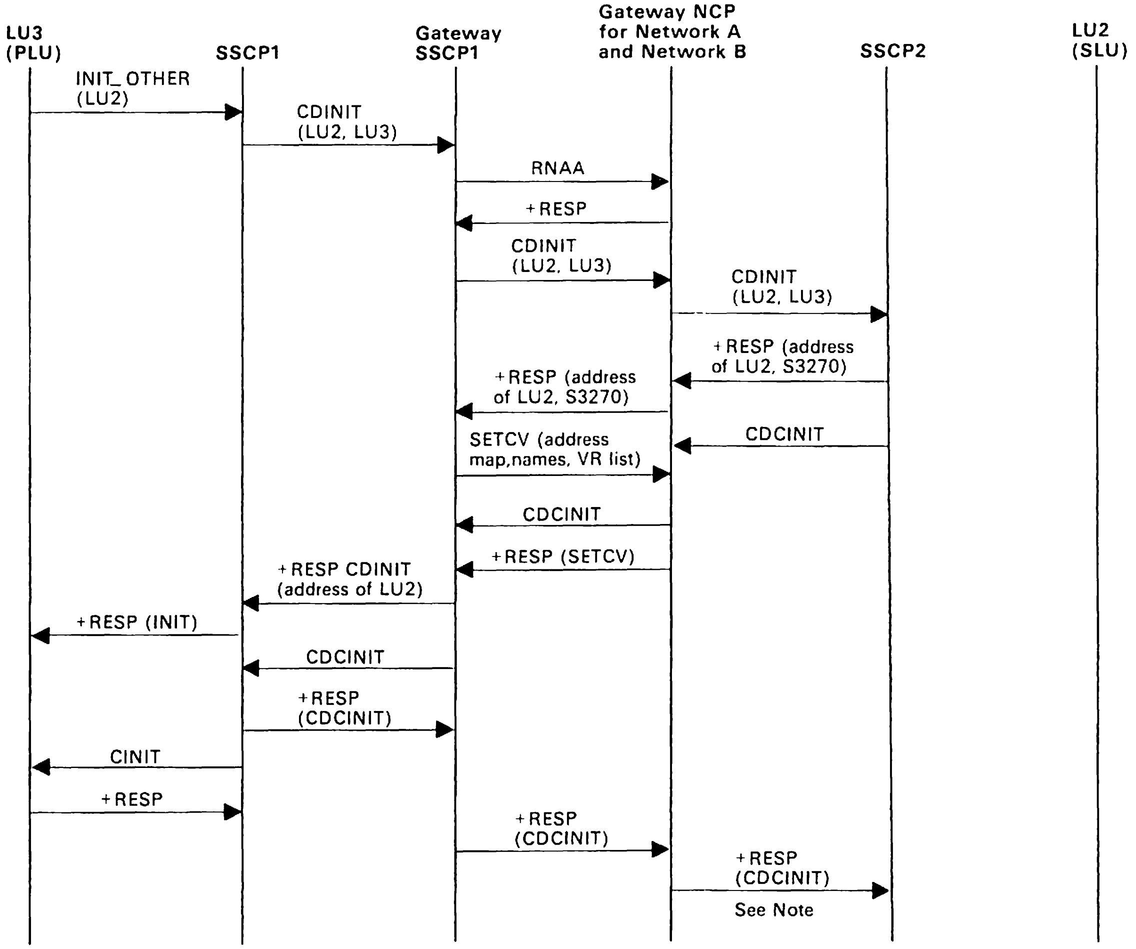


Figure 147. MVS VM Establishing a Cross-Network LU-LU Session: PLU-Initiated Request for Single GWSSCP and Single GWNCP (Pre-V3R2).

Note: Session setup continues as in the flow for an SLU-initiated session.

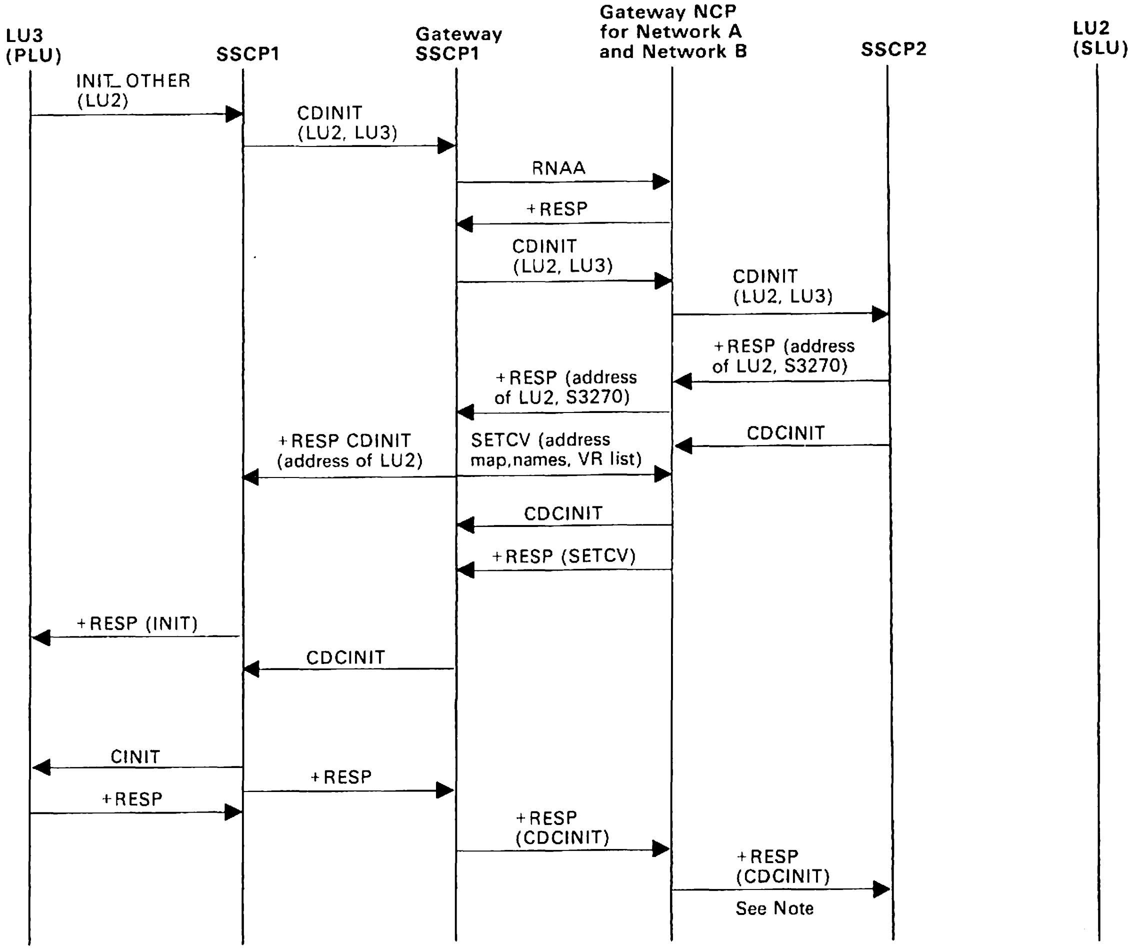


Figure 148. **MVS VM** Establishing a Cross-Network LU-LU Session: PLU-Initiated Request for Single GWSSCP and Single GWNCP (V3R2).

Note: Session setup continues as in the flow for an SLU-initiated session.

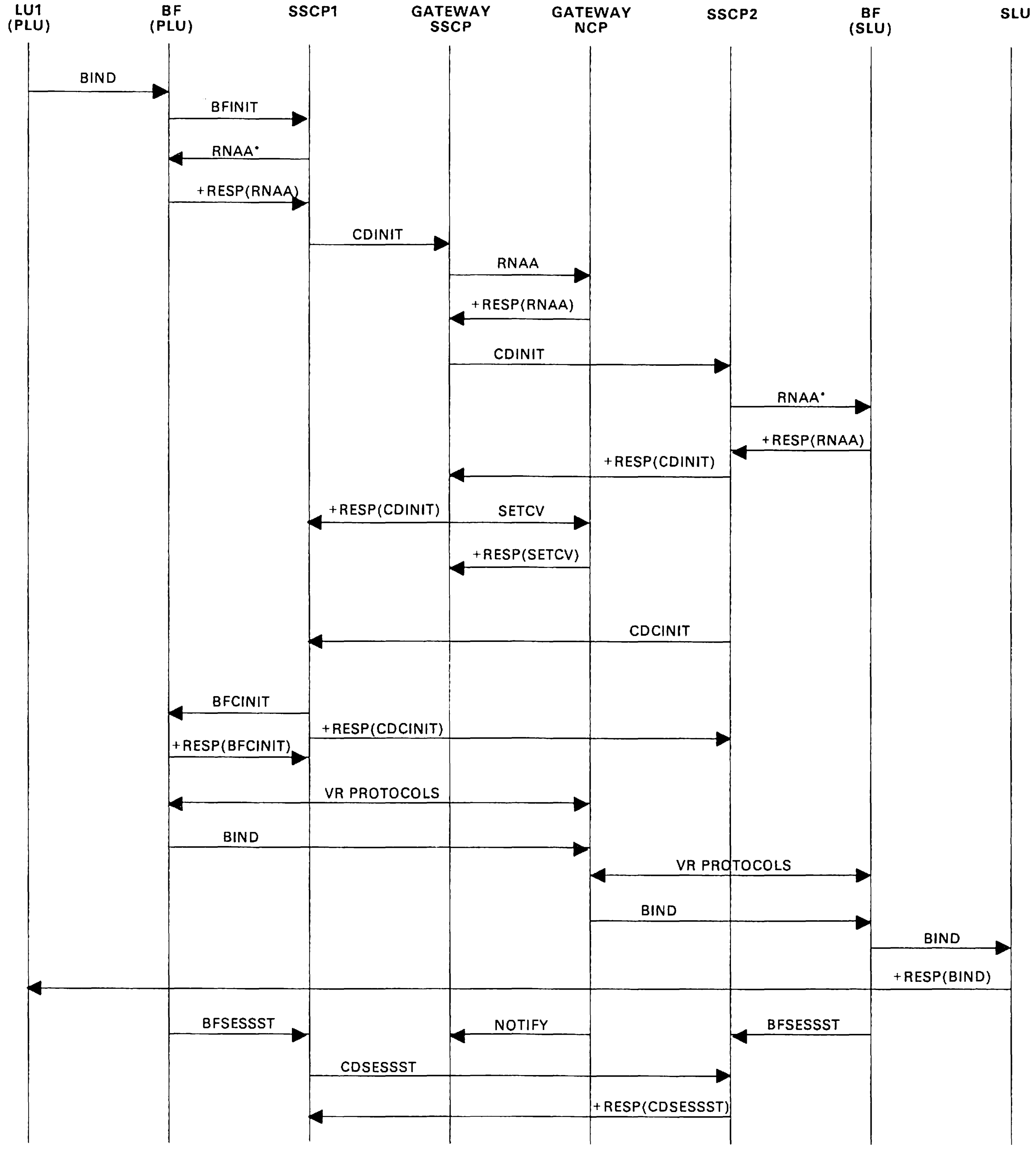


Figure 149. MVS VM Establishing a Cross-Network LU-LU Session: Independent PLU Request for Single GWNCP with an Independent SLU.

* RNAA flows only if the network address is needed.

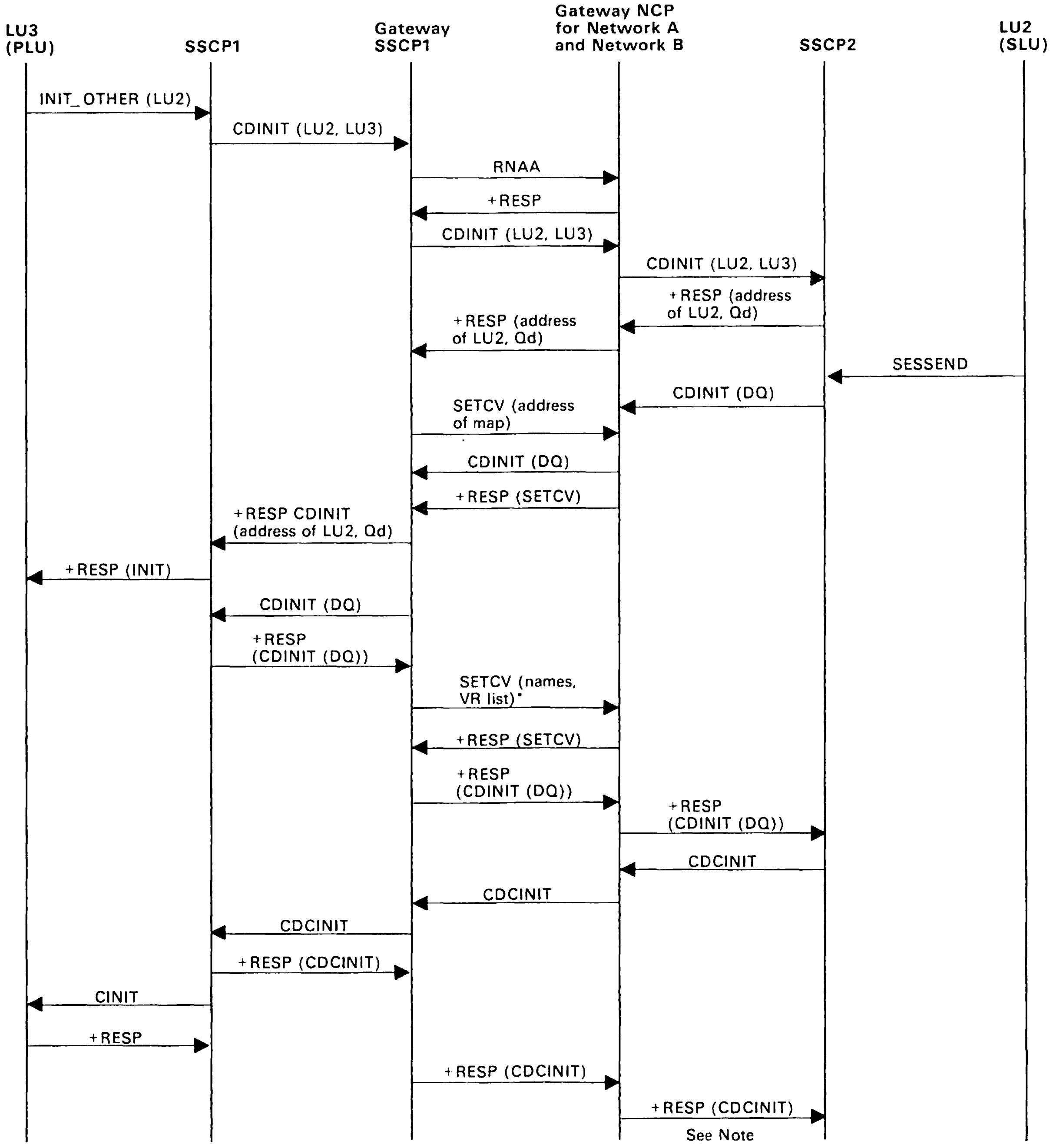


Figure 150. MVS VM Establishing a Cross-Network LU-LU Session: PLU-Initiated Request Setup Queued for Single GWSSCP/Single GWNCP (Pre-V3R2).

Note: Session setup continues as in the flow for an SLU-initiated session.

* Names only sent conditionally.

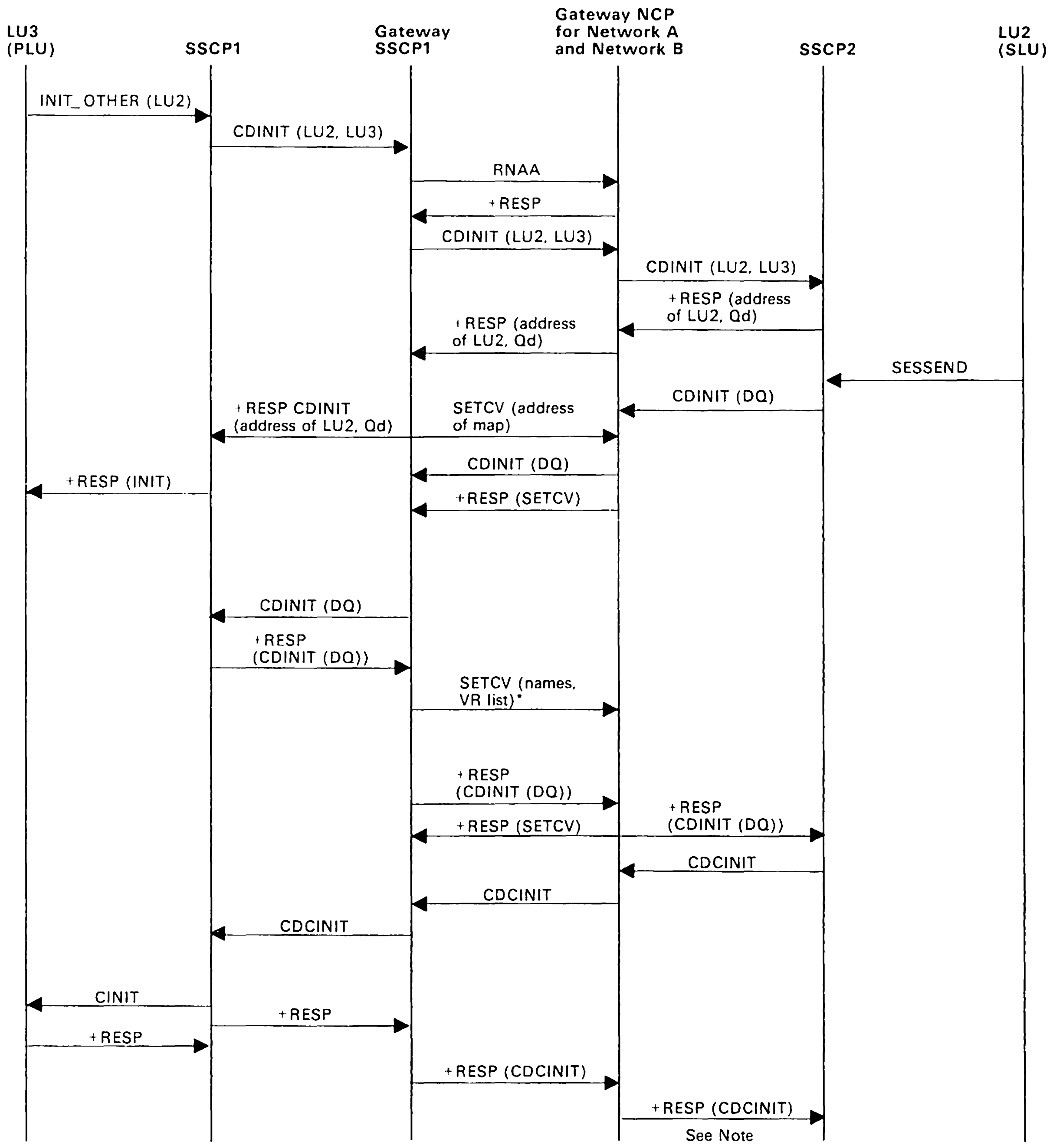


Figure 151. MVS VM Establishing a Cross-Network LU-LU Session: PLU-Initiated Request Setup Queued for Single GWSSCP/Single GWNCP (V3R2).

Note: Session setup continues as in the flow for an SLU-initiated session.

* Names only sent conditionally.

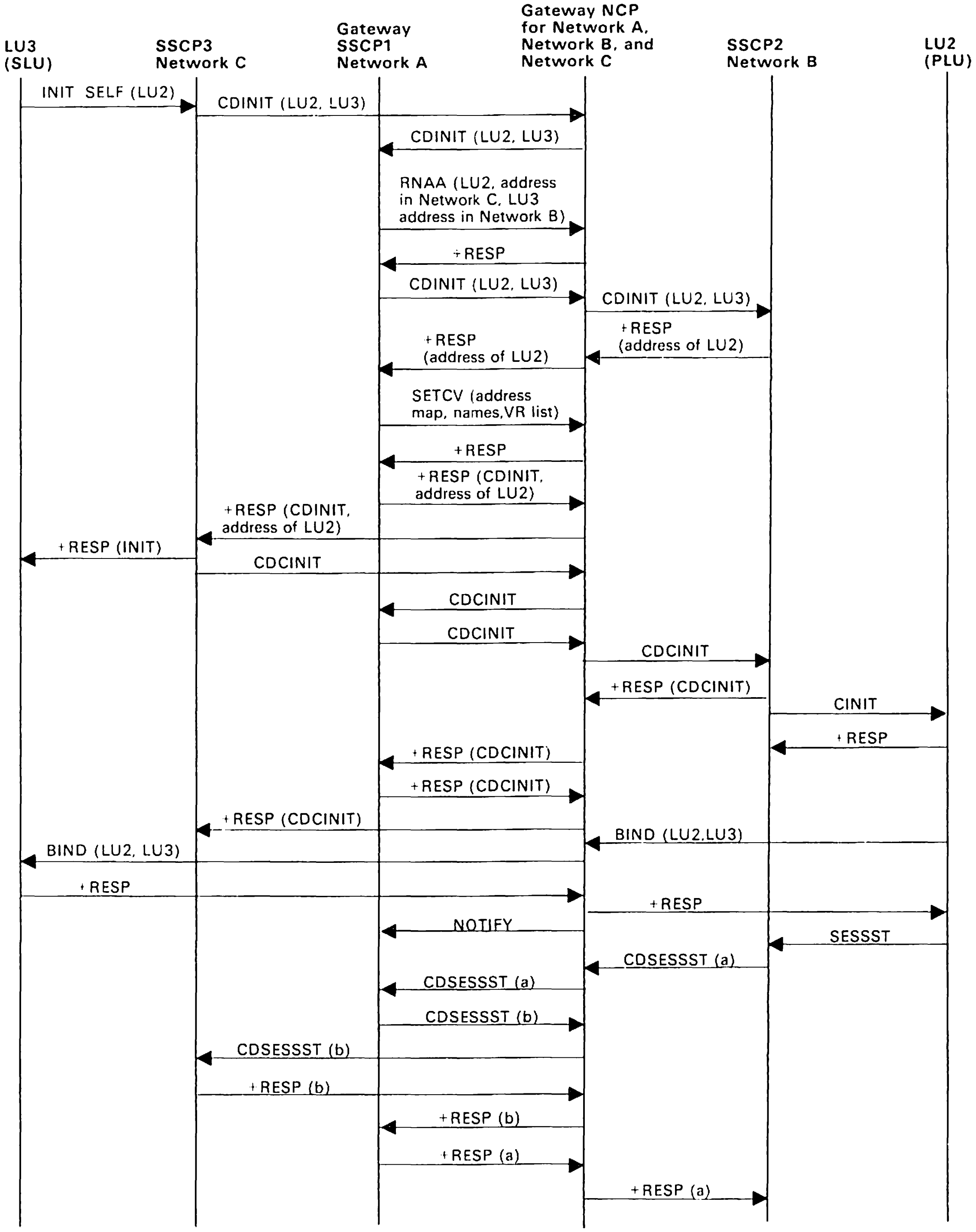


Figure 152. MVS VM Establishing a Cross-Network LU-LU Session: SLU-Initiated Request for Single Gateway Connecting Three or More Networks

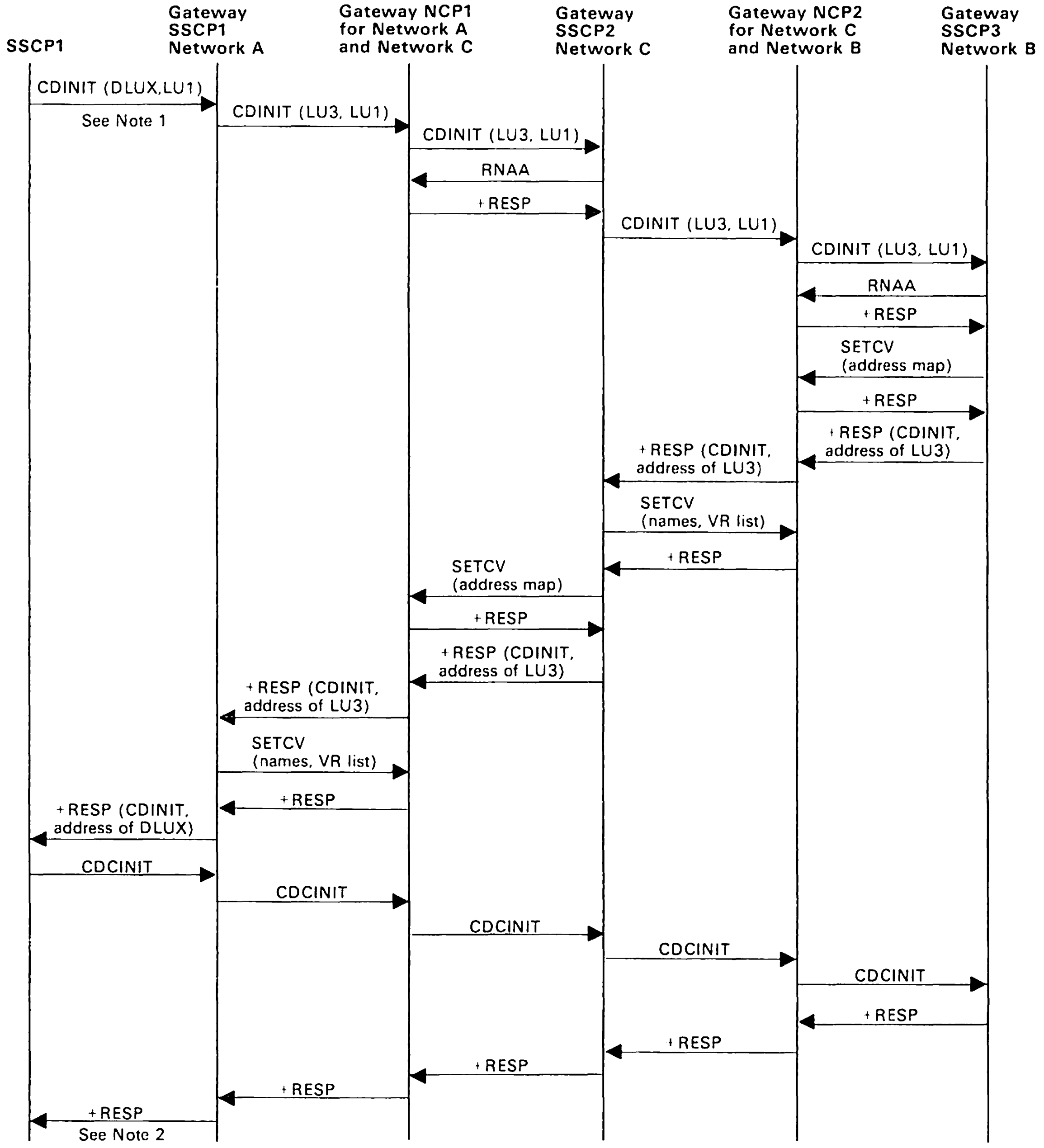


Figure 153. MVS VM Establishing a Cross-Network LU-LU Session: Default Partitioning of GWSSCP Responsibility Spanning Three Networks.

Note 1: Only the SSCP-SSCP session communication is shown. Assume LU1 (a logical unit owned by SSCP1 in Network A) requests a session with DLUX (an alias for LU3 in Network B; LU3 is a logical unit owned by gateway SSCP3). LU1 is the SLU in the request session.

Note 2: Session setup proceeds as shown in the basic flows. BIND flows from LU3 to gateway NCP2, to gateway NCP1, and to LU1.

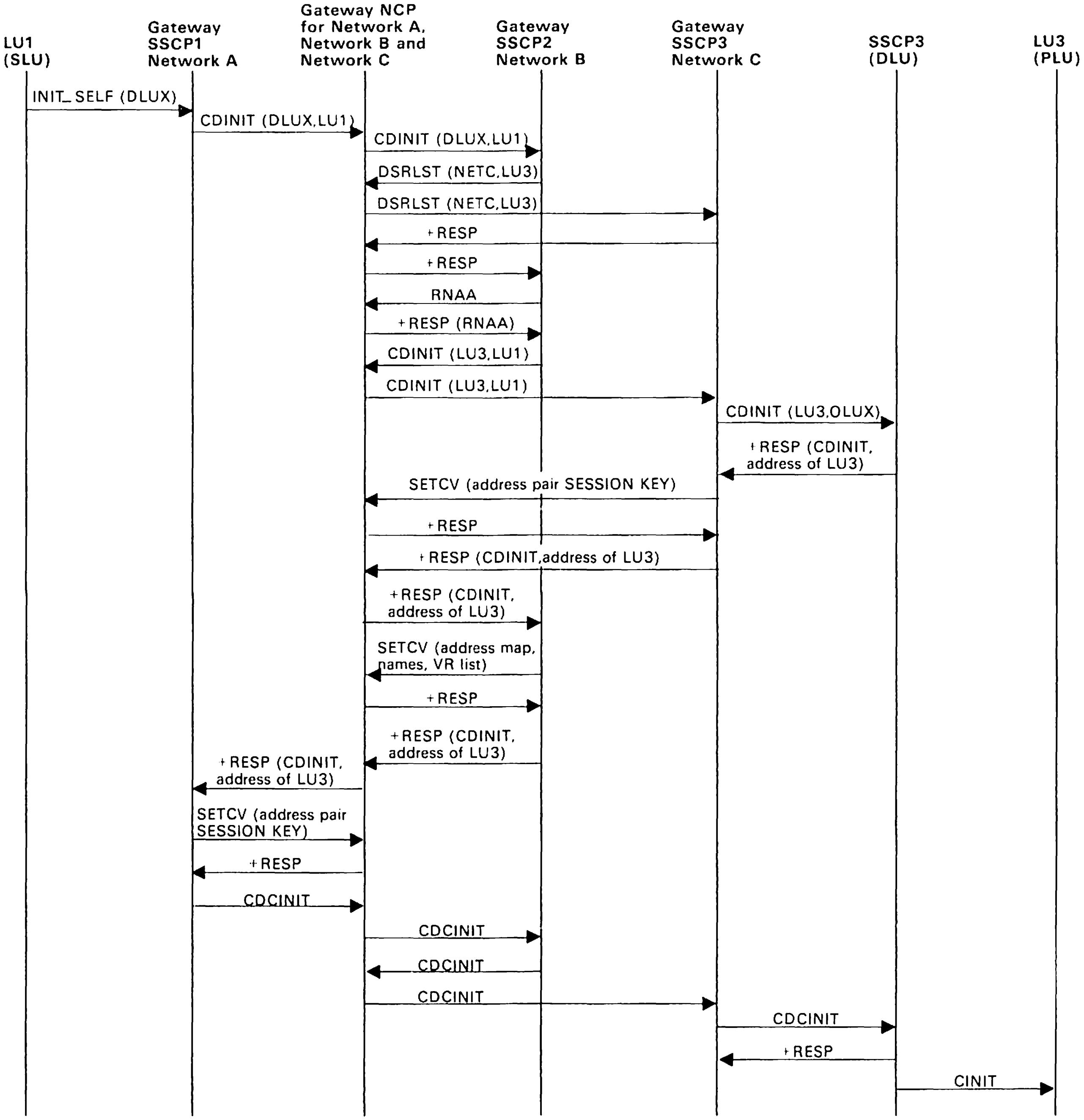


Figure 154 (Part 1 of 2). **MVS VM** Establishing a Cross-Network LU-LU Session: SLU-Initiated Request for Predesignated Control of GWNCP by Middle Host (Pre-V3R2)

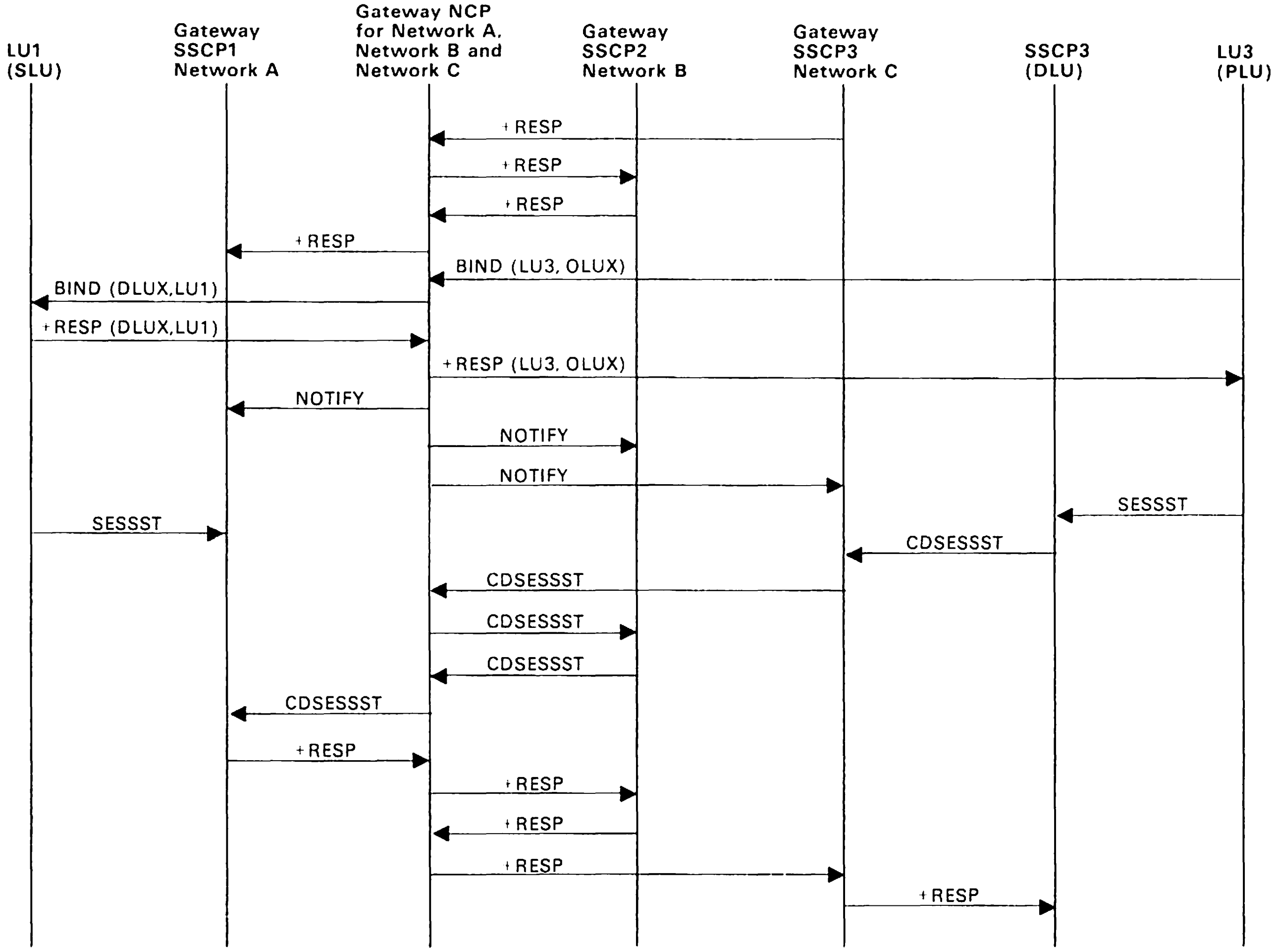


Figure 154 (Part 2 of 2). MVS VM Establishing a Cross-Network LU-LU Session: SLU-Initiated Request for Predesignated Control of GWNCP by Middle Host (Pre-V3R2)

