LY38-3021-0
File No. S370-30

Advanced Communications Function for VTAM (ACF/VTAM)

Program Product

Logic: Overview

ACF/VTAM Release 2

Program Numbers: 5746-RC3 (DOS/VSE)
5735-RC2 (OS/VS)

IBM

__First Edition (March 1979)__

This edition applies to ACF/VTAM Release 2 for DOS/VSE (Program Number 5746-RC3), OS/VS1, and OS/VS2 MVS (Program Number 5735-RC2). Information about the optional Multisystem Networking Facility of ACF/VTAM is included. Information about the optional ACF/VTAM Encrypt/Decrypt Feature, available for OS/VS1 and OS/VS2 MVS only, is also included.

A form for readers' comments is provided at the back of this publication. If the form has been removed, address comments to IBM Corporation, Department 63T, Neighborhood Road, Kingston, New York 12401. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

PREFACE

This publication gives a logic overview of the Advanced Communications Function for the Virtual Teleprocessing Access Method-Release 2, which is referred to in this publication as ACF/VTAM. It describes the high-level logic for ACF/VTAM as it operates with the system control programs OS/VS and DOS/VSE with the program product VSE/Advanced Function installed.

This publication should be read to gain a general understanding of ACF/VTAM logic. The unique logic for the Multisystem Networking Facility is described in *ACF/VTAM Logic: Multisystem Networking Facility* (see Related Publications below). The Teleprocessing Online Test Executive Program (TOLTEP) logic is described in the ACF/VTAM publications for each operating system. Figure P-1 shows the relationships among the ACF/VTAM service publications.

ORGANIZATION OF THIS PUBLICATION

This publication is divided into two parts:

- Part 1, "Overview of ACF/VTAM," describes the general way that ACF/VTAM works and the interfaces between major groups of routines. Control blocks necessary for ACF/VTAM functions are described in Chapter 2. For a complete description of the data areas, refer to the publication *ACF/VTAM Data Areas*.

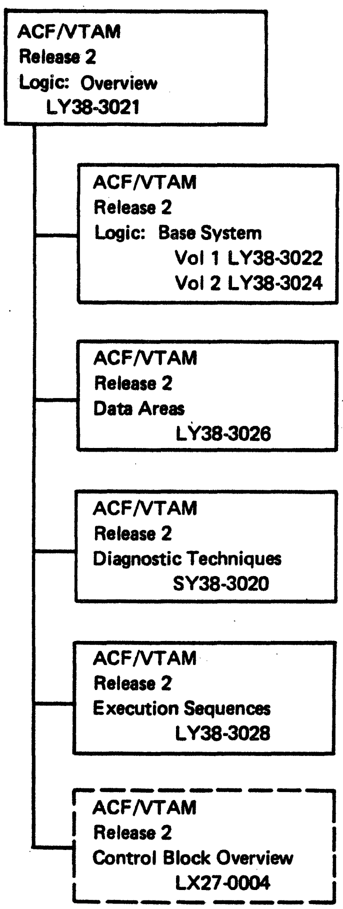


Figure P-1. ACF/VTAM Release 2 Service Publications

- Part 2, "Components of ACF/VTAM," consists of 16 chapters, one for each ACF/VTAM component. A component is a collection of ACF/VTAM modules that perform related functions. The ACF/VTAM logic manuals describe ACF/VTAM at the component, subcomponent, and module levels. With the coding documented in this manner, any function can be traced down to the level that is needed. For example, to find out how the Physical Unit Services operates, refer in this publication to Chapter 14, "Physical Unit Services." If more detailed information about Physical Unit Services is needed, refer to the detailed description in the ACF/VTAM logic publication for your system.

Most chapters in Part 2 contain an overview diagram of the method of operation of the component. The conventions used in these diagrams are explained at the beginning of Part 2.

RELATED PUBLICATIONS

Before reading this publication, you should be familiar with the external description of ACF/VTAM in _ACF/VTAM General Information: Concepts_, GC27-0463. To obtain a complete description of ACF/VTAM from the highest functional level down to the more detailed module level, you should use this publication in conjunction with the following appropriate publications:

- For DOS/VSE
  - _ACF/VTAM Logic: Base System, Volume 1_, LY38-3022
  - _ACF/VTAM Logic: Base System, Volume 2_, LY38-3024
- For OS/VS
  - _ACF/VTAM Logic: Base System, Volume 1_, LY38-3027
  - _ACF/VTAM Logic: Base System, Volume 2_, LY38-3032

The control blocks are described in detail in:

- _ACF/VTAM Data Areas (DOS/VSE)_, LY38-3026
- _ACF/VTAM Data Areas (OS/VS)_, LY38-3030

A graphic overview of ACF/VTAM control blocks is given in:

- _ACF/VTAM Control Block Overview (DOS/VSE)_, LX27-0004
- _ACF/VTAM Control Block Overview (OS/VS1)_, LX27-0009
- _ACF/VTAM Control Block Overview (OS/VS2 MVS)_, LX27-0013

The execution sequences of modules to process selected application program requests are described in:

- _ACF/VTAM Execution Sequences (DOS/VSE)_, LY38-3028
- _ACF/VTAM Execution Sequences (OS/VS1)_, LY38-3031
- _ACF/VTAM Execution Sequences (OS/VS2 MVS)_, LY38-3035

ACF/VTAM diagnostic information is given in:

- _ACF/VTAM Diagnostic Techniques (DOS/VSE)_, SY38-3020
- _ACF/VTAM Diagnostic Techniques (OS/VS)_, SY38-3029

The logic for the Multisystem Networking Facility of ACF/VTAM is described in:

- _ACF/VTAM Logic: Multisystem Networking Facility_, LY38-3023

The logic for the Encrypt/Decrypt Feature is described in:

- _ACF/VTAM Logic: Encrypt/Decrypt Feature_, LY38-3025

Notes:

1. In this publication, references to OS/VS are provided for planning purposes only.
2. All references to ACF/VTAM assume Release 2 in this publication.
3. Ensure that all DOS/VSE publications have been modified by the appropriate VSE/Advanced Function supplements.

CONTENTS

FIGURES

METHOD-OF-OPERATION DIAGRAMS

PART 1. OVERVIEW OF ACF/VTAM

CHAPTER 1. OVERVIEW OF ACF/VTAM OPERATIONS

INTRODUCTION TO ACF/VTAM

The Advanced Communications Function/Virtual Teleprocessing Access Method (ACF/VTAM) Release 2 is used by application programs to control data transmission to and from terminals in a data communications network. Communications controllers (3705s) are programmable devices that control the flow of information between ACF/VTAM and the remote terminals in the network. Because many access method functions are performed in the communications controllers, the application programs can communicate within the network without regard to how they are attached to the system. Thus, communication lines and communications controllers are transparent when coding application programs. ACF/VTAM provides the application programmer with macro facilities that allow for this communication.

In addition to its primary role of transmitting data, ACF/VTAM provides network-control and network-sharing capabilities. ACF/VTAM commands allow the network operator to redefine the network by activating or deactivating any part of the network. The operator can give ACF/VTAM the use of terminals, lines, communications controllers, or application programs or can terminate ACF/VTAM. The operator can monitor the data communications network by displaying status information about any device or application program. Diagnostic (debugging) facilities can also be controlled by altering the data communications network. The network-sharing capability allows multiple application programs to share leased, multipoint communication lines in the network. Because of this sharing capability, different programs can communicate with different terminals or nodes on the same line simultaneously. Figure 1-1 illustrates an ACF/VTAM network.

Four operations are involved in setting up and running an ACF/VTAM domain:

1. Initialization, during which ACF/VTAM is started in the host operating system.

2. Network configuration definition, during which the parts of the network and their relationships are defined and made available to ACF/VTAM.

3. Establishing sessions, which involves identifying application programs to ACF/VTAM and establishing sessions between logical units (for example, devices and application programs).

4. Processing I/O requests, issued by application programs or initiated by logical units in session with application programs, and controlling the flow of data through the domain.

The first two operations must be performed before ACF/VTAM can be used; the third and fourth operations are performed on a continuing basis.

This chapter describes how ACF/VTAM establishes the operating environment and processes transaction requests from the application programs. The chapters in Part 2 provide more detailed descriptions of the ACF/VTAM components (Figure 1-2). Details beyond this level are contained in the ACF/VTAM Logic Base System, Volumes 1 and 2. (See Figure 1-3.)

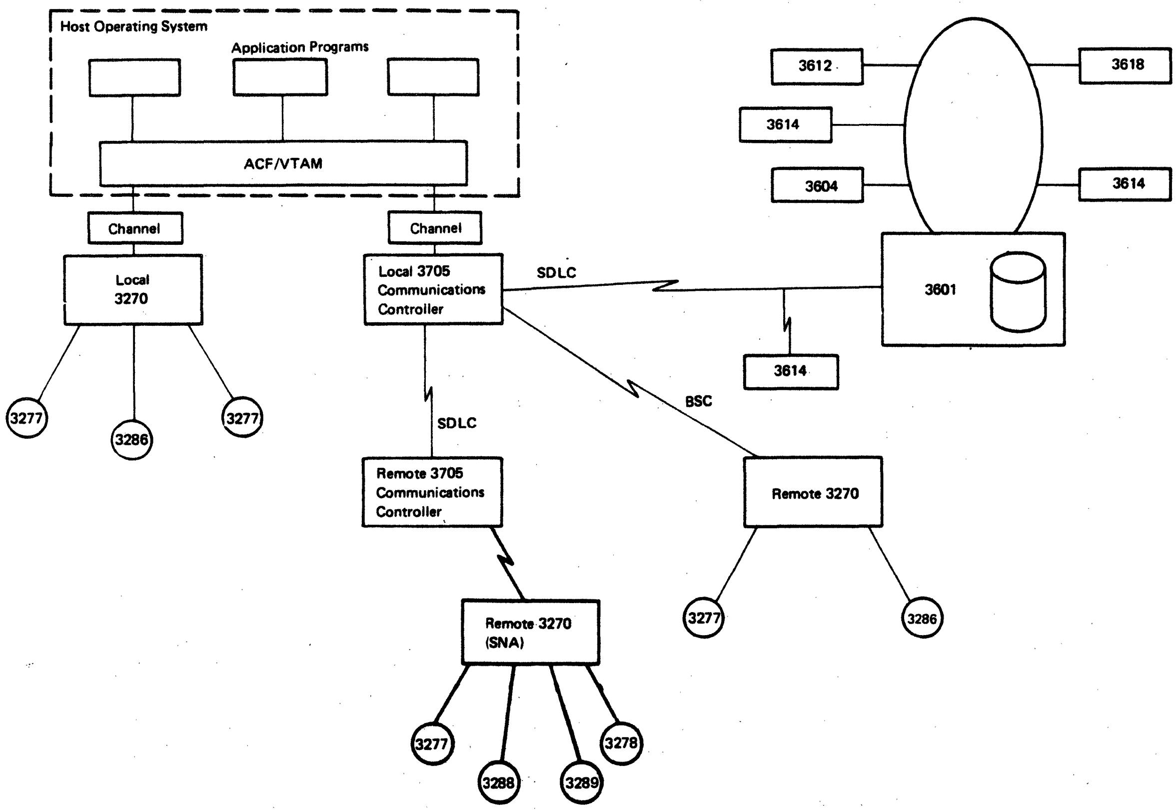


Figure 1-1. Illustration of an ACF/VTAM Network

INITIALIZING ACF/VTAM

The ACF/VTAM initialization routines, which start ACF/VTAM in the operating system, are invoked when the system operator issues an EXEC command (DOS/VSE) or a START command (OS/VS) that names the ACF/VTAM procedure. The ACF/VTAM host-attachment routines are given control first. The start options specified for the system are examined, and these options are used to establish various operating characteristics for the ACF/VTAM network. ACF/VTAM modules are loaded into the partition, storage is obtained for the ACF/VTAM communication vector table (ATCVT) and the configuration table (CONFT), and these tables are initialized. The buffer pools controlled by ACF/VTAM's storage management routines are built, ACF/VTAM queues are initialized, and ACF/VTAM files are opened. The ACF/VTAM initialization routines then attach sets of ACF/VTAM routines as subtasks (or threads) under ACF/VTAM control.

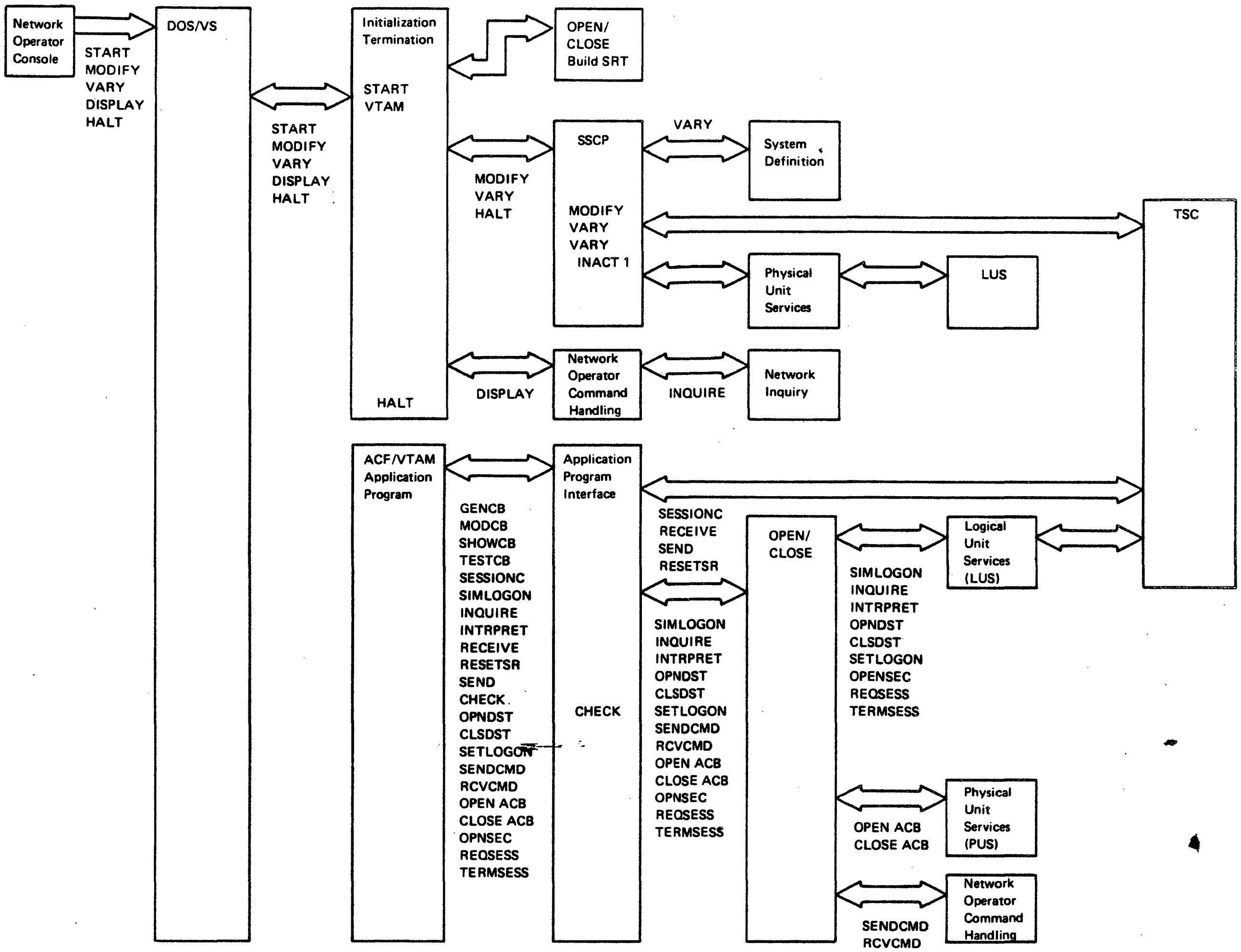


Figure 1-2. Use of ACF/VTAM Components in Processing Macros and Commands

In OS/VS, ACF/VTAM initialization may attach the following subtasks:

- Dump/load/restart, which dumps, loads, or restarts the NCP in a communications controller and performs other SSCP functions requiring task waits.
- System definition, which maintains tables that define the characteristics and status of each network node.
- TOLTEP, the Teleprocessing Online Test Executive Program.
- Tuning statistics, which collects data on the activity of each local programmable controller in the network.
- ACF/VTAM termination, which ensures that resources allocated to ACF/VTAM application programs and to ACF/VTAM control blocks are returned to the operating system at abnormal termination.