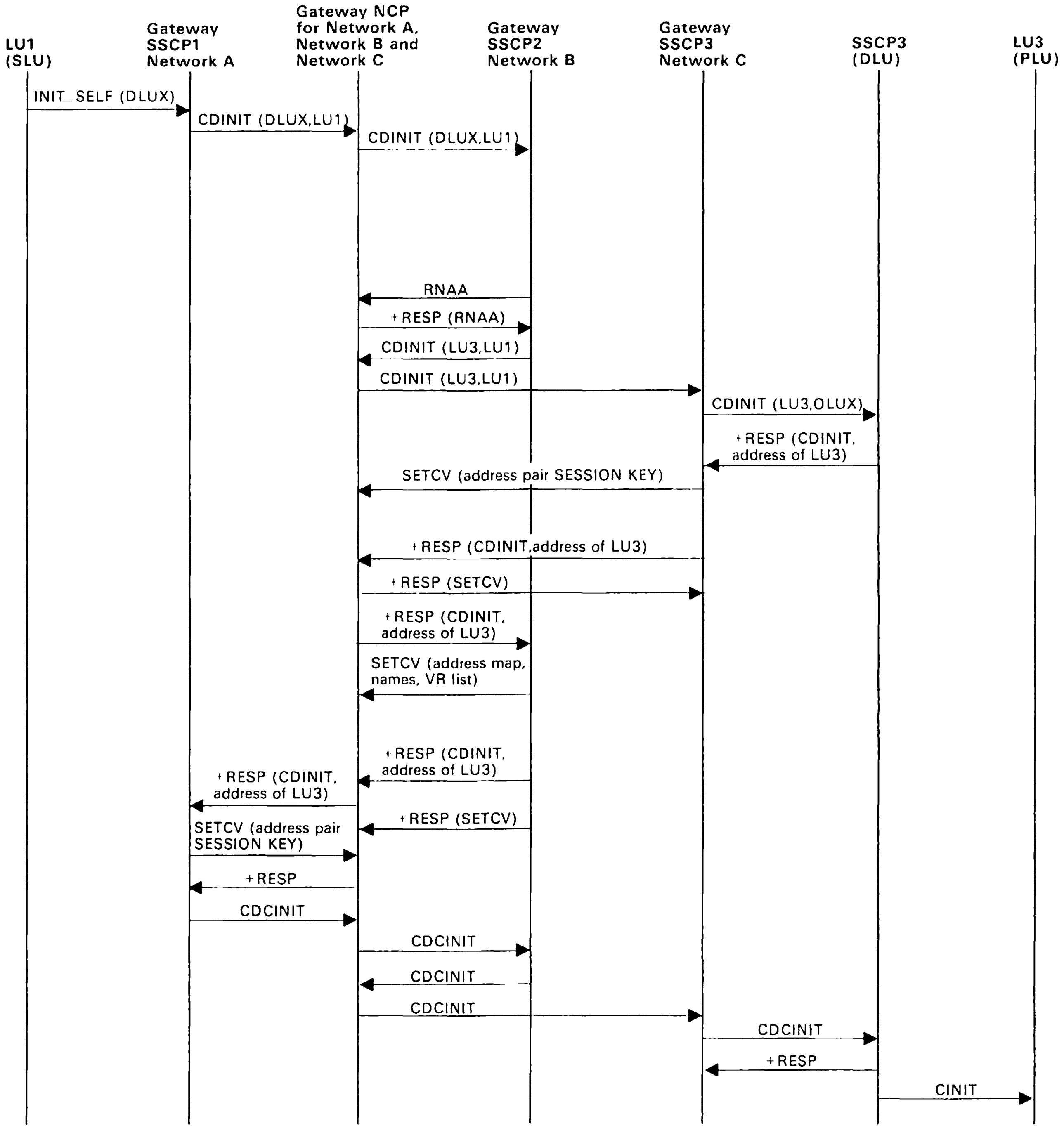


Figure 155 (Part 1 of 2). **MVS VM** Establishing a Cross-Network LU-LU Session: SLU-Initiated Request for Predesignated Control of GWNCP by Middle Host (V3R2)

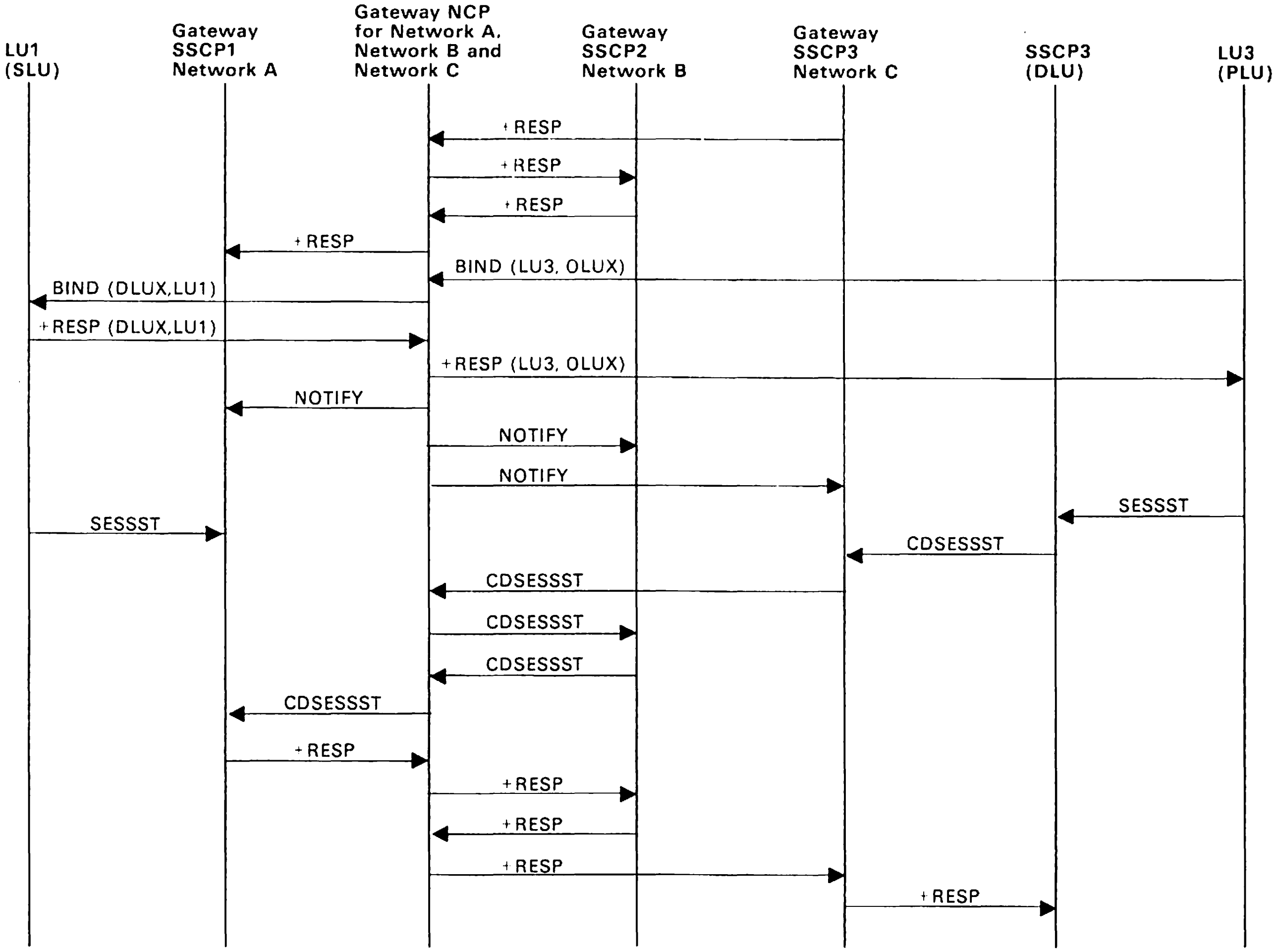


Figure 155 (Part 2 of 2). MVS VM Establishing a Cross-Network LU-LU Session: SLU-Initiated Request for Predesignated Control of GWNCP by Middle Host (V3R2)

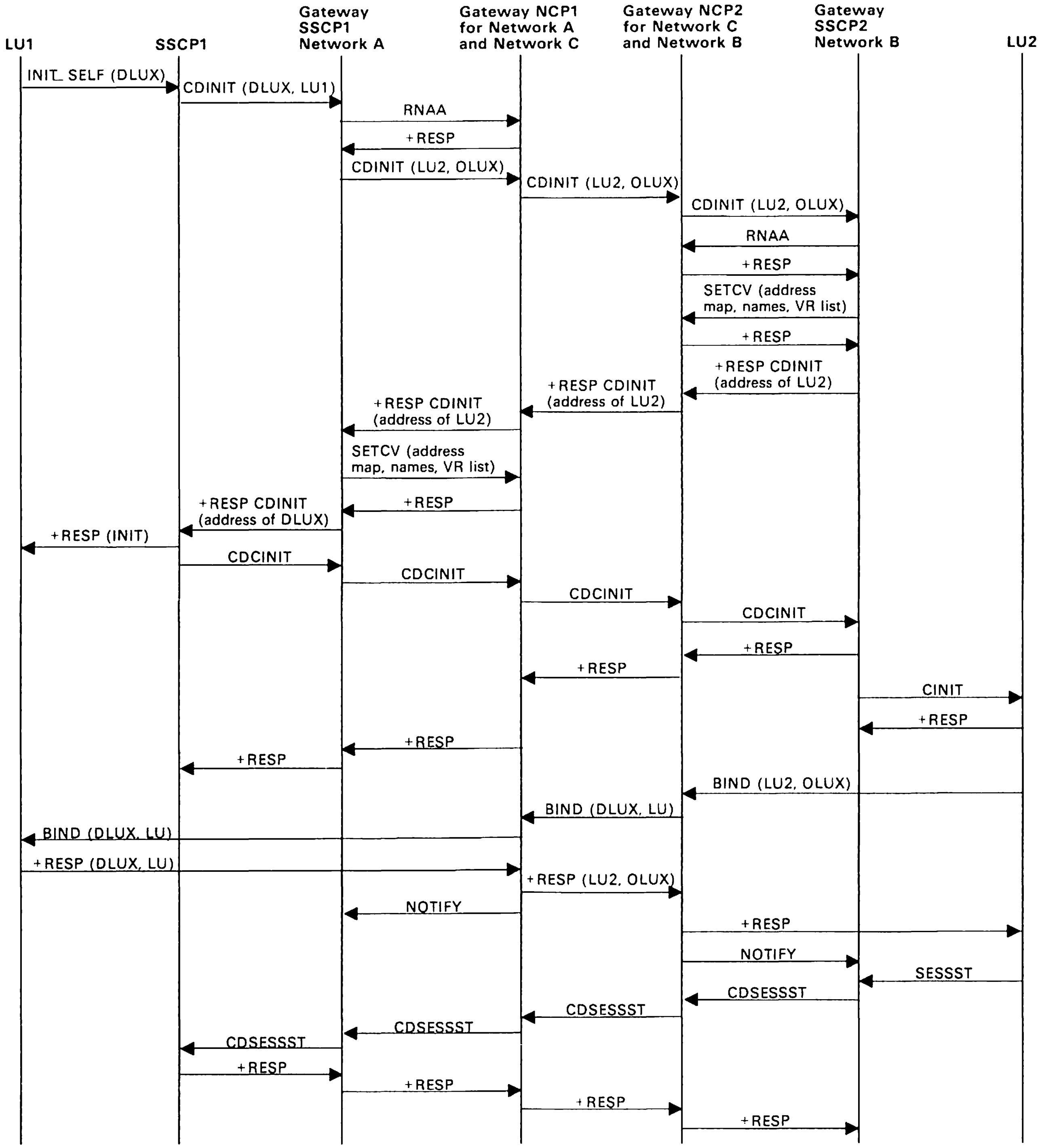


Figure 156. MVS VM Establishing a Cross-Network LU-LU Session: Multiple GWSSCPs and Back-to-Back GWNCPs

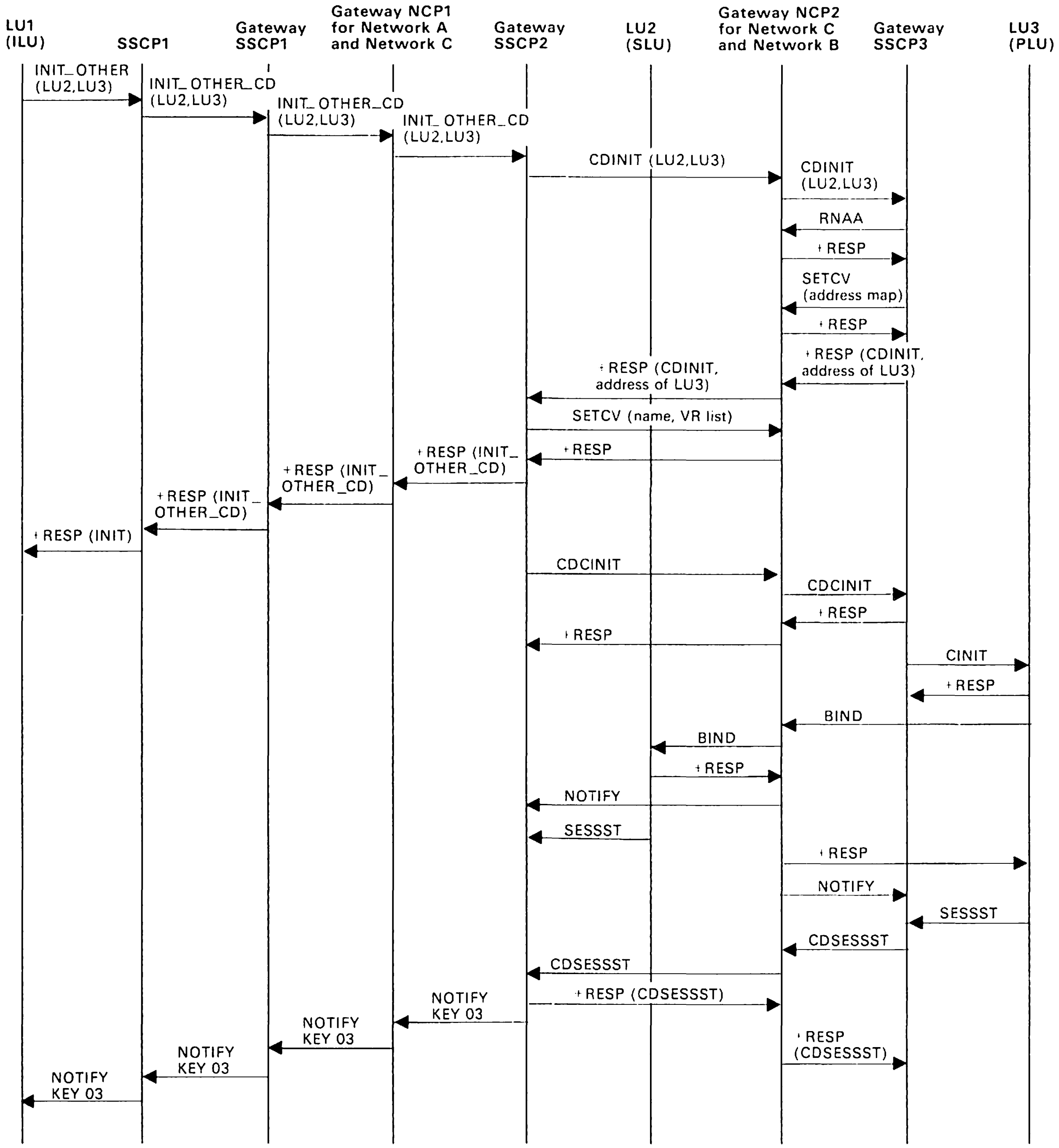


Figure 157. MVS VMEstablishing a Cross-Network LU-LU Session: Third Party-Initiated Request Spanning Three Networks

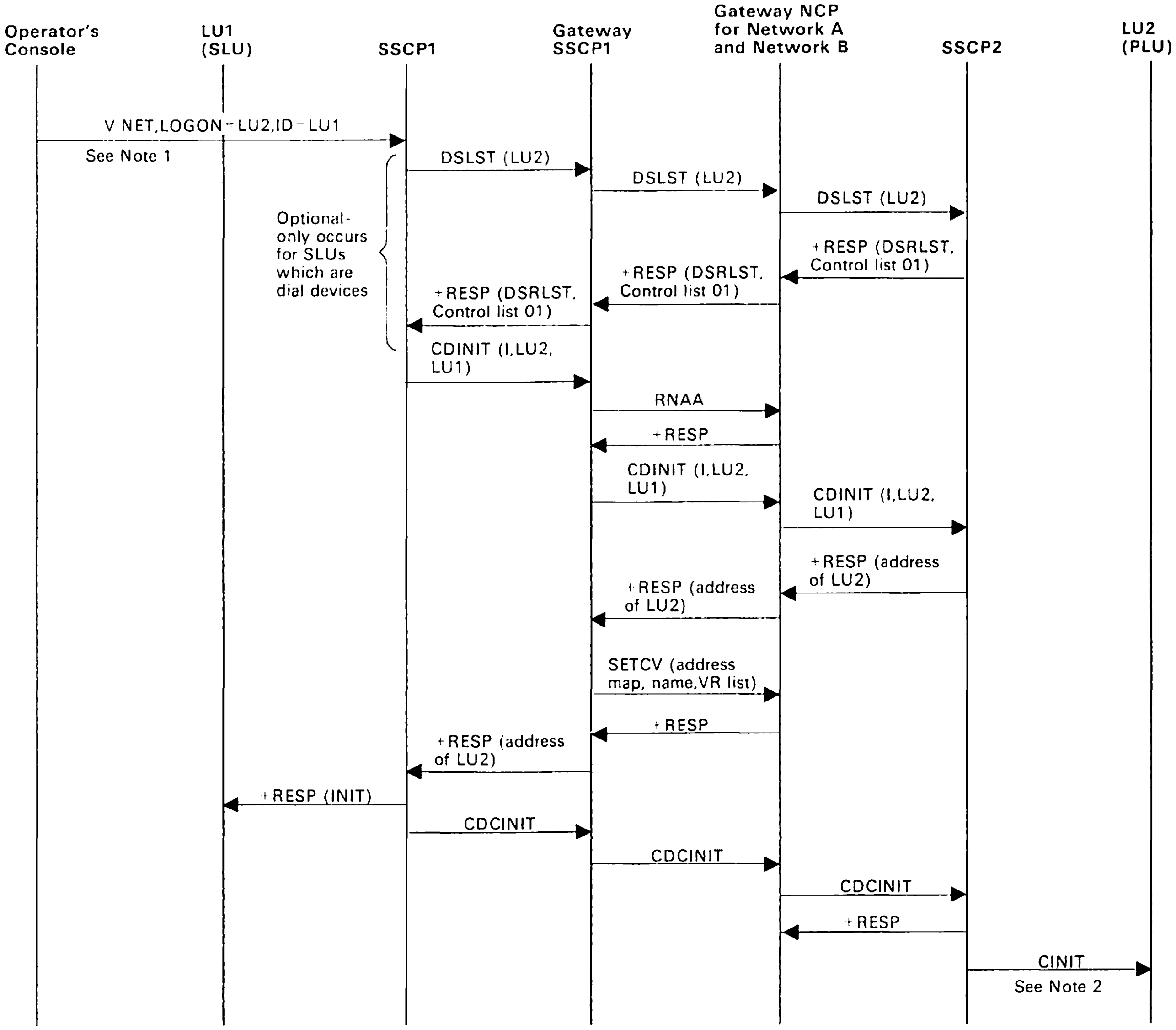


Figure 158. MVS VMEstablishing a Cross-Network LU-LU Session: Initiate with VARY LOGON or Controlling PLU for Single GWSSCP/Single GWNCP (Pre-V3R2).

Note 1: Assume LU1 is active.

Note 2: Session setup continues as shown in the basic flows.

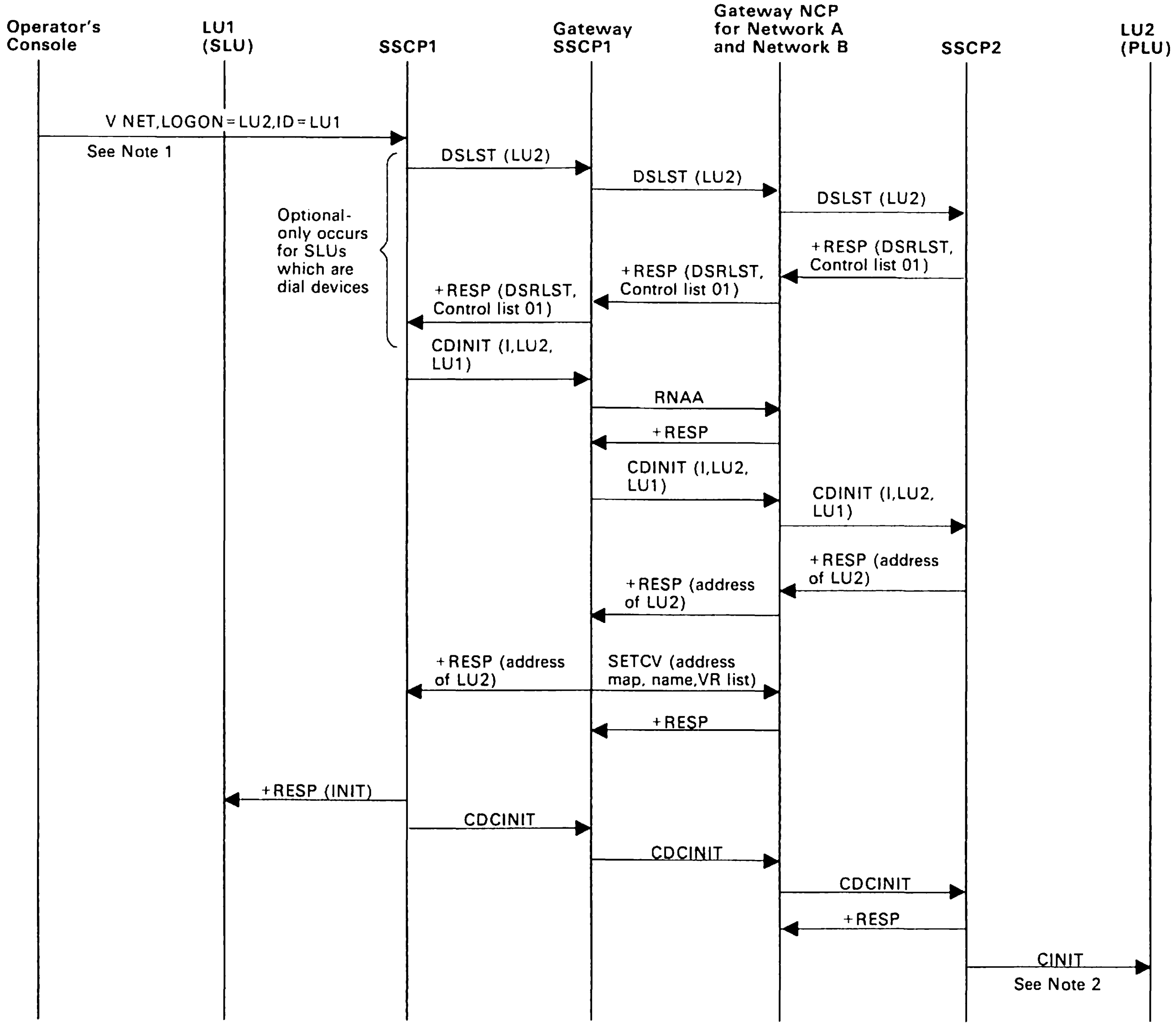


Figure 159. **MVS VM** Establishing a Cross-Network LU-LU Session: Initiate with VARY LOGON or Controlling PLU for Single GWSSCP/Single GWNCP (Pre-V3R2).

Note 1: Assume LU1 is active.

Note 2: Session setup proceeds as shown in the basic flows.

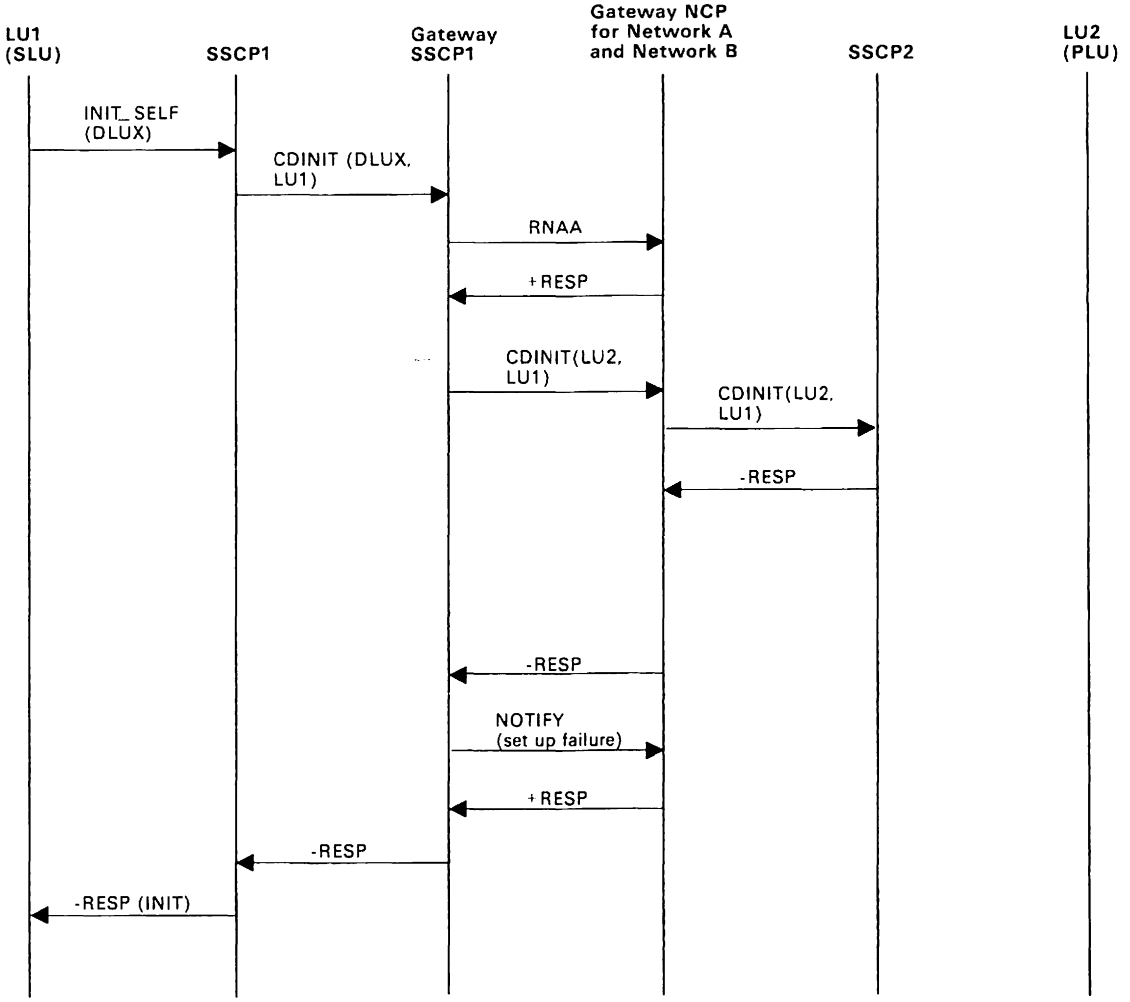


Figure 160. MVS VM Establishing a Cross-Network LU-LU Session: SLU-Initiated Session Initiation Failure (CDINIT Rejection) for Single GWNCP (Pre-V3R2)

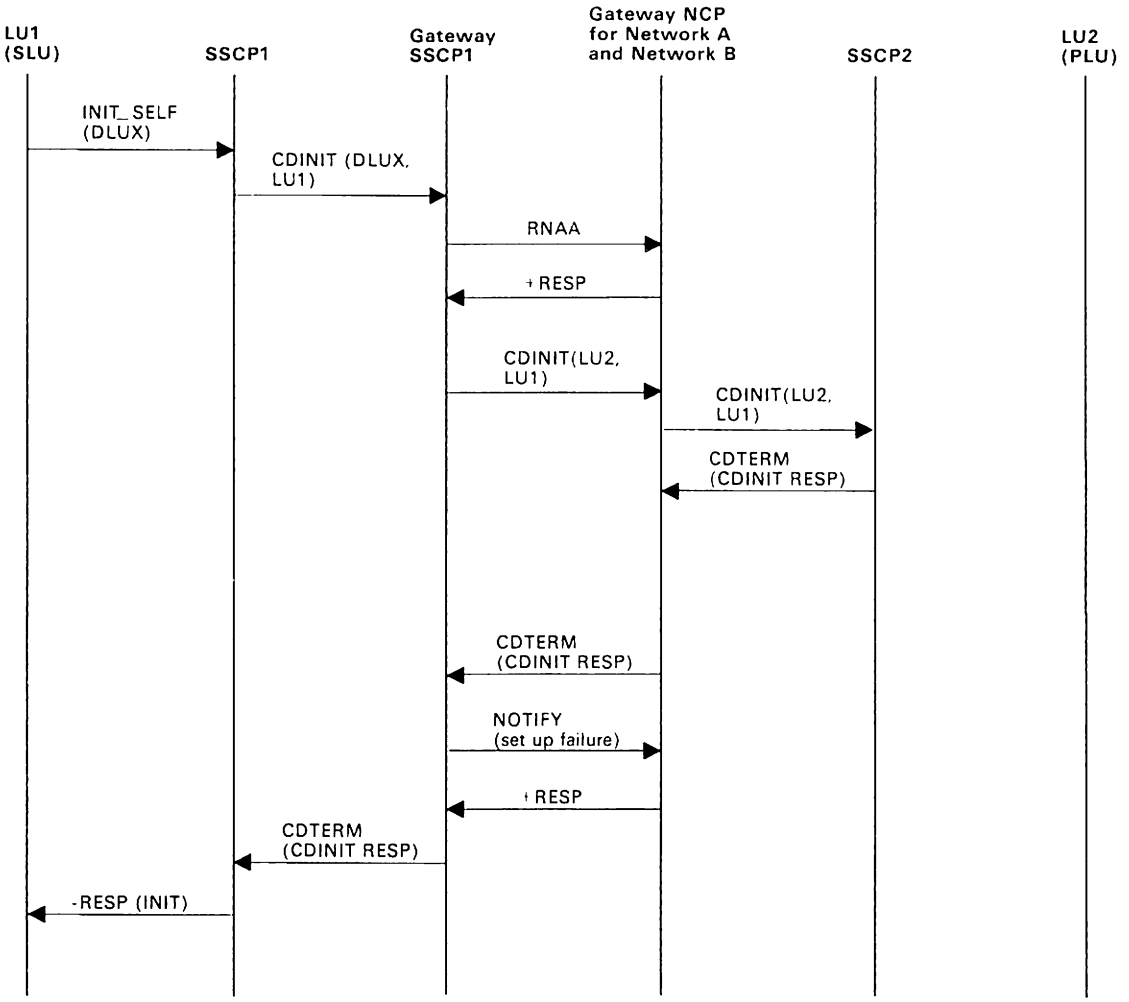


Figure 161. MVS VM Establishing a Cross-Network LU-LU Session: SLU-Initiated Session Initiation Failure (CDINIT Rejection) for Single GWNCP (V3R2)

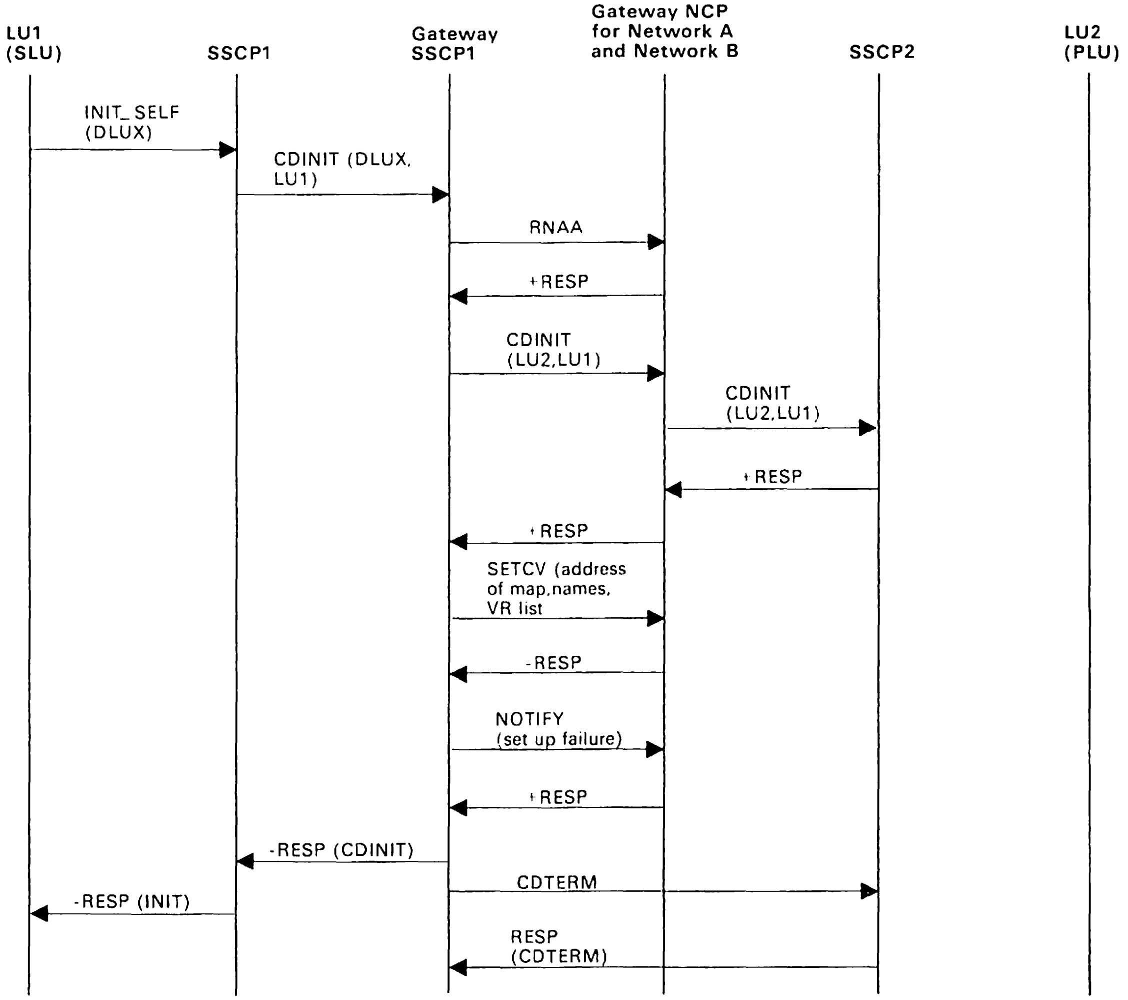


Figure 162. **MVS VM** Establishing a Cross-Network LU-LU Session: SLU-Initiated Session Initiation Failure (SETCV) for Single GWSSCP/Single GWNCP (Pre-V3R2)

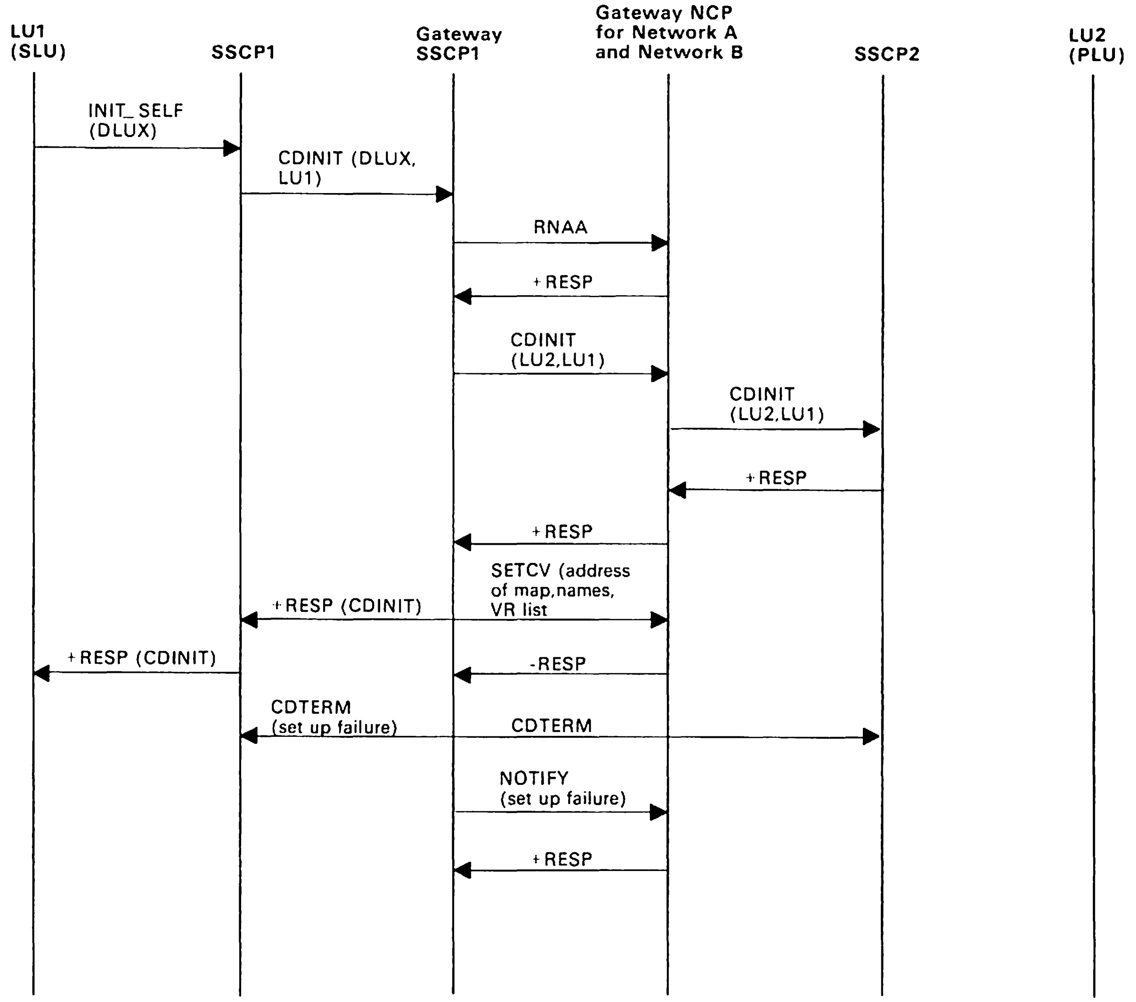


Figure 163. MVS VM Establishing a Cross-Network LU-LU Session: SLU-Initiated Session Initiation Failure (SETCV) for Single GWSSCP/Single GWNCP (V3R2)

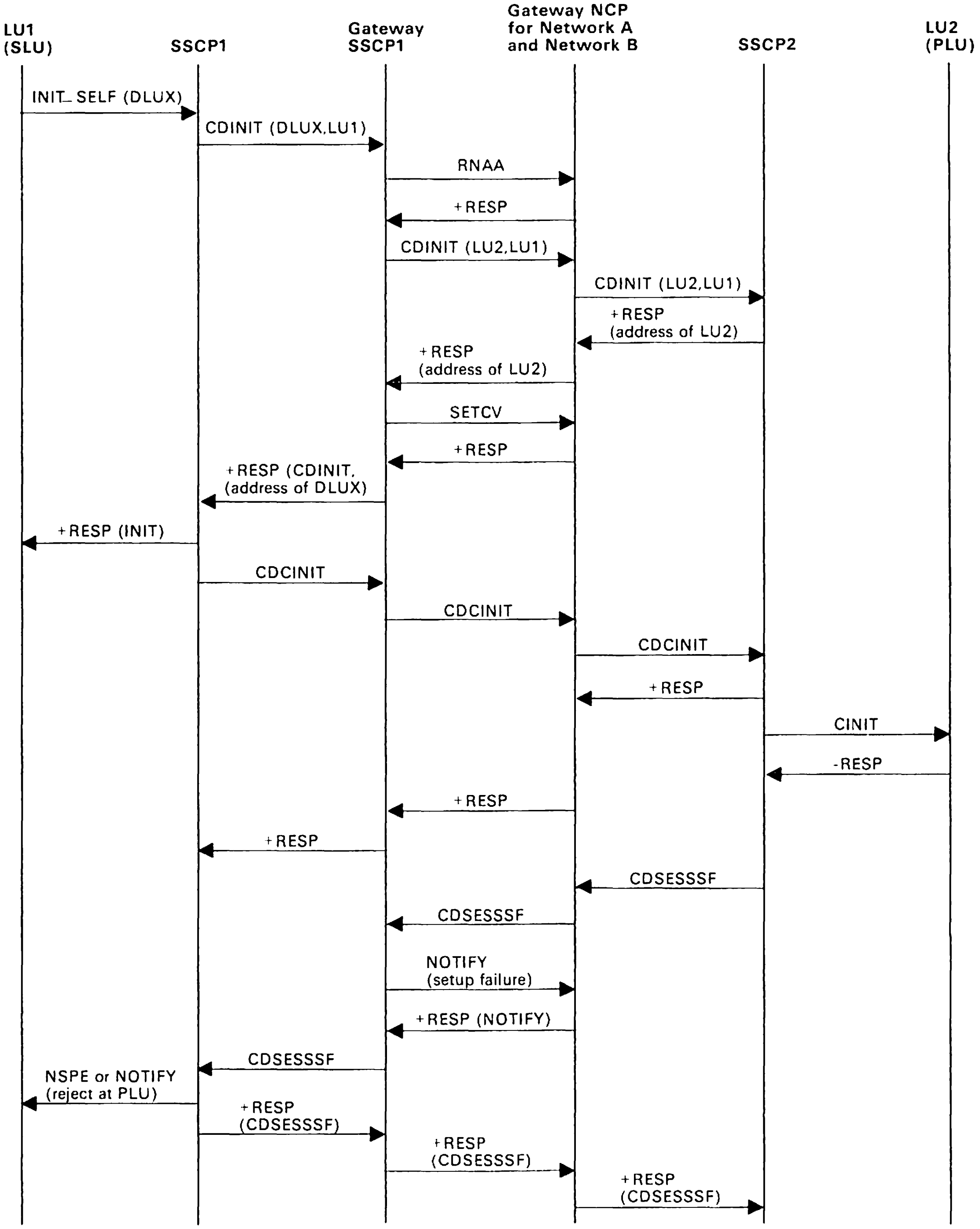


Figure 164. MVS VM Establishing a Cross-Network LU-LU Session: SLU-Initiated Setup Procedure Failure (CINIT Rejection) for Single GWSSCP/GWNCP (Pre-V3R2)

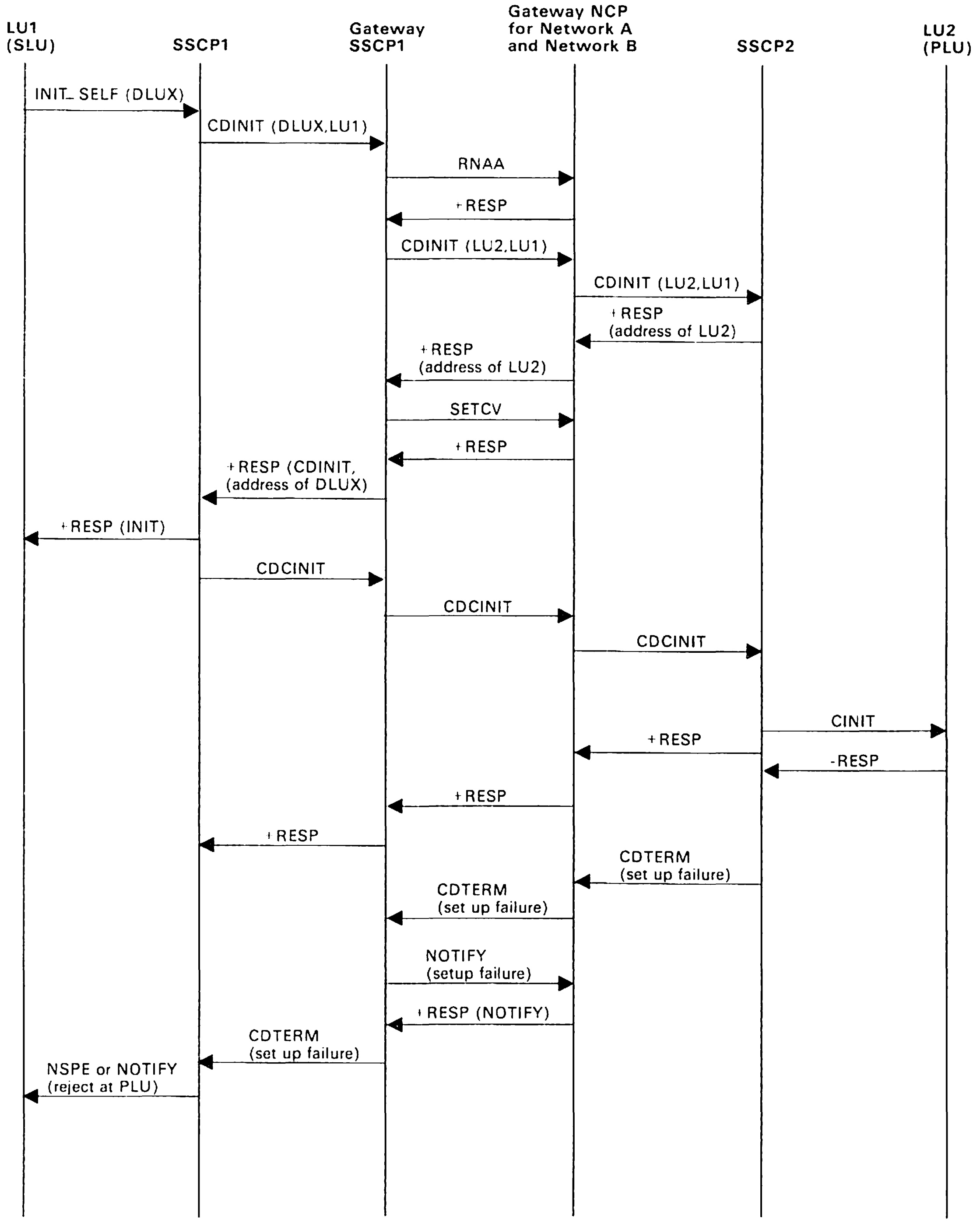


Figure 165. MVS VM Establishing a Cross-Network LU-LU Session: SLU-Initiated Setup Procedure Failure (CINIT Rejection) for Single GWSSCP/GWNCP (V3R2)

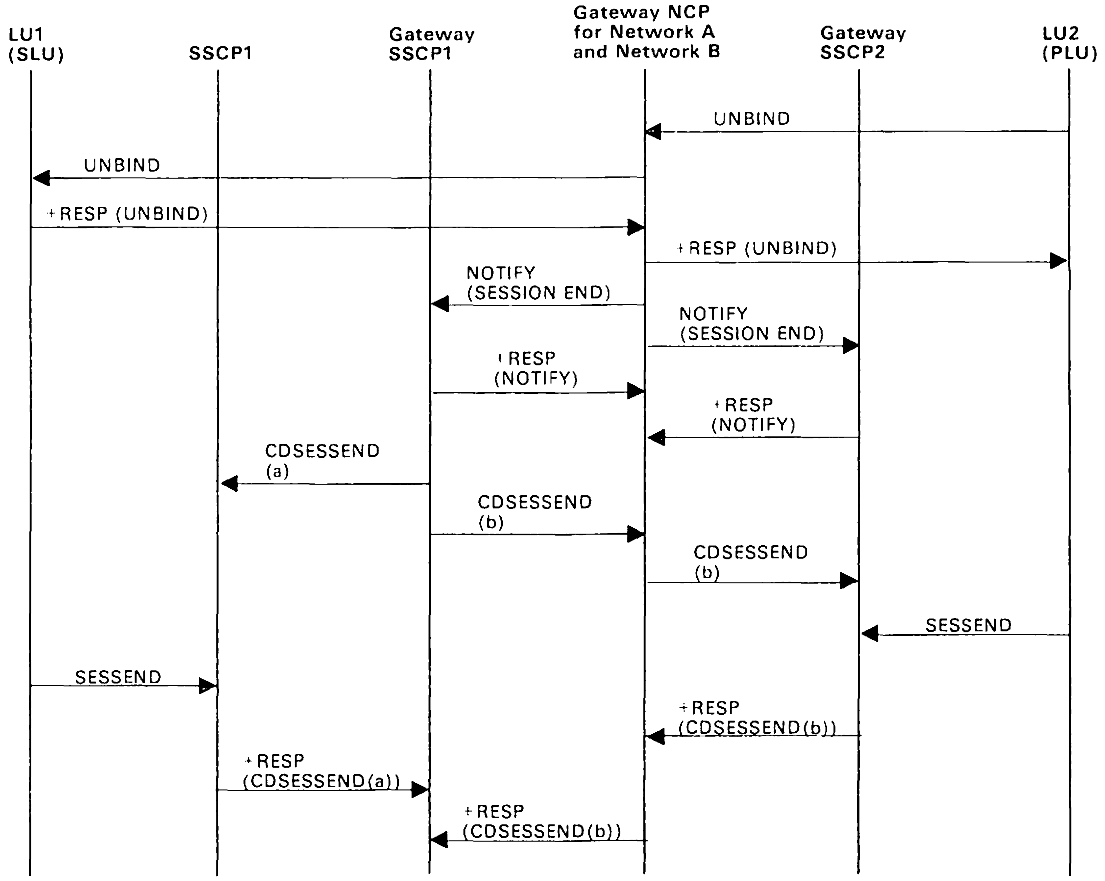


Figure 166. MVS VM Deactivating Resources and Ending Sessions: PLU sends UNBIND for Single GWSSCP and Single GWNCP

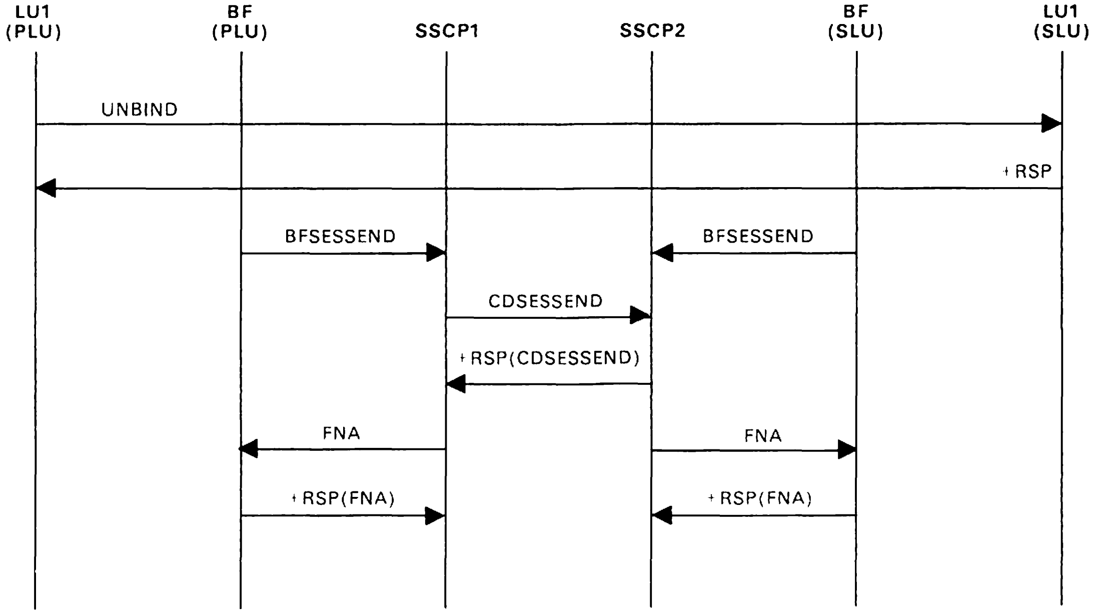


Figure 167. MVS VM Deactivating Resources and Ending Sessions: Independent PLU sends UNBIND for Cross-domain LU-LU session

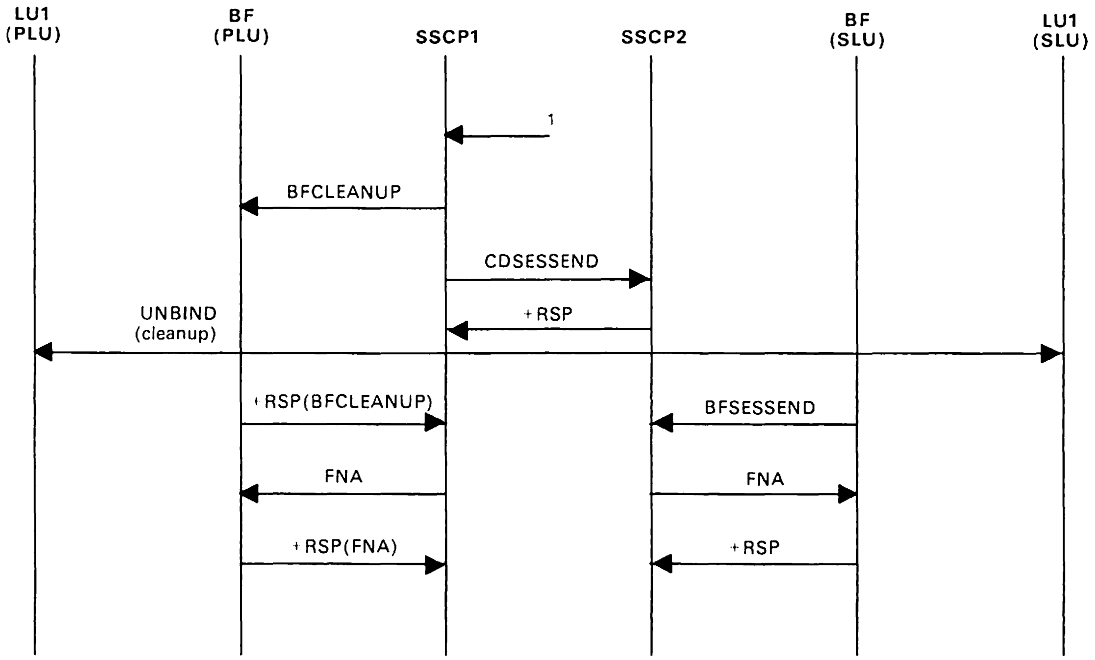


Figure 168. MVS VM Deactivating Resources and Ending Sessions: Independent PLU sends BFCLEANUP for Cross-domain LU-LU session.

[1] BFCLEANUP may be sent by the SSCP(PLU) for several reasons, including:

A network operator at the SSCP(PLU) may issue a VARY NET,TERM,UNCOND, generating an internal TERM-OTHER(forced).

A network operator at either SSCP may issue a VARY NET,INACT,ID=cdrm, deactivating all cross-domain sessions between the SSCPs.

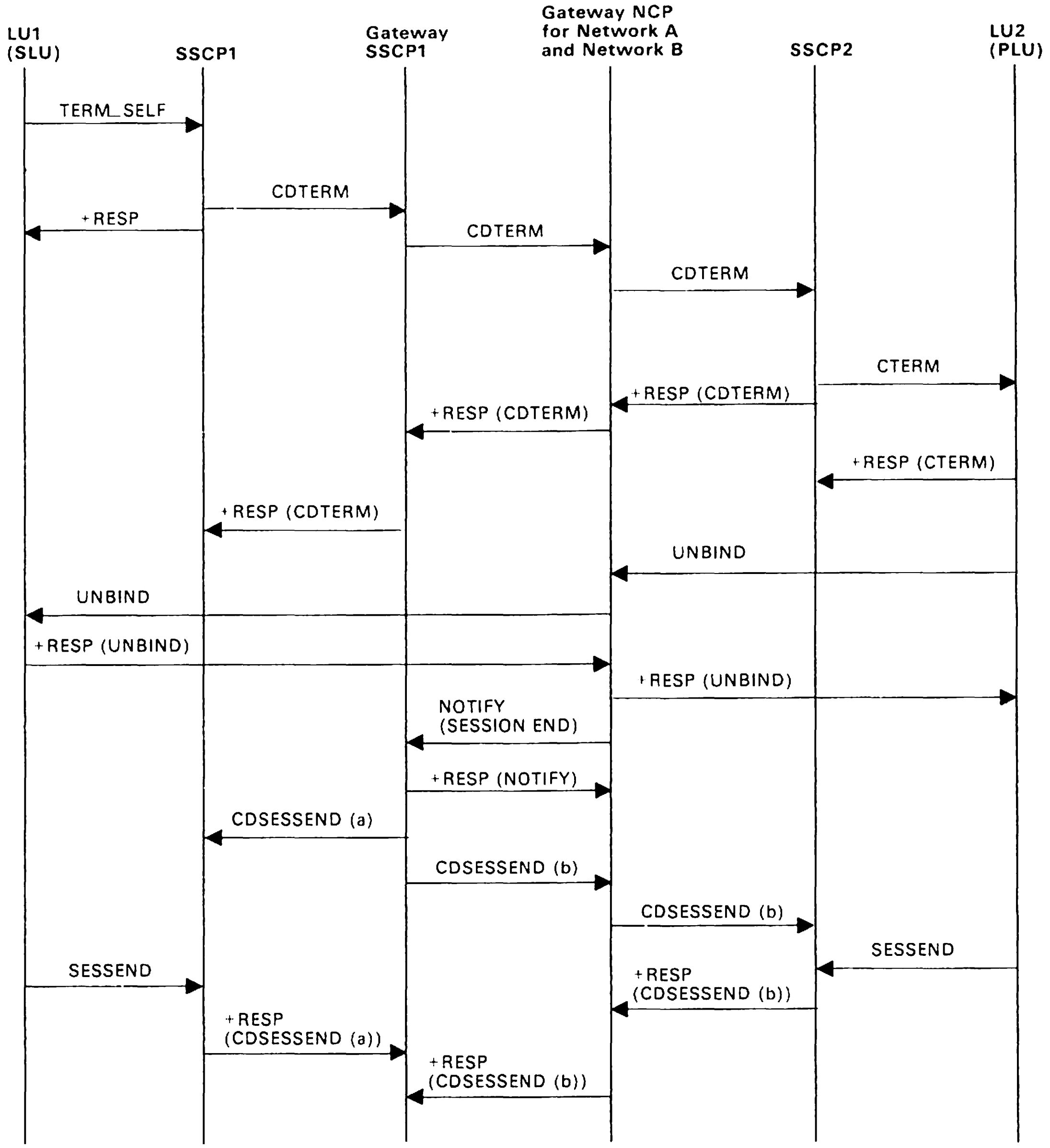


Figure 169. MVS VM Deactivating Resources and Ending Sessions: SLU Requests Terminate Self for Single GWSSCP and Single GWNCP

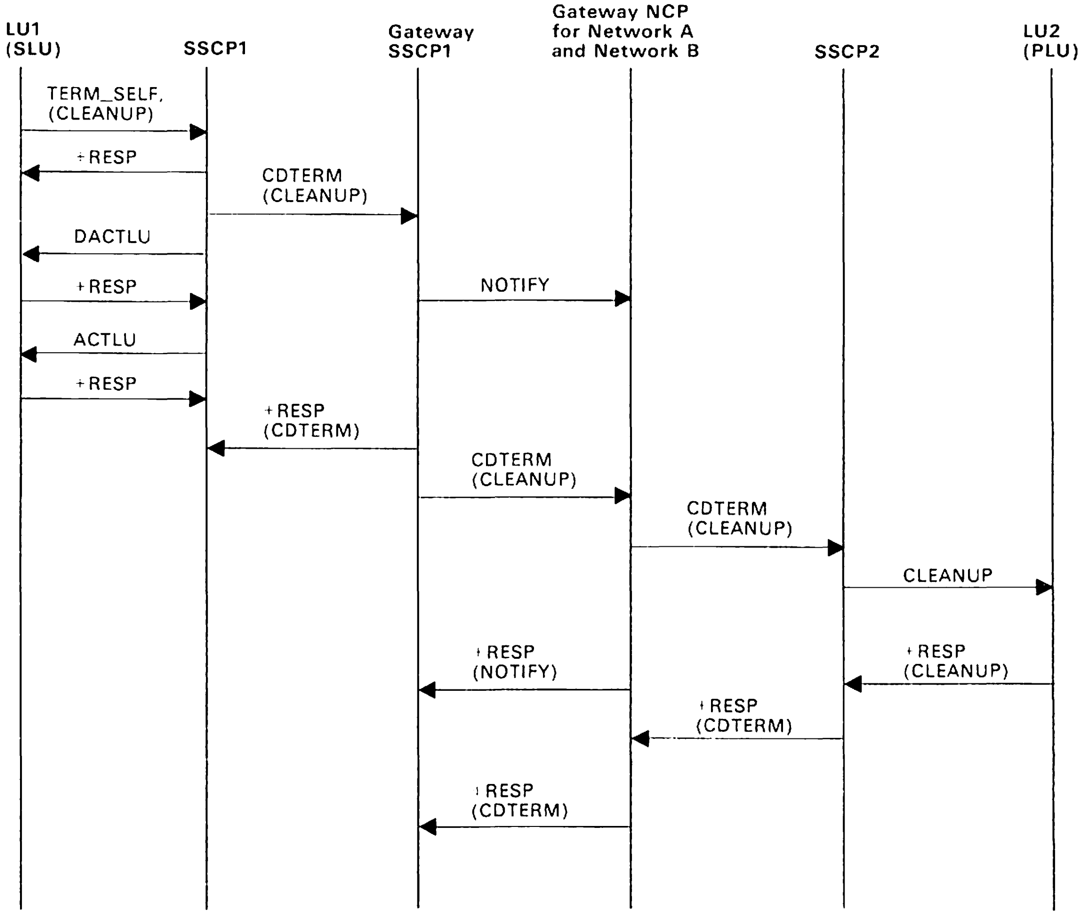


Figure 170. MVS VMDeactivating Resources and Ending Sessions: SLU Requests Terminate Self (Cleanup Option) for Single GWSSCP/GWNCP (Pre-V3R2)

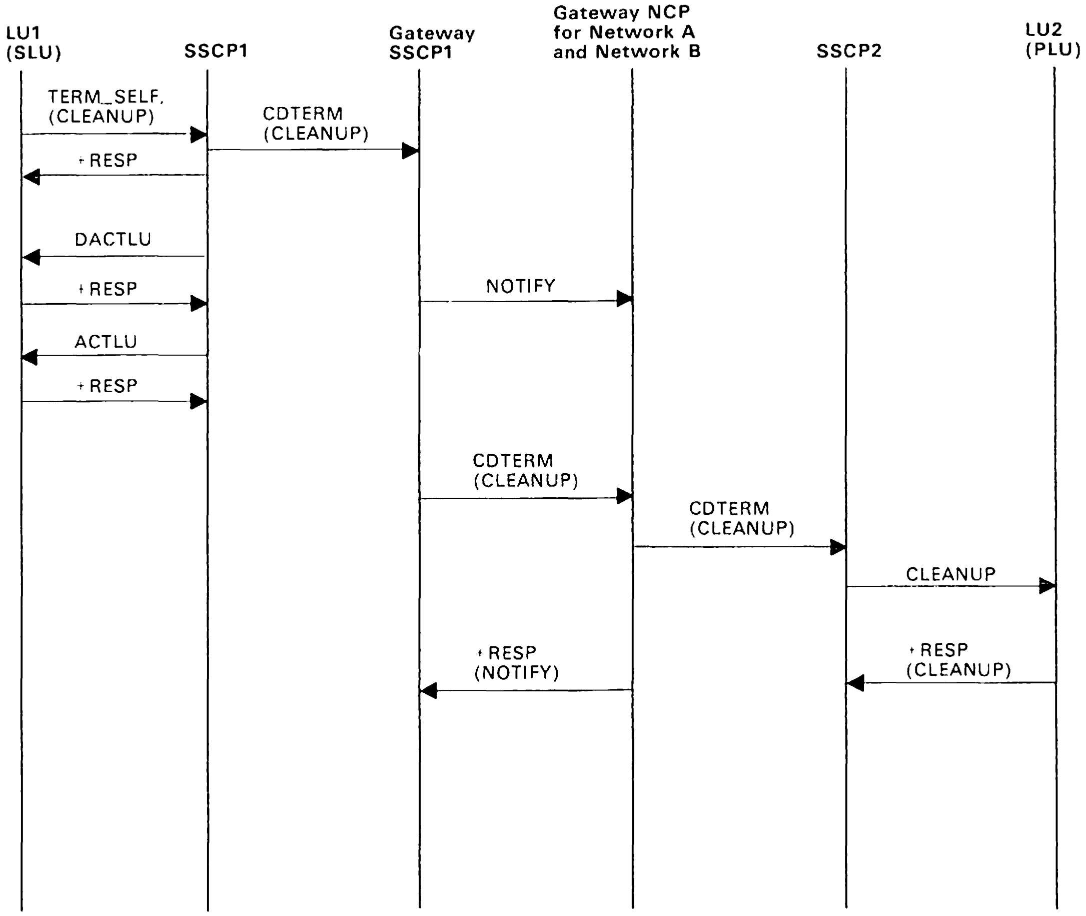


Figure 171. MVS VMDeactivating Resources and Ending Sessions: SLU Requests Terminate Self (Cleanup Option) for Single GWSSCP/GWNCP (V3R2)

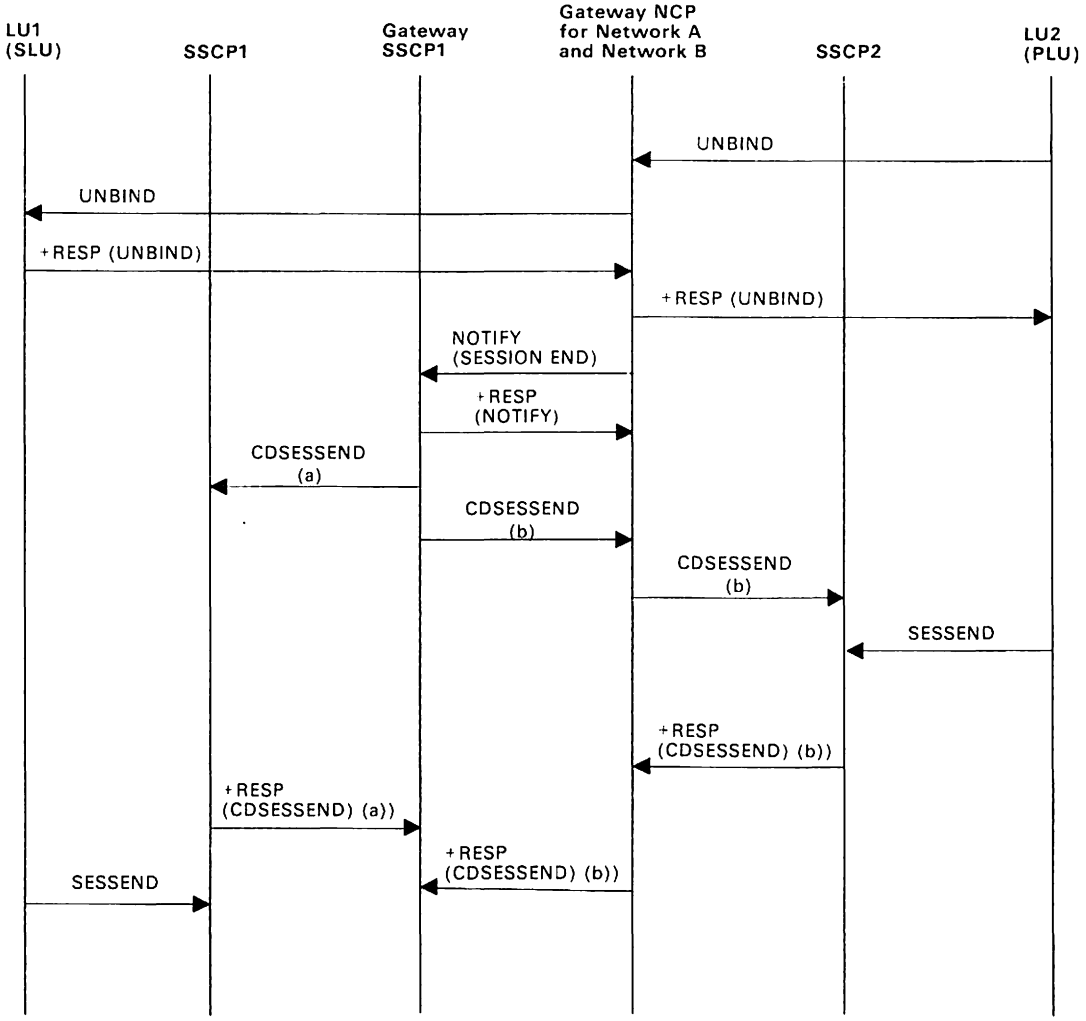


Figure 172. MVS VM Deactivating Resources and Ending Sessions: PLU Sends UNBIND for Multiple GWSSCPs and Single GWNCP

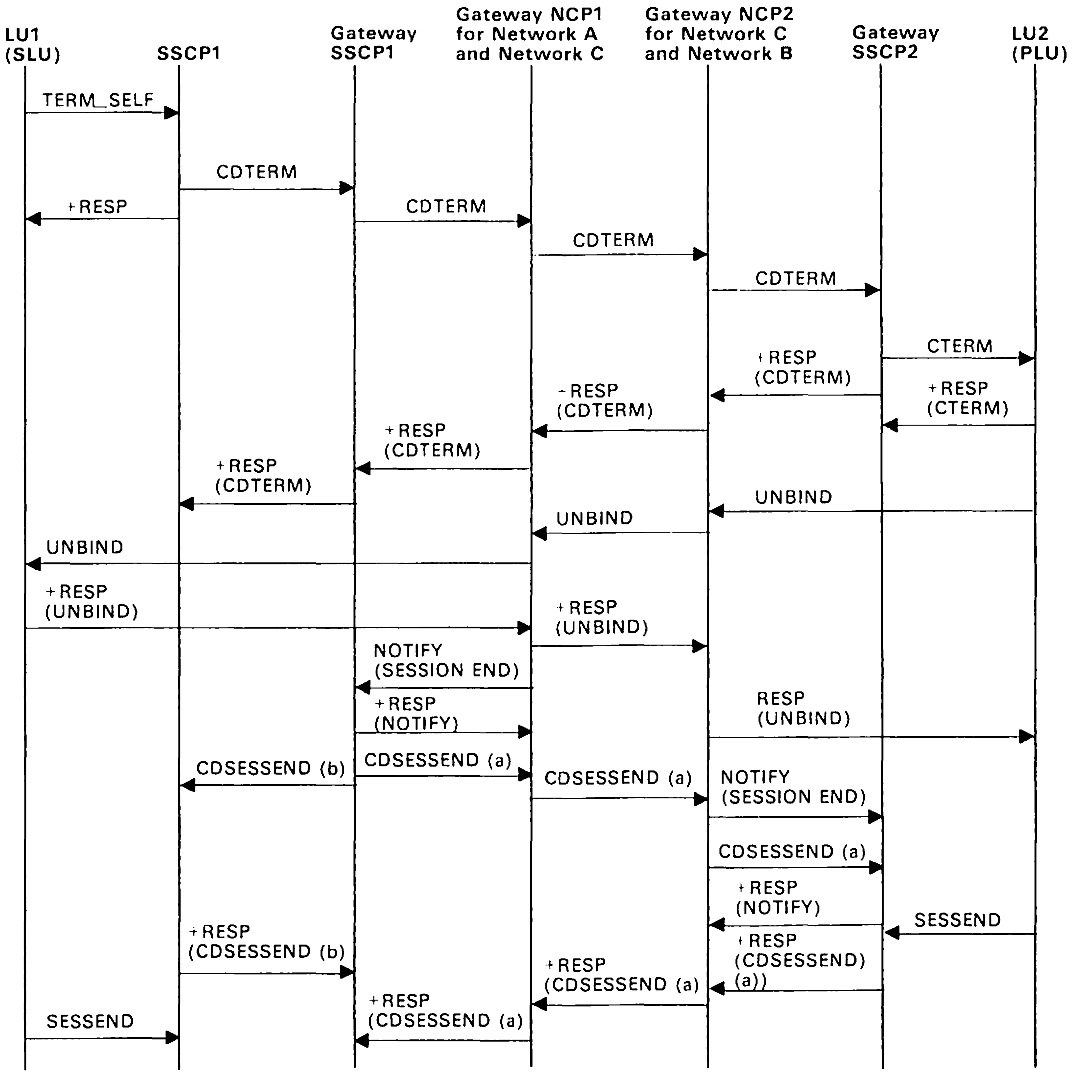
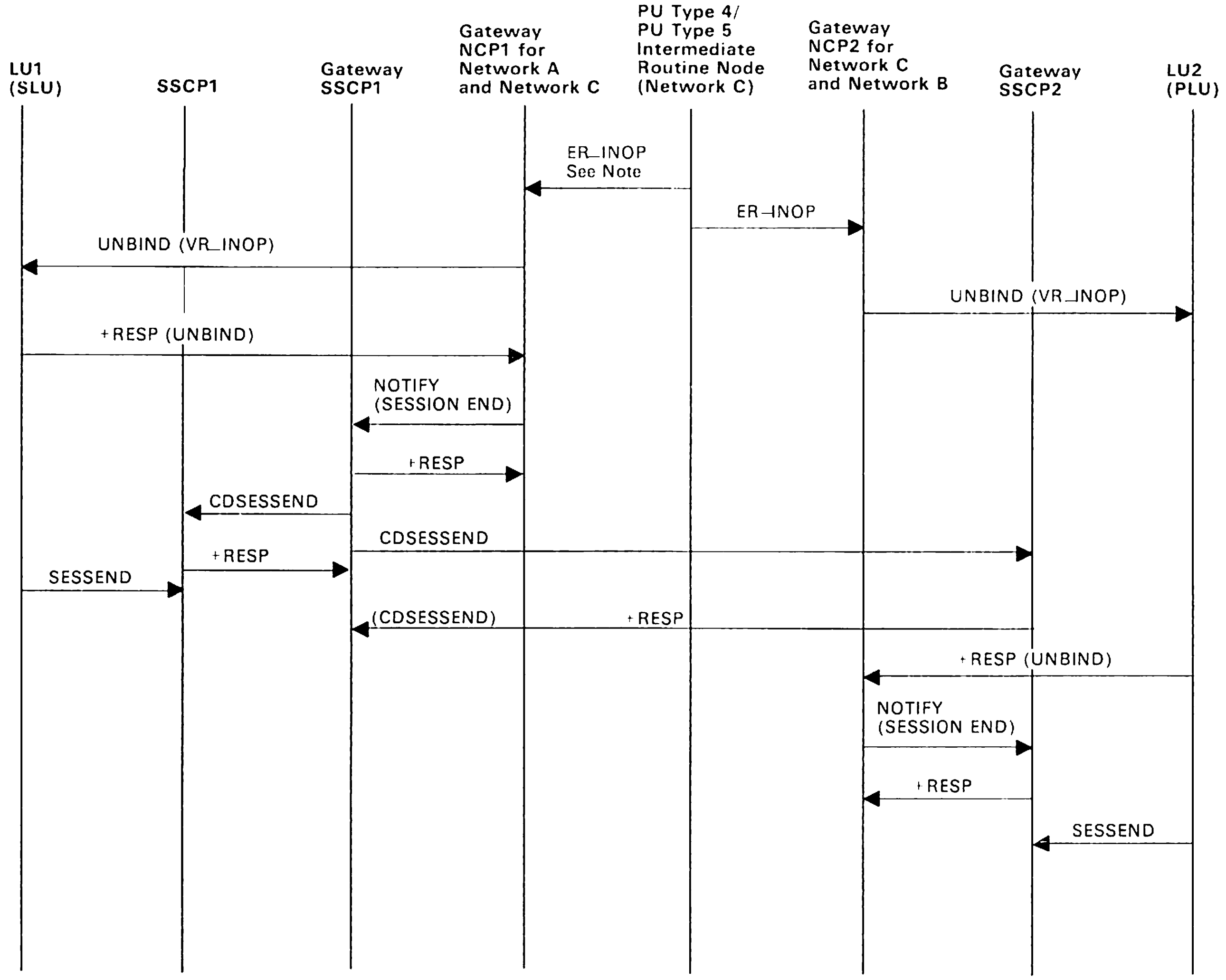