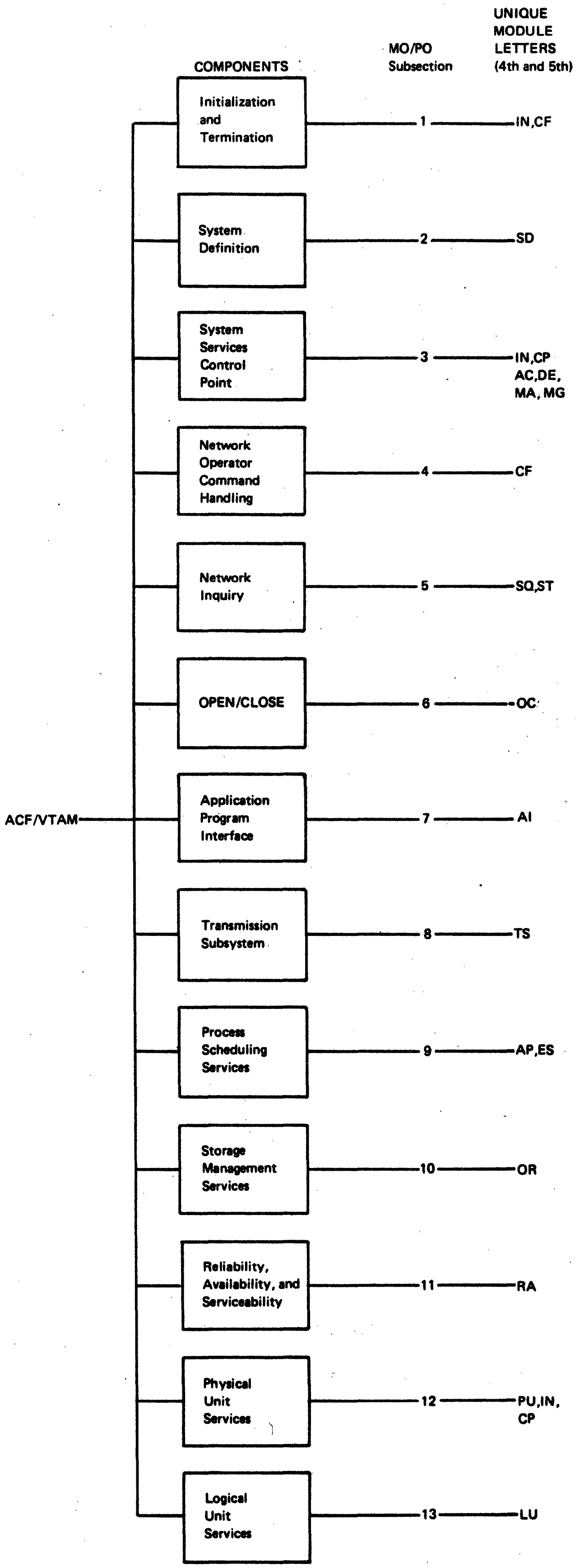


Figure 1-3. PLM Organization According to ACF/VTAM Components

In DOS/VSE, ACF/VTAM initialization can attach the following subtasks:

- ACF/VTAM request processor, which handles asynchronous requests.
- Trace I/O, which prints the output of the trace thread.
- TOLTEP, the Teleprocessing Online Test Executive Program.

ACF/VTAM initialization also creates the following ACF/VTAM threads:

- Initialization, which pseudo-attaches the other threads.
- System writer, which handles writing of messages to the system console.
- Operator control, which does initial processing of ACF/VTAM commands.
- Trace, which traces various events during ACF/VTAM processing.
- Tuning statistics, which collects data on the activity of each local programmable controller in the network.
- Dump/load/restart, which dumps, loads, or restarts the NCP in a communications controller and performs other SSCP functions requiring task waits.
- System definition, which builds or deletes tables that define the characteristics and status of each node in the domain.
- Subtask completion, which attaches and detaches TOLTEP and detaches other ACF/VTAM subtasks.

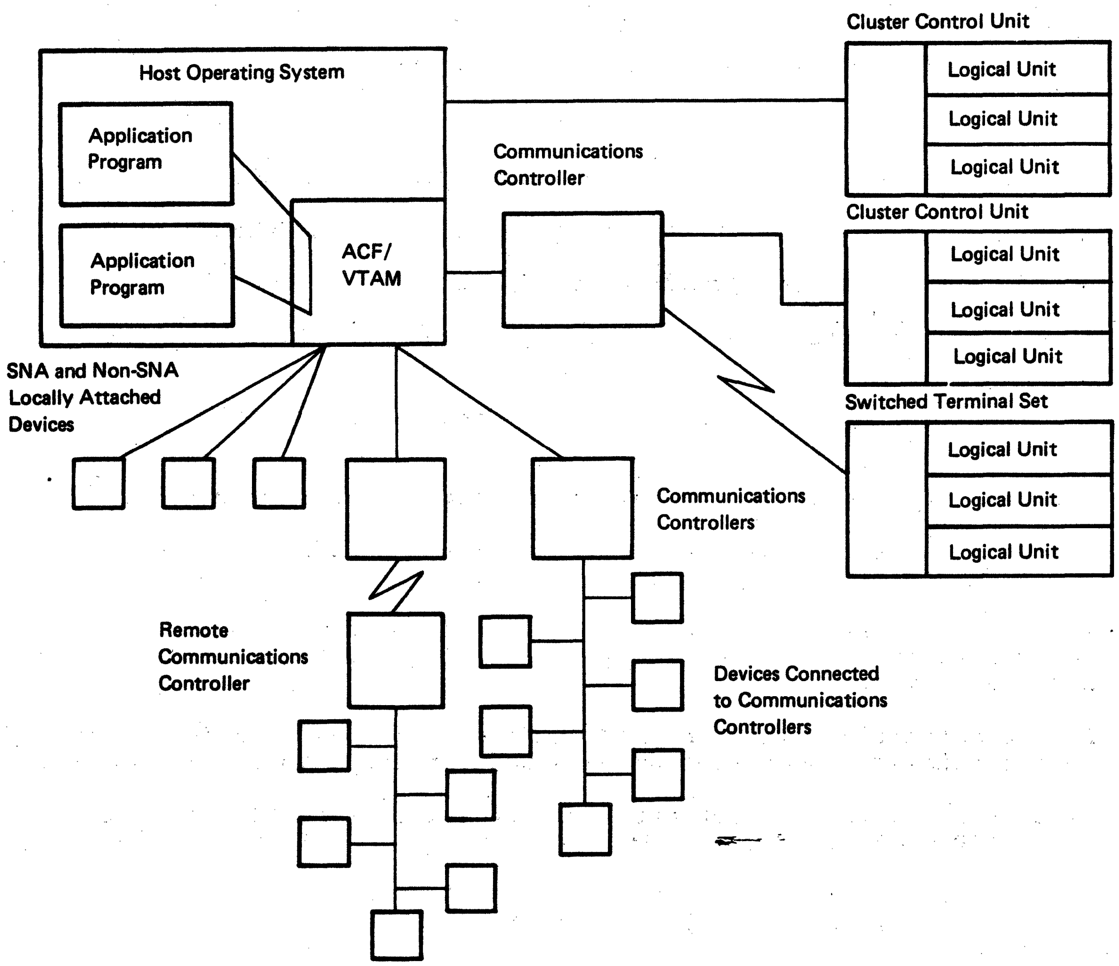
After this processing has been completed, ACF/VTAM is ready to accept ACF/VTAM commands, which can be used to further define the network configuration.

NETWORK CONFIGURATION DEFINITION

After the system has been initialized, the parts of the network and the way in which these parts are connected to the system are established (see Figure 1-4). The configuration of the network and the characteristics of its nodes are represented to ACF/VTAM by a series of control blocks called <u>resource definition table</u> (<u>RDT</u>) <u>segments</u>. Each RDT segment represents a major node in the domain with which ACF/VTAM will communicate. A resource definition table segment is composed of a header and a number of entries. Each entry represents a specific (minor) node in the network. There are seven types of major nodes:

<u>Application program major node</u>: An application program that can use ACF/VTAM is defined by an APPL statement. A group of application programs defined and named as a unit is an application program major node.

<u>NCP major node</u>: The specific nodes within the NCP major node are the lines, SNA physical units, logical units, cluster control units, and devices that are attached to the NCP.



Note: The devices in the network and their relationship to each other are specified in the ACF/VTAM definition library (the source statement library data set in a DOS/VSE system and the SYS1.VTAMLST data set in an OS/VS system). An entry for each network element is included in the symbol resolution table (SRT) and in the resource definition table (RDT). Except for switched terminal sets, an entry for each major node is made in the major node table (MNT). A specific node table is associated with each major node table entry. An entry for each device attached to the network through a communications controller, for each local device, and for each application program is made in the specific (minor) node table (SNT) associated with the major node.

Figure 1-4. Defining the Network Configuration

__Switched SNA major node__: The specific nodes of a switched SNA major node are SNA physical units and logical units that can be attached to the network through switched lines.

__Local SNA major node__: The specific nodes of a local SNA major node are the SNA physical and logical units attached to the data channel.

__Local Non-SNA major node__: Sets of local non-SNA terminals are defined to ACF/VTAM with LOCAL statements. The individual devices are the specific nodes. A set of such specific nodes defined and named as a group is a local non-SNA major node.

__Cross-domain resource major node__: One or more major nodes can be defined for cross-domain resources. Each major node represents a set of logical units in other domains that can be used for cross-domain communication; each minor node represents an application program or logical unit in another domain.

Cross-domain resource manager major node: One or more major nodes can be defined for cross-domain resource managers. Each major node represents a set of cross-domain resource managers; each minor node represents a cross-domain resource manager.

Initial Network Configuration Definition

Defining the initial configuration of the network can be a step in the series of operations initiated by a start procedure. The nodes in the network can be initially defined to ACF/VTAM through the CONFIG option in the start procedure. The CONFIG option identifies a file (DOS/VSE) or data set (OS/VS) in which the ACF/VTAM major node names to be activated are filed. These definition statements represent specific nodes in the network. The specific node definition statements that are filed together as a book of a file (DOS/VSE) represent a major node. In a multisystem network, communication paths can be defined as part of the initial configuration.

Building the Resource Definition Tables: The source statement library file is read, and definition statements are brought in and processed one at a time. The resource definition table (RDT) segments are built from the ACF/VTAM definition statements by the system definition component and chained off a queue anchor block (QAB).

Building the Minor (Specific) Node Tables: In addition to the resource definition table, the system definition component builds the specific node tables (SNTs). These are addressing tables that are used during ACF/VTAM execution to locate resource definition table entries and other control blocks that describe the characteristics of specific addressable nodes within this host's domain. There is a specific node table entry for each device and application program entry in the RDT. RDT entries for switched terminal sets, cross-domain resources, and cross-domain resource managers are not included in the major or specific node tables. The SNT for application programs is built during initialization. Its size depends on the MAXAPPL start option value.

ACF/VTAM configuration services routines activate and deactivate nodes in the network in response to operator commands and ACF/VTAM definition statements. They also perform automatic logon processing (that is, in addition to activating nodes, they allocate terminals to application programs). Once the initial system configuration has been specified, all the control blocks needed to process application program requests have been defined and initialized (see Figure 1-5).

Changing the Network's Configuration

ACF/VTAM's system definition routines set up control information about the network in such a way that the user can change the configuration while ACF/VTAM is running.

The configuration of the network is defined through the ACF/VTAM definition statements. The configuration is changed when the network operator issues some form of the VARY command. The VARY command can also be issued by certain application programs. These application programs use ACF/VTAM's program operator interface. An application using the program operator interface can perform most of the functions of a network operator.

The remainder of this section contains a general description of what ACF/VTAM does to change the network's configuration. The examples selected represent some of the more typical processing sequences.

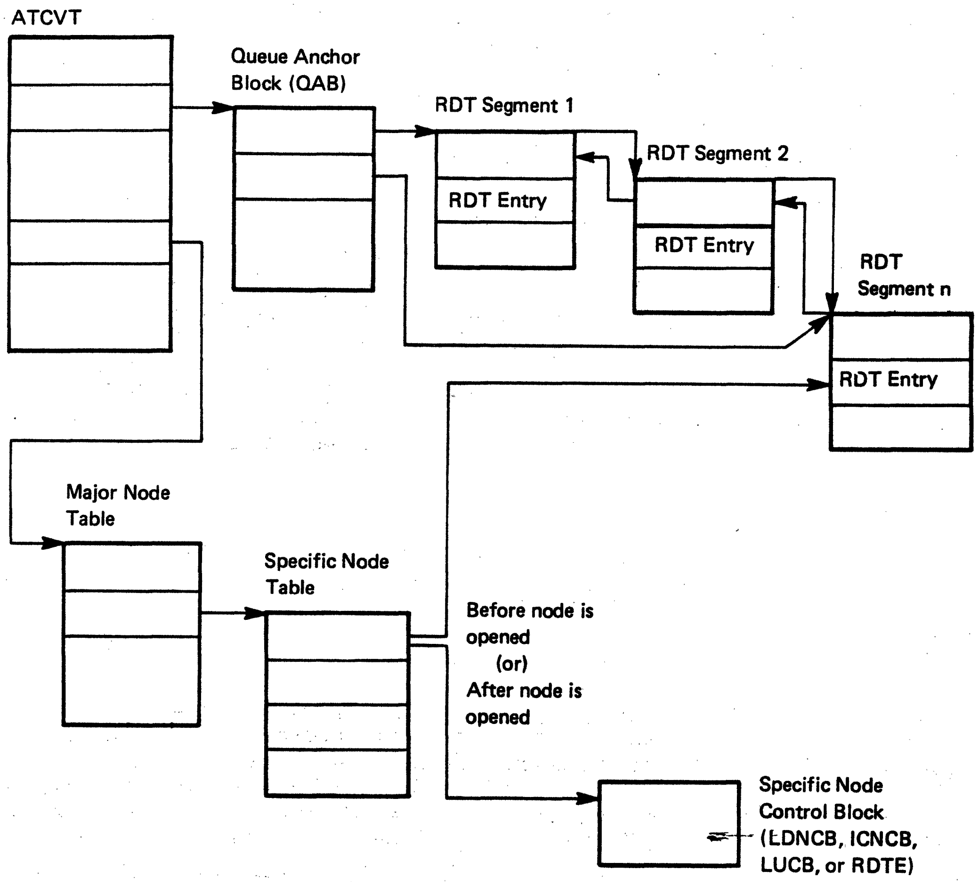


Figure 1-5. Control Block Structure after Initialization and Configuration Processing

The VARY ACT Command: When the VARY ACT command is passed from the operating system to ACF/VTAM, the command processor checks the syntax of the command; a code indicating that a node is to be activated and the ID of the node to be activated are then placed in the network configuration services parameter list (NCSPL). The NCSPL is the work element for most ACF/VTAM operator commands. The request/response unit processing element (RUPE) is the work element for most other ACF/VTAM commands. Most of the processing of the VARY command is done by the system services control point (SSCP) component. The SSCP first ensures that the command is valid. Finding the command valid, the SSCP determines whether a resource definition table entry (RDTE) exists for the node. If no entry exists for the node, the system definition component performs the same processing described earlier under "Initial Network Configuration Definition." After the RDT is built, entries exist for all the specific nodes (physical units, logical units, lines, application programs, cross-domain resource managers, etc.) in the major node. In addition to the RDT, control blocks such as the node control blocks, which describe characteristics of the nodes, and function management control blocks, which represent the sessions between the SSCP and the specific network nodes, are built. During activate processing, the SSCP activates the logical units when (1) the major node has been activated, (2) the links have been activated, and (3) the physical unit has been contacted and activated.

Examples of this sequence are given in several of the figures in Chapter 5.

The process of activating a node establishes sessions between the physical and logical units and the SSCP. A session is a logical connection between two addressable nodes that permits them to communicate with each other. Once a session is established between the SSCP and a logical unit, the SSCP aids in establishing and terminating sessions between that logical unit and other logical units.

The VARY INACT Command: The VARY INACT command reverses the effects of the VARY ACT command. It ends the session that exists between the SSCP and the logical or physical unit specified by the ID parameter.

One of two processing sequences occurs when VARY INACT is issued. If immediate deactivation was specified, the SSCP schedules the NSEXIT exit routine of the application program that is in session with the node being deactivated. If there is no NSEXIT routine, SSCP schedules the LOSTERM exit routine of the application program that is in session with the node being deactivated. Otherwise, if immediate deactivation was not specified, neither the NSEXIT nor the LOSTERM exit routine is scheduled. If the NSEXIT was not scheduled, the SSCP waits for the CLSDST macro instruction to be issued and for the session to be terminated by the OPEN/CLOSE component. After the session is terminated, the SSCP begins the deactivation sequence. This sequence ends the session between the SSCP and the node. The figures in Chapter 5 contain a detailed description of these sequences for various types of major nodes.

ESTABLISHING SESSIONS

Sessions between logical units are established and terminated in response to session requests. The logical unit that sends the session-activation request (BIND) is the primary half of the session and the logical unit receiving the session-activation request is the secondary half of the session. More than one session between two applications in the same or different domains is allowed. Any of these multiple (parallel) sessions may be initiated by the primary or secondary half of the session (Figure 1-6).

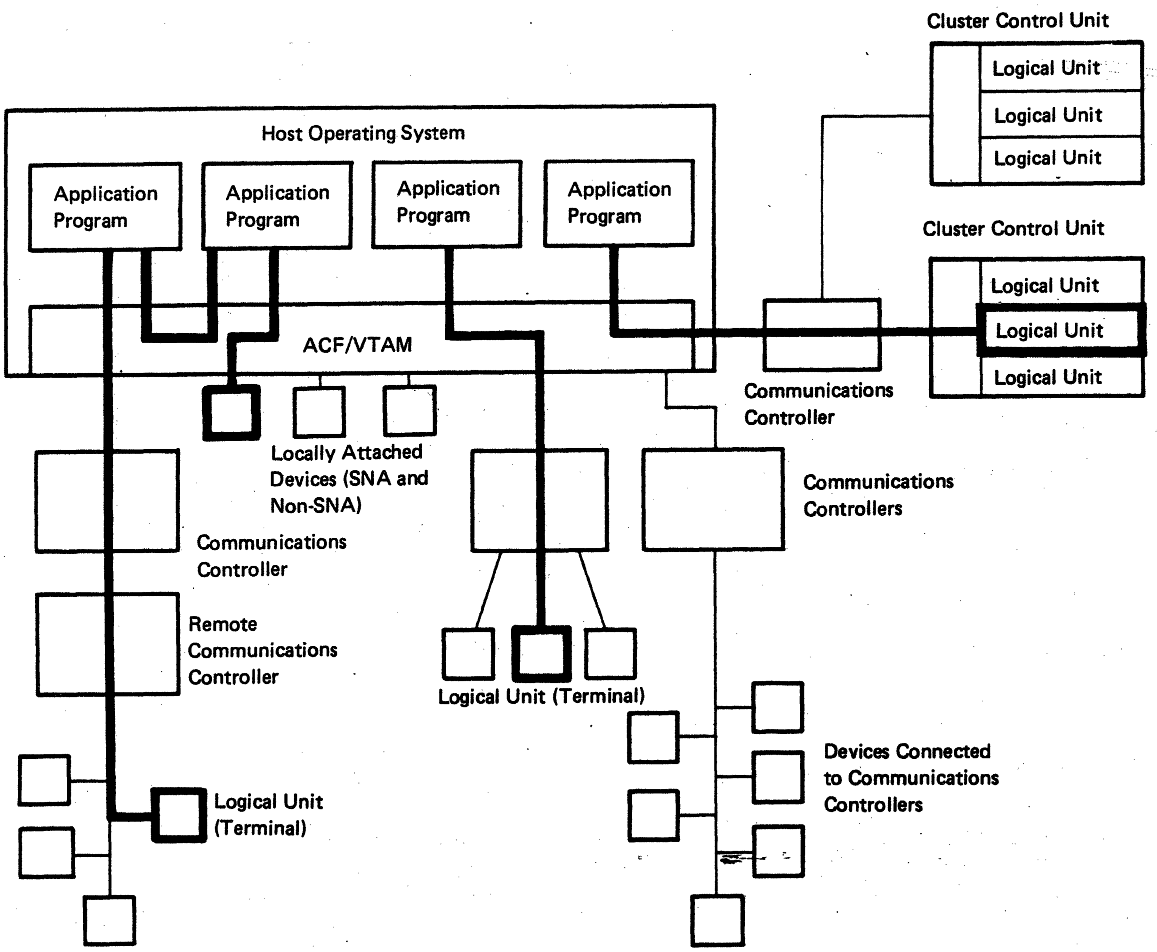
Before an application program can process I/O requests from another logical unit, three requirements must be met:

1. At least one of the logical units must be an application program that is known to ACF/VTAM. This knowledge is established as a result of an OPEN ACB macro instruction issued by the application program.

2. The logical units must be in session with each other. The logical unit that is to be the primary end of the session must issue an OPNDST macro instruction before this session can be established. (If the secondary end of the session is an application program, the secondary end can initiate the session by issuing a REQSESS macro instruction.) To establish a session between two logical units, ACF/VTAM must set up pointers, build and initialize session-establishment control blocks, and complete the necessary interface. This process is called binding. Two separate aspects of binding are:

   - Establishing a communication path between the two logical units

   - Selecting the ACF/VTAM routines needed to handle communication for these particular logical units



Note: A logical path to the logical unit (terminal) is allocated to the application program, and data can be transmitted between the application program and the logical unit.

Figure 1-6. Establishing Connection between Application Programs and Logical Units

3. If the logical units are in different domains, a session between the cross-domain resource managers (CDRMs) of each domain must be established and the cross-domain resources must be defined before the session can be established.

Establishing the Communication Path

When an application program issues an OPNDST macro instruction requesting a session with a logical unit that is not currently allocated to any other application program, ACF/VTAM's session-establishment routines allocate the logical unit to the application program. Except when the logical unit represents an application program, once a logical unit is allocated, requests for sessions with it are not honored until it becomes available. Those session requests must wait until the logical unit is released from its current allocation. The session-establishment routines build the control blocks required for the session. Information is obtained from the resource definition table (RDT) entries, the request parameter list (RPL), and the access method control block (ACB) to form a session information block (SIB).

Open destination routines then build the function management control block (FMCB), which associates the application program with the routines needed to support the session. The FMCB enables the application program and the logical unit to communicate with each other. An FMCB is queued to the logical unit control block (LUCB) for the application program.

Selecting ACF/VTAM Routines

The other aspect of binding is determining which routines shall be used to handle communications for the particular logical units in the session. The determination is made by using a control block called the skeleton destination vector table (DVT). This control block contains an entry for each routine that an application program can use for a session.

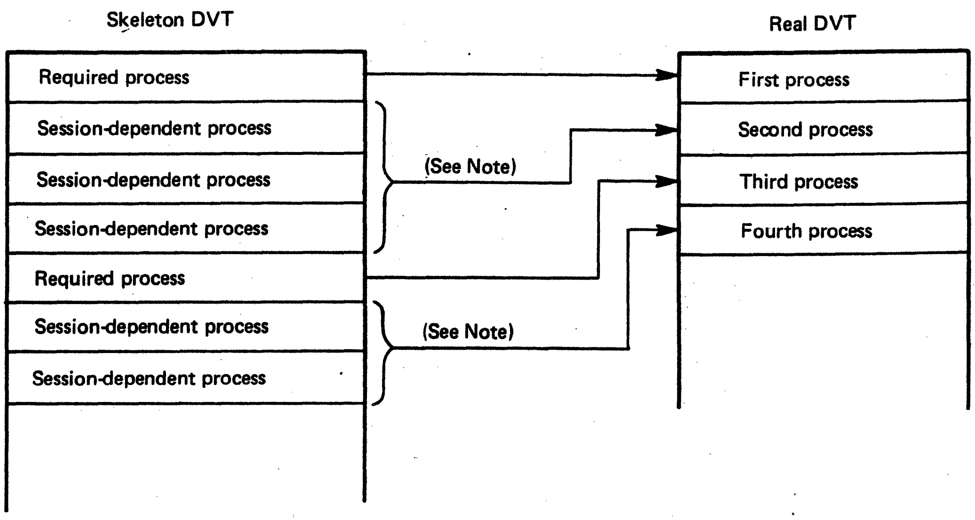
FMCBADD routines select those routines in the skeleton DVT that are required for this particular session, and then build a real DVT. The real DVT represents the routines to be used and the sequence in which they are to be executed.

After the FMCBADD routines have selected the required entries from the skeleton DVT, they convert the symbolic addresses of the routine to the storage addresses that were assigned to the routines when they were loaded during initialization. A pointer to the real DVT is then placed in the process anchor block (PAB) within the FMCB, and the name of the real DVT is placed in the DVT entry point table (EPT). ACF/VTAM's process scheduling routines use the real DVT to schedule execution of the routines in the proper order. Figure 1-7 illustrates the construction of a real DVT from a skeleton DVT. A skeleton DVT contains entries for required and optional processes. In the figure, the skeleton DVT contains required entries, which become the first and third entries in the real DVT. The skeleton DVT also contains a set of entries for three device-dependent routines. These routines perform equivalent functions for different types of logical units. The process that is necessary for the logical unit in the current session is selected and is then made the second entry in the real DVT. The skeleton DVT also contains another set of entries (two are shown) for device-dependent processes. The process that is needed for the logical unit (LU) in the current session is then selected and is made the fourth entry in the real DVT. The real DVT contains entries only for those processes that are needed to support a particular session.

Binding includes sending a Bind RU to the secondary resource. When binding is completed, I/O requests from application programs can be accepted and processed (see Figure 1-8).

PROCESSING I/O REQUESTS

An I/O operation can be initiated at either end of the session. Two independent flows (one normal, the other expedited) are available for sending and receiving requests and responses. Data can flow in both directions simultaneously because logical units can simultaneously receive input and send output. The direction of data flow at any one time is governed by the type of line and the line control discipline. Therefore, at the session level, requests and responses may be interspersed. At the link level, requests and responses will be interspersed only on a duplex line.



Note: When the real DVT is constructed from the skeleton DVT, only the processes required for communication with the terminal are selected.

Figure 1-7. Constructing a Real DVT from a Skeleton DVT

After an application program and a logical unit establish a session, data can be transferred between them. ACF/VTAM I/O-request processing consists of two general types of processing:

- Format conversion, which inserts control information within the requests so that requests sent to a particular type of logical unit are in the proper format and so that requests received from that type of logical unit are handled properly.

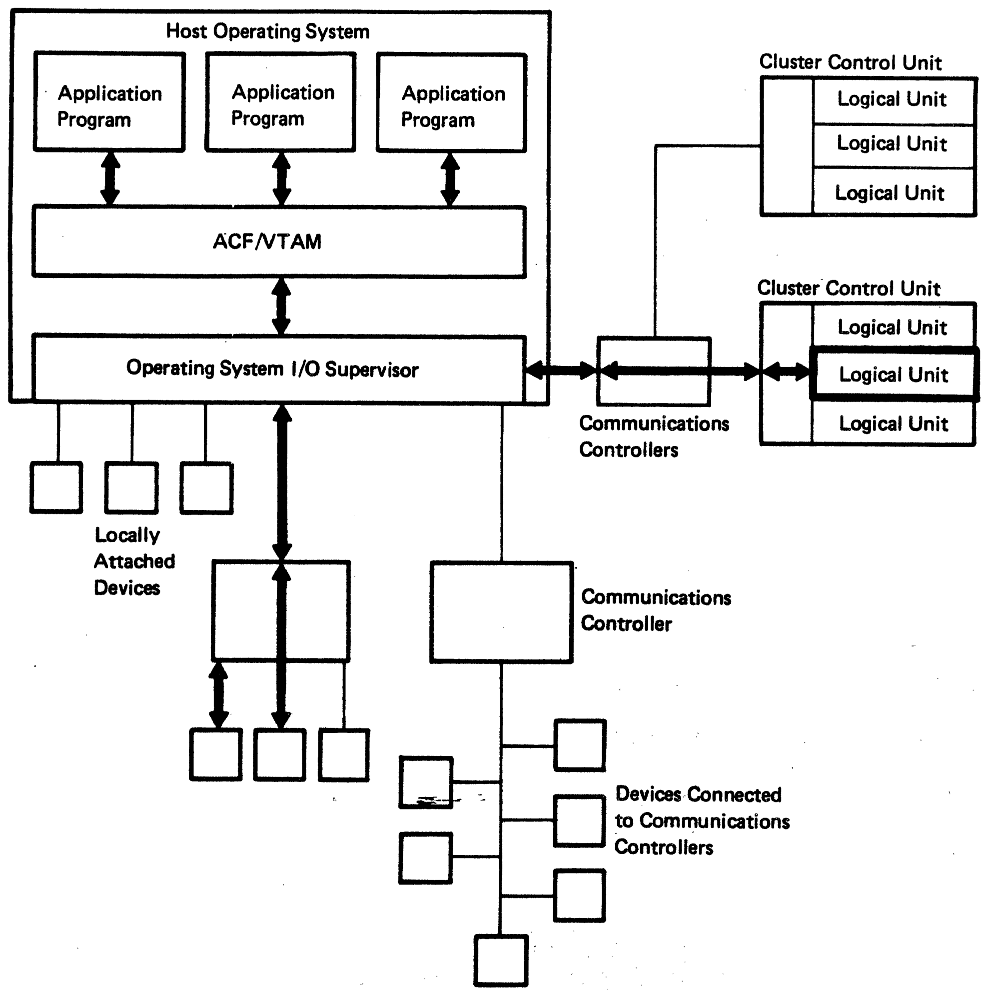
- Path control, which adds control information to the front of requests and responses so that they get to their destinations.

Format Conversion

Format conversion transforms information supplied to a logical unit into a request/response unit (RU), a request/response header (RH), and a transmission header (TH). Format conversion is done by the transmission subsystem component (TSC) and builds a request unit suitable for the request unit's destination. Format conversion processing consists of:

- Routines that insert control information into the request so that it is handled properly or displayed properly after it reaches its destination. The fundamental units of information in an SNA data communications system are request units and response units. Format conversion formats and attaches a control field (a request or response header) to the request or response unit. These control fields contain routing information, indications as to what type of response is desired (for example, respond only for error situation), and information as to whether a series of request units are related.