Figure 173. MVS VM Deactivating Resources and Ending Sessions: SLU Requests Terminate Self for Multiple GWSSCPs and Back-to-Back GWNCPs



Note: *An outage occurs on the route in Network C used by the LU1–LU2 session. ER_INOP reports the failure to gateway NCP1 and gateway NCP2.*

Figure 174. MVS VM Deactivating Resources and Ending Sessions: Terminating an SSCP-SSCP Session

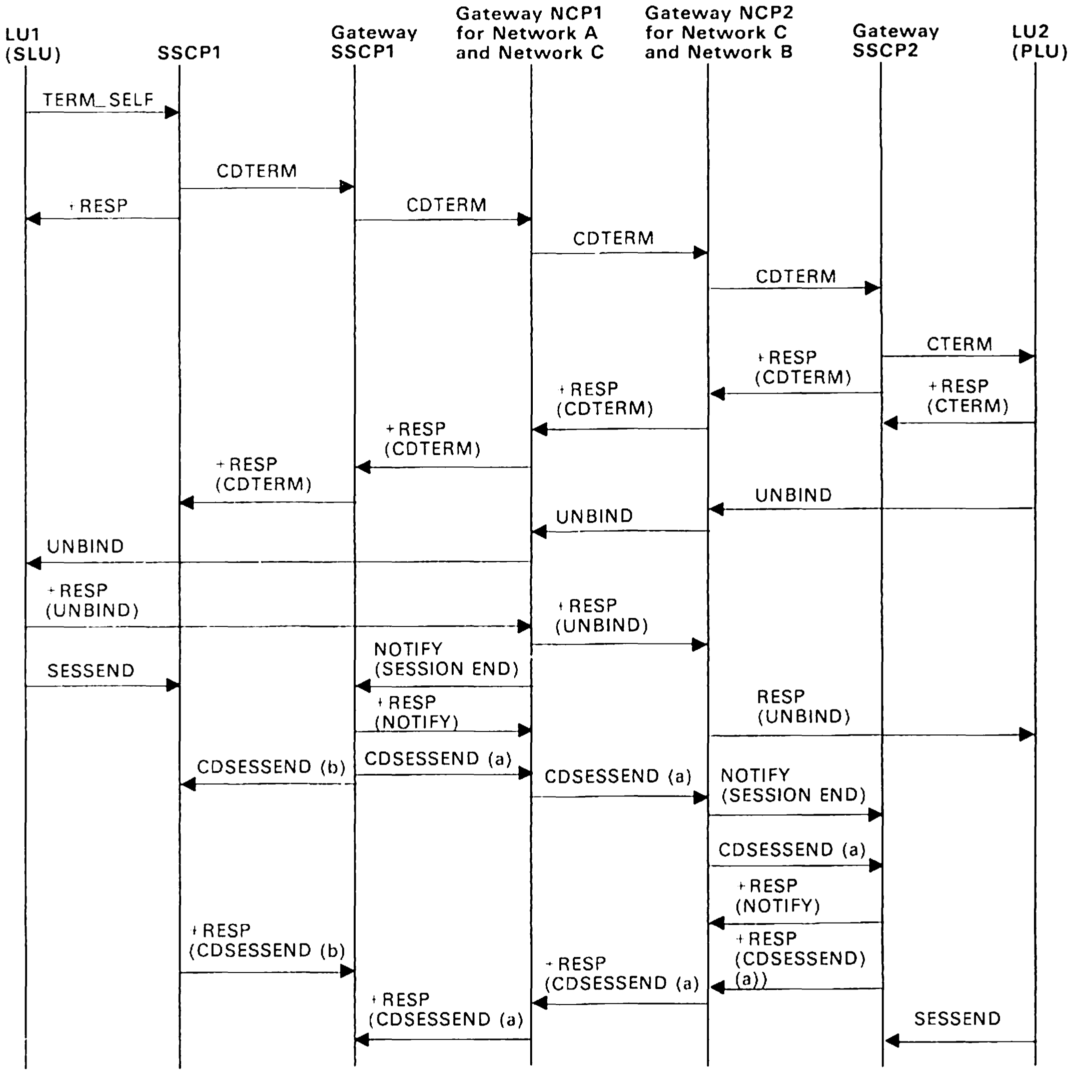


Figure 175. MVS VM Deactivating Resources and Ending Sessions: Active Session Termination of T2.1 Nodes

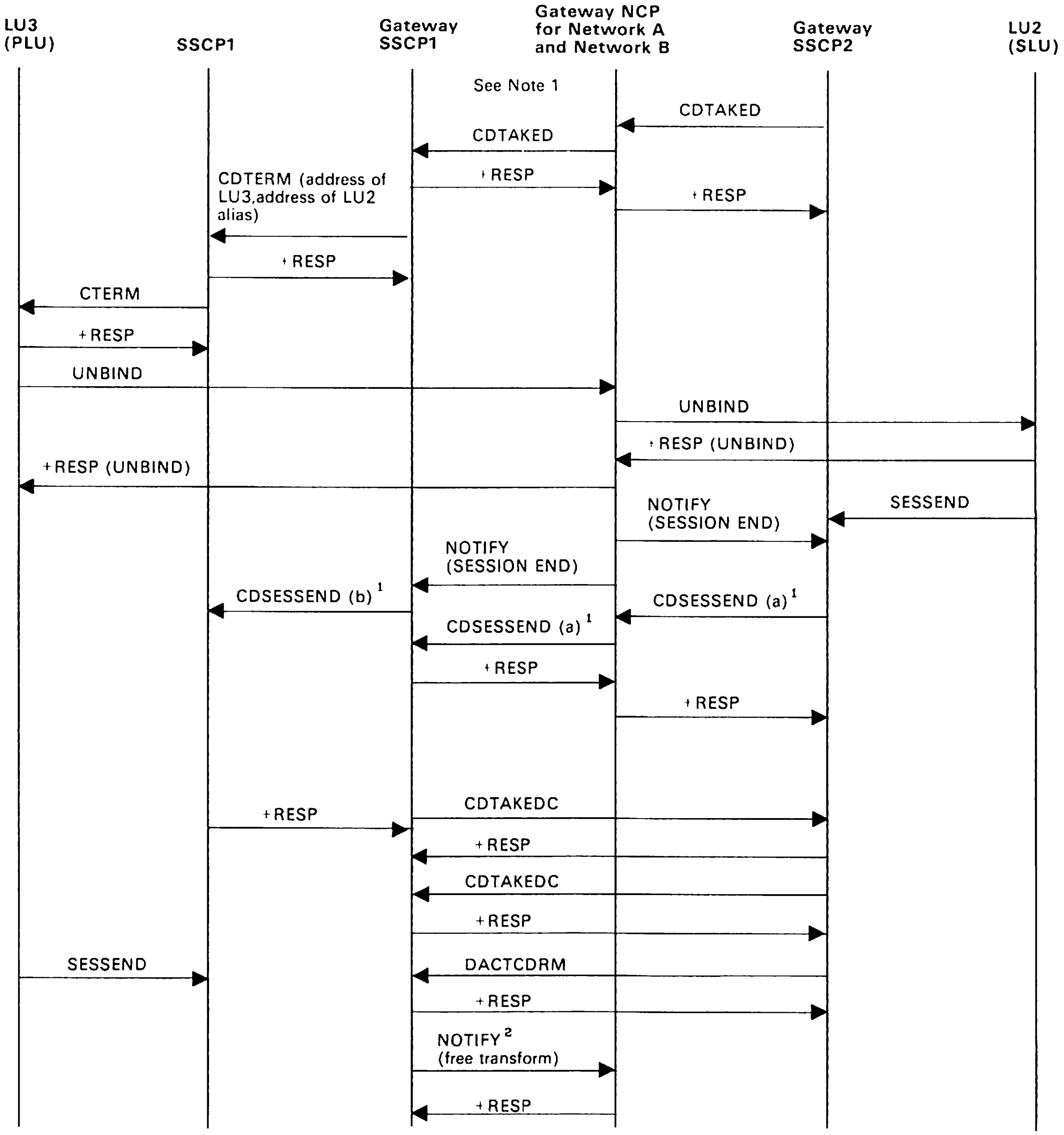


Figure 176. MVS VM Deactivating Resources and Ending Sessions: Route Failure Causes Session Termination (Pre-V3R2). **Note 1:** A cross-network LU-LU session exists between LU3 and LU2.

[1] See Figure 166 on page 633 through Figure 178 on page 644 for details of CDSESSEND processing.

[2] This flow assumes that the gateway SSCP1 established the network address translation for the gateway SSCP1-to-gateway SSCP2 session with the RNAA RU. The NOTIFY to free the transform is sent only if the RNAA that established the address specified "retain address."

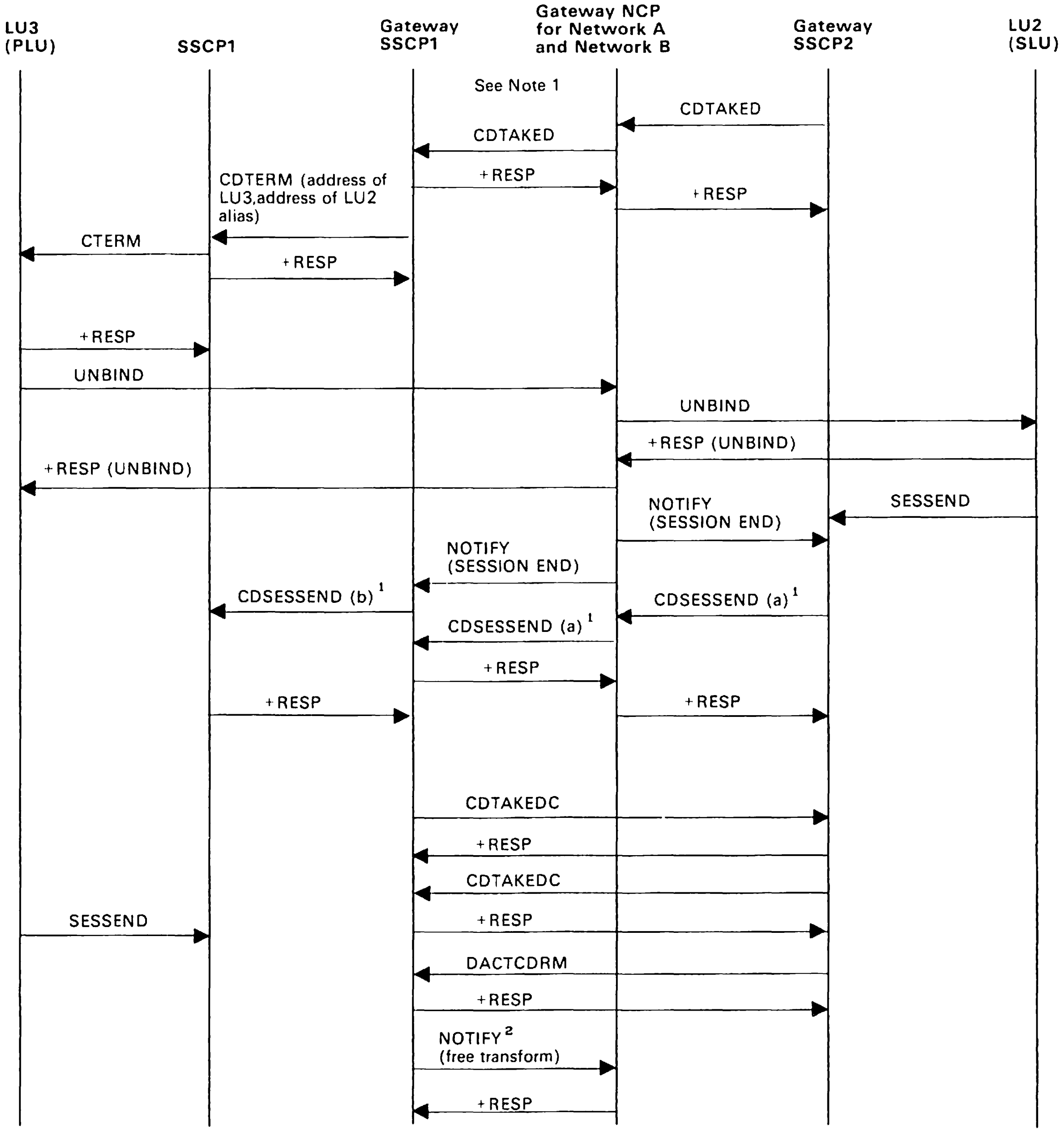


Figure 177. **MVS VM** Deactivating Resources and Ending Sessions: Route Failure Causes Session Termination (V3R2). **Note 1:** A cross-network LU-LU session exists between LU3 and LU2.

[1] See Figure 166 on page 633 through Figure 178 on page 644 for details of CDSESSEND processing.

[2] This flow assumes that the gateway SSCP1 established the network address translation for the gateway SSCP1-to-gateway SSCP2 session with the RNAA RU. The NOTIFY to free the transform is sent only if the RNAA that established the address specified "retain address."

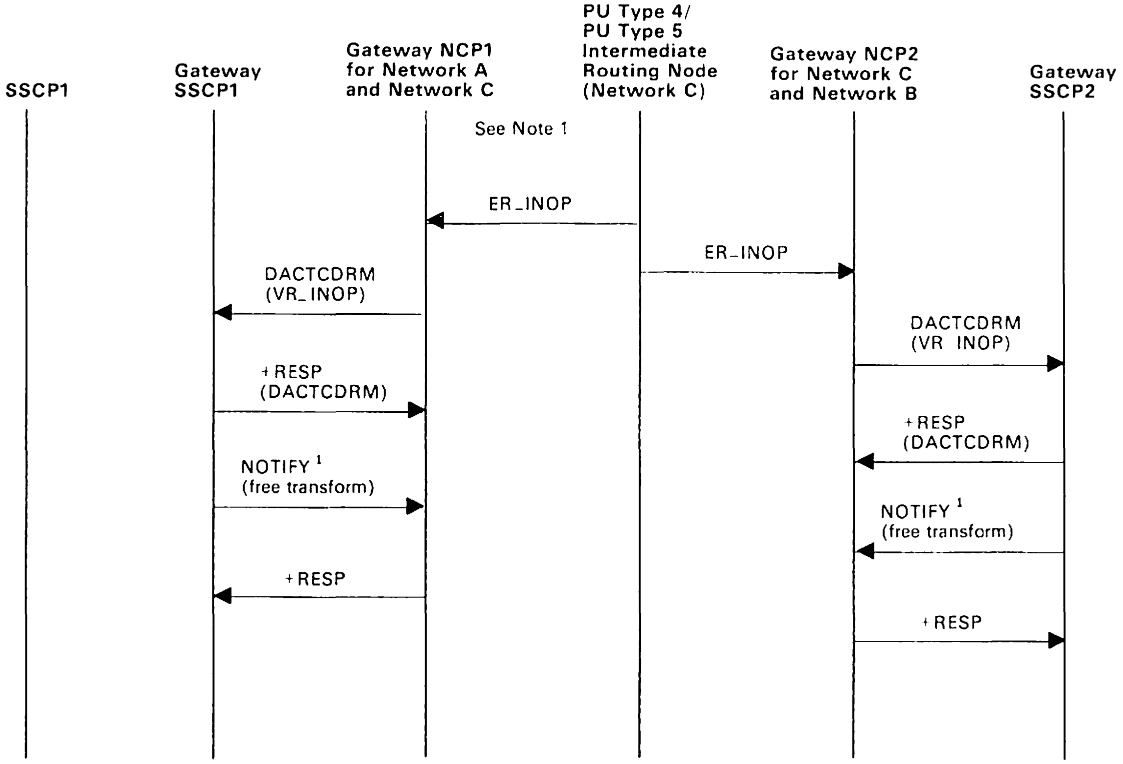

SSCP1
Gateway SSCP1
Gateway NCP1 for Network A and Network C
PU Type 4/ PU Type 5 Intermediate Routing Node (Network C)
Gateway NCP2 for Network C and Network B
Gateway SSCP2
See Note 1
ER_INOP
ER_INOP
DACTCDRM (VR_INOP)
DACTCDRM (VR INOP)
+RESP (DACTCDRM)
+RESP (DACTCDRM)
NOTIFY 1 (free transform)
NOTIFY 1 (free transform)
+RESP
+RESP

Figure 178. MVS VM Deactivating Resources and Ending Sessions: Route Failure Causes Session Termination for Multiple GWSSCPs and Back-to-Back GWNCPs.

Note 1: An outage occurs on route in Network C used by the gateway SSCP1-to-gateway SSCP2 session. The failure is reported to gateway NCP1 and gateway NCP2 with an ER INOP RU.

[1] The NOTIFY to free the transform is sent only if the RNAA that established the address specified "retain address."

Appendix D. CPCB Operation Codes

Many VTAM processes are represented by RUPE, NCSPL, DLRPL, CPCB, PPL, or TQE work elements. Each of these work elements contains a prefix called a CPCB at the beginning of the control block. The CPCB prefix contains a field called the CPCB operation code (CPCBOPC), which provides an exact indication of the type of VTAM process represented by the work element.

The CPCBOPC field is located at a displacement of hex 10 (decimal 16) from the beginning of each of the above-named control blocks. The CPCBOPC field is 4 bytes long and contains a category byte (CPCBCAT), followed by a 3-byte specific operation code field (CPCBFMH).

The possible contents of the CPCB operation code category are summarized below:

Category (hex)	Meaning
00	Operator command
01	Dump/load/restart request
02	I/O purge request
03	Timer management request
04	Unformatted RU
08	Function management data RU
09	Network control RU
0A	Data flow control RU
0B	Session control RU
0C	Function management data access method RU
0D	Network control access method RU
0E	Data flow control access method RU
0F	Session control access method RU

The following table summarizes the possible values of the CPCB operation code field. For each value, the character string representation which may appear in VTAM operator messages is given, along with the function of the associated work element.

CPCBOPC	Message	Function
00010000	VARY	VARY Command
00010001	VARY ACT	VARY Activate
00010002	VARY INACT	VARY Deactivate
00010004	VARY LOGON	VARY LOGON
00010005	VARY ACT	VARY Activate, LOGON Parameter Specified
00010008	VARY DRDS	VARY DRDS
00010012	VARY INACT	VARY Deactivate Immediate
00010032	VARY INACT	VARY Deactivate Immediate (Internal)
00010040	VARY ANS	VARY ANS
00010041	VARY ACT	VARY Activate, ANS Parameter Specified
00010045	VARY ACT	VARY Activate, ANS and LOGON Parameters Specified
00010100	VARY PATH	VARY PATH=USE
00010200	VARY PATH	VARY PATH=NOUSE
00010400	VARY INOP	VARY INOP
00010802	FORCE DEAC	Force Deactivate
00011002	FORCE REAC	Force Reactivate
00011802	VARY INACT	VARY Deactivate Giveback
00012000	VARY ACQ	VARY Acquire
00012001	VARY ACQ	VARY Activate, ACQ Parameter Specified
00012002	INACT SON	Deactivate (Session Outage Notification)
00012005	VARY ACQ	VARY Activate, ACQ and LOGON Parameters Specified
00014000	VARY REL	VARY Release
00014010	VARY REL	VARY Release Immediate
00018004	VARY NOLOGON	VARY NOLOGON
00020001	F DUMP	MODIFY DUMP
00020002	F ENCR	MODIFY ENCR
00020004	F CHANGE	MODIFY CHANGE

CPCBOPC	Message	Function
000200A0	F TRACE	MODIFY TRACE, NETCNTR
000200C0	F NOTRACE	MODIFY NOTRACE, NETCNTR
00020010	F TABLE	MODIFY TABLE
00020120	F AGPTRACE	MODIFY Activate generalized PIU trace
00020140	F DGPTRACE	MODIFY Deactivate generalized PIU trace
00020180	F LOAD ADD	MODIFY LOAD ADD
00020200	F CDRM	MODIFY CDRM
00020280	F LOAD REP	MODIFY LOAD REPLACE
00020401	TRANSFER NCP DUMP	Transfer DUMP (NCP)
00020408	F DR MOVE	MODIFY DR MOVE
00020480	F LOAD PRG	MODIFY LOAD PURGE
00020801	PRG NCP DUMP	PURGE DUMP (NCP)
00020808	F DR DELETE	MODIFY DR DELETE
00020820	F ALTRACE	MODIFY Activate Line Trace
00020840	F DLTRACE	MODIFY Deactivate Line Trace
00020880	F LOAD CNC	MODIFY LOAD CANCEL
00021001	F DUMP CSP	MODIFY DUMP (CSP)
00021801	PRG DUMP CSP	PURGE DUMP (CSP)
00022001	F DUMP MOS	MODIFY DUMP (MOSS)
00022801	PRG DUMP MOS	PURGE DUMP (MOSS)
00024001	F DUMP DYN	MODIFY Dump (Dynamic)
00024020	F ACT SIT	MODIFY Activate SIT Trace
00024040	F DACT SIT	MODIFY Deactivate SIT Trace
00028001	F DUMP DYN CH	MODIFY Dump (Dynamic-CH)
00028820	F ATGTRACE	MODIFY Activate TG Trace
00028840	F DTGTRACE	MODIFY Deactivate TG Trace
00030001	SOFT INOP	SOFT INOP
00030002	SSCP TKOVR	SSCP TAKEOVER
00030004	HARD INOP	HARD INOP
00040000	DISP INT	Display Internal Commands

CPCBOPC	Message	Function
00060001	INT SYNCH	Internal Synchronization Function
00060002	IPL INIT	IPL Initial Request
00060004	IPL TEXT	IPL Text Request
00060008	IPL FINAL	IPL Final Request
00060010	DUMP INIT	Dump Initial Request
00060020	DUMP TEXT	Dump Text Request
00060040	DUMP FINAL	Dump Final Request
00060100	HALT CDLNK	Process Cross-Domain Links During HALT
00061001	REQDMP CSP	Request CSP Dump
00062000	RESET LU	Reset LU
00062001	REQDMP MOS	Request MOSS Dump
00063001	INT TRF DH	Internal transfer of dump header
00063002	INT TRF DMS	Internal transfer of dump main storage
00064000	REQLOAD	Request Load
00064001	REQDUMP DY	Request Dynamic Dump Data
00068000	REQDUMP	Request Dump
00080001	DIAL START	Dial Start Request
012B0000	CKPTN	Checkpoint Node Status Function
014B0000	CHKPT	Checkpoint Resource Status Function
01BD0000	CPMSG	Internal WTOR Function
01DD0000	DLR PURGE	Dump/Load/Restart Purge
01EA0000	CPCRYPT	Cryptography Management Function
01ED0000	SELECT VR	Virtual Route Select
02510000	CDRM CLEAR	Clear CDRM-CDRM Session
02520000	CDRM ERP	CDRM ERP Internal Clear
03000000	TIMER REQ	Set Timer Request
04000000	CHAR CODED	Unformatted Request Unit
08010001	CHG TLIMIT	Change Transmission Limit
08010002	CHG NRSPOL	Change Negative Response to Poll Limit
08010003	CHG SESSLM	Change Session Limit
08010004	CHG POLLIM	Change Poll Limit

CPCBOPC	Message	Function
08010201	CONTACT	Contact
08010202	DISCONTACT	Discontact
08010203	IPL INIT	NC IPL Initial
08010204	IPL TEXT	NC IPL Text
08010205	IPL FINAL	NC IPL Final
08010206	DUMP INIT	Dump Initial
08010207	DUMP TEXT	Dump Text
08010208	DUMP FINAL	Dump Final
08010209	RMPO	Remote Power Off
0801020A	ACTLINK	Activate Link
0801020B	DACTLINK	Deactivate Link
0801020E	CONNOUT	Connect Out
0801020F	ABCONN	Abandon Connection
08010211	SCV	Set Control Vector
08010214	ENT SLOWDN	Enter Slowdown
08010215	EXT SLOWDN	Exit Slowdown
08010216	ACTCONNIN	Activate Connect In
08010217	DACTCONNIN	Deactivate Connect In
08010218	ABCONNOUT	Abandon Connect Out
08010219	ANA	Assign Network Address
0801021A	FNA	Free Network Address
0801021B	REQDISCONT	Request Discontact
08010280	CONTACTED	Contacted
08010281	INOP	Inoperative
08010284	REQCONT	Request Contact
08010285	NSLSA	Network Services Lost Subarea
08010302	ACTTRACE	Activate Trace
08010303	DACTTRACE	Deactivate Trace
08010331	DISP STOR	Display Storage
08010334	RECSTOR	Record Storage
08010381	RECMS	Record Maintenance Statistics

CPCBOPC	Message	Function
08010382	REC TEST	Record Test Data
08010383	REC TRACE	Record Line Trace Data
08010480	RECMD	Record Measurement Data
08010604	NSPE	Network Services Procedure Error
08010681	INIT SELF	Initiate(Self) Format 0
08010683	TERM SELF	Terminate(Self) Format 0
083F0233	INIT LOAD	NS Init Load
083F0234	LOAD STAT	NS Load Status
083F0814	TR-INQUIRY	Translate Inquiry
083F0816	TR-REPLY	Translate Reply
08410210	RNAA	Request Network Address Assignment
08410220	NFY SESSEND	Notify Sessend
08410237	LOAD REQRD	NS Load Required
08410240	ADDNR	Add Network Resource
08410243	IPL INIT	NS IPL Init
08410244	IPL TEXT	NS IPL Text
08410245	IPL FINAL	NS IPL Final
08410246	IPL ABORT	NS IPL Abort
08410286	RDELETENR	Request Delete Network Resource
08410287	LOST CTLPT	Lost Control Point
08410289	ROUTE-INOP	Network Services Route Inoperative
0841028A	REQACTCDRM	Request ACTCDRM
08410304	REQ MNT ST	Request Maintenance Statistics
08410305	LINKLVL2	Enter Test Mode (LL2)
08410307	REQ RTTEST	Request Route Test
08410311	MS SCV	Maintenance Services Set Control Vector
08410384	RECFMS	Record Formatted Maintenance Statistics
08410385	RECTR	Record Test Results
08410386	ER TESTED	Explicit Route Tested
08810387	REQ ECHO	Request Echo Test
08810389	ECHO TEST	Echo Test

CPCBOPC	Message	Function
08810601	CINIT	Control Initiate
08810602	CTERM	Control Terminate
08810620	NOTIFY	Notify
08810629	CLEANUP	Cleanup
08810680	INIT OTHER	Initiate(Other)
08810681	INIT SELF	Initiate(Self) Format 1
08810682	TERM OTHER	Terminate(Other)
08810683	TERM SELF	Terminate(Self)
08810685	BIND FAIL	Bind Failure
08810686	SESS START	Session Started
08810687	UBIND FAIL	Unbind Failure
08810688	SESS ENDED	Session Ended
08810810	FORWARD	Forward Request
08810812	DELIVER	Deliver Request
08810814	CNM REQ	CNM Request
08812601	BF CNTL INIT	BF Control Initiate
08812629	BF CLEANUP	BF Cleanup
08812681	BF INIT	BF Initiate
08812683	BF TERM	BF Terminate
08812686	BF SESS STRT	BF Session Start
08812688	BF SESS END	BF Session End
0881268C	BF SESS INFO	BF Session Information Request
08818620	CD NOTIFY	Cross-Domain Notify
08818627	CD DSEARCH	Cross-Domain Direct Search List
08818640	CDINIT OTH	Cross-Domain Initiate (Other)
08818641	CDINIT	Cross-Domain Initiate
08818643	CDTERM	Cross-Domain Terminate
08818645	CDSSF	Cross-Domain Session Setup Failure
08818646	CDSESSST	Cross-Domain Session Started
08818647	CDSTF	Cross-Domain Session Takedown Failure
08818648	CDSESSEND	Cross-Domain Session Ended

CPCBOPC	Message	Function
08818649	CDTAKEDOWN	Cross-Domain Takedown
0881864A	CDTD COMP	Cross-Domain Takedown Complete
0881864B	CDCINIT	Cross-Domain Control Initiate
09050000	NCLSA	Network Control Lost Subarea
09060000	ER INOP	Explicit Route Inoperative
09070000	ANSC	Auto Network Shutdown Complete
09080000	LOST PATH	Lost Path
09090000	ER TEST	Explicit Route Test
090A0000	ER TST RPY	Explicit Route Test Reply
090B0000	ER ACT	Explicit Route Activate
090C0000	ER ACT RPY	Explicit Route Activate Reply
090D0000	ACTVR	Activate Virtual Route
090E0000	DACTVR	Deactivate Virtual Route
090F0000	ER OP	Explicit Route Operative
09510000	SW TO NCP	Switch Line to NCP Mode
09520000	SW TO EP	Switch Line to EP Mode
0A040000	LUSTAT	LU Status
0A050000	RTR	Ready to Receive
0A700000	BIS	Bracket Initiation Stopped
0A710000	SBI	Stop Bracket Initiation
0A800000	QEC	Quiesce at End of Chain
0A810000	QC	Quiesce Complete
0A820000	RELQ	Release Quiesce
0A830000	CANCEL	Cancel
0A840000	CHASE	Chase
0AC00000	SHUTDOWN	Shutdown
0AC10000	SHUTC	Shutdown Complete
0AC20000	RSHUTD	Request Shutdown
0AC80000	BID	Bid
0AC90000	SIGNAL	Signal
0B0D0000	ACTLU	Activate LU

CPCBOPC	Message	Function
0B0E0000	DACTLU	Deactivate LU
0B110000	ACTPU	Activate PU
0B120000	DACTPU	Deactivate PU
0B140000	ACTCDRM	Activate CDRM
0B150000	DACTCDRM	Deactivate CDRM
0B310000	BIND	Bind Session
0B320000	UNBIND	Unbind Session
0BA00000	SDT	Start Data Traffic
0BA10000	CLEAR	Clear Session
0BA20000	STSN	Set and Test Sequence Numbers
0BA30000	RQR	Request Recovery
0BC00000	CRV	Cryptography Verify
0C0102A0	AM ALLORSC	Allocate Resource
0C0102A1	AM FREERSC	Free Resource
0C0102A2	AM SETRT	Set Routable State
0C0102A3	AM RESETRT	Reset Routable State
0C010480	AM RCRD MS DATA	Record Measurement Data
0C410206	AM GAINED GWN	Gained GWN
0C410207	AM LOST GWN	Lost GWN
0C410208	AM DACT TRF	Deactivate Transforms
0C410210	AM RNAA	Request Network Address Assignment
0C410212	AM CONN	Connect
0C410213	AM DISC	Disconnect
0C410214	AM INIT PU	Initiate PU
0C410266	XID	AM Exchange ID
0C4102BD	AM ADDLINK	Add Link
0C4102BE	AM ADDLSTA	Add Link Station
0C4102BF	DELETENR	Delete Network Resource
0C4102CD	AM REQDUMP	Request Dump
0C4102CE	AM CONDLOD	Request Conditional Load

CPCBOPC	Message	Function
0C4102CF	AM UNCDLOD	Request Unconditional Load
0C410601	AM OPNACB	Open ACB
0C410602	AM CLSACB	Close ACB
0C810619	AM ARCA	Address Request Complete
0C810620	AM RESUME	Resume
0C810629	AM CLEANUP	Cleanup
0C810643	Generic TERM	Termination Placeholder
0C810680	AM REALLOC	Reallocate
0C810681	GEN SESS INIT	Generic Session Initiation
0C810801	AM NOTIFY	Notify
0C810A00	API SETLST	SETLOGON(START)
0C810A01	API SETLSP	SETLOGON(STOP)
0C810A02	API SETLQS	SETLOGON(QUIESCE)
0C810A10	API SIMLOG	SIMLOGON
0C810A20	API OPNACQ	OPNDST(ACQUIRE)
0C810A21	API OPNACC	OPNDST(ACCEPT)
0C810A30	API INQLOG	INQUIRE(LOGONMSG)
0C810A31	API INQDVC	INQUIRE(DEVCHAR)
0C810A32	API INQCNT	INQUIRE(COUNTS)
0C810A33	API INQTOP	INQUIRE(TOPLOGON)
0C810A34	API INQCID	INQUIRE(CIDXLATE)
0C810A35	API INQTRM	INQUIRE(TERMS)
0C810A36	API INQAPS	INQUIRE(APPSTAT)
0C810A37	API INQSPM	INQUIRE(SESSPARMS)
0C810A38	API INQSKY	INQUIRE(SESSKEY)
0C810A39	API INQDSP	INQUIRE(DISPLAY)
0C810A40	API INTERP	INTRPRET
0C810A50	API CLSPAS	CLSDST(PASS)
0C810A51	API CLSRLS	CLSDST(RELEASE)
0C810A60	API SESONC	SESSIONC
0C810A70	API SNDCMD	SENDCMD

CPCBOPC	Message	Function
0C810A75	API SEND	SEND
0C810A80	API RCVCMD	RCVCMD
0C810A85	API RECEIVE	RECEIVE
0C810A90	API REQSES	REQSESS
0C810AA0	API OPNSEC	OPNSEC
0C810AB0	API TRMSES	TERMSESS
0C810AC0	API RESETSR	RESETSR
0D010000	AM VR INOP	Virtual Route Inoperative
0D0B0000	AM REQ ERA	Request Explicit Route Activate
0D0E0000	AM REQ VRD	Request Virtual Route Deactivate
0DFF0000	AM VR STAT	Virtual Route Status
0E010000	AM PCE	Purge Chain Element
0F010000	AM NFY SLT	Notify (Schedule LOSTERM Exit)
0F020000	AM SSA	Set Session Address
0F030000	AM SSADISC	Set Session Address and Disconnect
0F040000	AM OSA	Override Session Address
0F050000	AM PWQ	Purge Wait Queue
0F060000	AM FLUSH	Flush Virtual Route
0F310000	AM GBIND	Generic BIND
0F320000	AM GUNBIND	Generic UNBIND
FF000000	VECTOR	VECTOR FMD request units

Appendix E. Storage and Control Block ID Codes

This appendix lists the control block ID codes for VTAM and VSCS control blocks. It also includes storage ID codes for VSCS.

This appendix contains the following sections:

- "VTAM Control Block ID Codes" on page 658
- "VM VSCS Storage and Control Block Identifiers" on page 659.

VTAM Control Block ID Codes

You can identify certain VTAM control block types in a storage dump by examining an identification code in the first byte of the control block (offset 0). The control block identification codes are shown in the following table. In case of duplicate codes, use other means (such as the operating environment or the control block's context) to determine the type of control block.

ID	Control Block	ID	Control Block
00	RPL	50	DLRPL
00	SONCB	52	LUCB
01	RPH	54	RUPE
03	FMCB	54	TUNB
05	VRBLK	57	PPL
06	ICNCB	58	TQE
07	LDNCB	5F	CPCB
09	HALCB	60	NCSPL
0A	BSCLB	61	PST
0B	VLNCB (V3R1.2 and later)	62	LU 6.2 (V3R2)
0C	PCLCB (V3R1.2 and later)	62	NSICB, NSSCB
0D	PRWCB (V3R1.2 and later)	63	SMP
0E	TRGCB (V3R1.2 and later)	65	NACP
0F	ACDEB	68	RSQE
10	UECB	69	DTPL
11	DYPAB	6B	DSSIB (V3R2)
13	TRAC	6C	IOSIB (V3R2)
14	ERTE	6D	ASRIT
18	TIE	6E	GWIT
19	PDVT	75	WRE
1A	CHAIN	77	ADJSR
1B	TGCB	81	EXLST
24	OCW	96	SIBX
25	VCCB (V3R1.2 and later)	97	SIBIX (V3R2)
26	PICB (V3R1.2 and later)	98	SIB
29	LMPCB	99	TSCB
2B	RAQ	9A	PSPL (V3R1.2 and later)
2C	PAQ	9A	TSPL
2D	SAT	A0	ACB
3B	LICB (V3R1.2 and later)	A2	RNCA
41	PUSCB	A3	ALCA
43	PLSCB	BF	COS
45	POIA	C0	LOGMD
46	POCB	CA	GRPCB (V3R1.2 and later)
47	POMCB	D0	NIB
48	PORCB	EB	AUTOE
49	POWE	FE	XCNCB
4A	MALT	FF	OCA
4C	ERCT	FF	PLL (V3R1.2 and later)
4D	TGE	$PIO	PIO
4E	VRWSE		

Figure 179. Control Block ID Codes

VM VSCS Storage and Control Block Identifiers

You can identify certain VSCS control block types in a storage dump by examining an identification code in the first byte of the control block.

Storage Identifiers

ID	Storage Type
SS	Static storage
SD	Dynamic storage
ZZ	Dummy storage prefix

Control Block Identifiers

ID	Control Block
CG	DTICGB
CI	DTICIA
CL	DTICLB
CM	DTICMD
DTICXWB	DTISWBs for Communication Services
DTIPSWB	DTISWBs for Presentation Services
DTISPSB	DTISWBs for Presentation Services Abends
DTISUTB	DTISWBs for Utility Services Abends
DTISVSB	DTISWBs for VTAM Services Abends
DTITXWB	DTISWBs for Timer Services
DTIUSWB	DTISWBs for Utility Services
DTIVSWB	DTISWBs for VTAM Services
DTIVXWB	DTISWBs for VTAM Exit Services
IN	DTIWEB internal
PB	DTIPDB
PD	DTIDSB for Presentation Services
PG	DTIPGB
PL	DTIPLB
SA	DTISAB
SM	DTISDB
TA	DTITAB
TD	DTIGTD
TR	DTITHDR
UG	DTIUGB
VD	DTIDSB for VTAM Services
VE	DTIVEIB
VG	DTIVGB
VL	DTIVLB
4I	DTIWEB IREAD internal read buffer
1R	DTIWEB IRECEIVE one-way reply

ID	Control Block
2R	DTIWEB IRECEIVE one-way send
1S	DTIWEB ISEND one-way reply
2S	DTIWEB ISEND one-way send
3R	DTIWEB IRECEIVE two-way reply
4R	DTIWEB IRECEIVE two-way send
3S	DTIWEB ISEND two-way reply
4S	DTIWEB ISEND two-way send

Appendix F. PIU Discard Reason Codes

VTAM path information unit (PIU) discard reason codes can be found in NetView trace panels and in the VTAM Internal Trace (VIT) DSCD entry. When VTAM's transmission subsystem component (TSC) cannot forward a PIU towards its final destination, it generates a discard trace entry, consisting of a discard reason code and the discarded PIU. The TSC sends the discard trace entry to the NetView program, or to the VIT, or to both.

Note: NetView must be active to VTAM in order to accept discard trace entries. The VIT must also be active.

To reference all discard entries sent to NetView, issue the following NetView command:

SESS *DISCARD

NetView puts discarded PIUS in this file and, when possible, also puts them in the associated session trace file.

Because PIUS are often discarded without causing problems in the network, most discarded PIUS are not indicative of error conditions and do not need to be investigated. A discarded PIU should be investigated only if an error symptom, such as a "hang" condition, occurs.

For a description of NetView trace panels, see *NetView Installation and Operation*.

See Chapter 8, "Using the VTAM Internal Trace" on page 307 for a full description of the VIT DSCD entry.

Discard Reason Codes

Figure 180 lists the VTAM discard reason codes for each HS and TSC module that discards PIUs. HS and TSC modules, with their associated reason codes, are listed in alphanumeric order. For each reason code, a specific reason is given for the discard, followed by a probable cause. Probable causes are described in more detail in Figure 181 on page 670.

VTAM Module	Reason Code	Reason for Discard	Probable Cause
ISTHSCEP	0001	The session is being deactivated. Unsent PIUs must be discarded.	Should not occur
	0002	The session is being deactivated. PIUs on the pacing queue must be discarded.	Should not occur
	0003	The session is being deactivated. PIUs on the HSPAB must be discarded.	
	0004	The session is being deactivated. Expedited flow PIUs must be discarded.	Should not occur
ISTHSCGI	0001	Either insufficient resources, or LRM encounters an error while processing.	Should not occur
ISTTSCNS (No Session PAB Processor)	0001	No processing is required for this FID0 PIU.	Timing consideration with BSC 3270
ISTHSCRT	0001	A PIU was received from another VTAM component on the expedited work queue that is not a BIND or UNBIND.	
	0002	An SC PIU was received from the network that is not a BIND or UNBIND.	Should not occur
ISTHSCRU	0001	Insufficient resources.	Should not occur
ISTHSCRV	0001	Format error on received request PIU.	Should not occur
	0002	Format error on received response PIU.	Should not occur
	0003	PIU received in incorrect state.	Should not occur
	0004	Insufficient resources.	Should not occur
ISTHSCSN	0001	Chaining state error.	Should not occur
ISTHSCSR	0001	Chaining state error.	Should not occur
ISTHSCTP	0001	Insufficient resources.	Should not occur
	0002	Pacing response received when pacing is not supported.	Should not occur
ISTHSCTR	0001	Received PIU caused Transmission Control protocol violation.	Should not occur

Figure 180 (Part 1 of 8). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause
ISTTSCBR	0001	Invalid BDU command code.	
	0002	No response to the command in progress, or the command in progress has been sent again.	
	0003	Invalid control command.	
	0004	Not an error response, but SET MODE failed in processing completion of select BHSET command.	
	0005	CLEAR command in progress failed.	
ISTTSCCN (V3R1.2 and later)	0001	Device reset.	
ISTTSCCR (Connection Point Manager Receive)	0001	The request TSCB is a PRI.DT.SEND or a SEC.DT.RCV in pending reset state.	Timing on LU-LU session traffic
	0002	Inbound FMD request when SSCP-LU session is not active.	Timing on SSCP-LU session traffic
ISTTSCGR (General Response Routine)	0001	Response could not be generated because the PIU was FID0, was not first in the segment, was already a response, or a response was not allowed by RH.	Should not occur
ISTTSCHT (V3R1.2 and later)	0001	PIU is discarded from the PUT pending traffic queue.	
	0002	PIU is discarded from the NCB pending traffic queue.	
	0003	PIU is discarded from the NCB work queue.	
	0004	PIU is discarded from the NCB inbound router PAB.	
ISTTSCIP (Host IRN Processor)	0001	The route is not available to the destination.	Post-failure traffic purge
	0002	A response cannot be generated.	Post-failure traffic purge

Figure 180 (Part 2 of 8). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause
ISTTSCIS (Inbound Session Control Function Interpreter)	0001	The response is discarded when PRI.SESS.SEND is not in a pending active state.	Timing on LU-LU session traffic.
	0002	The response is discarded when PRI.DT.SEND is not in a pending active state.	Timing on LU-LU session traffic.
	0003	The response is discarded when PRI.DT.SEND is not in a pending reset state.	Timing on LU-LU session traffic.
	0004	The response is discarded when RQR.SEND is not in a pending state.	Timing on LU-LU traffic
	0005	The response is discarded when PRI.SESS.SEND or SEC.SESS.RCV is not in an active state.	Timing on LU-LU session traffic
	0006	The session control response is invalid.	Should not occur — logical unit error
	0007	Last in segment not received.	Should not occur — logical unit error
ISTTSCIU (Host IRN Utility Processor)	0001	The link is not contacted and active.	Post-failure traffic purge
	0002	No IRN extension exists on the NCB.	Post-failure traffic purge
ISTTSCJS (V3R1.2 and later)	0001	PIUs discarded from BCT pending traffic queue for logical unit session control request.	
	0002	PIUs discarded from BSCLB waiting printer end queue for logical unit session control request.	
	0003	PIUs discarded from BCT pending traffic queue for PU session control request.	
	0004	PIUs discarded from BSCLB waiting printer end queue for PU session control request.	

Figure 180 (Part 3 of 8). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause
ISTTSCJT (V3R1.2 and later)	0001	PIU is discarded from the BCT pending traffic queue.	
	0002	PIU is discarded from the BSCLB waiting printer end queue.	
	0003	PIU is discarded from the PCPAB work queue.	
	0004	PIU is discarded from the BSCLB inbound router PAB.	
ISTTSCLS (Local 3270 Write Scheduler)	0001	The link is not yet active.	Post-failure traffic purge
	0002	The request was purged because of the session control request that followed.	Timing on LU-LU session traffic
VM ISTTSCMB (Migration BIND, UNBIND, ACTCDRM, DACTCDRM Processor)	0001	This is not a migration route.	Should not occur, or route has been redefined.
	0002	The ICNCB was not found.	Post-failure traffic purge
	0003	The virtual route (VR) number and the transmission priority (TP) number are not 0, as they must be for migration.	Should not occur —VR and TP numbers must be 0 for migration
ISTTSCNC (Network Control Function Interpreter)	0001	The initialization complete request is the only inbound request allowed from network control.	Should not occur — error in network
ISTTSCNS (No Session PAB Processor)	0001	No processing is required for this FIDO PIU.	Timing consideration with BSC 3270
ISTTSCPI (Path Control Inbound Processor)	0001	The RU size is exceeded, and the PIU is not for session control.	Logical unit problem — logical unit sent a PIU larger than the BIND-specified maximum
ISTTSCPR (Path Control Router)	0001	A response cannot be generated for an IRN PIU.	Post-failure traffic purge
ISTTSCQD (Session Deactivation Request Processor)	0001	A contention ACTCDRM was discarded because a DACTCDRM arrived from the same subarea.	Operation — operator deactivated CDRM from one host after activation
ISTTSCQF (V3R1.2 and later)	0001	Abend recovery PIU discard	

Figure 180 (Part 4 of 8). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause
ISTTSCQP (Session Deactivation Processor)	0001	an ACTCDRM was discarded due to ACTCDRM contention session restart.	Operation — CDRM activated from both hosts simultaneously
ISTTSCRA (Abend Recovery)	0001	This is not a "first-in-segment" PIU.	Post-failure traffic purge after application program abend
ISTTSCRI (Virtual Route Path Control Inbound)	0001	The origin subarea is invalid.	Should not occur — adjacent NCP error
	0002	The route is not usable.	Post-failure traffic purge — ER not operative
	0003	The sequence number is invalid.	Should not occur — lost data or logical unit error
ISTTSCRO (Virtual Route Path Control Outbound)	0001	The route is not active.	Post-failure traffic purge
	0002	The activation number does not match the VR activation number.	Post-failure traffic purge
ISTTSCSA (Segment Assembler)	0001	The beginning of the PIU segment was lost.	Should not occur — network error
	0002	Two "First-in-Segment" PIUs were received. Discard the partial PIU segment and retransmit the PIU.	Should not occur — check logical unit
	0003	Hot I/O. Discard all PIU segments for this session.	Should not occur — check logical unit

Figure 180 (Part 5 of 8). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause
ISTTSCSR (Session Router)	0001	Incorrect virtual route ID.	Should not occur — PIU originator error
	0002	A complete DACTCDRM request caused by contention resolution, session override, or other session outage notification.	Network operator error or transmission group failure
	0003	Session deactivation response and state is not pending reset.	Should not occur — out of synchronization. Possibly misrouted.
	0004	Negative session activation response and flow direction does not match the direction indicator.	Should not occur — PIU originator error
	0005	Negative session activation response and invalid sequence number.	Should not occur — out of synchronization or lost traffic
	0006	Negative session activation response and state is not pending active.	Out of synchronization — multiple session activation and deactivation running concurrently
	0007	The session control response PIU is not flagged as expedited flow.	Protocol violation by the response originator

Figure 180 (Part 6 of 8). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause
ISTTSCTS (Boundary Function Transmission Control Inbound)	0001	(SSCP,PU).BF.SESS.RCV is not pending active for the ACTPU response. VTAM was not waiting for an ACTPU response.	Should not occur — out of synchronization on SSCP-PU session
	0002	(SSCP,PU).BF.SESS.RCV is not pending reset for the DACTPU response. VTAM was not waiting for a DACTPU response.	Should not occur — out of synchronization on SSCP-PU
	0003	(SSCP,SLU).BF.SESS.RCV is not pending active for the ACTLU response. VTAM was not waiting for an ACTLU response.	Should not occur — out of synchronization on SSCP-LU session
	0004	(SSCP,LU).BF.SESS.RCV is not pending reset for the DACTLU response. VTAM was not waiting for a DACTLU response.	Should not occur — out of synchronization on SSCP-LU session
	0005	(PLU,SLU).BF.SESS.RCV is not pending active for the BIND response. VTAM was not waiting for a BIND response.	Timing between SSCP sessions and LU-LU session — SSCP session reset during logical unit session activation
	0006	(PLU,SLU).BF.SESS.RCV is not pending active for the UNBIND response. VTAM was not waiting for an UNBIND response.	Timing between SSCP sessions and LU-LU session — SSCP session reset during logical unit session deactivation
ISTTSCWS (Communication Controller/Cluster Controller Write Scheduler)	0001	The channel link has not been contacted.	Post-failure traffic purge
ISTTSCXS (Channel-to-Channel Attachment Write Scheduler)	0001	The channel link is not in contacted or active state.	Post-failure traffic purge
ISTTSC3R (3270 Virtual PU/LU Transmission Control Outbound)	0001	DT.RCV is not active.	Timing on LU-LU session traffic
	0002	The session is being reset.	Timing on LU-LU session traffic

Figure 180 (Part 7 of 8). PIU Discard Reason Codes

VTAM Module	Reason Code	Reason for Discard	Probable Cause
ISTTSC3S (3270 Virtual PU/LU Transmission Control Inbound)	0001	The session was not active, and the PIU received did not contain a session control RU.	Timing on LU-LU session traffic
	0002	(SSCP,PU).SEC.SESS.RCV is not pending active for the ACTPU response. VTAM was not waiting for an ACTPU response.	Should not occur — out of synchronization on SSCP-PU session
	0003	(SSCP,PU).SEC.SESS.RCV is not pending reset for the DACTPU response. VTAM was not waiting for a DACTPU response.	Should not occur — out of synchronization on SSCP-PU session
	0004	(SSCP,SLU).SEC.SESS.RCV is not pending active for the ACTLU response. VTAM was not waiting for an ACTLU response.	Timing between SSCP-PU and SSCP-LU sessions
	0005	(SSCP,SLU).SEC.SESS.RCV is not pending reset for the DACTLU response. VTAM was not waiting for a DACTLU response.	Timing between SSCP-PU and SSCP-LU sessions
	0006	(PLU,SLU).SEC.SESS.RCV is not pending active for a BIND response. VTAM was not waiting for a BIND response.	Timing between SSCP sessions and LU-LU session — SSCP session reset during LU-LU session activation
	0007	(PLU,SLU).SEC.SESS.RCV is not pending reset for the UNBIND response. VTAM was not waiting for an UNBIND response.	Timing between SSCP sessions and LU-LU session — SSCP session reset during LU-LU session deactivation
	0008	For an LU-LU session, the DATATRAFFIC_RCV is not active.	Timing on LU-LU session traffic
ISTTSC6F		No discard codes from this module. Using TSALERT with reason code 0 for normal PIU discard.	
ISTTSC6Z (V3R1.2 and later)	0001	End of PIU lost	Should not occur — error in network
	0002	Retransmission of PIU, discard the partial PIU.	Should not occur — error in network
	0003	Segmenting length error	Should not occur — error in network
	0004	Queue freed — segmenting error	Should not occur — error in network

Figure 180 (Part 8 of 8). PIU Discard Reason Codes

Probable Causes of Discards

Probable Cause	Explanation of Cause
Timing on LU-LU session traffic	This discard can occur on the session between the application program and the SLU. It is usually caused by a reset request (for example, a Clear or a BIND) from the application program. This type of discard does not usually cause a problem in the network.
Timing between SSCP sessions and LU-LU session	This discard usually occurs when a reset of the SSCP-PU or SSCP-LU session affects a pending initiation or termination of the LU-LU session. For example, if the application program issues a CLSDST request to the logical unit, VTAM sends an UNBIND request to the logical unit. If the SSCP-LU session fails before the UNBIND response comes back, the UNBIND response is discarded. This type of discard does not usually cause a problem in the network.
Timing between SSCP-PU and SSCP-LU sessions	This discard usually occurs when a reset of the SSCP-PU session affects a pending initiation or termination of the SSCP-LU session. For example, if the operator deactivates the logical unit, VTAM sends a DACTLU to the logical unit. If the SSCP-PU session fails before the DACTLU response comes back, the DACTLU response is discarded. This type of discard does not cause a problem in the network.
Post-failure traffic purge	This discard may be caused by a network failure. Any PIU traffic arriving after the network failure but before the session is terminated can be purged using the discard function. For example, after a channel I/O error, PIUs that continue to be queued outbound over the channel can be discarded. This type of discard does not usually cause a problem in the network.
Should not occur	This discard indicates a hardware or software error in the network; it usually occurs after a major SNA protocol violation. For example, if VTAM is waiting for a sequence number of 0008 but receives a PIU with sequence number 0010, PIUs with sequence numbers 0008 and 0009 are missing. The missing PIUs could have been lost anywhere in the session path, or perhaps they were not sent. The logical unit, physical unit, line, NCP, channel, or VTAM might be responsible for the problem. Further investigation is usually required to determine where in the network the problem is. The session type and reason code determine what type of documentation is needed to resolve a problem identified by this discard code.

Figure 181. Probable Causes of Discards

Bibliography

VTAM Books

The following paragraphs briefly describe the library of books for VTAM V3R2. Several of these books also contain information about VTAM V3R1.2 for VM and VSE, V3R1.1 for MVS and VM, and V3R1 for VSE.

VTAM Installation and Resource Definition (SC23-0111)

This book enables a system programmer to install VTAM, define the resources in the network to VTAM, code start options, and test VTAM definitions. The appendixes in this book enable system programmers to quickly locate the detailed syntax of the macroinstructions and definition statements used to define a network to VTAM, as well as VTAM start options.

VTAM Customization (LY30-5614)

The objective of this book is to enable a system programmer to customize VTAM and tune it for better performance. It discusses modifying VTAM messages; modifying VTAM USS commands, installation exit routines, and replaceable modules; and tuning VTAM.

VTAM Directory of Programming Interfaces for Customers (GC31-6403)

This book provides a directory of all programming interfaces provided by VTAM for use by customers. It also identifies and specifies limitations on the use of certain "product-sensitive" interfaces, which are dependent on the detailed design and implementation of VTAM.

VTAM Operation (SC23-0113)

This is a reference book for VTAM network operators. It is also a guide for system programmers who must supply operators with the detailed information that they need to run the VTAM network.

This book includes an introduction to operating VTAM, a description of VTAM commands, and a description of how to use VTAM commands to perform network control functions.

VTAM Messages and Codes (SC23-0114)

This book contains, in alphanumerical order, all messages and codes issued by VTAM. These messages include VTAM messages for network operators, TSO/VTAM messages for network operators, TSO/VTAM messages for terminal users, USS messages for terminal users, and VSCS messages.

This book can be inserted into the operating system messages book, if desired, or used as a stand-alone book.

VTAM Programming (SC23-0115)

This book describes how to use VTAM macroinstructions to send data to and receive data from (1) a terminal in either the same or a different domain, or (2) another application program in either the same or a different domain. Also included is a dictionary of VTAM macroinstructions. This book assumes that the reader is familiar with assembler language and the programming facilities of the operating system.

VTAM Programming for LU 6.2 (SC30-3400)

This book describes VTAM's LU 6.2 programming interface for host application programs. This book pertains to programs that use only LU 6.2 sessions or that use LU 6.2 sessions along with other session types. (Only LU 6.2 sessions are covered in this book, however.)

VTAM Diagnosis (LY30-5601)

This book assists system programmers in identifying a VTAM problem, classifying it, and collecting information about the problem in preparation for calling the IBM Support Center. The information collected includes traces, dumps, and other problem documentation.

VTAM Data Areas for MVS (LY30-5592)

VTAM Data Areas for VM (LY30-5593)

VTAM Data Areas for VSE (LY30-5594)

These books describe VTAM data areas and can be used to read a VTAM dump. They are intended for IBM programming service representatives and customer personnel who are diagnosing problems with VTAM.

VTAM Reference Summary (LY30-5600)

This book is designed as a quick reference for system programmers and network programmers. This book contains selected reference information that includes VTAM and VSCS commands, VTAM definition statements, VTAM start options, VTAM macroinstructions, VTAM and VSCS trace formats, and selected SNA reference data.

VTAM V3R1.2 Books

VTAM Expanded Network Capabilities Support (LD21-0019)

This supplement is available as a stand-alone book for V3R1.2; the information has also been included in the V3R2 books.

VTAM Directory of Programming Interfaces for Customers (GC31-6402)

This book provides a directory of all programming interfaces provided by VTAM for use by customers. It also identifies and specifies limitations on the use of certain "product-sensitive" interfaces, which are dependent on the detailed design and implementation of VTAM.

VTAM V3R1.1 Books

You may order additional copies of the VTAM V3R1.1 books by the following order numbers:

VTAM Installation and Resource Definition (SC23-0111)

VTAM Customization (SC23-0112)

VTAM Operation (SC23-0113)

VTAM Message and Codes (ST23-0114)

VTAM Programming (SC23-0115)

VTAM Diagnosis Guide (SC23-0116)

VTAM Diagnosis Reference (LY30-5582)

VTAM Data Areas for MVS (LY30-5584)

VTAM Data Areas for VM (LY30-5583)

VTAM Reference Summary (SC23-0135)

VTAM V3R1 Books

You may order additional copies of the VTAM V3R1 books by using the following pseudonumbers:

VTAM Installation and Resource Definition (ST23-0110)

VTAM Customization (ST23-0112)

VTAM Operation (ST23-0113)

VTAM Programming (ST23-0115)

VTAM Diagnosis Reference (LT70-5582)

VTAM Message and Codes (ST23-0114)

You can still order the following VTAM V3R1 books by their original order numbers:

VTAM Messages and Codes for VM (SC30-3275)

VTAM Data Areas for MVS (LY30-5581)

VTAM Data Areas for VM (LY30-5580)

VTAM Data Areas for VSE (LY30-5579)

VTAM Library Supplement for X.21 SHM/MPS (SD21-0010)

Other Network Program Products Books

For more information about the books listed in this section, see the *Network Program Products Bibliography and Master Index.*

The following list shows the cross-product books for VTAM, NetView, NCP, SSP, and NetView/PC.

Network Program Products General Information (GC30-3350)

Network Program Products Planning (SC30-3351)

Network Program Products Samples (SC30-3352)

Network Program Products Bibliography and Master Index (GC30-3353)

Network Program Products Storage Estimates (SC30-3403)

NetView Books

The following list shows the books associated with Release 2 of the NetView program.

Learning about NetView (SK2T-0292)

NetView Installation and Administration Guide (SC30-3476)

NetView Administration Reference (SC30-3361)

NetView Customization (SC30-3462)

NetView Directory of Programming Interfaces for Customers (GC31-6005)

NetView Command Lists (SC30-3423)

NetView Operation Primer (SC30-3363)

NetView Operation (SC30-3364)

NetView Messages (SC30-3365)

NetView Operation Scenarios (SC30-3376)

NetView Command Summary (SX27-3620)

NetView Diagnosis (LY30-5587)

NetView Hardware Problem Determination Reference (SC30-3366)

NetView Problem Determination Supplement for Management Services Major Vectors 0001 and 0025 (LD21-0023)

NetView 5822 Supplement (SD21-0016)

Automated Operations Books

Automated Operations Planning Guide (SC30-3474)

Automated Operations Using NetView Command Lists (SC30-3477)

NetView/PC Books

The following list shows the books for the NetView/PC program.

NetView/PC Planning and Operation Guide (SC30-3408)

NetView/PC Installation Guide (SC30-3482)

NetView/PC Application Program Interface/Communications Services Reference (SC30-3313)

NCP V4R2 Books

The following list shows the books for NCP V4R2 and SSP V3R2.

NCP and SSP Generation and Loading Guide (SC30-3348)

NCP and SSP Migration (SC30-3252)

NCP and SSP Resource Definition Guide (SC30-3349)

NCP and SSP Resource Definition Reference (SC30-3254)

NCP and EP Reference Summary and Data Areas (LY30-5570)

NCP and SSP Customization (LY30-5571)

NCP and SSP Messages and Codes (SC30-3169)

NCP and SSP Diagnosis Guide (LY30-5591)

NCP Reference (LY30-5569)

SSP Diagnosis Reference (LY30-5564)

EP Installation, Resource Definition, and Diagnosis (SC30-3338)

NCP V4R3 Books

The following list shows the books for NCP V4R3 and SSP V3R3.

NCP, SSP, and EP Generation and Loading Guide (SC30-3348)

NCP Migration Guide (SC30-3252)

NCP, SSP, and EP Resource Definition Guide (SC30-3349)

NCP, SSP, and EP Resource Definition Reference (SC30-3254)

NCP and EP Reference Summary and Data Areas (LY30-5570)

NCP Customization Guide (LY30-5571)

NCP Customization Reference (LY30-5612)

SSP Customization (LY43-0021)

NCP and Related Products Directory of Programming Interfaces for Customers (GC31-6202)

NCP, SSP, and EP Messages and Codes (SC30-3169)

NCP, SSP, and EP Diagnosis Guide (LY30-5591)

NCP and EP Reference (LY30-5569)

NCP Version 5 Books

The following list shows the books for NCP Version 5 and SSP V3R3.

NCP, SSP, and EP Generation and Loading Guide (SC30-3348)

NCP Migration Guide (SC30-3440)

NCP, SSP, and EP Resource Definition Guide (SC30-3447)

NCP, SSP, and EP Resource Definition Reference (SC30-3448)

NCP and EP Reference Summary and Data Areas (LY30-5603)

NCP Customization Guide (LY30-5606)

NCP Customization Reference (LY30-5607)

SSP Customization (LY43-0021)

NCP and Related Products Directory of Programming Interfaces for Customers (GC31-6202)

NCP, SSP, and EP Messages and Codes (SC30-3169)

NCP, SSP, and EP Diagnosis Guide (LY30-5591)

NCP and EP Reference (LY30-5605)

Network Routing Facility Books

The following list shows the books for the Network Routing Facility.

NRF General Information (GC27-0594)

NRF Planning (SC27-0593)

NRF Resource Definition and Customization (SC30-3407)

NRF Diagnosis (LY30-5597)

NRF Trace Record Template (SX27-3711)

MVS/370 Books (MVS/SP Version 1)

IBM System/370 Principles of Operation, GA22-7000

Operator's Library: OS/VS2 MVS System Commands, GC28-1031

OS/VS2 SPL: Supervisor, GC28-1046

OS/VS2 Supervisor Services and Macros, GC28-1114

MVS/370 System Messages, Volume 1, GC28-1374; Volume 2, GC28-1375

OS/VS Message Library: VS2 System Codes, GC28-1008

OS/VS Message Library: VS2 Routing and Descriptor Codes, GC38-1102

MVS/370 Programming Library: Debugging Handbook, Volume 1, LC28-1385; Volume 2, LC28-1386; Volume 3, LC28-1387; Volume 4, LC28-1388; Volume 5, LC28-1389

OS/VS2 SPL: MVS Diagnostic Techniques, SY28-1133

OS/VS2 SPL: Debugging Handbook, Volume 1, GC28-1047; Volume 2, GC28-1048; Volume 3, GC28-1049

MVS IPCS User's Guide and Reference, GC28-1183

OS/VS2 SPL: Service Aids, GC28-0674

Environmental Record Editing and Printing User's Guide and Reference, GC28-1378

OS/VS2 SPL: SYS1.LOGREC Error Recording, GC28-0677

OS/VS2 TSO Guide to Writing a Terminal Monitor Program or a Control Program, GC28-0648

MVS/XA Books (MVS/SP Version 2)

MVS/System Product Version 2 General Information Manual, GC28-1118

MVS/Extended Architecture System Programming Library: System Macros and Facilities, Volume 1, GC28-1150; Volume 2, GC28-1151

MVS/Extended Architecture Diagnostic Techniques, LY28-1199

MVS/Extended Architecture Debugging Handbook, Volume 1, LC28-1164; Volume 2, LC28-1165; Volume 3, LC28-1166; Volume 4, LC28-1167; Volume 5, LC28-1168

MVS/Extended Architecture SYS1.LOGREC Error Recording, GC28-1162

MVS/Extended Architecture Interactive Problem Control System User's Guide and Reference, GC28-1297

MVS/Extended Architecture Message Library: System Codes, GC28-1157

MVS/Extended Architecture Message Library: System Messages, Volume 1, GC28-1376; Volume 2, GC28-1377

Environmental Record Editing and Printing User's Guide and Reference, GC28-1378

MVS/Extended Architecture System Programming Library: Service Aids, GC28-1159

MVS/Extended Architecture TSO Guide to Writing a Terminal Monitor Program or Command Processor, GC28-1295

MVS/Extended Architecture TSO Terminal Users Guide, GC28-1274

VM/SP Books (Release 5)

VM/SP Operator's Guide, SC19-6202-4

VM/SP System Messages and Codes, SC19-6204-4

VM/SP Terminal Reference, GC19-6206-4

VM/SP CP Command Reference, SC19-6211-4

VM/SP Group Control System Command and Macro Reference, SC24-5250-1

VM/SP System Messages Cross-Reference, SC24-5264-1

VM/SP Problem Reporting Guide, SC24-5282-2

VM/SP Group Control System Diagnosis Reference, LX24-5239-1

VM/SP System Logic & Problem Determination Guide, Volume 1 (CP), LY20-0892-4; Volume 2 (CMS), LY20-0893-4

VM/SP Data Areas and Control Block Logic, Volume 1 (CP), LY24-5220-3; Volume 2 (CMS), LY24-5221-2

VM Diagnosis Guide, LY24-5241-0

VSE/System Package Books (Release 3.1.1)

IBM VSE/SP Messages and Codes, SC33-6310

IBM VSE/SP Guide for Solving Problems, SC33-6311

VSE/Advanced Functions Books (Version 2)

VSE/Advanced Functions System Management Guide, SC33-6191

VSE/Advanced Functions Operation, SC33-6194

VSE/Advanced Functions Diagnosis: Service Aids, SC33-6195

VSE/Advanced Functions Application Programming: Macro User's Guide, SC33-6196

VSE/Advanced Functions Application Programming: Macro Reference, SC33-6197

VSE/Advanced Functions System Control Statements, SC33-6198

VSE/Advanced Functions Maintain System History Program, Reference, SC33-6199

VSE/System Package, Messages and Codes, SC33-6181

VSE/System Package, Diagnosis: Guide, SC33-6182

Host Processor Books

IBM 4361 Processor Communication Adapter, GA33-1571

IBM 9370 Information System: Telecommunications Subsystem Description and Reference, SA24-4041.

List of Abbreviations

ACB. Application control block.

ACDEB. VTAM data extent block.

ACF. Advanced Communications Function.

ACF/NCP. Advanced Communications Function for the Network Control Program.

ACF/SSP. Advanced Communications Function for the System Support Programs.

ACF/TAP. Advanced Communications Function for the Trace Analysis Program.

ACF/TCAM. Advanced Communications Function for the Telecommunications Access Method.

ACF/VTAM. Advanced Communications Function for the Virtual Telecommunications Access Method.

ACTLINK. Activate Link.

ACTLU. Activate logical unit.

ACTPU. Activate physical unit.

ANA. Assign network address.

AP. Alternate printer.

API. Application program interface.

APPC. Advanced Program-to-Program Communication.

APPL. Application program.

APUNS. Adjacent physical unit network services.

ASCII. American National Standard Code for Information Interchange.

ATCVT. VTAM communication vector table.

AVT. Address vector table.

BFT. Boundary function table.

BHSET. Block handler routines.

BIU. Basic information unit.

BNN. Boundary network node.

BSC. Binary synchronous communication.

BTU. Basic transmission unit.

CA. Channel adapter.

CA/CS. Continue any/continue specific.

CC. Command chaining.

CCP. Configuration control program facility.

CCS. Console communication services.

CCW. Channel command word.

CDRM. Cross-domain resource manager.

CDRSC. Cross-domain resource.

CDS. Configuration data set.

CDSESSEND. Cross-domain session ended.

CDSESSST. Cross-domain session started.

CGB. Communication services global block.

CICS. Customer Information Control System.

CID. Communication identifier.

CLB. Communication services local block.

CLSDST. Close destination.

CMS. Conversational Monitor System.

CNM. Communication network management.

CNOS. Change number of sessions.

CONFT. VTAM configuration table.

COS. Class of service.

CP. Control program.

CPCB. Control point control block.

CRA. Component recovery area.

CRR. Component recovery record.

CRV. Cryptography verification.

CS. Current state.

CSA. Common service area.

CSP. Communication scanner processor.

CSW. Channel status word.

CT. Control terminal.

CUA. Channel unit address.

CVT. Communication vector table.

DACTLINK. Deactivate link.

DACTLU. Deactivate logical unit.

DACTPU. Deactivate physical unit.

DAF. Destination address field.

DAN. Direct attachment node.

DBCS. Double-byte character set.

DC. Data chaining.

DCE. Data circuit-terminating equipment.

DET. Device entry table.

DEV. Device address field.

DFC. Data flow control.

DLC. Data link control.

DLRPL. Dump/load/restart parameter list.

DLU. Destination logical unit.

DOC. Documentation.

DPN. Destination program name.

DR. Definite response.

DRDS. Dynamic reconfiguration data set.

DS. Desired state.

DSB. Dispatcher/scheduler block

DTE. Data terminal equipment.

DVT. Destination vector table.

EBCDIC. Extended binary-coded decimal interchange code.

EC. Engineering change.

ECB. Event control block.

ED. Enciphered data.

EIA. Electronic Industries Association.

EIB. External interrupt block.

EID. Event identifier.

EIS. External interrupt support.

EL. Error loop.

EOT. End-of-transmission.

EP. Emulation Program.

EPCF. Extended primary control field.

ER. Explicit route.

ERACT. Error action.

ERCOD. Error code.

EREP. Environmental recording editing and printing.

ERP. Error recovery procedure.

E/T. Error-to-traffic.

EXLST. Exit list.

EXR. Exception request.

FDX. Full duplex.

FIC. First-in-chain.

FID. Format identification.

FIFO. First-in first-out.

FMCB. Function management control block.

FMD. Function management data.

FMH. Function management header.

FNA. Free network address.

FSSM. Full screen support mode.

GCS. Group control system.

GPT. Generalized path information unit trace.

GTD. Global timer data block.

HPO. High Performance Option.

GTF. Generalized trace facility.

GVM. Guest virtual machine.

HIO. Halt I/O.

ICNCB. Intelligent controller node control block.

ICW. Interface control word.

ID. Identification.

IMR. Intensive mode recording.

IMS. Information Management System/Virtual Storage.

IMS/VS. Information Management System/Virtual Storage.

I/O. Input/output.

IOPD. Input/output problem determination.

IPCS. Interactive problem control system.

IRN. Intermediate routing node.

IUCV. Inter-user communication vehicle.

JCL. Job control language.

LCB. Local block common.

LCD. Line control definer.

LDNCB. Local device node control block.

LFSM. Link function state machine.

LIFO. Last-in first-out.

LL2. Link level 2.

LMPEO. Large message performance enhancement outbound.

LPA. Link pack area.

LQAB. Lock queue anchor block.

LSID. Local session identification.

LU. Logical unit.

LUCB. Logical unit control block.

LUS. Logical unit services.

LUST. Logical unit status table.

LU 6.2. Logical unit 6.2.

MDR. Miscellaneous data record.

MI. Manual intervention.

MIC. Middle-in-chain.

MID. Machine identifier.

MOSS. Maintenance and operator subsystem.

MSG. Console messages.

MSHP. Maintain system history program.

MVS. Multiple Virtual Storage operating system.

MVS/XA. Multiple Virtual Storage for Extended Architecture operating system.

NAB. Network address block.

NAU. Network addressable unit.

NC. Network control.

NCB. Node control block.

NCCF. Network Communications Control Facility.

NCP. Network Control Program.

NCSPL. Network configuration services parameter list.

NFY. Notify.

NIB. Node initialization block.

NLDM. Network Logical Data Manager.

NMVT. Network management vector transport.

NPDA. Network Problem Determination Application.

NPR. No print.

NR. Negative response.

NRF. Network Routing Facility.

NS. Network services.

NSI. Non-SNA Interconnection.

NTO. Network Terminal Option.

OACBRU. Open ACB request/response unit.

OAF. Origin address field.

OBR. Outboard record.

OIC. Only-in-chain.

OLU. Origin logical unit.

OPNDST. Open destination.

OPT. Option.

OS. Operating system.

PAB. Process anchor block.

PC. Path control.

PCB. Pool control block.

PCF. Primary control field.

PCID. Procedure-correlation identifier.

PDB. Pool descriptor block.

PDF. Parallel data field.

PEP. Partitioned emulation program.

PERFM. Performance.

PGB. Presentation services global block.

PIB. Process information block.

PIU. Path information unit.

PLB. Presentation services local block.

PLM. Program logic manual.

PLU. Primary logical unit.

PP. Parallel print.

PR. Print error.

PROGCK. Program check.

PRN. Primary resource name.

PSS. Process scheduling services.

PST. Process scheduling table.

PSW. Program status word.

PTF. Program temporary fix.

PU. Physical unit.

PUNS. Physical unit services.

PUSCB. Physical unit services control block.

PUT. Program update tape.

PVI. Primitive VTAM interface.

PXB. Pool extension block.

QAB. Queue anchor block.

RAS. Reliability, availability, and serviceability.

RCA. Resident common area.

RDT. Resource definition table.

RDTE. Resource definition table entry.

REA. Request for Engineering Activity.

RECFMS. Record formatted maintenance statistics.

RECMS. Record maintenance statistics.

REGS. Registers.

REX. Route extension.

RFT. Request for test.

RH. Request/response header.

RNAA. Request network address assignment.

RPH. Request parameter header.

RPL. Request parameter list.

RTI. Response type indicator.

RTM. Response time monitor.

RU. Request/response unit.