- Routines that format the request and insert other internal control information.



Note: After an application program and a terminal have been connected, ACF/VTAM passes data received from the application program to the operating system's I/O supervisor for transmission to the terminal. ACF/VTAM accepts input that the terminal sends to the I/O supervisor and passes it to the application program. An application program can send data to and receive data from another application program with which it has established a session.

Figure 1-8. Sending and Receiving Data

Path Control

Path control uses the information in the RH and TH to route a request or response to its destination. Path control, which is done by the transmission subsystem component (TSC), consists of adding additional control information to the request to ensure that the request is handled properly as it passes through the network. A transmission header is added when communicating with SNA terminals. The path control routines must handle requests for local terminals, as well as terminals attached through the NCP.

Outbound Communication

After the application program's I/O request has been processed by TSC, it is passed by TSC to the host operating system's I/O supervisor, which then initiates the I/O operation.

Inbound Communication

Data sent from a logical unit to an application program is received by the host operating system's I/O supervisor, which passes it to TSC. TSC then takes the data and passes it to the application program for which the data is intended. TSC then converts completion status information to a form that is meaningful to an application program and passes it to the application program through the RPL.

COMPONENTS THAT PERFORM I/O-REQUEST PROCESSING

Two ACF/VTAM components perform the processing needed to convert an application program's I/O request into a form that can be used by the operating system's I/O supervisor, that is, into a channel program:

- The application program interface (API)
- The transmission subsystem component (TSC)

The functions of each of these components are summarized below.

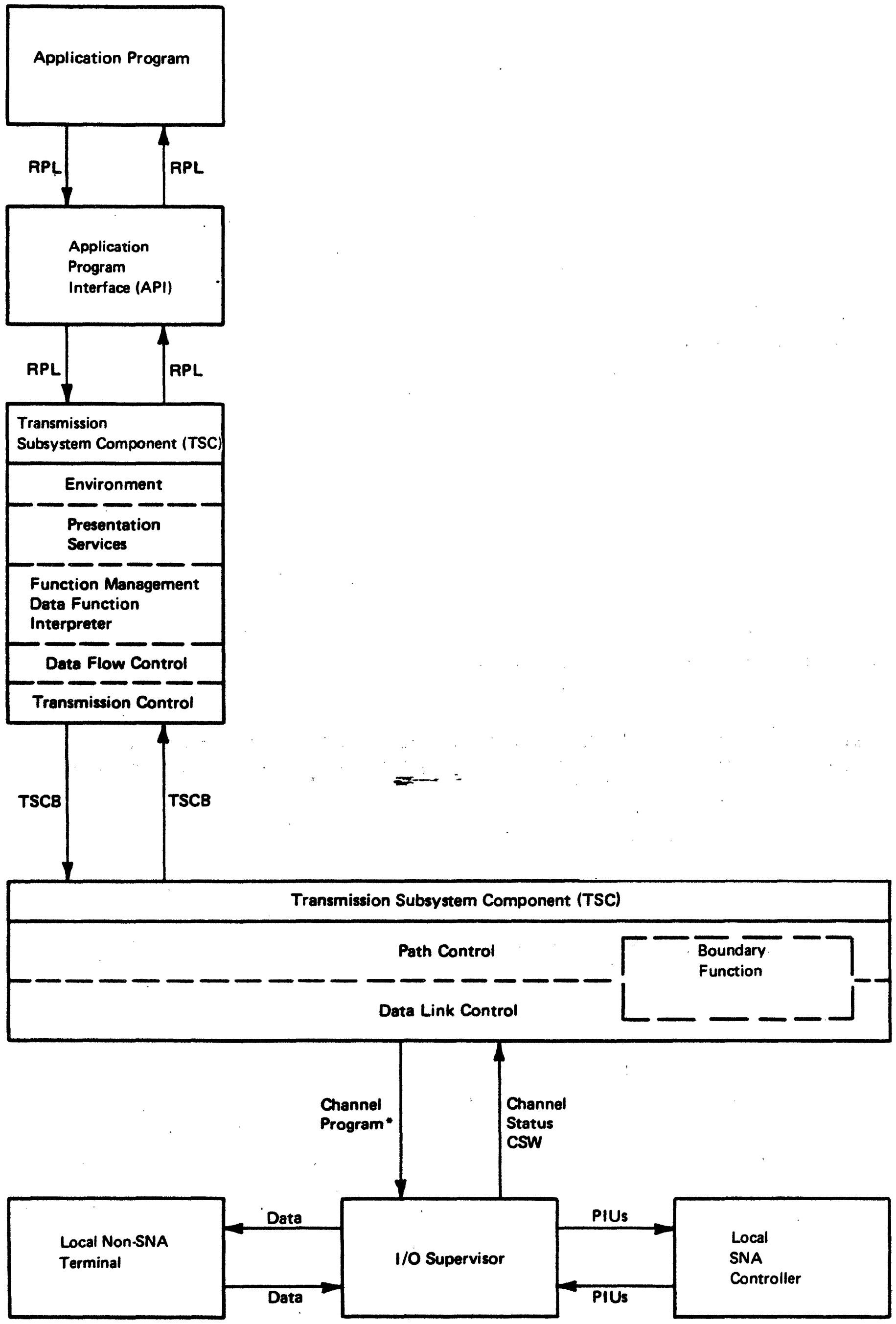
Application Program Interface (API)

These routines process requests (issued as macro instructions) for session-establishment, session termination, and data transfer. The API routines filter both outbound and inbound information between the application program and other ACF/VTAM routines. For requests coming from application programs, the API checks whether the request is valid, interprets it, and passes it to the appropriate subcomponent. After the requested operation has been performed, the API routines notify the application program of its completion and provide the feedback (status information) that resulted from the operation.

Transmission Subsystem Component (TSC)

The TSC is ACF/VTAM's interface with the operating system for outbound and inbound requests. The TSC receives an outbound request from the API, determines the type of request, ensures that it is a valid request, builds the appropriate channel program, and schedules it with the host operating system's I/O supervisor. The TSC receives an inbound request, response, or feedback from the I/O supervisor, and passes it to the API. When the TSC receives a request from a logical unit, if a RECEIVE request is not currently queued, the inbound request is queued. The request is sent to the application program only when it issues a RECEIVE request.

Figure 1-9 illustrates how these components (API and TSC) transmit data from an application program and process the resulting feedback. Detailed examples of processing I/O requests are in <u>ACF/VTAM Execution Sequences</u>.



[1]The channel program consists of WRITE (output) and READ (input) CCWs.

Figure 1-9. Flow of Control and Data for I/O Requests

I/O Facilities

These macro instructions control data transfer between an application program and an SNA terminal:

SEND: Causes ACF/VTAM to transmit a record or control information from an application program to a logical unit. The four types of records that can be transmitted are application program data, application program response to data received, a control request, and a response to a control request.

RECEIVE: Causes ACF/VTAM to transfer a record or control information from an ACF/VTAM buffer to a storage area in the application program. RECEIVE may be limited to one type of record, or it might handle all four types.

SESSIONC: Includes starting a transmission (Start Data Traffic), purging requests (Clear), resetting sequence numbers (STSN), requesting recovery (RQR), rejecting a Bind by a secondary end, and responding to an STSN or SDT received with the SCIP exit routine.

RESETSR: Cancels pending RECEIVE request(s) and/or switches a logical unit's continue-specific mode or continue-any mode.

Before an I/O request can be issued, an RPL must be constructed that points to the opened ACB, contains information describing the type of I/O request, and contains a communication ID (CID).

After a session has been set up, both nodes can start to send data immediately, provided that the session does not require a Start Data Traffic command. (If the session requires a Start Data Traffic command, neither node can send data until the primary end of the session issues the command.)

Application programs transfer data by issuing SEND and RECEIVE macro instructions. The flow of control and data that results from issuing these macro instructions is illustrated in Figures 1-10 and 1-11.

SEND

This macro instruction requests that ACF/VTAM transfer a request or a response to a specific logical unit. An FID1 (Format ID=1) PIU specifies the format used for SNA terminals. Data is transferred from an area pointed to by the RPL; response information is specified symbolically in the RPL. Assuming that data is being sent from an application program, the TSC copies the data, builds a request header (RH) and transmission header (TH), formats channel command words (CCWs), and schedules the channel program. The transmission header contains information for routing the data through the network. If a response is necessary, the response from the logical unit is queued. When the application program issues a RECEIVE for the logical unit, the response is then passed to the application program. Figure 1-10 illustrates the processing of a SEND request. The example assumes that POST=SCHED was specified for the SEND.

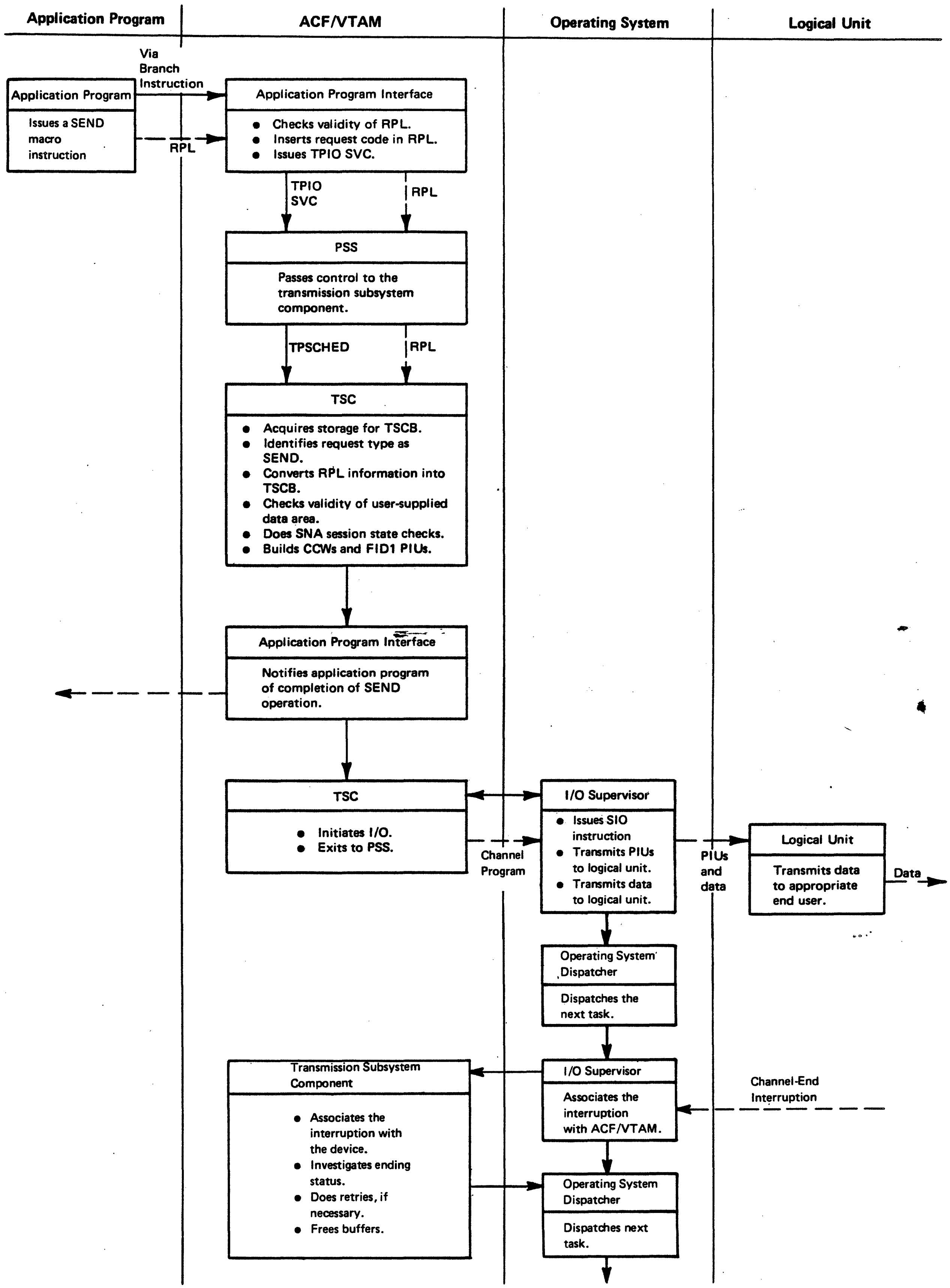


Figure 1-10. Processing a SEND Request

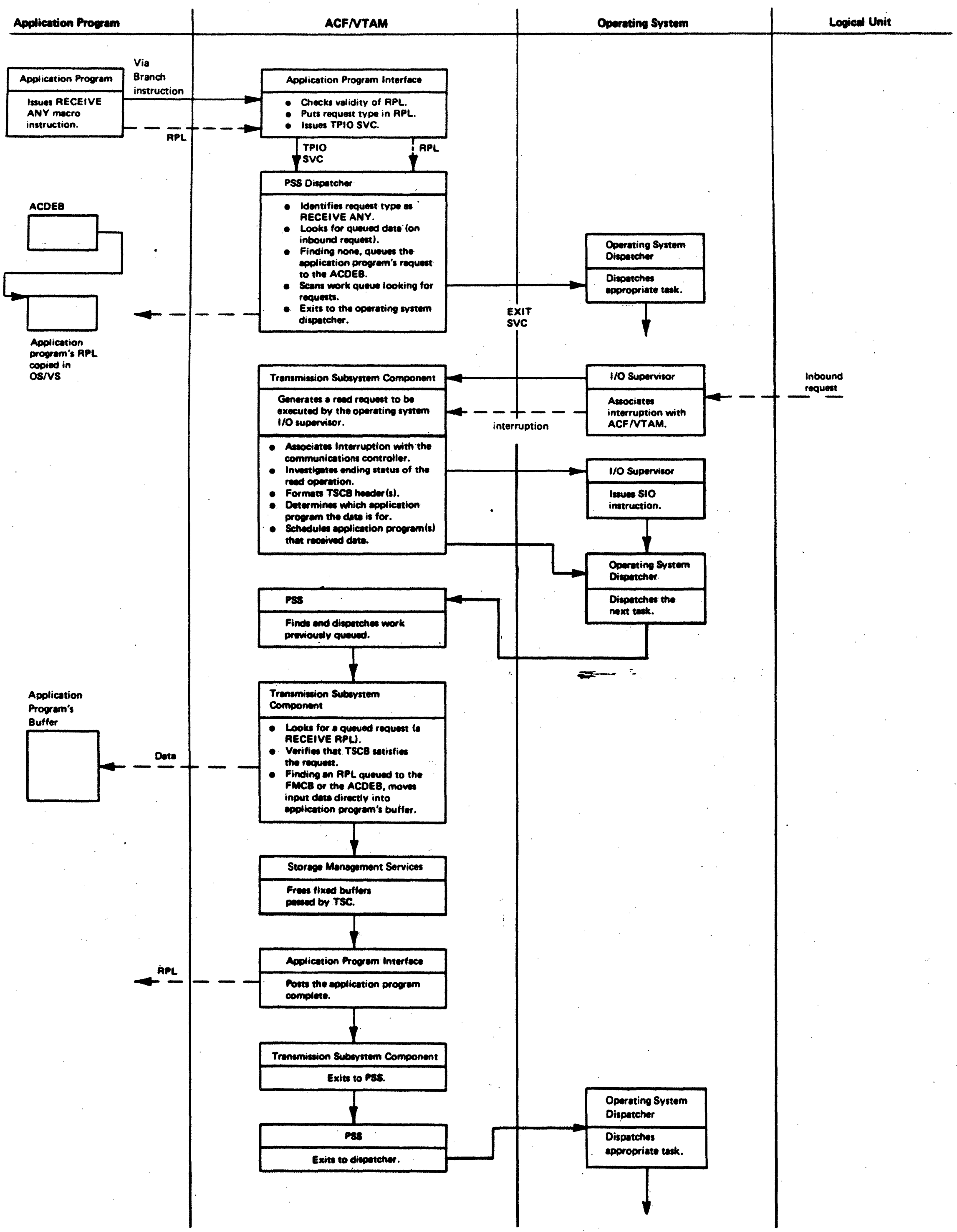


Figure 1-11. Processing a RECEIVE Request

RECEIVE

The RECEIVE macro instruction transfers data from internal ACF/VTAM buffers to the application program's input data area. RECEIVE can be issued in two formats: RECEIVE SPEC and RECEIVE ANY. RECEIVE SPEC requests data that was sent in by a specific logical unit, while RECEIVE ANY requests data that was sent in by any logical unit with which the application program is in session.

The RECEIVE operation code is passed to ACF/VTAM in the RPL:

- If data is available that will satisfy the RECEIVE request, the data is moved into the user's area.
- If no data is available to satisfy the RECEIVE request and if the queue option was specified, the RPL is placed on a wait queue until data is received to satisfy it.
- If no data is available to satisfy the RECEIVE request and the queue option was specified, RECEIVE will fail.

A RECEIVE request does not cause any activity in the network. All processing takes place in ACF/VTAM.

When the RECEIVE request is satisfied, the RPL is posted to indicate to the application program that the RPL can be reused.

Figure 1-11 illustrates the processing of a RECEIVE ANY request. The example assumes that no data is available at the time the RECEIVE is issued, so the RECEIVE is queued. Later, data arrives to satisfy the RECEIVE.

OTHER ACF/VTAM OPERATIONS

In addition to initialization, network configuration definition, establishing and terminating of sessions, and processing of I/O requests, ACF/VTAM also includes components that perform process scheduling services, storage management services, reliability, availability, and serviceability functions, termination, and multisystem networking functions. These components are described briefly below and in more detail in Part 2.

Process scheduling services and storage management services provide ACF/VTAM with a means of requesting resources from the host operating system. Because these components interact with host operating system routines, they are system-dependent and vary from one operating system to another. However, the services that they provide to other ACF/VTAM routines are independent of the host operating system. Since these routines act as a central facility for the services they provide, other components can request system services from this central facility and thereby remain independent of the operating system.

Process Scheduling Services (PSS)

Process scheduling services (PSS) routines control the flow of requests through ACF/VTAM by scheduling and dispatching the routines needed to process requests from application programs. Because these routines serve as a dispatcher, ACF/VTAM does not need to use the operating system dispatcher to schedule its own work.

When an application program issues a request for a session or data transfer, the application program interface ensures that the request is valid and then issues a TPIO SVC. The TPIO SVC indicates to PSS that there is an application program request to be processed. In response to the TPIO SVC, system SVC handlers provide the change from problem-program state to supervisor state needed to process the request. PSS then schedules the request and dispatches the routines needed to process the request.

To schedule the application program's request, PSS routines place the request parameter list (RPL) that represents the request on the appropriate queue. Then, when the request is to be satisfied, ACF/VTAM routines find the RPL and do the processing needed to satisfy the request that it represents.

To dispatch the routines needed to process the request, PSS routines first select the real DVTs and any device-dependent routines required for the request. The routines named in the real DVT are then dispatched to process the application program's request. PSS routines transfer control from one DVT to another and from one routine pointed to by a DVT to another routine pointed to by the same DVT. See "Scheduling an ACF/VTAM Process" in Chapter 2.

Storage Management Services (SMS)

ACF/VTAM's storage management routines operate with host operating system storage management routines to provide ACF/VTAM components with storage areas for control blocks and I/O buffers. Storage management services are not directly available to application programs, but only to ACF/VTAM components, which, in turn, perform the following services for the application programs:

- Building pools for buffers (either pageable or nonpageable), and variable-length storage areas
- Expanding and contracting buffer pools
- Allocating and releasing buffers

ACF/VTAM components issue the following macro instructions to request storage management services:

- BLDPOOL macro instruction, which is used during initialization to build a buffer pool and make an entry in the pool directory
- DELPOOL macro instruction, which is used during termination to delete a buffer pool from the system and remove its entry from the pool directory
- GETSTOR macro instruction, which requests variable-length storage from the appropriate storage area
- FREESTOR macro instruction, which releases storage obtained by GETSTOR and makes it available for reallocation
- REQSTORE macro instruction, which makes requests for fixed-length buffers to be used to hold data or control block information
- RELSTORE macro instruction, which releases a buffer, or a chain of buffers, previously acquired through the REQSTORE macro instruction
- VTALLOC macro instruction, which requests variable-length storage from the appropriate storage area

- VTFREE macro instruction, which releases storage obtained by VTALLOC and makes it available for reallocation

Reliability, Availability, and Serviceability (RAS)

ACF/VTAM can trace the I/O activity of any addressable node and can also trace the contents of buffers as they enter and leave ACF/VTAM control. The ACF/VTAM internal trace keeps a record of ACF/VTAM resource usage and flow of control.

Reconfiguring the Network Dynamically

Dynamic reconfiguration (DR) allows ACF/VTAM to add and delete cluster control units and their associated logical units to or from nonswitched links without the need to do a new NCP generation. This allows the physical relocation of PUs and LUs temporarily until the clusters have been returned to their original configuration or, in the case of a permanent move, a new NCP can be generated.

All PUs are added to or deleted from the physical unit dynamic reconfiguration pool (PUDRPOOL) and all LUs are added to or deleted from the logical unit dynamic reconfiguration pool (LUDRPOOL). The DR function is provided through the VARY command by requesting the addition or deletion of control records and resource definition statements in the dynamic reconfiguration data set (DRDS).

Shutting Down the Network

To shut down the network, the network operator issues a HALT command, which invokes ACF/VTAM's termination routines. These routines perform operations that are the reverse of those performed by the initialization routines. The termination routines delete the ACF/VTAM modules and free all system resources acquired for ACF/VTAM. Control is then returned to the operating system.

CHAPTER 2. ACF/VTAM CONTROL BLOCKS

A general understanding of the most-used control blocks is needed to understand the overall operation of ACF/VTAM. These control blocks can be grouped into the following categories:

- Configuration control blocks, which define the components and structure of the domain
- ACF/VTAM-application program interface control blocks, which permit application programs to use ACF/VTAM facilities
- Session control blocks, which describe sessions between logical units
- Process scheduling control blocks, which are used in scheduling and processing I/O requests

The control blocks in these groups are described in the following sections. For more details about individual control blocks, refer to *ACF/VTAM Data Areas*.

CONFIGURATION CONTROL BLOCKS

Control blocks that define the ACF/VTAM configuration are:

- ACF/VTAM communication vector table (ATCVT)
- Boundary function table (BFT)
- Intelligent controller node control block (ICNCB)
- Local device node control block (LDNCB)
- Resource definition table (RDT)
- Symbol resolution table (SRT)
- Major node table (MNT)
- Specific (minor) node table (SNT)

The relationship of these control blocks is shown in Figure 2-1.

ACF/VTAM COMMUNICATION VECTOR TABLE (ATCVT)

The ATCVT contains the addresses of routines that are not contained in destination vector tables (DVTs). The ATCVT also contains the addresses of other control blocks (such as the resource definition table and symbol resolution table) that are needed for ACF/VTAM execution.

BOUNDARY FUNCTION TABLE (BFT)

The BFT is used for path control and transmission control functions for logical units and physical units connected to ACF/VTAM through a locally attached SNA cluster controller. There is one BFT for each cluster controller, and one BFT entry for the physical unit and for each logical unit attached to that physical unit.

The BFT entry for a logical unit contains information that enables ACF/VTAM to maintain the status of each session flowing through the boundary function and to maintain information needed to support pacing.

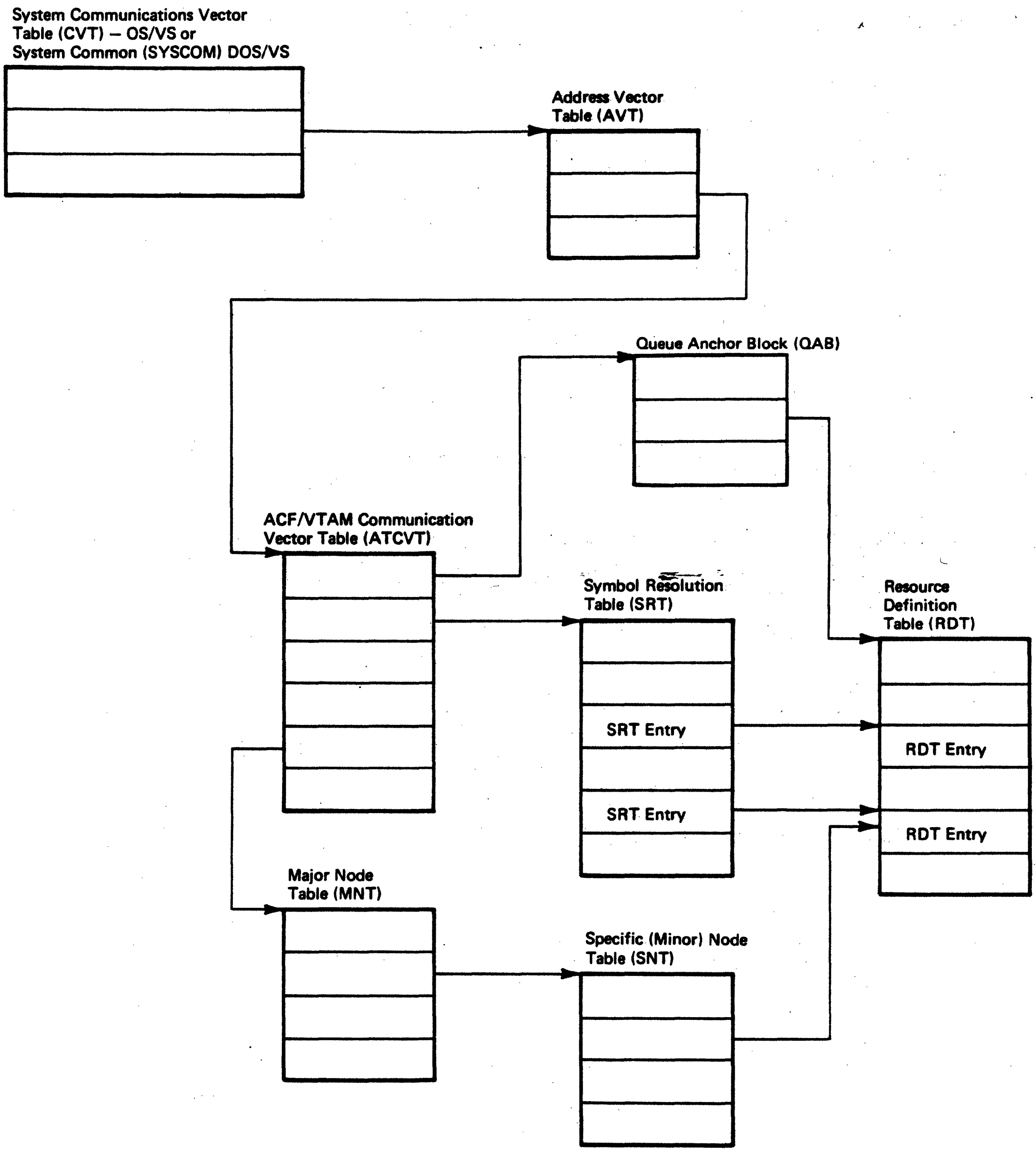


Figure 2-1. Tables That Define the Configuration of an ACF/VTAM Network

INTELLIGENT CONTROLLER NODE CONTROL BLOCK (ICNCB)

The ICNCB is ACF/VTAM's representation of a channel-attached physical unit. The ICNCB is created when the physical unit is activated under ACF/VTAM control (that is, when the operator issues a VARY ACT command). The ICNCB defines the characteristics and current status of the physical unit. This information is needed by ACF/VTAM to control I/O initiation and termination. It also defines scheduling queues and parameters required for communication with the I/O supervisor routines that actually perform the I/O operation.

LOCAL DEVICE NODE CONTROL BLOCK (LDNCB)

The LDNCB is ACF/VTAM's representation of a local non-SNA terminal. The LDNCB defines the characteristics and current status of the terminal. This information is needed by ACF/VTAM to control I/O initiation and termination for the terminal. It also defines scheduling queues and parameters required for communication with the I/O supervisor routines that actually perform the I/O operation. The LDNCB is created when the operator issues a VARY ACT command for the terminal.

RESOURCE DEFINITION TABLE (RDT)

The RDT describes the resources available to the system. It contains entries for communications controllers, groups, lines, cluster control units, terminals, terminal components, logical units, physical units, cross-domain resources, cross-domain resource managers, and application programs. The resource definition table is a segmented table: each segment contains information about a major node defined during system definition. A major node consists of a communications controller and its attached terminals, a group of local non-SNA terminals, a group of physical and logical units attached through the switched network, a group of physical and logical units attached through a data channel, a group of cross-domain resources, a group of cross-domain resource managers, or a group of application programs. Reference is made to these major node subcomponents during ACF/VTAM startup and during OPEN/CLOSE, VARY, MODIFY, and DISPLAY processing. Figure 2-2 illustrates the format of the RDT and the node entries for a communications controller's NCP. The RDT consists of a chain of RDT segments that represent communications controllers, lines, cluster control units, and devices. Each 3705 communications controller is defined as a separate segment. (See Figure 2-3.)

A description of the RDT segment header and the various parts of the RDT segment follows:

<u>RDT Segment Header</u>: Each RDT segment is preceded by a header that identifies whether this RDT segment applies to an application program major node, an NCP major node, a switched SNA major node, a local SNA major node, a local non-SNA major node, a CDRM major node, or a CDRSC major node. The header is also used to chain the RDT segments together.

<u>APPL Entry</u>: The APPL entry appears in the application program RDT segment. It defines a particular application program.

<u>CDRM Entry</u>: The CDRM entry appears in the CDRM RDT segment. It defines either the cross-domain resource manager for this domain or the cross-domain resource manager for another domain.