RUPE. Request/response unit processing element.

SAB. Service application block.

SADP. Serviceability aids and debugging procedures.

SAF. Source address field.

SAW. Session awareness.

SBA. Set buffer address.

SC. Session control.

SCF. Secondary control field.

SCT. Section control table.

SDA. Sense data included.

SDB. Storage descriptor block.

SDF. Serial data field.

SDLC. Synchronous Data Link Control.

SDT. Start data traffic.

SDVT. Skeleton destination vector table.

SDWA. System diagnostic work area.

SESSEND. Session ended.

SESSER. Session serialization.

SESSST. Session started.

SIB. Session information block.

SIO. Start I/O.

SIT. Scanner interface trace.

SLT. Section list table.

SLU. Secondary logical unit.

SMF. System management facility.

SMP. System Modification Program.

SMP/E. System Modification Program Extended.

SMS. Storage management services.

SNA. Systems Network Architecture.

SNI. SNA network interconnection.

SP. Service point.

SPANC. Storage pool anchor block.

SPCS. Service point command service.

SRT. Symbol resolution table.

SRTD. Symbol resolution table directory.

SRTE. Symbol resolution table entry.

SS. Start-stop.

SSCP. System services control point.

SSP. System Support Programs.

STD. Set control vector time and date.

STSN. Set and test sequence numbers.

SVA. Shared virtual area.

SVC. Switched virtual circuit.

SWB. Save/Work block.

SYSDEF. System definition.

SYSREC. System error file.

TAB. Terminal anchor block.

TAP. Trace analysis program.

TC. Transmission control.

TCAM. Telecommunications Access Method.

TCAS. Terminal control address space.

TCB. Task control block.

TECT. Temporary error counter.

TG. Transmission group.

TH. Transmission header.

TIB. Task information block.

TIC. Token-ring interface coupler.

TID. Task identifier.

TIK. Task identification key.

TL. Test loop.

TNSTAT. Tuning statistics.

TP. Transmission priority.

TPRINT. Trace print.

TR. Trace.

TRAC. Trace record.

TRCPL. Trace parameter list.

TRFILE. Trace file.

TSC. Transmission subsystem component.

TSCB. Transmission subsystem control block.

TSO. Time sharing option.

TSO/VTAM. Time sharing option for VTAM.

TSPL. Transmission subsystem parameter list.

TT. Test terminal.

TWA. Test work area.

TWX. Teletypewriter exchange service.

UECB. User exit control block.

UGB. Utilities services task global block.

USS. Unformatted system services.

VEIB. Virtual external interrupt block.

VGB. VTAM services global block.

VIT. VTAM internal trace.

VLB. VTAM services local block.

VM. Virtual Machine operating system.

VM/SP. Virtual Machine/System Product operating system.

VP. Variable pageable.

VR. Virtual route.

VS. Virtual storage.

VSAM. Virtual Storage Access Method.

VSCS. VM SNA console support.

VSE. Virtual Storage Extended operating system.

VSE/AF. Virtual Storage Extended/Advanced Functions operating system.

VSE/SP. Virtual Storage Extended/System Package operating system.

VSM. VTAM service machine.

VTAM. Virtual Telecommunications Access Method.

VTIOC. VTAM Terminal I/O Coordinator.

WEB. Work element block.

WRE. Waiting request element.

XID. Exchange identification.

XMT. Transmit buffer information.

XREF. cross-reference.

XRF. Extended recovery facility.

X.21. Recommendation X.21 (Geneva 1980)

X.25. Recommendation X.25 (Geneva 1980)

Glossary

This glossary defines important NCP, NetView, NetView/PC, SSP, and VTAM abbreviations and terms. It includes information from the *IBM Vocabulary for Data Processing, Telecommunications, and Office Systems*, GC20-1699. Definitions from the *American National Dictionary for Information Processing* are identified by an asterisk (*). Definitions from draft proposals and working papers under development by the International Standards Organization, Technical Committee 97, Subcommittee 1 are identified by the symbol **(TC97)**. Definitions from the *CCIT Sixth Plenary Assembly Orange Book, Terms and Definitions* and working documents published by the Consultative Committee on International Telegraph and Telephone of the International Telecommunication Union, Geneva, 1980 are preceded by the symbol **(CCITT/ITU)**. Definitions from published sections of the *ISO Vocabulary of Data Processing*, developed by the International Standards Organization, Technical Committee 97, Subcommittee 1 and from published sections of the *ISO Vocabulary of Office Machines*, developed by subcommittees of ISO Technical Committee 95, are preceded by the symbol **(ISO)**.

For abbreviations, the definition usually consists only of the words represented by the letters; for complete definitions, see the entries for the words.

Reference Words Used in the Entries

The following reference words are used in this glossary:

> *Deprecated term for*. Indicates that the term should not be used. It refers to a preferred term, which is defined.
>
> *Synonymous with*. Appears in the commentary of a preferred term and identifies less desirable or less specific terms that have the same meaning.
>
> *Synonym for*. Appears in the commentary of a less desirable or less specific term and identifies the preferred term that has the same meaning.
>
> *Contrast with*. Refers to a term that has an opposed or substantively different meaning.
>
> *See*. Refers to multiple-word terms that have the same last word.
>
> *See also*. Refers to related terms that have similar (but not synonymous) meanings.

abend. Abnormal end of task.

abnormal end of task (abend). Termination of a task before its completion because of an error condition that cannot be resolved by recovery facilities while the task is executing.

ACB. (1) In VTAM, application control block. (2) In NCP, adapter control block.

ACB name. (1) The name of an ACB macroinstruction. (2) A name specified in the ACBNAME parameter of a VTAM APPL statement. Contrast with *network name.*

accept. For a VTAM application program, to establish a session with a logical unit (LU) in response to a CINIT request from a system services control point (SSCP). The session-initiation request may begin when a terminal user logs on, a VTAM application program issues a macroinstruction, or a VTAM operator issues a command. See also *acquire (1).*

access method. A technique for moving data between main storage and input/output devices.

accounting exit routine. In VTAM, an optional installation exit routine that collects statistics about session initiation and termination.

ACF. Advanced Communications Function.

ACF/NCP. Advanced Communications Function for the Network Control Program. Synonym for *NCP.*

ACF/SSP. Advanced Communications Function for the System Support Programs. Synonym for *SSP.*

ACF/TAP. Advanced Communications Function for the Trace Analysis Program. Synonym for *TAP.*

ACF/TCAM. Advanced Communications Function for the Telecommunications Access Method. Synonym for *TCAM.*

ACF/VTAM. Advanced Communications Function for the Virtual Telecommunications Access Method. Synonym for *VTAM.*

acquire. (1) For a VTAM application program, to initiate and establish a session with another logical unit (LU). The acquire process begins when the application program issues a macroinstruction. See also *accept.* (2) To take over resources that were formerly controlled by an access method in another domain, or to resume control of resources that were controlled by this domain but released. Contrast with *release.* See also *resource takeover.*

activate. To make a resource of a node ready to perform the functions for which it was designed. Contrast with *deactivate*.

active. (1) The state a resource is in when it has been activated and is operational. Contrast with *inactive, pending,* and *inoperative*. (2) Pertaining to a major or minor node that has been activated by VTAM. Most resources are activated as part of VTAM start processing or as the result of a VARY ACT command.

active application. The application subsystem currently in an extended recovery facility (XRF) session with a terminal user. See *alternate application*.

adapter. Hardware card that allows a device, such as a PC, to communicate with another device, such as a monitor, a printer, or other I/O device.

adapter control block (ACB). In NCP, a control block that contains line control information and the states of I/O operations for BSC lines, SS lines, or SDLC links.

adaptive session pacing. Synonym for *adaptive session-level pacing*.

adaptive session-level pacing. A form of session-level pacing in which session components exchange pacing windows that may vary in size during the course of a session. This allows transmission to adapt dynamically to variations in availability and demand of buffers on a session by session basis. Session pacing occurs within independent stages along the session path according to local congestion at the intermediate nodes. Synonymous with *adaptive session pacing*. See *pacing, session-level pacing*, and *virtual route pacing*.

address translation. See *network address translation*.

adjacent networks. Two SNA networks joined by a common gateway NCP.

adjacent nodes. Two nodes that are connected by one or more data links with no intervening nodes.

Advanced Communications Function (ACF). A group of IBM licensed programs (principally VTAM, TCAM, NCP, and SSP) that use the concepts of Systems Network Architecture (SNA), including distribution of function and resource sharing.

Advanced Program-to-Program Communication (APPC). A synonym for logical unit (LU) 6.2 and its implementations.

alert. (1) In SNA, a record sent to a system problem management focal point to communicate the existence of an alert condition. (2) In the NetView program, a high priority event that warrants immediate attention. This data base record is generated for certain event types that are defined by user-constructed filters.

alert condition. A problem or impending problem for which some or all of the process of problem determination, diagnosis, and resolution is expected to require action at a control point.

alias name. A name defined in a host used to represent a logical unit name, logon mode table name, or class-of-service name in another network. This name is defined to a name translation program when the alias name does not match the real name. The alias name translation program is used to associate the real and alias names.

allocate. A logical unit (LU) 6.2 application program interface (API) verb used to assign a session to a conversation for the conversation's use. Contrast with *deallocate*.

alternate application. The subsystem that is prepared to take over a particular active application's extended recovery facility (XRF) sessions with terminal users in case the application fails. See *active application*.

API. Application program interface.

APPC. Advanced Program-to-Program Communication.

application control block (ACB). A control block that links an application program to VSAM or VTAM.

application program. (1) A program written for or by a user that applies to the user's work. (2) A program used to connect and communicate with stations in a network, enabling users to perform application-oriented activities.

application program exit routine. In VTAM, a user-written exit routine that performs functions for a particular application program and is run as part of the application program. Examples are the RPL exit routine, the EXLST exit routine, and the TESTCB exit routine. Contrast with *installation exit routine*.

application program interface (API). (1) The formally defined programming language interface between an IBM system control program or licensed program and its user. (2) The interface through which an application program interacts with an access method. In VTAM, it is the language structure used in control blocks so that application programs can reference them and be identified to VTAM.

application program major node. A group of application program minor nodes. In the VTAM definition library, it is a member, book, or file that contains one or more APPL statements, which represent application programs. In MVS, it is a member of the library; in

VSE, it is a book; and in VM, it is a CMS file of filetype VTAMLST.

ASCII. American National Standard Code for Information Interchange.

asynchronous operation. Simultaneous operations of software or hardware. In software, an operation, such as a request for session establishment or data transfer, in which the application program is allowed to continue execution while the operation is performed. The access method informs the application program after the operation is completed. Contrast with *synchronous operation.*

asynchronous request. In VTAM, a request for an asynchronous operation. Contrast with *synchronous request.*

authorization exit routine. In VTAM, an optional installation exit routine that approves or disapproves requests for session initiation.

authorized path. In VTAM for MVS, a facility that enables an application program to specify that a data transfer or related operation be carried out in a privileged and more efficient manner.

automatic logon. (1) A process by which VTAM automatically creates a session-initiation request to establish a session between two logical units (LUs). The session will be between a designated primary logical unit (PLU) and a secondary logical unit (SLU) that is neither queued for nor in session with another PLU. See also *controlling application program* and *controlling logical unit.* (2) In VM, a process by which a virtual machine is initiated by other than the user of that virtual machine. For example, the primary VM operator's virtual machine is activated automatically during VM initialization.

available. In VTAM, pertaining to a logical unit that is active, connected, enabled, and not at its session limit.

back-level. Pertaining to an earlier release of an IBM product, which may not support a particular, current function.

backup session. The session that replaces the failing primary extended recovery facility (XRF) session between a terminal user and the active subsystem.

basic conversation. A conversation that supports the functions of the basic conversation protocol boundary defined by LU 6.2. That format requires data to be sent as logical records consisting of a 2-byte length prefix followed by the data. See also *mapped conversation.*

basic information unit (BIU). In SNA, the unit of data and control information that is passed between half-sessions. It consists of a request/response header (RH) followed by a request/response unit (RU).

basic transmission unit (BTU). In SNA, the unit of data and control information passed between path control components. A BTU can consist of one or more path information units (PIUs). See also *blocking of PIUs.*

begin bracket. In SNA, the value (binary 1) of the begin-bracket indicator in the request header (RH) of the first request in the first chain of a bracket; the value denotes the start of a bracket. Contrast with *end bracket.* See also *bracket.*

bidder. In SNA, the LU-LU half-session defined at session activation as having to request and receive permission from the other LU-LU half-session to begin a bracket. Contrast with *first speaker.* See also *bracket protocol* and *contention.*

binary synchronous communication (BSC). (1) Communication using binary synchronous line discipline. (2) A uniform procedure, using a standardized set of control characters and control character sequences, for synchronous transmission of binary-coded data between stations.

bind. In SNA, a request to activate a session between two logical units (LUs). See also *session activation request.* Contrast with *UNBIND.*

BIU. Basic information unit.

BIU segment. In SNA, the portion of a basic information unit (BIU) that is contained within a path information unit (PIU). It consists of either a request/response header (RH) followed by all or a portion of a request/response unit (RU), or only a portion of an RU.

blocking of PIUs. In SNA, an optional function of path control that combines multiple path information units (PIUs) into a single basic transmission unit (BTU).

BNN. Boundary network node.

boundary function. (1) A capability of a subarea node to provide protocol support for attached peripheral nodes, such as: (a) interconnecting subarea path control and peripheral path control elements, (b) performing session sequence numbering for low-function peripheral nodes, and (c) providing session-level pacing support. (2) The component that provides these capabilities. See also *boundary node, network addressable unit (NAU), peripheral path control, subarea node,* and *subarea path control.*

boundary network node (BNN). Deprecated term for *boundary node.*

boundary node. (1) A subarea node with boundary function. See *subarea node* (including illustration).

See also *boundary function*. (2) The programming component that performs FID2 (format identification type 2) conversion, channel data link control, pacing, and channel or device error recovery procedures for a locally attached station. These functions are similar to those performed by a network control program for an NCP-attached station.

bracket. In SNA, one or more chains of request units (RUs) and their responses that are exchanged between the two LU-LU half-sessions and that represent a transaction between them. A bracket must be completed before another bracket can be started. Examples of brackets are data base inquiries/replies, update transactions, and remote job entry output sequences to work stations. See also *begin bracket* and *end bracket*.

bracket protocol. In SNA, a data flow control protocol in which exchanges between the two LU-LU half-sessions are achieved through the use of brackets, with one LU designated at session activation as the first speaker and the other as the bidder. The bracket protocol involves bracket initiation and termination rules. See also *bidder* and *first speaker*.

BSC. Binary synchronous communication.

BTU. Basic transmission unit.

buffer. A portion of storage for temporarily holding input or output data.

cancel closedown. A closedown in which VTAM is abnormally terminated either because of an unexpected situation or as the result of an operator command. See also *orderly closedown* and *quick closedown*.

CCP. Configuration control program facility.

CCS. Console communication services.

CDRM. Cross-domain resource manager.

CDRSC. Cross-domain resource.

chain. See *RU chain*.

channel. * A path along which signals can be sent, for example, data channel, output channel. See *data channel* and *input/output channel*. See also *link*.

channel adapter. A communication controller hardware unit used to attach the controller to a System/360 or a System/370 channel.

channel-attached. Pertaining to the attachment of devices directly by System 370 input-output channels to a host processor.

channel-attachment major node. (1) A major node that includes an NCP that is channel-attached to a data host. (2) A major node that may include minor nodes that are the line groups and lines that represent a channel attachment to an adjacent (channel-attached) host. (3) In VM or VSE operating systems, a major node that may include minor nodes that are resources (host processors, NCPs, line groups, lines, SNA physical units and logical units, cluster controllers, and terminals) attached through a communication adapter.

channel link. A System/370 I/O channel to control unit interface that has an SNA network address. A channel link can be either a subarea link or a peripheral link. See also *link* and *subarea link*.

character-coded. Synonym for *unformatted*.

CICS. Customer Information Control System.

CID. Communication identifier.

CINIT. A network services request sent from a system services control point (SSCP) to a logical unit (LU) asking that LU to establish a session with another LU and to act as the primary end of the session.

class of service (COS). In SNA, a designation of the path control network characteristics, such as path security, transmission priority, and bandwidth, that apply to a particular session. The end user designates class of service at session initiation by using a symbolic name that is mapped into a list of virtual routes, any one of which can be selected for the session to provide the requested level of service.

cleanup. A network services request, sent by a system services control unit (SSCP) to a logical unit (LU), that causes a particular LU-LU session with that LU to be ended immediately and without the participation of either the other LU or its SSCP.

clear data. Data that is not enciphered. Synonymous with *plaintext*.

clear session. A session in which only clear data is transmitted or received. Contrast with *cryptographic session*.

closedown. The deactivation of a device, program, or system. See *cancel closedown, orderly closedown*, and *quick closedown*.

cluster controller. A device that can control the input/output operations of more than one device connected to it. A cluster controller may be controlled by a program stored and executed in the unit; for example, the IBM 3601 Finance Communication Controller. Or it may be controlled entirely by hardware; for example, the IBM 3272 Control Unit.

CMS. Conversational Monitor System.

CNM. Communication network management.

CNOS. Change number of sessions.

code point. In the NetView/PC program and in the NetView program, a 1- or 2-byte hexadecimal value that indexes a text string stored at an alert receiver and is used by the alert receiver to create displays of alert information.

command. (1) A request from a terminal for the performance of an operation or the execution of a particular program. (2) In SNA, any field set in the transmission header (TH), request header (RH), and sometimes portions of a request unit (RU), that initiates an action or that begins a protocol; for example: (a) Bind Session (session-control request unit), a command that activates an LU-LU session, (b) the change-direction indicator in the RH of the last RU of a chain, (c) the virtual route reset window indicator in a FID4 transmission header. See also *VTAM operator command*.

command facility. The component of the NetView program that is a base for command processors that can monitor, control, automate, and improve the operation of a network.

command list (CLIST). In the NetView program, a sequential list of commands and control statements that is assigned a name. When the name is invoked (as a command) the commands in the list are executed.

command processor. A program that performs an operation specified by a command.

communication adapter. An optional hardware feature, available on certain processors, that permits communication lines to be attached to the processors.

communication control unit. A communication device that controls the transmission of data over lines in a network. Communication control units include transmission control units (such as the 2702 Transmission Control Unit) and communication controllers (such as the 3720 or 3725).

communication controller. A type of communication control unit whose operations are controlled by one or more programs stored and executed in the unit; for example, the IBM 3725 Communication Controller. It manages the details of line control and the routing of data through a network.

communication identifier (CID). In VTAM, a key for locating the control blocks that represent a session. The key is created during the session-establishment procedure and deleted when the session ends.

communication line. Deprecated term for *telecommunication line* and *transmission line*.

communication management configuration (CMC). (1) In VTAM, a technique for configuring a network that allows for the consolidation of many network management functions for the entire network in a single host processor. (2) A multiple-domain network configuration in which one of the hosts, called the communication management configuration host, performs most of the controlling functions for the network, thus allowing the other hosts, called data hosts, to process applications. This is accomplished by configuring the network so that the communication management host owns most of the resources in the network that are not application programs. The resources that are not owned by the communication management host are the resources that are channel-attached stations of data hosts.

communication management configuration host node. The type 5 host processor in a communication management configuration that does all network-control functions in the network except for the control of devices channel-attached to data hosts. Synonymous with *communication management host*. Contrast with *data host node*.

communication management host. Synonym for *communication management configuration host node*. Contrast with *data host*.

communication network management (CNM). The process of designing, installing, operating, and managing the distribution of information and controls among end users of communication systems.

communication network management (CNM) application program. A VTAM application program that issues and receives formatted management services request units for physical units. For example, NetView.

communication network management (CNM) interface. The interface that the access method provides to an application program for handling data and commands associated with communication system management. CNM data and commands are handled across this interface.

communication network management (CNM) processor. A program that manages one of the functions of a communications system. A CNM processor is executed under control of NetView.

communication scanner processor (CSP). A processor in the 3725 Communication Controller that contains a microprocessor with control code. The code controls transmission of data over links attached to the CSP.

composite end node (CEN). A group of nodes made up of a single type 5 node and its subordinate type 4 nodes that together support type 2.1 protocols. To a type 2.1 node, a CEN appears as one end node. For example, NCP and VTAM act as a composite end node.

configuration. (1) (TC97) The arrangement of a computer system or network as defined by the nature, number, and the chief characteristics of its functional units. The term may refer to a hardware or a software configuration. (2) The devices and programs that make up a system, subsystem, or network. (3) In CCP, the arrangement of controllers, lines, and terminals attached to an IBM 3710 Network Controller. Also, the collective set of item definitions that describe such a configuration.

configuration control program (CCP) facility. An SSP interactive application program facility by which configuration definitions for the IBM 3710 Network Controller can be created, modified, and maintained.

configuration services. In SNA, one of the types of network services in the control point (CP) and in the physical unit (PU); configuration services activate, deactivate, and maintain the status of physical units, links, and link stations. Configuration services also shut down and restart network elements and modify path control routing tables and address-translation tables. See also *maintenance services*, *management services*, *network services*, and *session services*.

connected. In VTAM, pertaining to a physical unit (PU) or logical unit (LU) that has an active physical path to the host processor containing the system services control point (SSCP) that controls the PU or LU.

connection. Synonym for *physical connection*.

console communications services (CCS). The SNA facility that acts as an interface between the control program and the VSCS component of VTAM for VM.

contention. A situation in which two logical units (LUs) that are connected by an LU 6.2 session both attempt to allocate the session for a conversation at the same time. The control operator assigns "winner" and "loser" status to the LUs so that processing may continue on an orderly basis. The contention loser requests permission from the contention winner to allocate a conversation on the session, and the contention winner either grants or rejects the request. See also *bidder*.

control block. (ISO) A storage area used by a computer program to hold control information.

control operator. For logical unit (LU) 6.2, a service transaction program that describes and controls the availability of certain resources. For example, it describes network resources accessed by the local LU and it controls session limits between the LU and its partners. See also *contention*.

control point (CP). (1) A system services control point (SSCP) that provides hierarchical control of a group of nodes in a network. (2) A control point (CP) local to a specific node that provides control of that node, either in the absence of SSCP control (for type 2.1 nodes engaged in peer to peer communication) or to supplement SSCP control.

control program (CP). The VM operating system that manages the real processor's resources and is responsible for simulating System/370s for individual users.

controlling application program. In VTAM, an application program with which a secondary logical unit (other than an application program) is automatically put in session whenever the secondary logical unit is available. See also *automatic logon* and *controlling logical unit*.

controlling logical unit. In VTAM, a logical unit with which a secondary logical unit (other than an application program) is automatically put in session whenever the secondary logical unit is available. A controlling logical unit can be either an application program or a device-type logical unit. See also *automatic logon* and *controlling application program*.

control statement. A statement in a command list that controls the processing sequence of the command list or allows the command list to send messages to the operator and receive input from the operator.

conversation. In SNA, a logical connection between two transaction programs using an LU 6.2 session. Conversations are delimited by brackets to gain exclusive use of a session.

Conversational Monitor System (CMS). A VM application program for general interactive time sharing, problem solving, and program development.

converted command. An intermediate form of a character-coded command produced by VTAM through use of an unformatted system services definition table. The format of a converted command is fixed; the unformatted system services definition table must be constructed in such a manner that the character-coded command (as entered by a logical unit) is converted into the predefined, converted command format. See also *unformatted*.

COS. Class of service.

coupler. A hardware device that connects a modem to a public phone system in much the same way that a telephone does.

CP. (1) Control program. (2) Control point.

cross-domain. In SNA, pertaining to control of resources involving more than one domain.

cross-domain resource (CDRSC). A resource owned by a cross-domain resource manager (CDRM) in another domain but known by the CDRM in this domain by network name and associated CDRM.

cross-domain resource manager (CDRM). In VTAM, the function in the system services control point (SSCP) that controls initiation and termination of cross-domain sessions.

cross-network. In SNA, pertaining to control or resources involving more than one SNA network.

cross-network LU-LU session. In SNA, a session between logical units (LUs) in different networks.

cross-network session. An LU-LU or SSCP-SSCP session whose path traverses more than one SNA network.

CRV. Cryptography verification.

cryptographic. Pertaining to the transformation of data to conceal its meaning. See also *encipher* and *decipher*.

cryptographic session. In SNA products, an LU-LU session in which a function management data (FMD) request may be enciphered before it is transmitted and deciphered after it is received. Contrast with *clear session*. See *required cryptographic session* and *selective cryptographic session*.

cryptographic session key. In SNA, deprecated term for *session cryptography key*.

cryptography verification (CRV) request. A request unit sent by the primary logical unit (PLU) to the secondary logical unit (SLU) as part of cryptographic session establishment, to allow the SLU to verify that the PLU is using the correct cryptographic session key.

CSP. Communication scanner processor.

Customer Information Control System (CICS). A licensed program that enables transactions entered at remote terminals to be processed concurrently by user-written application programs. It also includes facilities for building, using, and maintaining data bases.

DAF. Destination address field.

data channel. Synonym for *input/output channel*. See *channel*.

data check. An indication that a transmission is faulty. For example, in SDLC a frame check sequence (FCS) error.

data circuit-terminating equipment (DCE). (TC97) The equipment installed at the user's premises that provides all functions required to establish, maintain, and terminate a connection, and the signal conversion and coding between the data terminal equipment (DTE) and the line. The DCE may be separate equipment or an integral part of other equipment.

data-encrypting key. A key used to encipher and decipher data transmitted in a cryptographic session. Contrast with *key-encrypting key*. See *session cryptography key*.

data flow control (DFC). In SNA, a request/response unit (RU) category used for requests and responses exchanged between the data flow control layer in one half-session and the data flow control layer in the session partner.

data flow control (DFC) layer. In SNA, the layer within a half-session that (1) controls whether the half-session can send, receive, or concurrently send and receive request units (RUs); (2) groups related RUs into RU chains; (3) delimits transactions via the bracket protocol; (4) controls the interlocking of requests and responses in accordance with control modes specified at session activation; (5) generates sequence numbers; and (6) correlates requests and responses.

data flow control (DFC) protocol. In SNA, the sequencing rules for requests and responses by which network addressable units (NAUs) in a session coordinate and control data transfer and other operations. For example, see *bracket protocol*.

data host. Synonym for *data host node*. Contrast with *communication management configuration host*.

data host node. In a communication management configuration, a type 5 host node that is dedicated to processing applications and does not control network resources, except for its channel-attached or communication adapter-attached devices. Synonymous with *data host*. Contrast with *communication management configuration host node*.

data link. In SNA, synonym for *link*.

data link control (DLC) layer. In SNA, the layer that consists of the link stations that schedule data transfer over a transmission medium connecting two nodes and perform error control for the link connection. Examples of data link control are SDLC for serial-by-bit link connection and data link control for the System/370 channel.

data link control protocol. In SNA, a set of rules used by two nodes on a data link to accomplish an orderly exchange of information. Synonymous with *line control*.

data terminal equipment (DTE). (TC97) That part of a data station that serves as a data source, data link, or both, and provides for the data communication control function according to protocols.

data types. In the NetView program, a concept to describe the organization of panels. Data types are defined as alerts, events, and statistics. Data types are combined with resource types and display types to describe NetView's display organization. See also *display types* and *resource types*.

DBCS. Double-byte character set.

DCE. Data circuit-terminating equipment.

deactivate. To take a resource of a node out of service, rendering it inoperable, or to place it in a state in which it cannot perform the functions for which it was designed. Contrast with *activate*.

deallocate. A logical unit (LU) 6.2 application program interface (API) verb that terminates a conversation, thereby freeing the session for a future conversation. Contrast with *allocate*.

decipher. To convert enciphered data into clear data. Contrast with *encipher*. Synonymous with *decrypt*.

decrypt. To convert encrypted data into clear data. Contrast with *encrypt*. Synonym for *decipher*.

decryption. The unscrambling of data using an algorithm which works under the control of a key. The key allows data to be protected even when the algorithm is unknown. Data is unscrambled after transmission. Contrast with *encryption*.

definite response (DR). In SNA, a value in the form-of-response-requested field of the request header. The value directs the receiver of the request to return a response unconditionally, whether positive or negative, to that request. Contrast with *exception response* and *no response*.

definition statement. (1) In VTAM, the statement that describes an element of the network. (2) In NCP, a type of instruction that defines a resource to the NCP. See Figure 182, Figure 183, and Figure 184. See also *macroinstruction*.

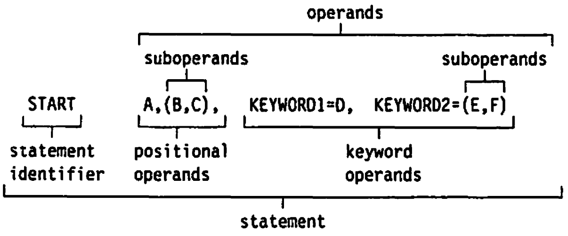


Figure 182. Example of a Language Statement

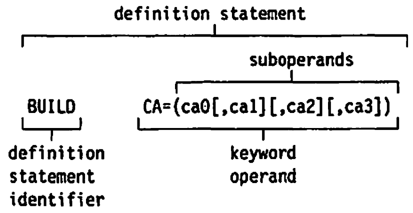


Figure 183. NCP Examples

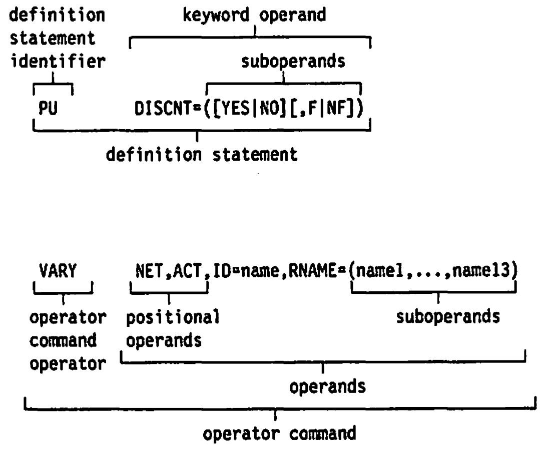


Figure 184. VTAM Examples

definition statement identifier. A specific character string that identifies the purpose of a definition statement.

dependent LU. Any logical unit (LU) that receives an ACTLU over a link. Such LUs can act only as secondary logical units (SLUs) and can have only one LU-LU session at a time. Contrast with *independent LU*.

destination address field (DAF). In SNA, a field in a FID0 or FID1 transmission header that contains the network address of the destination.

destination logical unit (DLU). The logical unit to which data is to be sent. Contrast with *origin logical unit (OLU).*

device-type logical unit. In VTAM, a logical unit that has a session limit of 1 and usually acts as the secondary end of a session. It is typically a logical unit (LU) in an SNA terminal, such as a 3270. It could be the primary end of a session, for example, the logical unit representing the Network Routing Facility (NRF) logical unit.

DFC. Data flow control.

dial-in. Refers to the direction in which a switched connection is requested by any node or terminal other than the receiving host or an NCP.

dial-out. Refers to the direction in which a switched connection is requested by a host or an NCP.

directory. In VM, a control program (CP) disk that defines each virtual machine's normal configuration.

disabled. In VTAM, pertaining to a logical unit (LU) that has indicated to its system services control point (SSCP) that it is temporarily not ready to establish LU-LU sessions. An initiate request for a session with a disabled logical unit (LU) can specify that the session be queued by the SSCP until the LU becomes enabled. The LU can separately indicate whether this applies to its ability to act as a primary logical unit (PLU) or a secondary logical unit (SLU). See also *enabled* and *inhibited.*

disconnection. The termination of a physical connection.

display. (1) To present information for viewing, usually on a terminal screen or a hard-copy device. (2) A device or medium on which information is presented, such as a terminal screen. (3) Deprecated term for *panel.*

display levels. Synonym for *display types.*

display types. In NetView, a concept to describe the organization of panels. Display types are defined as total, most recent, user action, and detail. Display types are combined with resource types and data types to describe NetView's panel organization. See *data types* and *resource types.* Synonymous with *display levels.*

DLC. Data link control.

DLU. Destination logical unit.

domain. (1) An access method, its application programs, communication controllers, connecting lines, modems, and attached terminals. (2) In SNA, a system services control point (SSCP) and the physical units (PUs), logical units (LUs), links, link stations, and all the associated resources that the SSCP has the ability to control by means of activation requests and deactivation requests. See *system services control point domain* and *type 2.1 node control point domain.* See also *single-domain network* and *multiple-domain network*

domain operator. In a multiple-domain network, the person or program that controls the operation of the resources controlled by one system services control point. Contrast with *network operator* (2).

double-byte character set (DBCS). A set of graphic characters where two bytes are used to represent the character set of such languages as Chinese and Japanese. The shift-in (SI) and shift-out (SO) control codes are used when DBCS is mixed with standard one-byte character sets.

downstream. In the direction of data flow from the host to the end user. Contrast with *upstream.*

downstream device. For the IBM 3710 Network Controller, a device located in a network such that the 3710 is positioned between the device and a host. A display terminal downstream from the 3710 is an example of a downstream device. Contrast with *upstream device.*

downstream line. For the IBM 3710 Network Controller, a telecommunication line attaching a downstream device to a 3710. Contrast with *upstream line.*

Downstream Load Utility (DSLU). A licensed program that uses the communication network management (CNM) interface to support the load requirements of certain type 2 physical units, such as the IBM 3644 Automatic Data Unit and the IBM 8775 Display Terminal.

DR. (1) In NCP and CCP, dynamic reconfiguration. (2) In SNA, definite response.

DRDS. Dynamic reconfiguration data set.

DTE. Data terminal equipment.

dump. (1) Computer printout of storage. (2) To write the contents of all or part of storage to an external medium as a safeguard against errors or in connection with debugging. (3) (ISO) Data that have been dumped.

duplex. * In data communication, pertaining to a simultaneous two-way independent transmission in both directions. Synonymous with *full duplex.* Contrast with *half duplex.*

dynamic reconfiguration (DR). The process of changing the network configuration (peripheral PUs and LUs) without regenerating complete configuration tables.

dynamic reconfiguration data set (DRDS). In VTAM, a data set used for storing definition data that can be applied to a generated communication controller configuration at the operator's request. A dynamic reconfiguration data set can be used to dynamically add PUs and LUs, delete PUs and LUs, and move PUs. It is activated with the VARY DRDS operator command. See also *dynamic reconfiguration.*

EBCDIC. * Extended binary-coded decimal interchange code. A coded character set consisting of 8-bit coded characters.

ECB. Event control block.

echo. The return of characters to the originating SS device to verify that a message was sent correctly.

echo check. A check to determine the correctness of the transmission of data in which the received data are returned to the source for comparison with the originally transmitted data.

ED. Enciphered data.

EIA. Electronic Industries Association. Provides interface standards for electrical and electronic equipment.

element. (1) A field in the network address. (2) The particular resource within a subarea identified by the element address. See also *subarea.*

element address. In SNA, a value in the element address field of the network address identifying a specific resource within a subarea. See *subarea address.*

emulation mode. The function of a network control program that enables it to perform activities equivalent to those performed by a transmission control unit. Contrast with *network control mode.*

Emulation Program (EP). An IBM control program that allows a channel-attached 3705 or 3725 communication controller to emulate the functions of an IBM 2701 Data Adapter Unit, an IBM 2702 Transmission Control, or an IBM 2703 Transmission Control. See also *network control program.*

enabled. In VTAM, pertaining to a logical unit (LU) that has indicated to its system services control point (SSCP) that it is now ready to establish LU-LU sessions. The LU can separately indicate whether this prevents it from acting as a primary logical unit (PLU) or as a secondary logical unit (SLU). See also *disabled* and *inhibited.*

encipher. (1) To scramble data or convert it, before transmission, to a secret code that masks the meaning of the data to any unauthorized recipient. (2) In VTAM, to convert clear data into enciphered data. Contrast with *decipher.* Synonymous with *encrypt.*

enciphered data (ED). Data whose meaning is concealed from unauthorized users.

encrypt. Synonym for *encipher.*

encryption. The scrambling or encoding of data using an algorithm which works under the control of a key. The key allows data to be protected even when the algorithm is unknown. Data is scrambled prior to transmission. Contrast with *decryption.*

end bracket. In SNA, the value (binary 1) of the end bracket indicator in the request header (RH) of the first request of the last chain of a bracket; the value denotes the end of the bracket. Contrast with *begin bracket.* See also *bracket.*

end-of-transmission (EOT). The specific character, or sequence of characters, that indicates no more data.

end-of-transmission (EOT) handshaking. When a 3710 sends EOT characters over an idle line and waits for return characters. If no EOT response is returned, the 3710 breaks the session.

end node. A type 2.1 node that does not provide any intermediate routing or session services to any other node. For example, APPC/PC is an end node. See *composite end node, node,* and *type 2.1 node.*

end user. In SNA, the ultimate source or destination of application data flowing through an SNA network. An end user may be an application program or a terminal operator.

entry point. An SNA node that provides distributed network management support. It may be a type 2, type 2.1, type 4, or type 5 node. It sends SNA-formatted network management data about itself and the resources it controls to a focal point for centralized processing, and it receives and executes focal point initiated commands to manage and control its resources.

EOT. End-of-transmission.

EP. Emulation Program.