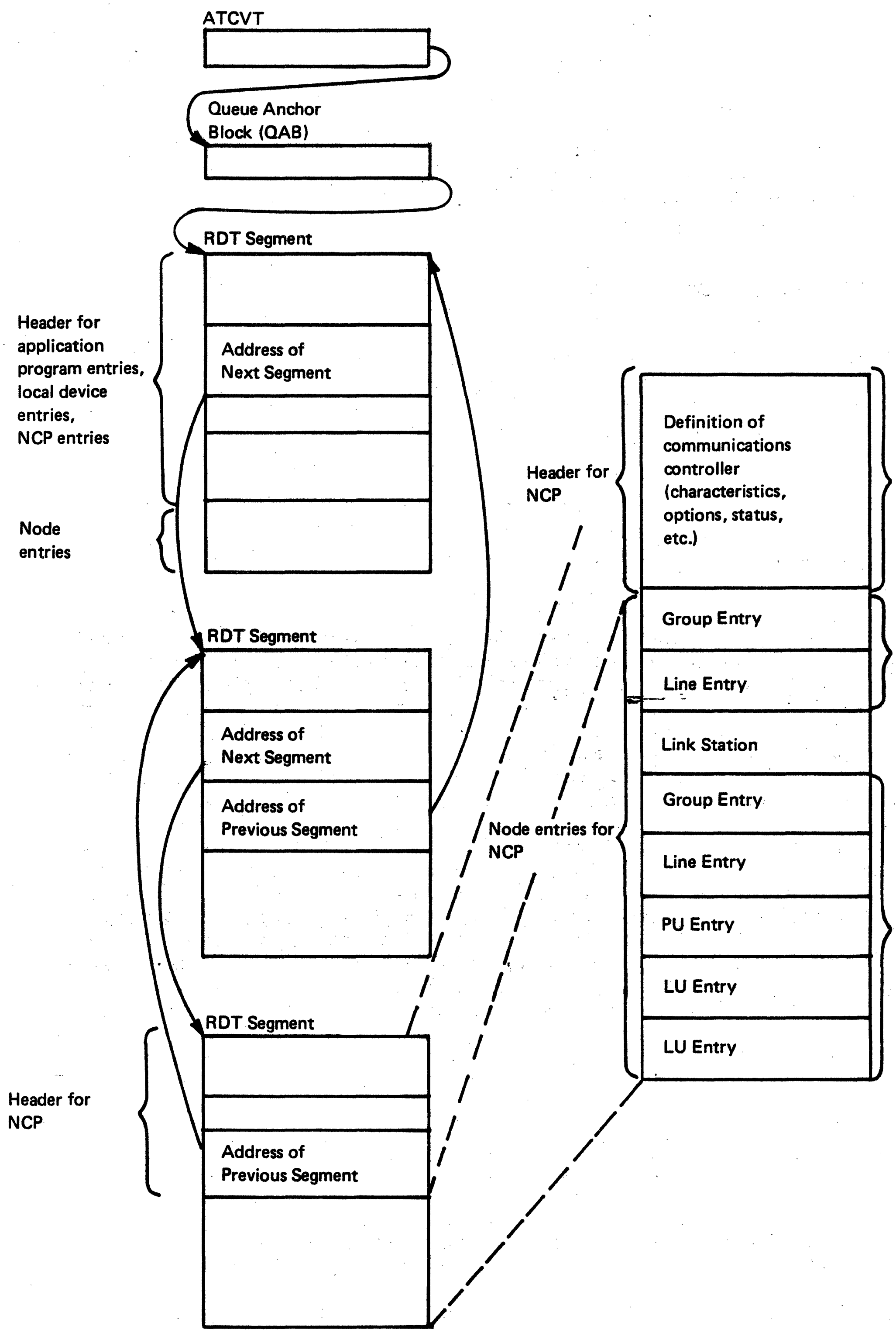


Figure 2-2. Resource Definition Table (RDT), Showing Node Entries for a Communications Controller NCP

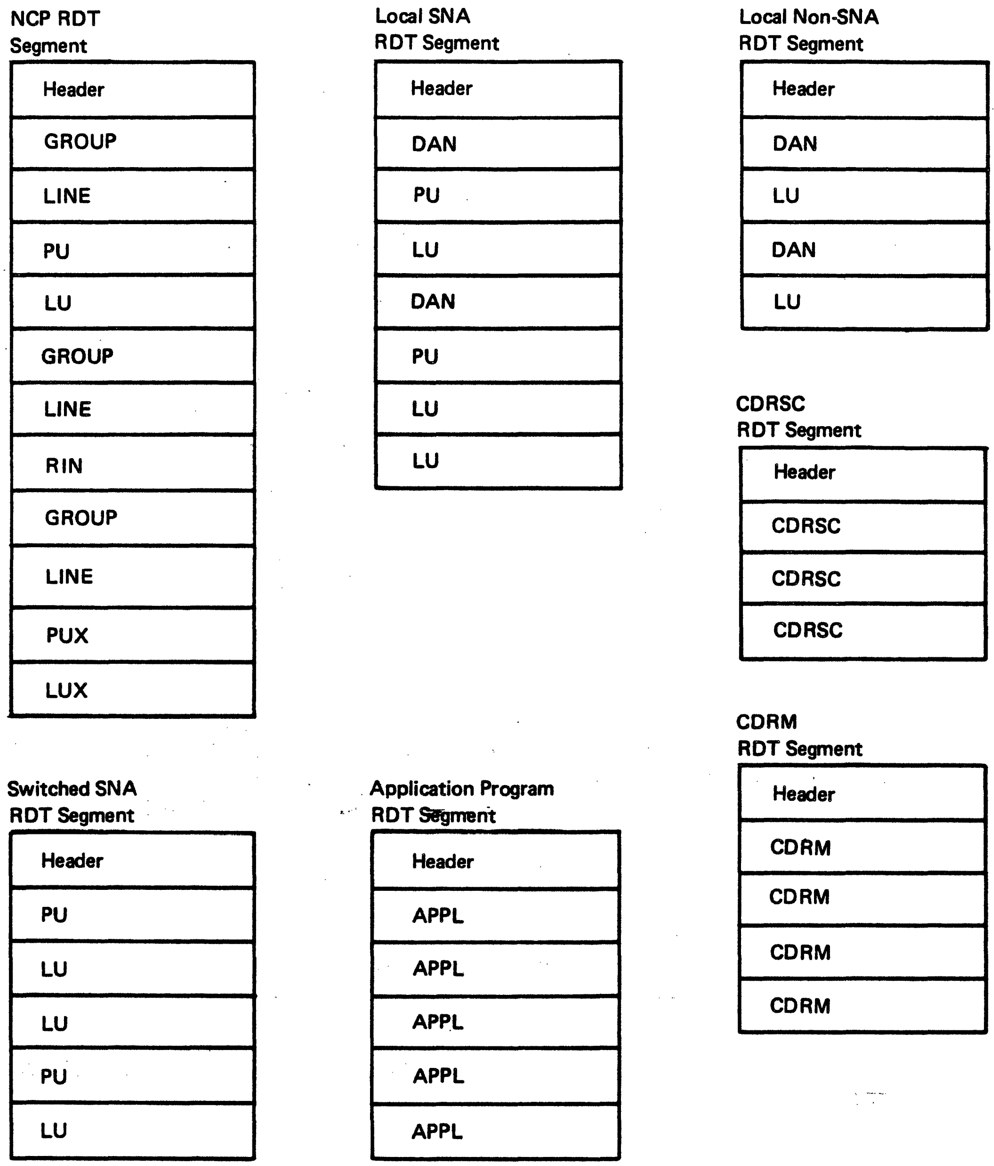


Figure 2-3. Structure of RDT Segments

CDRSC Entry: The CDRSC entry appears in the CDRSC RDT segment. It defines a particular cross-domain resource. A cross-domain resource is a logical unit in another domain with which logical units in this domain can communicate.

Channel Attachment Entry (DAN): The DAN entry appears in the local SNA and local non-SNA RDT segments. It defines a channel attachment to the host processor. There is one DAN entry for every channel-attached device defined in the major node. In a local SNA RDT segment, each DAN entry is followed by a PU entry. In a local non-SNA RDT segment, each DAN entry is followed by a single LU entry.

GROUP Entry: The GROUP entry appears in the NCP RDT segment. It defines a group of switched or nonswitched lines. Switched and nonswitched lines cannot appear together in the same GROUP. Each GROUP entry is followed by one or more LINE entries.

LINE Entry: The LINE entry appears in the NCP RDT segment. It defines a particular switched or nonswitched line. Each LINE entry is followed by one or more PU, PUX, or RIN entries. PU, PUX, and RIN entries cannot be mixed together on the same LINE.

LU Entry: The LU entry appears in the NCP, switched SNA, local SNA, and local non-SNA RDT segments. It defines a logical unit.

LUX Entry: The LUX entry appears in the NCP RDT segment. It defines a network address for a switched-network logical unit.

PU Entry: The PU entry appears in the NCP, switched SNA, local SNA, and RDT segments. It defines a physical unit. Each PU entry is followed by one or more LU entries.

PUX Entry: The PUX entry appears in the NCP RDT segment. It defines a network address for a switched-network physical unit. When a switched-network connection is made between the physical unit and ACF/VTAM, the PUX is associated with a PU entry in the switched SNA major node.

RIN Entry: The RIN entry appears in the NCP RDT segment. It defines a cross-subarea link station.

SYMBOL RESOLUTION TABLE (SRT)

The symbol resolution table (SRT) is used to convert the symbolic name of a node or table into the address of the node or table. The SRT is used most often for opening and closing the I/O interface between an application program node and a destination node, and for displaying and altering the status of nodes.

The SRT, illustrated in Figure 2-4, is a multilevel table created during the activation process (by VARY ACT) and during ACF/VTAM initialization. The first level is the SRT directory (SRTD), which contains the addresses of the beginning of the SRT entry queues. There are 1023 such queues. Whenever ACF/VTAM creates an SRT entry, it applies an algorithm to the name field of the SRT entry to randomly allocate the SRT entry to one of the SRT queues. When ACF/VTAM has to find the SRT entry associated with a given name, it uses the same algorithm to find the appropriate SRT queue to search.

The SRT contains an entry for each symbolic node name that is known to ACF/VTAM. Entries are included for all nodes defined in an active NCP major node, all application programs, all cross-domain resources (CDRSCs), all cross-domain resource managers (CDRMs), and all local devices that are defined to ACF/VTAM. Entries are also included for all USS definition tables, all logon mode tables, and all interpret tables. SRT entries exist for all nodes, not just for addressable nodes. Nonaddressable nodes include lines, line groups, and the logical units and physical units in switched SNA major nodes.

There are four types of SRT entries: those that are part of an RDT entry, those that point to an RDT entry, those that point to a table, and those that point to the RDT entry of an application program that is to receive a network services request unit. Within the SRT entry is a type field that indicates which type it is. Each SRT entry also has a name field that corresponds to the name of the associated node.

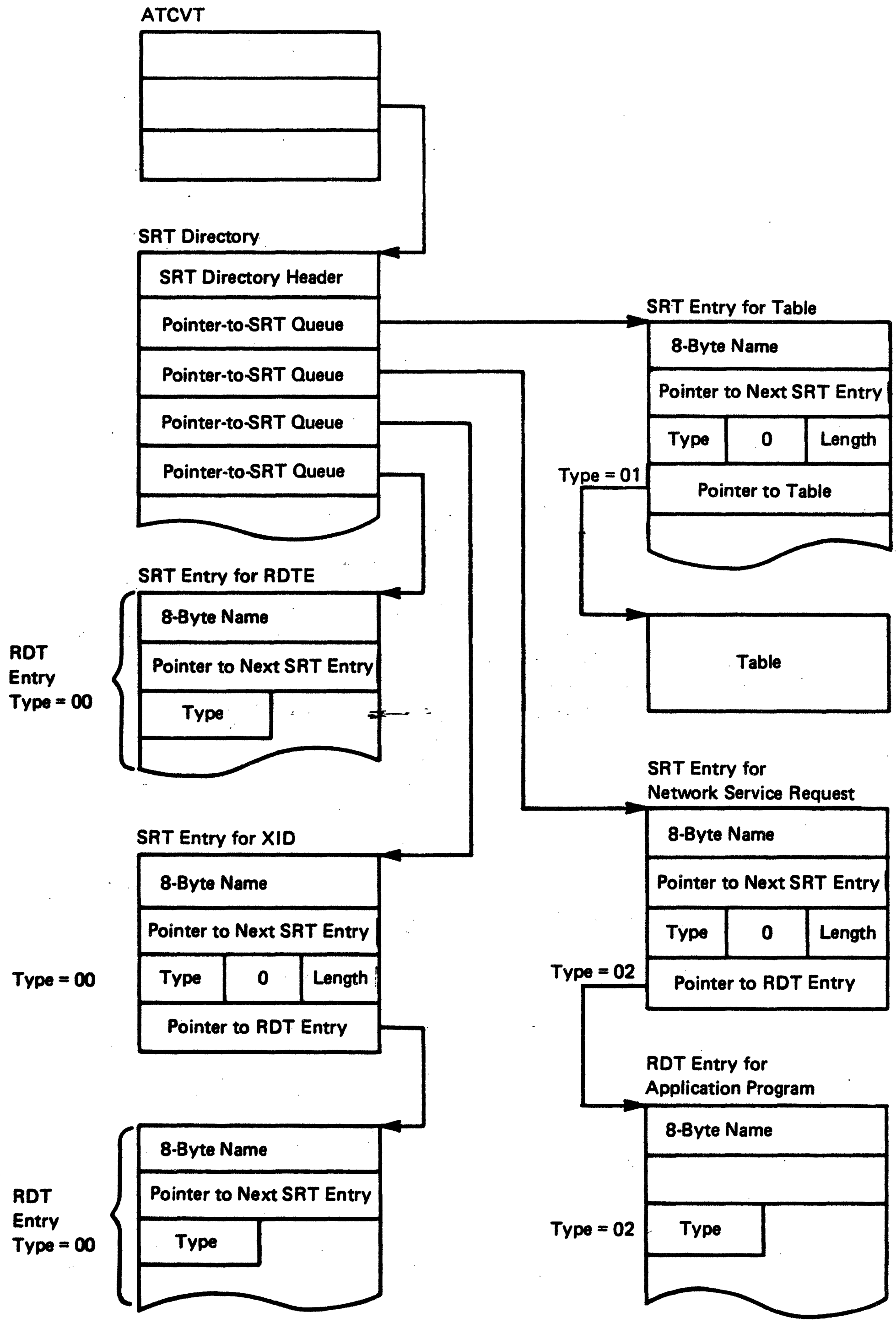


Figure 2-4. Symbol Resolution Table (SRT)

Licensed Material—Property of IBM

Every RDT entry begins with an SRT entry. Therefore, every RDT entry can be found through the SRT directory by use of the appropriate node name. For example, when ACF/VTAM needs to find the RDT entry for a logical unit named LOGUNIT5, it runs this name through the algorithm to find the appropriate SRT queue. ACF/VTAM then searches each SRT entry on that queue until it finds the SRT entry whose name field contains LOGUNIT5. ACF/VTAM also checks the type field to ensure that what it has found is really an RDT entry and not a table by the same name. The RDT entry for that logical unit immediately follows the SRT entry.

Additional SRT entries can exist that point to the RDT entry, but are not part of the RDT. These additional SRT entries are used for the physical units in switched SNA major nodes. When a session is established with such a physical unit, the physical unit provides an exchange ID (XID). ACF/VTAM builds an SRT entry with the XID in the name field and chains this SRT entry to the SRT entry for the physical unit name. Therefore, the RDT entry for a switched physical unit can be found either through the physical unit name or through the XID supplied by the physical unit.

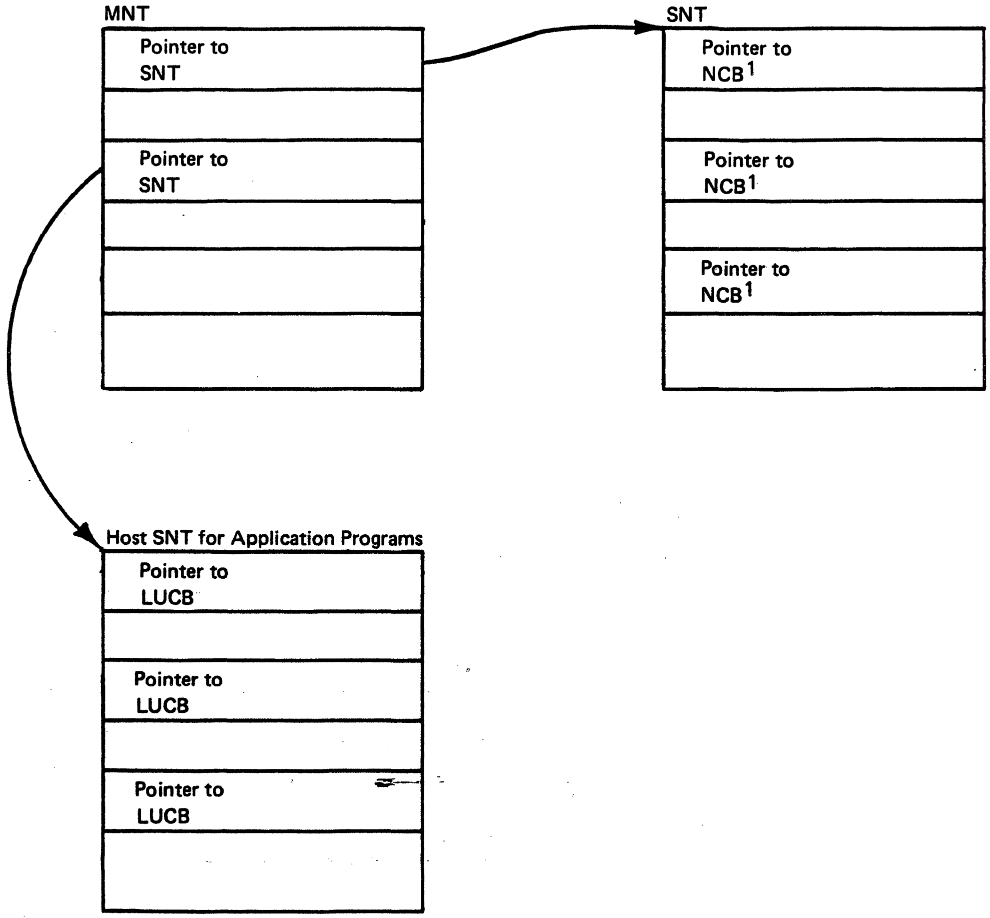
SRT entries can also point to a table. This table can be a USS definition table, a logon mode table, a CSMI routing table, or an interpret table. ACF/VTAM locates the table, using the name of the table and the SRT directory, in the same way as it locates an RDT entry.

SRT entries are also used to route a network services request unit. An application program can be authorized to receive certain network services request units. When ACF/VTAM processes the definition statements that indicate which application programs are to receive each of these request units, ACF/VTAM builds an SRT entry for each request unit that has the request unit identifier in the name field and points this SRT entry to the RDT entry for the application program name. Should ACF/VTAM receive such a network services request unit, it searches the SRT for a matching entry. If such an entry exists, it points to the application program that is to receive this request unit. If the entry does not exist, ACF/VTAM discards the request unit.

MAJOR NODE TABLE (MNT) AND SPECIFIC NODE TABLE (SNT)

The MNT and the base portion of the SNT are built during ACF/VTAM initialization. The MNT contains entries for each subarea in the network (as specified by the MAXSUBA parameter). The SNT contains entries for each specific node within a major node that can be addressed in the domain. However, the SNT does not contain entries for logical units and physical units in switched major nodes, because they are not supported as addressable nodes; nor do they contain entries for specific nodes in other domains, because their addresses are not known in this domain until a session is established with them. Although the addressing tables are built during initialization or when a VARY ACT command is issued for a node, they are updated when an application program issues an OPNDST macro instruction for the particular node. Figure 2-5 shows the node addressing table structure.

The node addressing tables are used to match the destination address portion of a communication address with the address of an internal control block that represents the destination node. The node addressing tables serve as a directory of all destination addresses for which one or more I/O interfaces exist.



[1] Before OPNDST, this field contains the address of an RDTE. Before OPEN for the application program, this field contains the address of an application program RDTE.

Figure 2-5. Node Addressing Table Structure

ACF/VTAM-TO-APPLICATION PROGRAM INTERFACE CONTROL BLOCKS

The following control blocks define the interface between ACF/VTAM and application programs:

- Access method control block (ACB)
- ACF/VTAM data extent block (ACDEB)
- Asynchronous process table (APT)
- Logical unit control block (LUCB)

The relationship of these control blocks is shown in Figure 2-6.

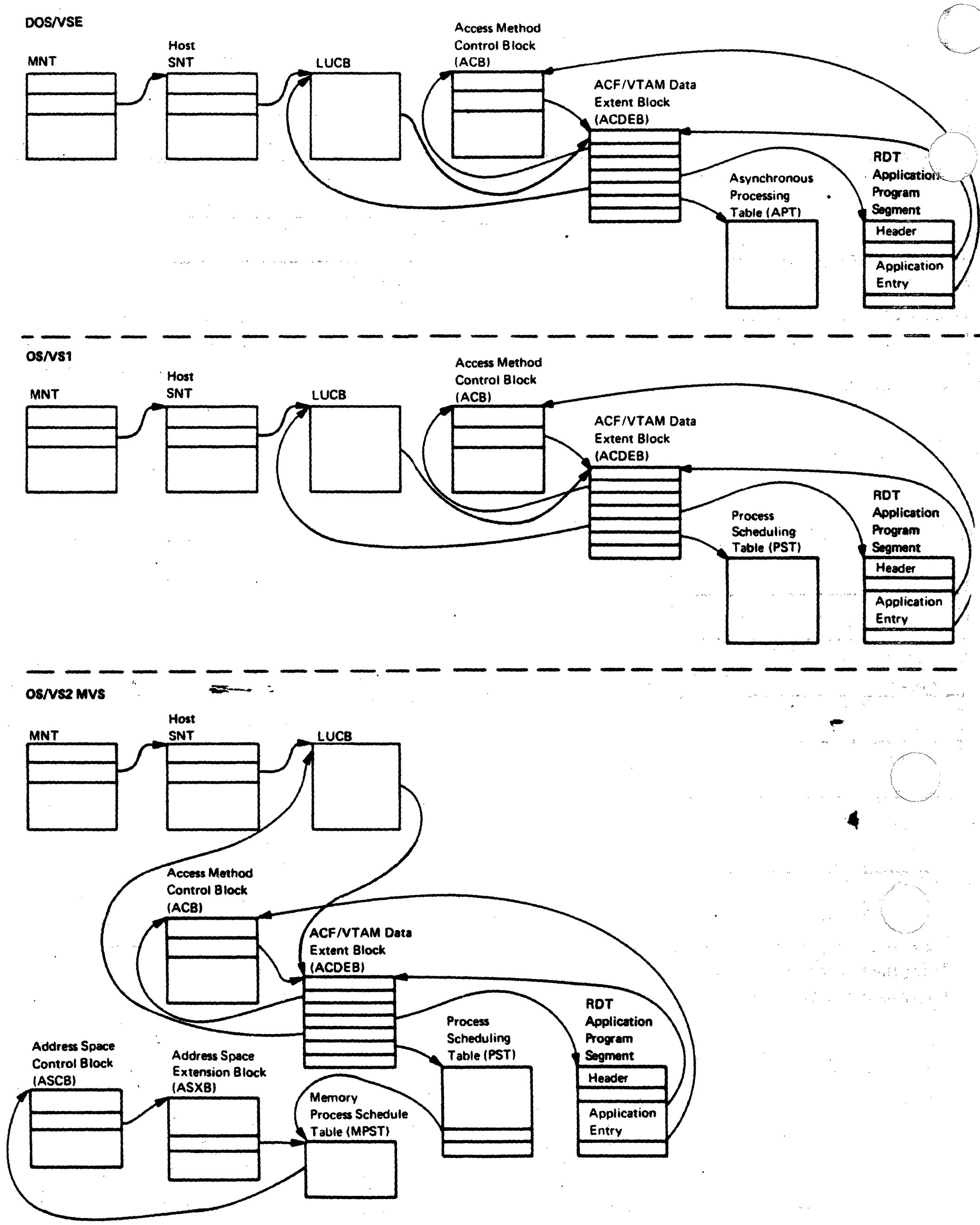


Figure 2-6. Control Blocks That Represent an Active Applicati

ACCESS METHOD CONTROL BLOCK (ACB)

The ACB represents an application program to ACF/VTAM. The ACB is initialized in response to an application program's OPEN ACB request. The ACB is a part of the user's application program; it defines the interface between the problem state application program code (generated by macro instructions in the application program) and the supervisor state ACF/VTAM routines that support the application program. Once this interface has been established, the application program has access to such system facilities as destination open and close and the I/O interface routines.

ACF/VTAM DATA EXTENT BLOCK (ACDEB)

The ACDEB contains information needed by ACF/VTAM to service application program requests. For example, the ACDEB provides the application program with information needed for scheduling the open destination and close destination operations, which are performed in supervisor state. The ACDEB also provides the required information for special forms of I/O requests.

ACF/VTAM control blocks that represent the application program's association with ACF/VTAM routines are chained off the ACDEB, which serves as an anchor point for request-processing operations. The ACDEB also serves as the control point for terminating sessions when an application program is terminated abnormally or issues a CLOSE ACB macro instruction while sessions still exist.

ASYNCHRONOUS PROCESS TABLE (APT)

The APT is ACF/VTAM's representation of an application program's task. This control block serves as the control point for scheduling all asynchronous functions related to the application program. These functions include scheduling I/O-request processing, completion processing, session-request completion processing, and asynchronous user exit routines.

LOGICAL UNIT CONTROL BLOCK (LUCB)

The LUCB is ACF/VTAM's representation of an active application program. There is one LUCB for each active application program. Chained off the application program's LUCB are the FMCBs that represent the other half of the application program's active sessions. The LUCB is created when the application program is activated by OPEN ACB processing and is deleted by CLOSE ACB processing. After the LUCB is created, its address is placed in the specific node table.

The size of the LUCB depends on the value specified for the EAS parameter in the APPL statement that defines the application program. The EAS parameter specifies the greatest number of active sessions that the application program is expected to have at any one time.

The content of the LUCB also depends on the EAS value specified. If the EAS value is small, the application program is represented by an LUCB that contains an entry for each active session. These entries consist of the 2-byte address of the logical unit and the 3-byte address of the associated FMCB. In this case, the FMCB is found by searching the entries in the LUCB for one whose network address matches the one for the specified node. If the EAS value is not specified or is large, the LUCB contains pointers to rows of FMCB look-up table entries. A special algorithm is used to add, delete, and find entries in the FMCB look-up table.

SESSION CONTROL BLOCKS

Several control blocks support four fundamental types of session:

- SSCP-SSCP sessions
- SSCP-PU sessions
- SSCP-LU sessions
- LU-LU sessions

For the most part, the same types of control blocks are used for all four types of sessions. These control blocks are:

- Session information block (SIB)
- Function management control block (FMCB)
- Node identification block (NIB)

The relationship of these control blocks is shown in Figures 2-7 and 2-8.

SESSION INFORMATION BLOCK (SIB)

The SIB indicates the status of an LU-LU session. SIBs are used by the session services subcomponent of the SSCP to keep track of which sessions exist and how far session establishment or termination has proceeded for a particular session. One SIB is created for every session request received by ACF/VTAM.

Each RDTE has two SIB queues: one for SIBs that represent sessions in which the logical unit is the primary end and another for SIBs that represent sessions in which the logical unit is the secondary end. Therefore, each SIB is queued off two RDTEs, one of which represents the primary end and the other the secondary end. The SIB contains pointers to the RDTEs for both logical units participating in the session.

To find all the sessions in which a logical unit is participating, one examines both SIB queues that are associated with the logical unit's RDTE. This reveals not only the active sessions (those for which OPNDST or OPNSEC has completed), but also the pending active sessions (those for which the logon exit has been scheduled, but for which the session has not been established) and the queued sessions (those for which the logon exit has not been scheduled). The SIB indicates whether the session is active, pending active, or queued.

FUNCTION MANAGEMENT CONTROL BLOCK (FMCB)

The FMCB is ACF/VTAM's representation of a half-session. The FMCB is associated with the application program by an LUCB, which contains pointers to FMCBs representing sessions with the application program. The FMCB contains the address of the ACDEB.