ER. (1) Explicit route. (2) Exception response.

error-to-traffic (E/T). The number of temporary errors compared to the traffic associated with a resource.

E/T. Error-to-traffic.

event. (1) In the NetView program, a record indicating irregularities of operation in physical elements of a network. (2) An occurrence of significance to a task; typically, the completion of an asynchronous operation, such as an input/output operation.

event control block (ECB). A control block used to represent the status of an event.

exception request (EXR). In SNA, a request that replaces another message unit in which an error has been detected.

exception response (ER). In SNA, a value in the form-of-response-requested field of a request header (RH). An exception response is sent only if a request is unacceptable as received or cannot be processed. Contrast with *definite response* and *no response*. See also *negative response*.

exchange identification (XID). A data link control command and response passed between adjacent nodes that allows the two nodes to exchange identification and other information necessary for operation over the data link.

EXEC. In a VM operating system, a user-written command file that contains CMS commands, other user-written commands, and execution control statements, such as branches.

exit list (EXLST). In VSAM and VTAM, a control block that contains the addresses of routines that receive control when specified events occur during execution; for example, routines that handle session-establishment request processing or I/O errors.

exit routine. Any of several types of special-purpose user-written routines. See *accounting exit routine, authorization exit routine, logon-interpret routine, virtual route selection exit routine, EXLST exit routine,* and *RPL exit routine*.

EXLST exit routine. In VTAM, a routine whose address has been placed in an exit list (EXLST) control block. The addresses are placed there with the EXLST macroinstruction, and the routines are named according to their corresponding operand; hence DFASY exit routine, TPEND exit routine, RELREQ exit routine, and so forth. All exit list routines are coded by the VTAM application programmer. Contrast with *RPL exit routine*.

expedited flow. In SNA, a data flow designated in the transmission header (TH) that is used to carry network control, session control, and various data flow control request/response units (RUs); the expedited flow is separate from the normal flow (which carries primarily end-user data) and can be used for commands that affect the normal flow. Contrast with *normal flow*.

explicit command. In the NetView program, using a direct command to start an operation or to request information instead of stepping through the panel hierarchy to do so.

explicit route (ER). In SNA, the path control network elements, including a specific set of one or more transmission groups, that connect two subarea nodes. An explicit route is identified by an origin subarea address, a destination subarea address, an explicit route number, and a reverse explicit route number. Contrast with *virtual route (VR)*. See also *path* and *route extension*.

explicit route length. In SNA, the number of transmission groups in an explicit route.

EXR. Exception request.

extended recovery facility (XRF). Software designed to minimize the effect of failures in MVS, VTAM, the host processor, or IMS/VS on sessions between IMS/VS and designated terminals. It provides an alternate subsystem to take over failing sessions.

FDX. Full duplex.

feature. A particular part of an IBM product that a customer can order separately.

FIC. First-in-chain.

FID. Format identification.

field-formatted. Pertaining to a request or response that is encoded into fields, each having a specified format such as binary codes, bit-significant flags, and symbolic names. Contrast with *character-coded*.

field-formatted request. In SNA, a request that is encoded into fields, each having a specified format such as binary codes, binary counts, bit-significant flags, and symbolic names; a format indicator in the request/response header (RH) for the request is set to zero. Contrast with *character-coded*.

first-in-chain (FIC). A request unit (RU) whose request header (RH) begin chain indicator is on and whose RH end chain indicator is off. See also *RU chain*.

first speaker. In SNA, the LU-LU half-session defined at session activation as: (1) able to begin a bracket without requesting permission from the other LU-LU half-session to do so, and (2) winning contention if both half-sessions attempt to begin a bracket simultaneously. Contrast with *bidder*. See also *bracket protocol*.

flow control. In SNA, the process of managing the rate at which data traffic passes between components of the network. The purpose of flow control is to optimize the rate of flow of message units, with minimum congestion in the network; that is, to neither overflow the buffers at the receiver or at intermediate routing nodes, nor leave the receiver waiting for more message units. See also *adaptive session-level pacing, pacing, session-level pacing,* and *virtual route pacing*.

FMD. Function management data.

FMH. Function management header.

focal point. An entry point that provides centralized management and control for other entry points for one or more network management categories.

format identification (FID) field. In SNA, a field in each transmission header (TH) that indicates the format of the TH; that is, the presence or absence of certain fields. TH formats differ in accordance with the types of nodes between which they pass. The six FID types are:

> FID0, used for traffic involving non-SNA devices between adjacent subarea nodes when either or both nodes do not support explicit route and virtual route protocols.
>
> FID1, used for traffic between adjacent subarea nodes when either or both nodes do not support explicit route and virtual route protocols.
>
> FID2, used for traffic between a subarea node and an adjacent type 2 peripheral node.
>
> FID3, used for traffic between a subarea node and an adjacent type 1 peripheral node.
>
> FID4, used for traffic between adjacent subarea nodes when both nodes support explicit route and virtual route protocols.
>
> FIDF, used for certain commands (for example, for transmission group control) sent between adjacent subarea nodes when both nodes support explicit route and virtual route protocols.

formatted system services. A portion of VTAM that provides certain system services as a result of receiving a field-formatted command, such as an Initiate or Terminate command. Contrast with *unformatted system services (USS)*. See also *field-formatted*.

frame. (1) The unit of transmission in some local area networks, including the IBM Token-Ring Network. It includes delimiters, control characters, information, and checking characters. (2) In SDLC, the vehicle for every command, every response, and all information that is transmitted using SDLC procedures.

full duplex (FDX). Synonym for *duplex*.

full-screen mode. A form of panel presentation in NetView where the contents of an entire terminal screen can be displayed at once. Full-screen mode can be used for fill-in-the-blanks prompting. Contrast with *line mode*.

function management data (FMD). In SNA, a request unit (RU) category used for end-user data exchanged between logical units (LUs) and for requests and responses exchanged between network services components of LUs, physical units (PUs), and system services control points (SSCPs).

function management header (FMH). In SNA, one or more headers, optionally present in the leading request units (RUs) of an RU chain, that allow one half-session in an LU-LU session to: (1) select a destination at the session partner and control the way in which the end-user data it sends is handled at the destination, (2) change the destination or the characteristics of the data during the session, and (3) transmit between session partners status or user information about the destination (for example, a program or device). Function management headers (FMHs) can be used on LU-LU types 1, 4, and 6.2.

gateway. The combination of machines and programs that provide address translation, name translation, and system services control point (SSCP) rerouting between independent SNA networks to allow those networks to communicate. A gateway consists of one gateway NCP and at least one gateway SSCP.

gateway NCP. An NCP that performs address translation to allow cross-network session traffic. The gateway NCP connects two or more independent SNA networks. Synonymous with *gateway node*.

gateway node. Synonym for *gateway NCP*.

gateway SSCP. An SSCP that is capable of cross-network session initiation, termination, takedown, and session outage notification. A gateway SSCP is in session with the gateway NCP; it provides network name translation and assists the gateway NCP in setting up alias network addresses for cross-network sessions.

GCS. Group control system.

generalized path information unit trace (GPT). A record of the flow of path information units (PIUs) exchanged between the network control program and its attached resources. PIU trace records consist of up to 44 bytes of transmission header (TH), request/response header (RH), and request/response unit (RU) data.

generation. The process of assembling and link editing definition statements so that resources can be identified to all the necessary programs in a network.

generic unbind. Synonym for *session deactivation request*.

GPT. Generalized path information unit trace.

group. In the NetView/PC program, to identify a set of application programs that are to run concurrently.

group control system (GCS). A component of VM that provides multi-programming and shared memory support to virtual machines. It is a saved system intended for use with SNA products.

half-duplex. * In data communication, pertaining to an alternate, one way at a time, independent transmission. Contrast with *duplex*.

half-session. In SNA, a component that provides function management data (FMD) services, data flow control, and transmission control for one of the sessions of a network addressable unit (NAU). See also *primary half-session* and *secondary half-session*.

hard copy. A printed copy of machine output in a visually readable form; for example, printed reports, listings, documents, summaries, or network logs.

hardware monitor. The component of the NetView program that helps identify network problems, such as hardware, software, and microcode, from a central control point using interactive display techniques.

help panel. An online display that tells you how to use a command or another aspect of a product. See *task panel*.

High Performance Option (HPO). A licensed program that is an extension of VM/SP. It provides performance and operation enhancements for large system environments.

hierarchy. In the NetView program, the resource types, display types, and data types that make up the organization, or levels, in a network.

host node. A node providing an application program interface (API) and a common application interface. See *boundary node*, *node*, *peripheral node*, *subarea host node*, and *subarea node*. See also *boundary function* and *node type*.

host processor. (1) (TC97) A processor that controls all or part of a user application network. (2) In a network, the processing unit in which the data communication access method resides.

HPO. High Performance Option.

hung terminal. A terminal to which a session is disrupted and that cannot send or receive commands.

I-frame. A DLC frame type for transmitting data. Other DLC frame types are for control, status, and supervisory information.

IMR. Intensive mode recording.

IMS. Information Management System/Virtual Storage. Synonymous with *IMS/VS*.

IMS/VS. Information Management System/Virtual Storage. Synonym for *IMS*.

inactive. Describes the state of a resource that has not been activated or for which the VARY INACT command has been issued. Contrast with *active*. See also *inoperative*.

independent LU. A logical unit (LU) that does not receive an ACTLU over a link. Such LUs can act as primary logical units (PLUs) or secondary logical units (SLUs) and can have one or more LU-LU sessions at a time. Contrast with *dependent LU*.

Information/Management. A feature of the Information/System licensed program that provides interactive systems management applications for problem, change, and configuration management.

Information Management System (IMS). A general purpose system whose full name is Information Management System/Virtual Storage (IMS/VS). It enhances the capabilities of OS/VS for batch processing and telecommunication and allows users to access a computer-maintained data base through remote terminals.

Information/System. An interactive retrieval program with related utilities designed to provide systems programmers with keyword access to selected technical information contained in either of its companion products, Information/MVS or Information/VM-VSE.

inhibited. In VTAM, pertaining to a logical unit (LU) that has indicated to its system services control point (SSCP) that it is not ready to establish LU-LU sessions. An initiate request for a session with an inhibited LU will be rejected by the SSCP. The LU can separately indicate whether this applies to its ability to act as a primary logical unit (PLU) or as a secondary logical unit (SLU). See also *enabled* and *disabled*.

initiate. A network services request sent from a logical unit (LU) to a system services control point (SSCP) requesting that an LU-LU session be established.

inoperative. The condition of a resource that has been active, but is not. The resource may have failed, received an INOP request, or is suspended while a reactivate command is being processed. See also *inactive*.

input/output channel. (1) **(ISO)**. In a data processing system, a functional unit that handles the transfer of data between internal and peripheral equipment. (2) In a computing system, a functional unit, controlled by a processor, that handles the transfer of data between processor storage and local peripheral devices. Synonymous with *data channel*. See *channel*. See also *link*.

installation exit routine. In VTAM, a user-written exit routine that can perform functions related to initiation and termination of sessions and is run as part of VTAM rather than as part of an application program. Examples are the accounting, authorization, logon-interpret, and virtual route selection exit routines. Contrast with *application program exit routine*.

intensive mode recording (IMR). An NCP function that forces recording of temporary errors for a specified resource.

interactive problem control system (IPCS). A VM facility for diagnosing problems, and managing problem information and status. IPCS is the principal means for diagnosing virtual machine dumps.

interconnection. See *SNA network interconnection*.

interface. * A shared boundary. An interface might be a hardware component to link two devices or it might be a portion of storage or registers accessed by two or more computer programs.

intermediate routing function (IRF). In SNA, a path control capability in a subarea node that receives and routes path information units (PIUs) that neither originate in nor are destined for network addressable units (NAUs) in the subarea. See also *boundary function*.

intermediate routing node (IRN). In SNA, a subarea node with intermediate routing function.

interpret table. In VTAM, an installation-defined correlation list that translates an argument into a string of eight characters. Interpret tables can be used to translate logon data into the name of an application program for which the logon is intended.

inter-user communication vehicle (IUCV). A VM facility for passing data between virtual machines and VM components.

IPCS. Interactive problem control system.

IRN. Intermediate routing node.

item. In CCP, any of the components, such as communication controllers, lines, cluster controllers, and terminals, that comprise an IBM 3710 Network Controller configuration.

IUCV. Inter-user communication vehicle.

JCL. Job control language.

job control language (JCL). * A problem-oriented language designed to express statements in a job that are used to identify the job or describe its requirements to an operating system.

Kanji. A character set of symbols used for Japanese ideograms. See also *double-byte character set*.

key-encrypting key. A key used in sessions with cryptography to encipher and decipher other keys. Contrast with *data-encrypting key*.

keyword. (1) **(TC97)** A lexical unit that, in certain contexts, characterizes some language construction. (2) * One of the predefined words of an artificial language. (3) One of the significant and informative words in a title or document that describes the content of that document. (4) A name or symbol that identifies a parameter. (5) A part of a command operand that consists of a specific character string (such as DSNAME=). See also *definition statement* and *keyword operand*. Contrast with *positional operand*.

keyword operand. An operand that consists of a keyword followed by one or more values (such as DSNAME=HELLO). See also *definition statement*. Contrast with *positional operand*.

keyword parameter. (1) A parameter that consists of a keyword followed by one or more values.

large message performance enhancement outbound (LMPEO). In VTAM, a facility in which VTAM reformats function management data (FMD) that exceed the maximum request unit (RU) size (as specified in the BIND) into a chain or partial chain of RUs.

last-in-chain (LIC). A request unit (RU) whose request header (RH) end chain indicator is on and whose RH begin chain indicator is off. See also *RU chain*.

LIC. (1) Last-in-chain. (2) In NCP, line interface card.

line. See *communication line*.

line mode. A form of screen presentation in which the information is presented a line at a time in the message area of the terminal screen. Contrast with *full-screen mode*.

line control. Synonym for *data link control protocol*.

line group. One or more telecommunication lines of the same type that can be activated and deactivated as a unit.

link. In SNA, the combination of the link connection and the link stations joining network nodes; for example: (1) a System/370 channel and its associated protocols, (2) a serial-by-bit connection under the control of Synchronous Data Link Control (SDLC). A link connection is the physical medium of transmission. A link, however, is both logical and physical. Synonymous with *data link*. See Figure 185 on page 697.

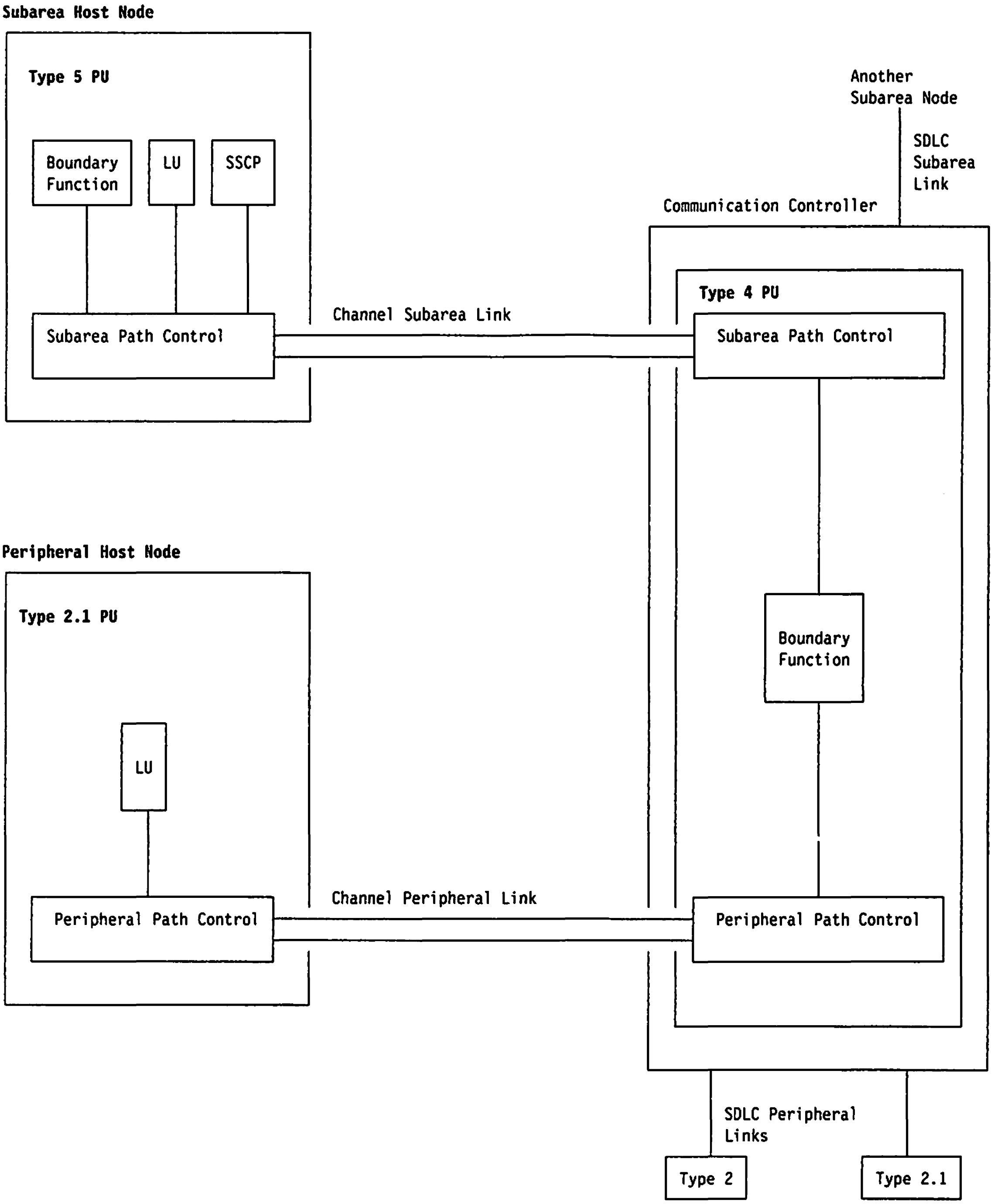


Figure 185. Links and Path Controls

link connection. In SNA, the physical equipment providing two-way communication between one link station and one or more other link stations; for example, a telecommunication line and data circuit terminating equipment (DCE).

link connection segment. A portion of the configuration that is located between two resources listed consecutively in the service point command service (SPCS) query link configuration request list.

link level 2 test. See *link test*.

link station. (1) In SNA, the combination of hardware and software that allows a node to attach to and provide control for a link. (2) In VTAM, a named resource within a subarea node that represents another subarea node that is attached by a subarea link. In the resource hierarchy, the link station is subordinate to the subarea link.

link test. In SNA, a test in which one link station returns data received from another link station without changing the data in order to test the operation of the link. Three tests can be made; they differ in the resources that are dedicated during the test.

LMPEO. Large message performance enhancement outbound.

load module. (ISO) A program unit that is suitable for loading into main storage for execution; it is usually the output of a linkage editor.

local address. In SNA, an address used in a peripheral node in place of an SNA network address and transformed to or from an SNA network address by the boundary function in a subarea node.

local area network (LAN). (1) A network in which a set of devices are connected to one another for communication and that can be connected to a larger network. See also *token ring*. (2) A network in which communications are limited to a moderately sized geographic area such as a single office building, warehouse, or campus, and which do not generally extend across public rights-of-way. Contrast with *wide area network*.

local session identification (LSID). In SNA, a field in a FID3 (format identification type 3) transmission header that contains an indication of the type of session (SSCP-PU, SSCP-LU, or LU-LU) and the local address of the peripheral logical unit (LU) or physical unit (PU).

logic error. In VTAM, an error condition that results from an invalid request; a program logic error.

logical record. (1) **(TC 97)** A set of related data or words considered to be a record from a logical viewpoint. (2) A unit of information normally pertaining to a single subject; a logical record is that user record requested of or given to the data management function. See also *basic conversation*.

logical unit (LU). In SNA, a port through which an end user accesses the SNA network and the functions provided by system services control points (SSCPs). An LU can support at least two sessions—one with an SSCP and one with another LU—and may be capable of supporting many sessions with other LUs. See also *network addressable unit (NAU)*, *peripheral LU*, *physical unit (PU)*, *system services control point (SSCP)*, *primary logical unit (PLU)*, and *secondary logical unit (SLU)*.

logical unit (LU) services. In SNA, capabilities in a logical unit to: (1) receive requests from an end user and, in turn, issue requests to the system services control point (SSCP) in order to perform the requested functions, typically for session initiation; (2) receive requests from the SSCP, for example to activate LU-LU sessions via Bind Session requests; and (3) provide session presentation and other services for LU-LU sessions. See also *physical unit (PU) services*.

logical unit (LU) 6.2. A type of logical unit that supports general communication between programs in a distributed processing environment. LU 6.2 is characterized by (1) a peer relationship between session partners, (2) efficient utilization of a session for multiple transactions, (3) comprehensive end-to-end error processing, and (4) a generic application program interface (API) consisting of structured verbs that are mapped into a product implementation.

logmode table. Synonym for *logon mode table*.

log off. To request that a session be terminated.

logoff. In VTAM, an unformatted session termination request.

log on. To initiate a session.

logon. In VTAM, an unformatted session initiation request for a session between two logical units. See *automatic logon* and *simulated logon*. See also *session-initiation request*.

logon data. In VTAM: (1) The user data portion of a field-formatted or unformatted session-initiation request. (2) The entire logon sequence or message from a logical unit (LU). Synonymous with *logon message*.

logon message. Synonym for *logon data*.

logon mode. In VTAM, a subset of session parameters specified in a logon mode table for communication with a logical unit. See also *session parameters*.

logon mode table. In VTAM, a set of entries for one or more logon modes. Each logon mode is identified by a logon mode name. Synonymous with *logmode table*.

logon-interpret routine. In VTAM, an installation exit routine, associated with an interpret table entry, that translates logon information. It may also verify the logon.

LSID. Local session identification.

LU. Logical unit.

LU 6.2 session. A session that is initiated by VTAM on behalf of a logical unit (LU) 6.2 application program, or a session initiated by a remote LU in which the application program specifies that VTAM is to control the session by using the APPCCMD macroinstruction.

LU type. In SNA, the classification of an LU-LU session in terms of the specific subset of SNA protocols and options supported by the logical units (LUs) for that session, namely:

> The mandatory and optional values allowed in the session activation request.
>
> The usage of data stream controls, function management headers (FMHs), request unit (RU) parameters, and sense codes.
>
> Presentation services protocols such as those associated with FMH usage.

LU types 0, 1, 2, 3, 4, 6.1, 6.2, and 7 are defined.

LU-LU session. In SNA, a session between two logical units (LUs) in an SNA network. It provides communication between two end users, or between an end user and an LU services component.

LU-LU session type. A deprecated term for *LU type*.

LU 6.2. Logical unit 6.2.

macroinstruction. (1) An instruction that when executed causes the execution of a predefined sequence of instructions in the same source language. (2) In assembler programming, an assembler language statement that causes the assembler to process a predefined set of statements called a macro definition. The statements normally produced from the macro definition replace the macroinstruction in the program. See also *definition statement*.

maintain system history program (MSHP). A program that facilitates the process of installing and servicing a VSE system.

maintenance and operator subsystem (MOSS). A subsystem of an IBM communication controller, such as the 3725 or the 3720, that contains a processor and operates independently of the rest of the controller. It loads and supervises the controller, runs problem determination procedures, and assists in maintaining both hardware and software.

maintenance services. In SNA, one of the types of network services in system services control points (SSCPs) and physical units (PUs). Maintenance services provide facilities for testing links and nodes and for collecting and recording error information. See also *configuration services*, *management services*, *network services*, and *session services*.

major node. In VTAM, a set of resources that can be activated and deactivated as a group. See *node* and *minor node*.

management services. In SNA, one of the types of network services in control points (CPs) and physical units (PUs). Management services are the services provided to assist in the management of SNA networks, such as problem management, performance and accounting management, configuration management and change management. See also *configuration services*, *maintenance services*, *network services*, and *session services*.

mandatory cryptographic session. Synonym for *required cryptographic session*.

mapped conversation. A type of conversation in which the data to be sent or received can be in a user-defined format. A logical unit (LU) that supports mapped conversations converts the user data to a format suitable for the basic conversation protocol boundary. See also *conversation* and *basic conversation*.

MDR. Miscellaneous data record.

message. (1) (TC97) A group of characters and control bit sequences transferred as an entity. (2) In VTAM, the amount of function management data (FMD) transferred to VTAM by the application program with one SEND request.

message unit. In SNA, the unit of data processed by any layer; for example, a basic information unit (BIU), a path information unit (PIU), or a request/response unit (RU).

MIC. Middle-in-chain.

middle-in-chain (MIC). A request unit (RU) whose request header (RH) begin chain indicator and RH end chain indicator are both off. See also *RU chain*.

migration. Installing a new version or release of a program when an earlier version or release is already in place.

minor node. In VTAM, a uniquely-defined resource within a major node. See *node* and *major node*.

miscellaneous data record (MDR). A record of a network hardware error recorded by the NCP and sent to the VTAM host that owns the failing component. Then VTAM writes the error on the operating system error data set.

modem. A device that modulates and demodulates signals transmitted over data communication facilities. The term is a contraction for modulator-demodulator.

mode name. A symbolic name for a set of session characteristics. For LU 6.2, a mode name and a partner LU name together define a group of parallel sessions having the same characteristics.

MOSS. Maintenance and operator subsystem.

MSHP. Maintain system history program.

multiple-domain network. In SNA, a network with more than one system services control point (SSCP). Contrast with *single-domain network*.

Multiple Virtual Storage (MVS). An IBM licensed program whose full name is the Operating System/Virtual Storage (OS/VS) with Multiple Virtual Storage/System Product for System/370. It is a software operating system controlling the execution of programs.

Multiple Virtual Storage for Extended Architecture (MVS/XA). An IBM licensed program whose full name is the Operating System/Virtual Storage (OS/VS) with Multiple Virtual Storage/System Product for Extended Architecture. Extended architecture allows 31-bit storage addressing. MVS/XA is a software operating system controlling the execution of programs.

MVS. Multiple Virtual Storage operating system.

MVS/XA. Multiple Virtual Storage for Extended Architecture operating system.

name translation. In SNA network interconnection, converting logical unit names, logon mode table names, and class-of-service names used in one network into equivalent names to be used in another network. This function can be provided through NetView and invoked by a gateway system services control point (SSCP) when necessary. See also *alias name*.

NAU. Network addressable unit.

NC. Network control.

NCCF. Network Communications Control Facility.

NCP. (1) Network Control Program (IBM licensed program). Its full name is Advanced Communications Function for the Network Control Program. Synonymous with *ACF/NCP*. (2) Network control program (general term).

NCP major node. In VTAM, a set of minor nodes representing resources, such as lines and peripheral nodes, controlled by a network control program. See *major node*.

negative response (NR). In SNA, a response indicating that a request did not arrive successfully or was not processed successfully by the receiver. Contrast with *positive response*. See *exception response*.

NetView. A system 370-based IBM licensed program used to monitor a network, manage it, and diagnose its problems.

NetView/PC. A PC-based IBM licensed program through which application programs can be used to monitor, manage, and diagnose problems in IBM Token-Ring networks, non-SNA communication devices, and voice networks.

network. (1) (TC97) An interconnected group of nodes. (2) In data processing, a user application network. See *path control network*, *public network*, *SNA network*, *subarea network*, *type 2.1 network*, and *user-application network*.

network address. In SNA, an address, consisting of subarea and element fields, that identifies a link, a link station, or a network addressable unit. Subarea nodes use network addresses; peripheral nodes use local addresses. The boundary function in the subarea node to which a peripheral node is attached transforms local addresses to network addresses and vice versa. See *local address*. See also *network name*.

network address translation. In SNA network interconnection, conversion of the network address assigned to a logical unit in one network into an address in an adjacent network. This function is provided by the gateway NCP that joins the two networks. See also *alias network address* and *real network address*.

network addressable unit (NAU). In SNA, a logical unit, a physical unit, or a system services control point. It is the origin or the destination of information transmitted by the path control network. Each NAU has a network address that represents it to the path control network. See also *network name*, *network address*, and *path control network*.

Network Communications Control Facility (NCCF). An IBM licensed program that is a base for command processors that can monitor, control, automate, and improve the operations of a network. Its function is included and enhanced in NetView's command facility.

network control (NC). In SNA, an RU category used for requests and responses exchanged for such purposes as activating and deactivating explicit and virtual routes and sending load modules to adjacent peripheral nodes. See also *data flow control layer* and *session control*.

network control mode. The functions of a network control program that enable it to direct a communication controller to perform activities such as polling, device addressing, dialing, and answering. Contrast with *emulation mode*.

Network Control Program (NCP). An IBM licensed program that provides communication controller support for single-domain, multiple-domain, and interconnected network capability. Its full name is Advanced Communications Function for the Network Control Program.

network control program. A program, generated by the user from a library of IBM-supplied modules, that controls the operation of a communication controller.

network controller. A concentrator and protocol converter used with SDLC links. By converting protocols, which manage the way data is sent and received, the IBM 3710 Network Controller allows the use of non-SNA devices with an SNA host processor.

network identifier (network ID). The network name defined to NCPs and hosts to indicate the name of the network in which they reside. It is unique across all communicating SNA networks. communication among domains.

Network Logical Data Manager (NLDM). An IBM licensed program that collects and correlates session-related data and provides online access to this information. It runs as an NCCF communication network management (CNM) application program. Its function is included and enhanced in NetView's session monitor.

network management vector transport (NMVT). A management services request/response unit (RU) that flows over an active session between physical unit management services and control point management services (SSCP-PU session).

network name. (1) In SNA, the symbolic identifier by which end users refer to a network addressable unit (NAU), a link, or a link station. See also *network address*. (2) In a multiple-domain network, the name of the APPL statement defining a VTAM application program is its network name and it must be unique across domains. Contrast with *ACB name*. See *uninterpreted name*.

network operator. (1) A person or program responsible for controlling the operation of all or part of a network. (2) The person or program that controls all the domains in a multiple-domain network. Contrast with *domain operator*.

Network Problem Determination Application (NPDA). An IBM licensed program that helps you identify network problems, such as hardware, software, and microcode, from a central control point using interactive display techniques. It runs as an NCCF communication network management (CNM) application program. Its function is included and enhanced in NetView's hardware monitor.

Network Routing Facility (NRF). An IBM licensed program that resides in the NCP, which provides a path for messages between terminals, and routes messages over this path without going through the host processor.

network services (NS). In SNA, the services within network addressable units (NAUs) that control network operation through SSCP-SSCP, SSCP-PU, and SSCP-LU sessions. See *configuration services, maintenance services, management services*, and *session services*.

network services (NS) header. In SNA, a 3-byte field in a function management data (FMD) request/response unit (RU) flowing in an SSCP-LU, SSCP-PU, or SSCP-SSCP session. The network services header is used primarily to identify the network services category of the request unit (RU) (for example, configuration services, session services) and the particular request code within a category.

Network Terminal Option (NTO). An IBM licensed program used in conjunction with NCP that allows certain non-SNA devices to participate in sessions with SNA application programs in the host processor. NTO converts non-SNA protocol to SNA protocol when data is sent to the host from a non-SNA device and reconverts SNA protocol to non-SNA protocol when data is sent back to the device.

NIB. Node initialization block.

NLDM. Network Logical Data Manager.

NMVT. Network management vector transport.

node. (1) In SNA, an endpoint of a link or junction common to two or more links in a network. Nodes can be distributed to host processors, communication controllers, cluster controllers, or terminals. Nodes can vary in routing and other functional capabilities. See *boundary node, host node, peripheral node*, and *subarea node* (including illustration). (2) In VTAM, a point in a network defined by a symbolic name. See *major node* and *minor node*.

node initialization block (NIB). In VTAM, a control block associated with a particular node or session that contains information used by the application program to identify the node or session and to indicate how communication requests on a session are to be handled by VTAM.

node name. In VTAM, the symbolic name assigned to a specific major or minor node during network definition.

node type. In SNA, a designation of a node according to the protocols it supports and the network addressable units (NAUs) that it can contain. Five types are defined: 1, 2.0, 2.1, 4, and 5. Type 1, type 2.0, and type

2.1 nodes are peripheral nodes; type 4 and type 5 nodes are subarea nodes. See also *type 2.1 node*.

Non-SNA Interconnection (NSI). An IBM licensed program that provides format identification (FID1/4) support for selected non-SNA facilities. Thus, it allows SNA and non-SNA facilities to share SDLC links. It also allows the remote concentration of selected non-SNA devices along with SNA devices.

nonswitched line. A telecommunication line on which connections do not have to be established by dialing. Contrast with *switched line*.

no response. In SNA, a value in the form-of-response-requested field of the request header (RH) indicating that no response is to be returned to the request, whether or not the request is received and processed successfully. Contrast with *definite response* and *exception response*.

normal flow. In SNA, a data flow designated in the transmission header (TH) that is used primarily to carry end-user data. The rate at which requests flow on the normal flow can be regulated by session-level pacing. Normal and expedited flows move in both the primary-to-secondary and secondary-to-primary directions. Contrast with *expedited flow*.

notify. A network services request that is sent by an SSCP to a logical unit (LU) to inform the LU of the status of a procedure requested by the LU.

NPDA. Network Problem Determination Application.

NR. Negative response.

NRF. Network Routing Facility.

NS. Network services.

NSI. Non-SNA Interconnection.

NTO. Network Terminal Option.

OAF. Origin address field.

OBR. Outboard record.

OIC. Only-in-chain.

OLU. Origin logical unit.

online. Stored in a computer and accessible from a terminal.

only-in-chain (OIC). A request unit for which the request header (RH) begin chain indicator and RH end chain indicator are both on. See also *RU chain*.

operand. (1) **(ISO)** An entity on which an operation is performed. (2) * That which is operated upon. An operand is usually identified by an address part of an instruction. (3) Information entered with a command name to define the data on which a command processor operates and to control the execution of the command processor. (4) An expression to whose value an operator is applied. See also *definition statement*, *keyword*, *keyword parameter*, and *parameter*.

operator. (1) In a language statement, the lexical entity that indicates the action to be performed on operands. (2) A person who operates a machine. See *network operator*. See also *definition statement*.

operator profile. In the NetView program, the resources and activities a network operator has control over. The statements defining these resources and activities are stored in a file that is activated when the operator logs on.

orderly closedown. The orderly deactivation of VTAM and its domain. An orderly closedown does not complete until all application programs have closed their application control blocks (ACBs). Until then, RPL-based operations continue; however, no new sessions can be established and no new ACBs can be opened. Contrast with *cancel closedown* and *quick closedown*.

origin address field (OAF). In SNA, a field in a FID0 or FID1 transmission header that contains the address of the originating network addressable unit (NAU). Contrast with *destination address field*. See also *format identification (FID) field* and *local session identification (LSID)*.

origin logical unit (OLU). The logical unit from which data is sent. Contrast with *destination logical unit (DLU)*.

outboard record. A record originated by I/O and communication components and supported by the access methods. It describes permanent errors or reports statistical data.

PAB. Process anchor block.

pacing. In SNA, a technique by which a receiving component controls the rate of transmission of a sending component to prevent overrun or congestion. See *session-level pacing*, *send pacing*, and *virtual route (VR) pacing*. See also *flow control*.

pacing group. In SNA, (1) The path information units (PIUs) that can be transmitted on a virtual route before a virtual-route pacing response is received, indicating that the virtual route receiver is ready to receive more PIUs on the route. Synonymous with *window*. (2) The requests that can be transmitted on the normal flow in one direction on a session before a session-level pacing response is received, indicating that the receiver is ready to accept the next group of requests.

pacing response. In SNA, an indicator that signifies a receiving component's readiness to accept another pacing group; the indicator is carried in a response header (RH) for session-level pacing, and in a transmission header (TH) for virtual route pacing.

packet switching. (TC97) The process of routing and transferring data by means of addressed packets so that a channel is occupied only during the transmission of a packet; upon completion of the transmission, the channel is made available for the transfer of other packets.

page. (1) The portion of a panel that is shown on a display surface at one time. (2) To move back and forth among the pages of a multiple-page panel. See also *scroll*. (3) (ISO) In a virtual storage system, a fixed-length block that has a virtual address and that can be transferred between real storage and auxiliary storage. (4) To transfer instructions, data, or both between real storage and external page or auxiliary storage.

panel. (1) A formatted display of information that appears on a terminal screen. See also *help panel* and *task panel*. Contrast with *screen*. (2) In computer graphics, a display image that defines the locations and characteristics of display fields on a display surface.

parameter. (1) **(ISO)** A variable that is given a constant value for a specified application and that may denote the application. (2) An item in a menu for which the user specifies a value or for which the system provides a value when the menu is interpreted. (3) Data passed to a program or procedure by a user or another program, namely as an operand in a language statement, as an item in a menu, or as a shared data structure. See also *keyword, keyword parameter*, and *operand*.

parallel links. In SNA, two or more links between adjacent subarea nodes.

parallel sessions. In SNA, two or more concurrently active sessions between the same two logical units (LUs) using different pairs of network addresses. Each session can have independent session parameters.

partitioned emulation program (PEP) extension. A function of a network control program that enables a communication controller to operate some telecommunication lines in network control mode while simultaneously operating others in emulation mode.

path. (1) In SNA, the series of path control network components (path control and data link control) that are traversed by the information exchanged between two network addressable units (NAUs). See also *explicit route (ER)*, *route extension*, and *virtual route (VR)*. (2) In VTAM when defining a switched major node, a potential dial-out port that can be used to reach that node. (3) In the NetView/PC program, a complete line in a configuration that contains all of the resources in the service point command service (SPCS) query link configuration request list.

path control (PC). The function that routes message units between network addressable units (NAUs) in the network and provides the paths between them. It converts the BIUs from transmission control (possibly segmenting them) into path information units (PIUs) and exchanges basic transmission units (BTUs) and one or more PIUs with data link control. Path control differs for peripheral nodes, which use local addresses for routing, and subarea nodes, which use network addresses for routing. See *peripheral path control* and *subarea path control*. See also *link*, *peripheral node*, and *subarea node*.

path control (PC) layer. In SNA, the layer that manages the sharing of link resources of the SNA network and routes basic information units (BIUs) through it. See also *BIU segment*, *blocking of PIUs*, *data link control layer*, and *transmission control layer*.

path control (PC) network. In SNA, the part of the SNA network that includes the data link control and path control layers. See *SNA network* and *user application network*. See also *boundary function*.

path information unit (PIU). In SNA, a message unit consisting of a transmission header (TH) alone, or of a TH followed by a basic information unit (BIU) or a BIU segment. See also *transmission header*.

PC. (1) Path control. (2) Personal Computer. Its full name is the IBM Personal Computer.

PCID. Procedure-correlation identifier.

PEP. Partitioned emulation program.

performance class. In the NetView program, a description of an objective or commitment of performance. It consists of a performance class name, boundary definitions, response time definition, response time ranges, and response time percentage objectives. Sessions may be assigned performance classes.

performance error. Synonym for *temporary error*.

peripheral host node. A node that provides an application program interface (API) for running application programs but does not provide SSCP functions and is not aware of the network configuration. The peripheral host node does not provide subarea node services. It has boundary function provided by its adjacent subarea. See *boundary node*, *host node*, *node*, *periph-*

eral node, *subarea host node*, and *subarea node*. See also *boundary function* and *node type*.

peripheral LU. In SNA, a logical unit representing a peripheral node.

peripheral node. In SNA, a node that uses local addresses for routing and therefore is not affected by changes in network addresses. A peripheral node requires boundary-function assistance from an adjacent subarea node. A peripheral node is a physical unit (PU) type 1, 2.0, or 2.1 node connected to a subarea node with boundary function within a subarea. See *boundary node*, *host node*, *node*, *peripheral host node*, *subarea host node*, and *subarea node*. See also *boundary function* and *node type*.

peripheral path control. The function in a peripheral node that routes message units between units with local addresses and provides the paths between them. See *path control* and *subarea path control*. See also *boundary function*, *peripheral node*, and *subarea node*.

peripheral PU. In SNA, a physical unit representing a peripheral node.

permanent error. A resource error that cannot be resolved by error recovery programs. Contrast with *temporary error*.

Personal Computer (PC). The IBM Personal Computer line of products including the 5150 and subsequent models.

physical connection. In VTAM, a point-to-point connection or multipoint connection. Synonymous with *connection*.

physical unit (PU). In SNA, a type of network addressable unit (NAU). A physical unit (PU) manages and monitors the resources (such as attached links) of a node, as requested by a system services control point (SSCP) through an SSCP-PU session. An SSCP activates a session with the physical unit in order to indirectly manage, through the PU, resources of the node such as attached links. See also *peripheral PU* and *subarea PU*.

physical unit (PU) services. In SNA, the components within a physical unit (PU) that provide configuration services and maintenance services for SSCP-PU sessions. See also *logical unit (LU) services*.

PIU. Path information unit.

plaintext. Synonym for *clear data*.

PLU. Primary logical unit.

polling. (1) * Interrogation of devices for purposes such as to avoid contention, to determine operational status, or to determine readiness to send or receive data. (2) (TC97) The process whereby stations are invited, one at a time, to transmit.

positional operand. An operand in a language statement that has a fixed position. See also *definition statement*. Contrast with *keyword operand*.

positive response. A response indicating that a request was received and processed. Contrast with *negative response*.

primary application program. In VTAM, an application program acting as the primary end of an LU-LU session.

primary half-session. In SNA, the half-session that sends the session activation request. See also *primary logical unit*. Contrast with *secondary half-session*.

primary logical unit (PLU). In SNA, the logical unit (LU) that contains the primary half-session for a particular LU-LU session. Each session must have a PLU and secondary logical unit (SLU). The PLU is the unit responsible for the bind and is the controlling LU for the session. A particular LU may contain both primary and secondary half-sessions for different active LU-LU sessions. Contrast with *secondary logical unit (SLU)*.

primary session. An extended recovery facility (XRF) session between the active application subsystem and a terminal user.

problem determination. The process of identifying the source of a problem; for example, a program component, a machine failure, telecommunication facilities, user or contractor-installed programs or equipment, an environment failure such as a power loss, or a user error.

process anchor block (PAB). In VTAM, a process scheduling services dispatch point.

procedure-correlation identifier (PCID). In SNA, a value used by a control point to correlate requests and replies.

profile. In the Conversational Monitor System (CMS) or the group control system (GCS), the characteristics defined by a PROFILE EXEC file that executes automatically after the system is loaded into a virtual machine. See also *operator profile*.

program temporary fix (PTF). A temporary solution or bypass of a problem diagnosed by IBM in a current unaltered release of the program.

protocol. (1) (CCITT/ITU) A specification for the format and relative timing of information exchanged between communicating parties. (2) (TC97) The set of rules governing the operation of functional units of a communication system that must be followed if communication is to be achieved. (3) In SNA, the meanings of, and the sequencing rules for, requests and responses used for managing the network, transferring data, and synchronizing the states of network components. See also *bracket protocol*. Synonymous with *line control discipline* and *line discipline*. See also *link protocol*.

protocol boundary. (1) A synonym for the architecturally defined logical unit (LU) 6.2 application program interface (API). (2) Any internal interface between components in a node.

PTF. Program temporary fix.

PU. Physical unit.

PU type. Physical unit type.

public network. A network established and operated by communication common carriers or telecommunication Administrations for the specific purpose of providing circuit-switched, packet-switched, and leased-circuit services to the public. Contrast with *user-application network*.

PU-PU flow. In SNA, the exchange between physical units (PUs) of network control requests and responses.

quick closedown. In VTAM, a closedown in which any RPL-based communication macroinstruction is terminated (posted complete with an error code) and no new sessions can be established and no new ACBs can be opened. See also *cancel closedown* and *orderly closedown*.

quiesce. (1) * To bring a system or a device to a halt by rejecting new requests for work. (2) In a VTAM application program, for one node to stop another node from sending synchronous-flow messages.

RDT. Resource definition table.

real name. The name by which a logical unit (LU), logon mode table, or class-of-service (COS) table is known within the SNA network in which it resides.

real network address. The address by which a logical unit (LU) is known within the SNA network in which it resides.

receive pacing. In SNA, the pacing of message units that the component is receiving. See also *send pacing*.

RECFMS. Record formatted maintenance statistics.

RECMS. Record maintenance statistics.

Recommendation X.21 (Geneva 1980). A Consultative Committee on International Telegraph and Telephone (CCITT) recommendation for a general purpose interface between data terminal equipment and data circuit equipment for synchronous operations on a public data network.

Recommendation X.25 (Geneva 1980). A Consultative Committee on International Telegraph and Telephone (CCITT) recommendation for the interface between data terminal equipment and packet-switched data networks. See also *packet switching*.

recommended action. Procedures suggested by NetView that can be used to determine the causes of network problems.

record formatted maintenance statistics (RECFMS). A statistical record built by an SNA controller and usually solicited by the host.

record maintenance statistics (RECMS). An SNA error event record built from an NCP or line error and sent unsolicited to the host.

release. For VTAM, to relinquish control of resources (communication controllers or physical units). See also *resource takeover*. Contrast with *acquire (2)*.

remote. Concerning the peripheral parts of a network not centrally linked to the host processor and generally using telecommunication lines with public right-of-way.

request header (RH). In SNA, control information preceding a request unit (RU). See also *request/response header (RH)*.

request parameter list (RPL). In VTAM, a control block that contains the parameters necessary for processing a request for data transfer, for establishing or terminating a session, or for some other operation.

request unit (RU). In SNA, a message unit that contains control information, end-user data, or both.

request/response header (RH). In SNA, control information, preceding a request/response unit (RU), that specifies the type of RU (request unit or response unit) and contains control information associated with that RU.

request/response unit (RU). In SNA, a generic term for a request unit or a response unit. See also *request unit (RU)* and *response unit*.

required cryptographic session. A cryptographic session in which all outbound data is enciphered and all inbound data is deciphered. Synonymous with *mandatory cryptographic session*. Contrast with *selective cryptographic session* and *clear session*.

resource. (1) Any facility of the computing system or operating system required by a job or task, and including main storage, input/output devices, the processing unit, data sets, and control or processing programs. (2) In the NetView program, any hardware or software that provides function to the network.

resource definition table (RDT). In VTAM, a table that describes the characteristics of each node available to VTAM and associates each node with a network address. This is the main VTAM network configuration table.

resource hierarchy. In VTAM, the relationship among network resources in which some resources are subordinate to others as a result of their position in the network structure and architecture; for example, the logical units (LUs) of a peripheral physical unit (PU) are subordinate to that PU, which, in turn, is subordinate to the link attaching it to its subarea node.

resource takeover. In VTAM, action initiated by a network operator to transfer control of resources from one domain to another. See also *acquire (2)* and *release*. See *takeover*.

resource types. In the NetView program, a concept to describe the organization of panels. Resource types are defined as central processing unit, channel, control unit, and I/O device for one category; and communication controller, adapter, link, cluster controller, and terminal for another category. Resource types are combined with data types and display types to describe display organization. See also *data types* and *display types*.

response header (RH). In SNA, a header, optionally followed by a response unit (RU), that indicates whether the response is positive or negative and that may contain a pacing response. See also *negative response*, *pacing response*, and *positive response*.

response time. (1) The amount of time it takes after a user presses the enter key at the terminal until the reply appears at the terminal. (2) For response time monitoring, the time from the activation of a transaction until a response is received, according to the response time definition coded in the performance class.

response time monitor (RTM). A feature available with certain hardware devices to allow measurement of response times, which may be collected and displayed by NetView.

response unit (RU). In SNA, a message unit that acknowledges a request unit; it may contain prefix information received in a request unit. If positive, the response unit may contain additional information (such as session parameters in response to Bind Session), or if negative, contains sense data defining the exception condition.

return code. * A code [returned from a program] used to influence the execution of succeeding instructions.

REX. Route extension.

RH. Request/response header.

ring. A network configuration where a series of attaching devices are connected by unidirectional transmission links to form a closed path.

route. See *explicit route* and *virtual route*.

route extension (REX). In SNA, the path control network components, including a peripheral link, that make up the portion of a path between a subarea node and a network addressable unit (NAU) in an adjacent peripheral node. See also *path*, *explicit route (ER)*, and *virtual route (VR)*.

RPL. Request parameter list.

RPL exit routine. In VTAM, an application program exit routine whose address has been placed in the EXIT field of a request parameter list (RPL). VTAM invokes the routine to indicate that an asynchronous request has been completed. See *EXLST exit routine*.

RPL-based macroinstruction. In VTAM, a macroinstruction whose parameters are specified by the user in a request parameter list.

RTM. Response time monitor.

RU. Request/response unit.

RU chain. In SNA, a set of related request/response units (RUs) that are consecutively transmitted on a particular normal or expedited data flow. The request RU chain is the unit of recovery: if one of the RUs in the chain cannot be processed, the entire chain is discarded. Each RU belongs to only one chain, which has a beginning and an end indicated by means of control bits in request/response headers within the RU chain. Each RU can be designated as first-in-chain (FIC), last-in-chain (LIC), middle-in-chain (MIC), or only-in-chain (OIC). Response units and expedited-flow request units are always sent as only-in-chain.

SAW data. Synonym for *session awareness (SAW) data*.

SC. Session control.

scanner. (1) A device capable of electronically reviewing amounts of data and translating the data into a machine readable form. (2) For the 3725 communication controller, a processor dedicated to controlling a small number of telecommunication lines. It provides the connection between the line interface coupler hardware and the central control unit.

scanner interface trace (SIT). A record of the activity within the communication scanner processor (CSP) for a specified data link between a 3725 Communication Controller and a resource.

screen. An illuminated display surface; for example, the display surface of a CRT or plasma panel. Contrast with *panel.*

scroll. To move all or part of the display image vertically to display data that cannot be observed within a single display image. See also *page (2).*

SDLC. Synchronous Data Link Control.

secondary half-session. In SNA, the half-session that receives the session-activation request. See also *secondary logical unit (SLU).* Contrast with *primary half-session.*

secondary logical unit (SLU). In SNA, the logical unit (LU) that contains the secondary half-session for a particular LU-LU session. An LU may contain secondary and primary half-sessions for different active LU-LU sessions. Contrast with *primary logical unit (PLU).*

secondary logical unit (SLU) key. A key-encrypting key used to protect a session cryptography key during its transmission to the secondary half-session.

segment. See *link connection segment.*

selective cryptographic session. A cryptographic session in which an application program is allowed to specify the request units to be enciphered. Contrast with *required cryptographic session* and *clear session.*

send pacing. In SNA, pacing of message units that a component is sending. See also *receive pacing.*

service point (SP). An entry point that supports applications that provide network management for resources not under the direct control of itself as an entry point. Each resource is either under the direct control of another entry point or not under the direct control of any entry point. A service point accessing these resources is not required to use SNA sessions (unlike a focal point). A service point is needed when entry point support is not yet available for some network management function.

service point command service (SPCS). An extension of the command facility in the NetView program that allows the host processor to communicate with a service point by using the communication network management (CNM) interface.

session. In SNA, a logical connection between two network addressable units (NAUs) that can be activated, tailored to provide various protocols, and deactivated, as requested. Each session is uniquely identified in a transmission header (TH) by a pair of network addresses, identifying the origin and destination NAUs of any transmissions exchanged during the session. See *half-session, LU-LU session, SSCP-LU session, SSCP-PU session,* and *SSCP-SSCP session.* See also *LU-LU session type* and *PU-PU flow.*

session activation request. In SNA, a request that activates a session between two network addressable units (NAUs) and specifies session parameters that control various protocols during session activity; for example, BIND and ACTPU. Contrast with *session deactivation request.*

session awareness (SAW) data. Data collected by NetView about a session that includes the session type, the names of session partners, and information about the session activation status. It is collected for LU-LU, SSCP-LU, SSCP-PU, and SSCP-SSCP sessions and for non-SNA terminals not supported by NTO. It can be displayed in various forms, such as most recent sessions lists.

session control (SC). In SNA, (1) One of the components of transmission control. Session control is used to purge data flowing in a session after an unrecoverable error occurs, to resynchronize the data flow after such an error, and to perform cryptographic verification. (2) A request unit (RU) category used for requests and responses exchanged between the session control components of a session and for session activation and deactivation requests and responses.

session control in-bound processing exit (SCIP). A user exit that receives control when certain request units (RUs) are received by VTAM.

session cryptography key. In SNA, a data encrypting key used to encipher and decipher function management data (FMD) requests transmitted in an LU-LU session that uses cryptography.

session data. Data about a session, collected by NetView, that consists of session awareness data and session trace data.

session deactivation request. In SNA, a request that deactivates a session between two network addressable units (NAUs); for example, UNBIND and DACTPU. Synonymous with *generic unbind.* Contrast with *session activation request.*

session information block (SIB). A control block that contains information about a particular SNA session.

session-initiation request. In SNA, an Initiate or logon request from a logical unit (LU) to a control point (CP) that an LU-LU session be activated.

session-level pacing. In SNA, a flow control technique that permits a receiver to control the data transfer rate (the rate at which it receives request units) on the normal flow. It is used to prevent overloading a receiver with unprocessed requests when the sender can generate requests faster than the receiver can process them. See also *pacing* and *virtual route pacing*.

session limit. (1) In SNA, (a) the maximum number of concurrently active LU-LU sessions a particular logical unit can support; (b) the limit that determines how many sessions may be active between two logical units (LUs) that are using LU 6.2 protocols and a given mode name. Each partner LU is allocated a minimum share of contention-winner sessions within this limit. (2) In the network control program, the maximum number of concurrent line-scheduling sessions on a non-SDLC, multipoint line.

session monitor. The component of NetView that collects and correlates session-related data and provides online access to this information.

session parameters. In SNA, the parameters that specify or constrain the protocols (such as bracket protocol and pacing) for a session between two network addressable units. See also *logon mode*.

session partner. In SNA, one of the two network addressable units (NAUs) having an active session.

session services. In SNA, one of the types of network services in the control point (CP) and in the logical unit (LU). These services provide facilities for an LU or a network operator to request that the SSCP initiate or terminate sessions between logical units. See *configuration services*, *maintenance services*, and *management services*.

session-termination request. In VTAM, a request that an LU-LU session be terminated.

session trace. In the NetView program, the function that collects session trace data for sessions involving specified resource types or involving a specific resource.

session trace data. Data relating to sessions that is collected by NetView whenever a session trace is started and that consists of session activation parameters, VTAM path information unit (PIU) data, and NCP data.

shadow resource. In VTAM, an alternate representation of a network resource that is retained as a definition for possible future use.

shared. Pertaining to the availability of a resource to more than one use at the same time.

shutdown. To stop or quiesce a NetView/PC or a NetView/PC application program.

SIB. Session information block.

simulated logon. A session-initiation request generated when a VTAM application program issues a SIMLOGON macroinstruction. The request specifies a logical unit (LU) with which the application program wants a session in which the requesting application program will act as the primary logical unit (PLU).

single-domain network. In SNA, a network with one system services control point (SSCP). Contrast with *multiple-domain network*.

SIT. Scanner interface trace.

SLU. Secondary logical unit.

SMF. System management facility.

SMP. System Modification Program.

SMP/E. System Modification Program Extended.

SNA. Systems Network Architecture.

SNA network. The part of a user-application network that conforms to the formats and protocols of Systems Network Architecture. It enables reliable transfer of data among end users and provides protocols for controlling the resources of various network configurations. The SNA network consists of network addressable units (NAUs), boundary function components, and the path control network.

SNA network interconnection. The connection, by gateways, of two or more independent SNA networks to allow communication between logical units in those networks. The individual SNA networks retain their independence.

SP. Service point.

SPCS. Service point command service.

span. In the NetView program, a user-defined group of network resources within a single domain. Each major or minor node is defined as belonging to one or more spans. See also *span of control*.

span of control. The total network resources over which a particular network operator has control. All the network resources listed in spans associated through profile definition with a particular network operator are within that operator's span of control.

SS. Start-stop.

SSCP. System services control point.

SSCP ID. In SNA, a number that uniquely identifies a system services control point (SSCP). The SSCP ID is used in session activation requests sent to physical units (PUs) and other SSCPs.

SSCP rerouting. In SNA network interconnection, the technique used by the gateway system services control point (SSCP) to send session-initiation request units (RUs), by way of a series of SSCP-SSCP sessions, from one SSCP to another, until the owning SSCP is reached.

SSCP-LU session. In SNA, a session between a system services control point (SSCP) and a logical unit (LU); the session enables the LU to request the SSCP to help initiate LU-LU sessions.

SSCP-PU session. In SNA, a session between a system services control point (SSCP) and a physical unit (PU); SSCP-PU sessions allow SSCPs to send requests to and receive status information from individual nodes in order to control the network configuration.

SSCP-SSCP session. In SNA, a session between the system services control point (SSCP) in one domain and the SSCP in another domain. An SSCP-SSCP session is used to initiate and terminate cross-domain LU-LU sessions.

SSP. System Support Programs (IBM licensed program). Its full name is Advanced Communications Function for System Support Programs. Synonymous with *ACF/SSP*.

ST. Session configuration screen abbreviation.

start option. In VTAM, a user-specified or IBM-supplied option that determines certain conditions that are to exist during the time a VTAM system is operating. Start options can be predefined or specified when VTAM is started.

statement. A language syntactic unit consisting of an operator, or other statement identifier, followed by one or more operands. See *definition statement*.

statement identifier. The lexical entity in a language statement that indicates the purpose of the statement, such as the action to be performed or the resource being defined. See also *definition statement*, *definition statement identifier*, and *operator*.

station. (1) One of the input or output points of a network that uses communication facilities; for example, the telephone set in the telephone system or the point where the business machine interfaces with the channel on a leased private line. (2) One or more computers, terminals, or devices at a particular location.

status code. In VTAM, information on the status of a resource as shown in a 10-character state code; for example, STATEACTIV for active.

status modifier. (1) In a System/370 status code, an indicator of the input/output (I/O) status. (2) In VTAM, a specific character appearing in specific positions of the status code; for example, B in the 10th position indicates a backup.

subarea. A portion of the SNA network consisting of a subarea node, any attached peripheral nodes, and their associated resources. Within a subarea node, all network addressable units, links, and adjacent link stations (in attached peripheral or subarea nodes) that are addressable within the subarea share a common subarea address and have distinct element addresses.

subarea address. In SNA, a value in the subarea field of the network address that identifies a particular subarea. See also *element address*.

subarea host node. A host node that provides both subarea function and an application program interface (API) for running application programs. It provides system services control point (SSCP) functions, subarea node services, and is aware of the network configuration. See *boundary node*, *communication management configuration host node*, *data host node*, *host node*, *node*, *peripheral node*, and *subarea node*. See also *boundary function* and *node type*.

subarea link. In SNA, a link that connects two subarea nodes. See *channel link* and *link*.

subarea node. In SNA, a node that uses network addresses for routing and whose routing tables are therefore affected by changes in the configuration of the network. Subarea nodes can provide gateway function, and boundary function support for peripheral nodes. Type 4 and type 5 nodes are subarea nodes. See *boundary node*, *host node*, *node*, *peripheral node*, and *subarea host node*. See also *boundary function* and *node type*.

subarea path control. The function in a subarea node that routes message units between network addressable units (NAUs) and provides the paths between them. See *path control* and *peripheral path control*. See also *boundary function*, *peripheral node*, and *subarea node*.

subarea PU. In SNA, a physical unit (PU) in a subarea node.

subsystem. A secondary or subordinate system, usually capable of operating independent of, or asynchronously with, a controlling system.

supervisor. The part of a control program that coordinates the use of resources and maintains the flow of processing unit operations.

supervisor call (SVC). A request that serves as the interface into operating system functions, such as allocating storage. The SVC protects the operating system from inappropriate user entry. All operating system requests must be handled by SVCs.

supervisor call (SVC) instruction. An instruction that interrupts the program being executed and passes control to the supervisor so that it can perform a specific service indicated by the instruction.

SVC. (1) Supervisor call. (2) Switched virtual circuit.

switched line. A communication line in which the connection between the communication controller and a remote link station is established by dialing.

switched major node. In VTAM, a major node whose minor nodes are physical units and logical units attached by switched SDLC links.

switched virtual circuit (SVC). An X.25 circuit that is dynamically established when needed. The X.25 equivalent of a switched line.

symptom string. A structured character string written to a file when VTAM detects certain error conditions.

Synchronous Data Link Control (SDLC). A discipline for managing synchronous, code-transparent, serial-by-bit information transfer over a link connection. Transmission exchanges may be duplex or half-duplex over switched or nonswitched links. The configuration of the link connection may be point-to-point, multipoint, or loop. SDLC conforms to subsets of the Advanced Data Communication Control Procedures (ADCCP) of the American National Standards Institute and High-Level Data Link Control (HDLC) of the International Standards Organization.

synchronous operation. In VTAM, a communication, or other operation in which VTAM, after receiving the request for the operation, does not return control to the program until the operation is completed. Contrast with *asynchronous operation*.

synchronous request. In VTAM, a request for a synchronous operation. Contrast with *asynchronous request*.

system management facility (SMF). A standard feature of MVS that collects and records a variety of system and job-related information.

System Modification Program (SMP). An operating system component that facilitates the process of installing and servicing an MVS system. See also *System Modification Program Extended.*

System Modification Program Extended (SMP/E). An IBM licensed program that facilitates the process of installing and servicing an MVS system. See also *System Modification Program*.

system services control point (SSCP). In SNA, a central location point within an SNA network for managing the configuration, coordinating network operator and problem determination requests, and providing directory support and other session services for end users of the network. Multiple SSCPs, cooperating as peers, can divide the network into domains of control, with each SSCP having a hierarchical control relationship to the physical units and logical units within its domain.

system services control point (SSCP) domain. The system services control point and the physical units (PUs), logical units (LUs), links, link stations and all the resources that the SSCP has the ability to control by means of activation requests and deactivation requests.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through and controlling the configuration and operation of networks.

System Support Programs (SSP). An IBM licensed program, made up of a collection of utilities and small programs, that supports the operation of the NCP.

takeover. The process by which the failing active subsystem is released from its extended recovery facility (XRF) sessions with terminal users and replaced by an alternate subsystem. See *resource takeover*.

TAP. Trace analysis program. Synonymous with *ACF/TAP*.

task. A basic unit of work to be accomplished by a computer. The task is usually specified to a control program in a multiprogramming or multiprocessing environment.

task panel. Online display from which you communicate with the program in order to accomplish the program's function, either by selecting an option provided on the panel or by entering an explicit command. See *help panel*.

TC. Transmission control.

TCAM. (1) Telecommunications Access Method. Synonymous with *ACF/TCAM*. (2) The IBM licensed program whose full name is Advanced Communications Function for TCAM and that provides queued message handling. TCAM Versions 1 and 2 are access methods, but TCAM Version 3 is a message handling subsystem.

TCAS. Terminal control address space.

telecommunication line. Any physical medium such as a wire or microwave beam, that is used to transmit data. Synonymous with *transmission line*.

teletypewriter exchange service (TWX). Teletypewriter service in which suitably arranged teletypewriter stations are provided with lines to a central office for access to other such stations throughout the U.S. and Canada. Both baudot and ASCII-coded machines are used. Business machines may also be used with certain restrictions.

temporary error. A resource failure that can be resolved by error recovery programs. Synonymous with *performance error*. Contrast with *permanent error*.

terminal. A device that is capable of sending and receiving information over a link; it is usually equipped with a keyboard and some kind of display, such as a screen or a printer.

terminal control address space (TCAS). The part of TSO/VTAM that provides logon services for TSO/VTAM users.

TERMINATE. In SNA, a request unit that is sent by a logical unit (LU) to its system services control point (SSCP) to cause the SSCP to start a procedure to end one or more designated LU-LU sessions.

TG. Transmission group.

TH. Transmission header.

threshold. In the NetView program, refers to a percentage value set for a resource and compared to a calculated error-to-traffic ratio.

TIC. Token-ring interface coupler.

time sharing option (TSO). An optional configuration of the operating system that provides conversational time sharing from remote stations.

time sharing option for VTAM (TSO/VTAM). An optional configuration of the operating system that provides conversational time sharing from remote stations in a network using VTAM.

token. A sequence of bits passed from one device to another along the token ring. When the token has data appended to it, it becomes a frame.

token ring. A network with a ring topology that passes tokens from one attaching device to another. For example, the IBM Token-Ring Network.

token-ring interface coupler (TIC). An adapter that can connect a 3725 Communication Controller to an IBM Token-Ring Network.

trace analysis program (TAP). An SSP program service aid that assists in analyzing trace data produced by VTAM, TCAM, and NCP and provides network data traffic and network error reports.

transaction program. (1) A program that is executed by or within an application program and performs services related to the processing of a transaction. (2) In VTAM, a program that performs services related to the processing of a transaction. One or more transaction programs may operate within a VTAM application program that is using the VTAM application program interface (API). In that situation, the transaction program would request services from the application program, using protocols defined by that application program. The application program, in turn, could request services from VTAM by issuing the APPCCMD macroinstruction.

transmission control (TC) layer. In SNA, the layer within a half-session that synchronizes and paces session-level data traffic, checks session sequence numbers of requests, and enciphers and deciphers end-user data. Transmission control has two components: the connection point manager and session control. See also *half-session*.

transmission control unit (TCU). A communication control unit whose operations are controlled solely by programmed instructions from the computing system to which the unit is attached; no program is stored or executed in the unit. Examples are the IBM 2702 and 2703 Transmission Controls. Contrast with *communication controller*.

transmission group (TG). In SNA, a group of links between adjacent subarea nodes, appearing as a single logical link for routing of messages. A transmission group may consist of one or more SDLC links (parallel links) or of a single System/370 channel.

transmission header (TH). In SNA, control information, optionally followed by a basic information unit (BIU) or a BIU segment, that is created and used by path control to route message units and to control their flow within the network. See also *path information unit*.

transmission line. Synonym for *telecommunication line*.

transmission priority. In SNA, a rank assigned to a path information unit (PIU) that determines its precedence for being selected by the transmission group control component of path control for forwarding to the next subarea node of the route used by the PIU.

transmission subsystem component (TSC). The component of VTAM that comprises the transmission control, path control, and data link control layers of SNA.

TSC. Transmission subsystem component.

TSO. Time sharing option.

TSO/VTAM. Time sharing option for VTAM.

TWX. Teletypewriter exchange service.

type 2.1 node (T2.1 node). A node that can attach to an SNA network as a peripheral node using the same protocols as type 2.0 nodes. Type 2.1 nodes can be directly attached to one another using peer-to-peer protocols. See *end node*, *node*, and *subarea node*. See also *node type*.

type 2.1 node (T2.1 node) control point domain. The CP, its logical units (LUs), links, link stations, and all resources that it activates and deactivates.

UNBIND. In SNA, a request to deactivate a session between two logical units (LUs). See also *session deactivation request*. Contrast with *BIND*.

unformatted. In VTAM, pertaining to commands (such as LOGON or LOGOFF) entered by an end user and sent by a logical unit in character form. The character-coded command must be in the syntax defined in the user's unformatted system services definition table. Synonymous with *character-coded*. Contrast with *field-formatted*.

unformatted system services (USS). In SNA products, a system services control point (SSCP) facility that translates a character-coded request, such as a logon or logoff request into a field-formatted request for processing by formatted system services and translates field-formatted replies and responses into character-coded requests for processing by a logical unit. Contrast with *formatted system services*. See also *converted command*.

uninterpreted name. In SNA, a character string that a system services control point (SSCP) is able to convert into the network name of a logical unit (LU). Typically, an uninterpreted name is used in a logon or Initiate request from a secondary logical unit (SLU) to identify the primary logical unit (PLU) with which the session is requested.

upstream. In the direction of data flow from the end user to the host. Contrast with *downstream*.

upstream device. For the IBM 3710 Network Controller, a device located in a network such that the device is positioned between the 3710 and a host. A communication controller upstream from the 3710 is an example of an upstream device. Contrast with *downstream device*.

upstream line. For the IBM 3710 Network Controller, a telecommunication line attaching a 3710 to an upstream device. Contrast with *downstream line*.

user. Anyone who requires the services of a computing system.

user-application network. A configuration of data processing products, such as processors, controllers, and terminals, established and operated by users for the purpose of data processing or information exchange, which may use services offered by communication common carriers or telecommunication Administrations. Contrast with *public network*.

user exit. A point in an IBM-supplied program at which a user routine may be given control.

USS. Unformatted system services.

value. (1) **(TC97)** A specific occurrence of an attribute, for example, "blue" for the attribute "color." (2) A quantity assigned to a constant, a variable, a parameter, or a symbol.

variable. In the NetView program, a character string beginning with & that is coded in a command list and is assigned a value during execution of the command list.

verb. (1) In SNA, the general name for a transaction program's request for communication services. (2) In VTAM, a programming language element in the logical unit (LU) 6.2 application program interface (API) that causes an LU 6.2 function to be performed.

virtual machine. A functional simulation of a computer and its associated devices.

Virtual Machine (VM). A licensed program whose full name is the Virtual Machine/System Product (VM/SP). It is a software operating system that manages the resources of a real processor to provide virtual machines to end users. As a time-sharing system control program, it consists of the virtual machine control program (CP), the conversational monitor system (CMS), the group control system (GCS), and the interactive problem control system (IPCS).

virtual route (VR). In SNA, a logical connection (1) between two subarea nodes that is physically realized as a particular explicit route, or (2) that is contained wholly within a subarea node for intranode sessions. A virtual route between distinct subarea nodes imposes a transmission priority on the underlying explicit route, provides flow control through virtual-route pacing, and provides data integrity through sequence numbering of path information units (PIUs). See also *explicit route (ER)*, *path*, and *route extension*.

virtual route (VR) pacing. In SNA, a flow control technique used by the virtual route control component of path control at each end of a virtual route to control the rate at which path information units (PIUs) flow over the virtual route. VR pacing can be adjusted according to traffic congestion in any of the nodes along the route. See also *pacing* and *session-level pacing*.

virtual route selection exit routine. In VTAM, an optional installation exit routine that modifies the list of virtual routes associated with a particular class of service before a route is selected for a requested LU-LU session.

virtual storage. (ISO) The notion of storage space that may be regarded as addressable main storage by the user of a computer system in which virtual addresses are mapped into real addresses. The size of virtual storage is limited by the addressing scheme of the computer system and by the amount of auxiliary storage available, not by the actual number of main storage locations.

Virtual Storage Access Method (VSAM). An access method for direct or sequential processing of fixed and variable-length records on direct access devices. The records in a VSAM data set or file can be organized in logical sequence by a key field (key sequence), in the physical sequence in which they are written on the data set or file (entry-sequence), or by relative-record number.

Virtual Storage Extended (VSE). An IBM licensed program whose full name is the Virtual Storage Extended/Advanced Function. It is a software operating system controlling the execution of programs.

Virtual Telecommunications Access Method (VTAM). An IBM licensed program that controls communication and the flow of data in an SNA network. It provides single-domain, multiple-domain, and interconnected network capability.

VIT. VTAM internal trace.

VM. Virtual Machine operating system. Its full name is Virtual Machine/System Product. Synonymous with *VM/SP*.

VM SNA console support (VSCS). A VTAM component for the VM environment that provides Systems Network Architecture (SNA) support. It allows SNA terminals to be virtual machine consoles.

VM/SP. Virtual Machine/System Product operating system. Synonym for *VM*.

VR. Virtual route.

VSAM. Virtual Storage Access Method.

VSCS. VM SNA console support.

VSE. Virtual Storage Extended operating system. Synonymous with *VSE/AF*.

VSE/AF. Virtual Storage Extended/Advanced Function operating svstem. Synonym for *VSE*.

VTAM. Virtual Telecommunications Access Method (IBM licensed program). Its full name is Advanced Communications Function for the Virtual Telecommunications Access Method. Synonymous with *ACF/VTAM*.

VTAM application program. A program that has opened an ACB to identify itself to VTAM and can now issue VTAM macroinstructions.

VTAM definition. The process of defining the user application network to VTAM and modifying IBM-defined characteristics to suit the needs of the user.

VTAM definition library. The operating system files or data sets that contain the definition statements and start options filed during VTAM definition.

VTAM internal trace (VIT). A trace used in VTAM to collect data on channel I/O, use of locks, and storage management services.

VTAM operator. A person or program authorized to issue VTAM operator commands. See *domain operator, program operator*, and *network operator (2)*.

VTAM operator command. A command used to monitor or control a VTAM domain. See also *definition statement.*

VTAM Terminal I/O Coordinator (VTIOC). The part of TSO/VTAM that converts TSO TGET, TPUT, TPG, and terminal control macroinstructions into SNA request units.

VTIOC. VTAM Terminal I/O Coordinator.

wide area network. A network that provides data communication capability in geographic areas larger than those serviced by local area networks. Wide area networks may extend across public rights-of-way. Contrast with *local area network*.

window. (1) In SNA, synonym for *pacing group*. (2) On a visual display terminal, a small amount of information in a framed-in area on a panel that overlays part of the panel.

wrap. In general, to go from the maximum to the minimum in computer storage. For example, the continuation of an operation from the maximum value in storage to the first minimal value.

XID. Exchange identification.

XRF. Extended recovery facility.

X.21. See *Recommendation X.21 (Geneva 1980).*

X.25. See *Recommendation X.25 (Geneva 1980).*

31-bit storage addressing. The storage address structure available in an MVS/XA operating system.

Index

Reader's Comment Form

VTAM
Diagnosis

Publication No. LY30-5601-1

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Note: Copies of IBM Publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.

Possible topics for comment are: clarity, accuracy, completeness, organization, coding, retrieval, and legibility.

Comments: ______________________________

____

____

____

____

____

____

What is your occupation? ______________________________

If you wish a reply, give your name, company, mailing address, and date:

____

____

____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the title page.)

LY30-5601-1

Reader's Comment Form

Fold and tape   **Please Do Not Staple**   Fold and tape

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS   PERMIT NO. 40   ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Dept. E15
P.O. Box 12195
Research Triangle Park, N.C. 27709-9990

Fold and tape   **Please Do Not Staple**   Fold and tape

IBM®

Publication Number
LY30-5601-1

File Number
S370/4300/30XX-50

Program Numbers
5665-313 (MVS/370)
5665-289 (MVS/XA)
5664-280 (VM/SP)
5666-313 (VSE)

Printed in USA

LY30-5601-01