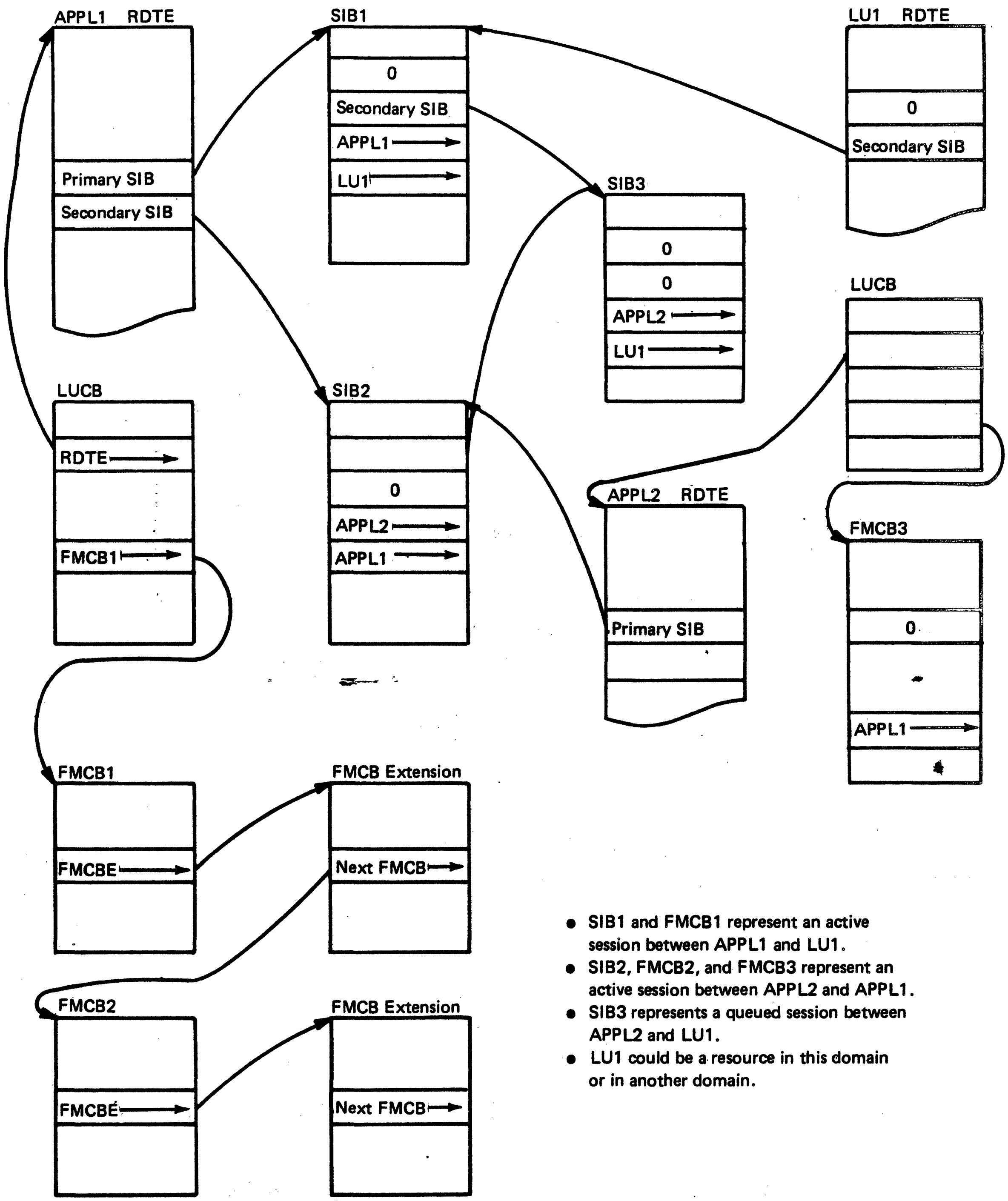


Figure 2-7. Control Blocks That Represent a Session

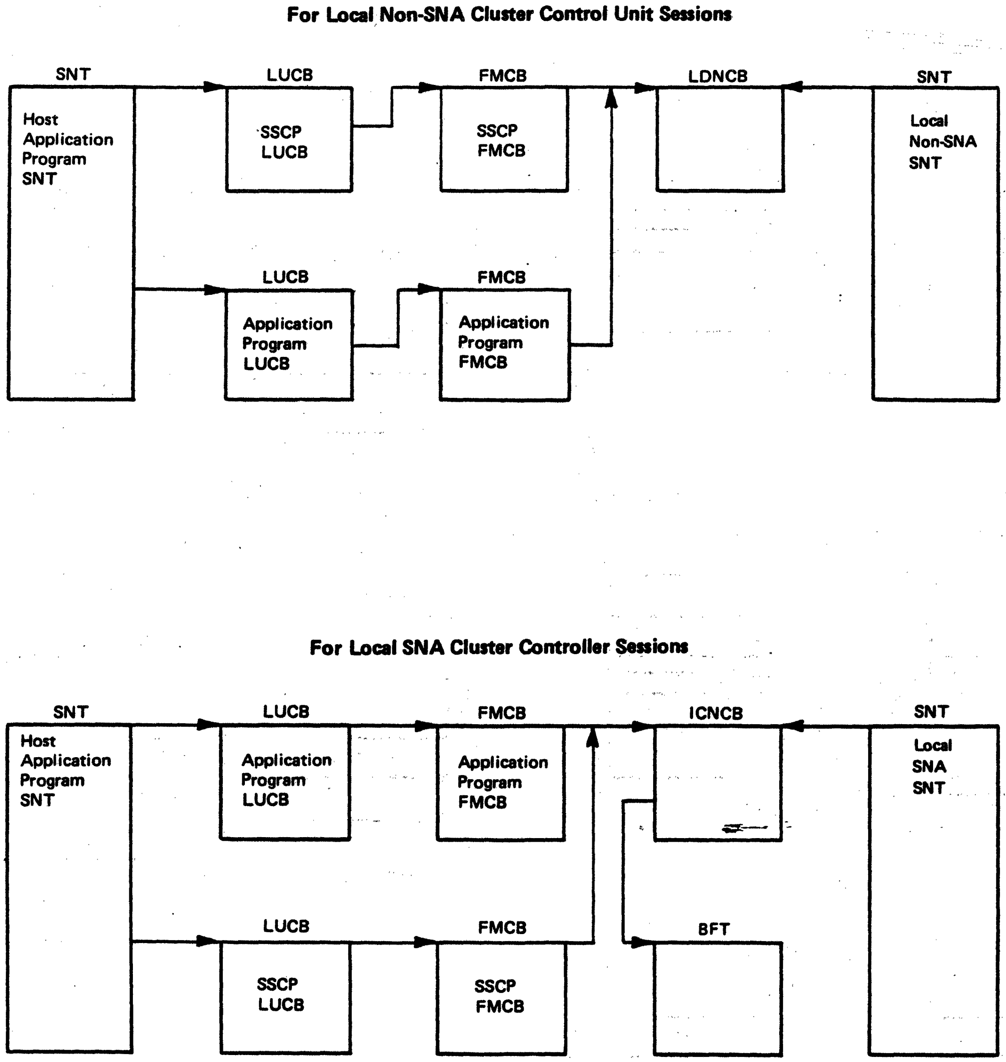


Figure 2-8. Control Blocks That Represent Sessions for Local Devices

The major contents of the FMCB are the queue anchors to which requests and responses (including data) are queued, the destination vector table (DVT) addresses that define the processing routines that were selected at session establishment, and the status of the half-session that this FMCB represents.

The major users of the FMCB are the TSC and process scheduling routines.

NODE IDENTIFICATION BLOCK (NIB)

The NIB describes the characteristics of a session request and identifies the logical unit for which the session is to be established or terminated.

PROGRAM OPERATOR CONTROL BLOCKS

The following control blocks are used for the interface between ACF/VTAM and a program operator application program:

- Program operator control block (POCB)
- Program operator message header (POHD)
- Program operator interface area (POIA)
- Program operator message control block (POMCB)
- Program operator parameter work area (POPWA)
- Program operator reply control block (PORCB)
- Program operator work element (POWE)

The relationship of program operator control blocks is shown in Figure 2-9.

Program Operator Control Block (POCB): The POCB is the primary program operator control block. It represents an application program in session with the program operator interface (POI). Each POCB serves as an anchor point for all control blocks dealing with messages to be received (POMCBs) and replies to be sent (PORCBs) by that application program.

Program Operator Message Header (POHD): The POHD is the header associated with each command sent by ACF/VTAM to the program operator and each message received by ACF/VTAM from the program operator. The POHD is used for message association in the program operator application program.

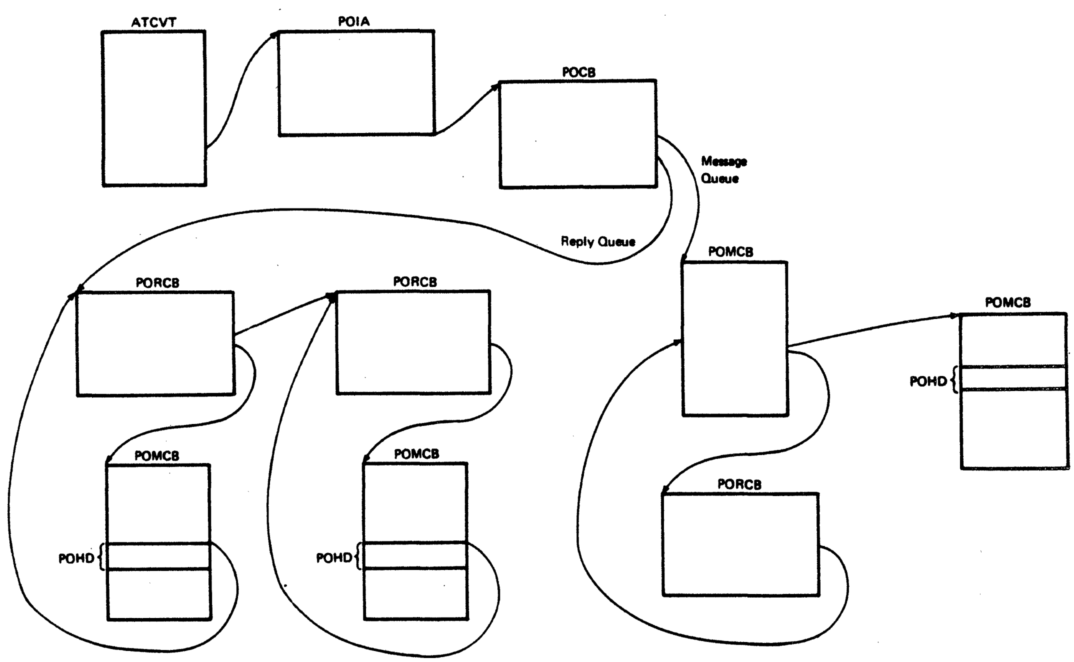


Figure 2-9. Relationship of Program Operator Control Blocks

Program Operator Interface Area (POIA): The POIA acts as an anchor block for the program operator control block (POCB) chain and the program operator interface (POI).

Program Operator Message Control Block (POMCB): The POMCB contains a message generated by ACF/VTAM to be passed to the program operator application program.

Program Operator Parameter Work Area (POPWA): The POPWA is a data area used by the program operator interface (POI) to pick out command or reply text and do initial validation.

Program Operator Reply Control Block (PORCB): The PORCB represents ACF/VTAM waiting for a reply to a WTOR macro instruction.

Program Operator Work Element (POWE): The POWE represents a WTO or WTOR macro instruction issued by ACF/VTAM to a program operator. A POMCB and, if necessary, a PORCB are built from the POWE.

PROCESS SCHEDULING CONTROL BLOCKS

The following control blocks are used in scheduling and dispatching ACF/VTAM processes:

- Process anchor block (PAB)
- Skeleton destination vector table (skeleton DVT or SDVT)
- Destination vector table (DVT)
- Request parameter header (RPH)
- Lock queue anchor block (LQAB)
- Network address block (NAB)
- Waiting request element (WRE)

Control blocks that contain PABs are the:

- Asynchronous processing table (APT)
- ACF/VTAM data extent block (ACDEB)
- ACF/VTAM communication vector table (ATCVT)
- Dynamic process anchor block (DYPAB)
- Function management control block (FMCB)
- Logical unit control block (LUCB)
- Node control block (NCB)
- User exit control block (UECB)

Work elements queued to PABs are the:

- Dump/Load/Restart Parameter List (DLRPL)
- Network configuration services parameter list (NCSPL)
- Pool control block (PCB)

- Program operator work element (POWE)
- Request parameter list (RPL)
- Request/response unit processing element (RUPE)
- TOLTEP interface element (TIE)
- Trace record (TRAC)
- Trace parameter list (TRCPL)
- Transmission subsystem control block (TSCB)
- User exit control block (UECB)

Process Anchor Block (PAB): Process anchor blocks are dispatching points for process scheduling services (PSS); for example, a PAB may be scheduled to cause execution of the modules addressed by the DVT associated with the PAB. A PAB may be explicitly scheduled with no work elements, or a work element may be queued to a PAB to cause it to be scheduled. PABs are not stand-alone control blocks; they are always contained within another control block.

Skeleton Destination Vector Table (Skeleton DVT): The skeleton DVT contains a list of all routines needed to support a session. Routines named in a skeleton DVT are marked either as unconditional (always needed) or as conditional (for example, routines used only for a particular type of session). A real DVT is constructed from a skeleton DVT by selecting those routines needed for each session.

Destination Vector Table (DVT): The DVT is not a control block in the usual sense. Rather, it is a parameter list for the ACF/VTAM execution sequence controller. A DVT lists, in the order they are to be executed, the addresses of the routines that must be executed to pass a request to the specified destination. The execution sequence controller uses the DVT to determine the routine that should next be given control whenever execution of a preceding routine is completed.

Request Parameter Header (RPH): The RPH is an internal parameter list and work area created by process scheduling routines. It contains the address of a queued work request element that is ready to be scheduled and other parameters necessary to define the processing environment for the routines that service the request.

Lock Queue Anchor Block (LQAB): The LQAB is the anchor for a queue of waiting request elements (WREs) and network address blocks (NABs).

Network Address Block (NAB): The network address block (NAB) is used to find the RDTE for a resource, given the network address of the resource. Each NAB contains a network address and a pointer to the RDTE for the resource associated with that network address. A queue of NABs is searched until the NAB containing the required network address is found; then the location of RDTE is obtained from the NAB. With parallel sessions, there can be multiple network addresses for a network resource and, therefore, multiple NABs for that resource's RDTE.

Waiting Request Element (WRE): A WRE represents an ACF/VTAM process that is waiting for the completion of an event after having issued a CPWAIT macro instruction. WREs are queued to the LQAB specified in the CPWAIT macro instruction. When a CPPOST is issued for the event, the WRE is dequeued, and the waiting process is resumed, if appropriate.

Major Control Blocks

Major control blocks in ACF/VTAM are defined as control blocks that contain PABs. See the heading "PABs Used by Major Control Blocks" for a list of the major control blocks, the PABs they contain, and the work elements associated with the PABs.

Asynchronous Process Table (APT): This control block, discussed under the heading "ACF/VTAM-to-Application Program Interface Control Blocks," is also an important processing control block. It serves as the control point for scheduling all asynchronous functions related to an ACF/VTAM application program.

ACF/VTAM Data Extent Block (ACDEB): The ACDEB, discussed under the heading "ACF/VTAM-to-Application Program Interface Control Blocks," contains the RECEIVE ANY PAB and the system service PAB.

ACF/VTAM Communication Vector Table (ATCVT): The ATCVT, discussed under the heading "Configuration Control Blocks," contains several PABs, as shown under "PABs Used by Major Control Blocks."

Dynamic Process Anchor Block (DYPAB): The DYPAB is a control block used to contain a PAB that would not otherwise reside within a major control block. It contains a header and a PAB used to schedule an ACF/VTAM process.

Function Management Control Block (FMCB): The FMCB, discussed under the heading "Session Control Blocks," contains inbound and outbound TSC PABs.

Logical Unit Control Block (LUCB): The LUCB, discussed under the heading "Session Control Blocks," is also used in process scheduling.

Node Control Block (NCB): The LDNCB and the ICNCB, discussed under the heading "Session Control Blocks," each contain a path control routing PAB, a PU services PAB, and a utility function PAB for TSC.

User Exit Control Block (UECB): The UECB is both a major control block and a work element. It contains the PAB used to schedule the user exit routine, and it is the work element that defines input to the user exit routine.

Work Elements

Dump/Load/Restart Parameter List (DLRPL): The DLRPL contains the parameters used by certain modules of the dump/load/restart thread.

Network Configuration Services Parameter List (NCSPL): The NCSPL represents an ACF/VTAM command (such as VARY) that is being processed. The NCSPL contains the symbolic name of the resource (the RDT entry) and the address of a storage area used both as a save area and working storage. The NCSPL reflects the status of processing for the command and is used to store status information when processing of the command is interrupted.

Pool Control Block (PCB): This control block describes the ACF/VTAM buffer pools and serves as the anchor for a queue of unallocated buffers.

Program Operator Work Element (POWE): The POWE represents a WTO or WTOR macro instruction issued by ACF/VTAM to the program operator. It is the work element for the program operator interface.

Request Parameter List (RPL): The RPL performs the following for an I/O or session request:

- Indicates (with a node initialization block [NIB] pointer) the nodes that are to be put in session with the application program when an OPNDST is issued or (with the communication ID [CID]) the node that is to receive data when output requests are made, and from which data is to be obtained when input requests are made.

- Indicates the location and length of data supplied by the application program for output requests, and indicates the location, maximum possible length, and length of data actually received following an input request.

- Indicates the location of the ECB to be posted by ACF/VTAM and checked by the application program after an I/O request, or, alternatively, indicates an exit routine to receive control when the ECB would otherwise have been posted.

- Indicates which attributes of an I/O request are to be in effect (for example, processing is synchronous rather than asynchronous, or output requests are automatically followed by input requests).

An RPL can be created during assembly with the RPL macro instruction, or it can be created during execution with the GENCB macro instruction. Except for indicating the CID, the functions listed above are controlled by the application program through the operands of the RPL macro instruction.

Request/Response Unit Processing Element (RUPE): The RUPE is the major work element for ACF/VTAM mainly in configuration services, logical unit services, and PSS. It is the work element for request/response unit processing and contains the information needed to process an RU.

TOLTEP Interface Element (TIE): The TIE is the work element for interfacing with TOLTEP. It contains a request code that indicates the function to be performed.

Trace Record (TRAC): The TRAC is the record that the trace writer is to write on the trace file.

Trace Parameter List (TRCPL): The TRCPL contains flags that indicate the type of trace and the node affected.

Transmission Subsystem Control Block (TSCB): The TSCB is the work element for TSC inbound and outbound processing.

User Exit Control Block (UECB): The UECB is both a major control block and a work element for the user exit routine.

Scheduling an ACF/VTAM Process

An ACF/VTAM process is a unit of work done by a group of routines to satisfy, or partially satisfy, a request from an application program, the network operator, the network (for example, a cross-domain or USS logon request), or an ACF/VTAM component. The request is represented by a control block that is designated as the work element. See Figure 2-10 for an illustration of the control blocks used to schedule a process.

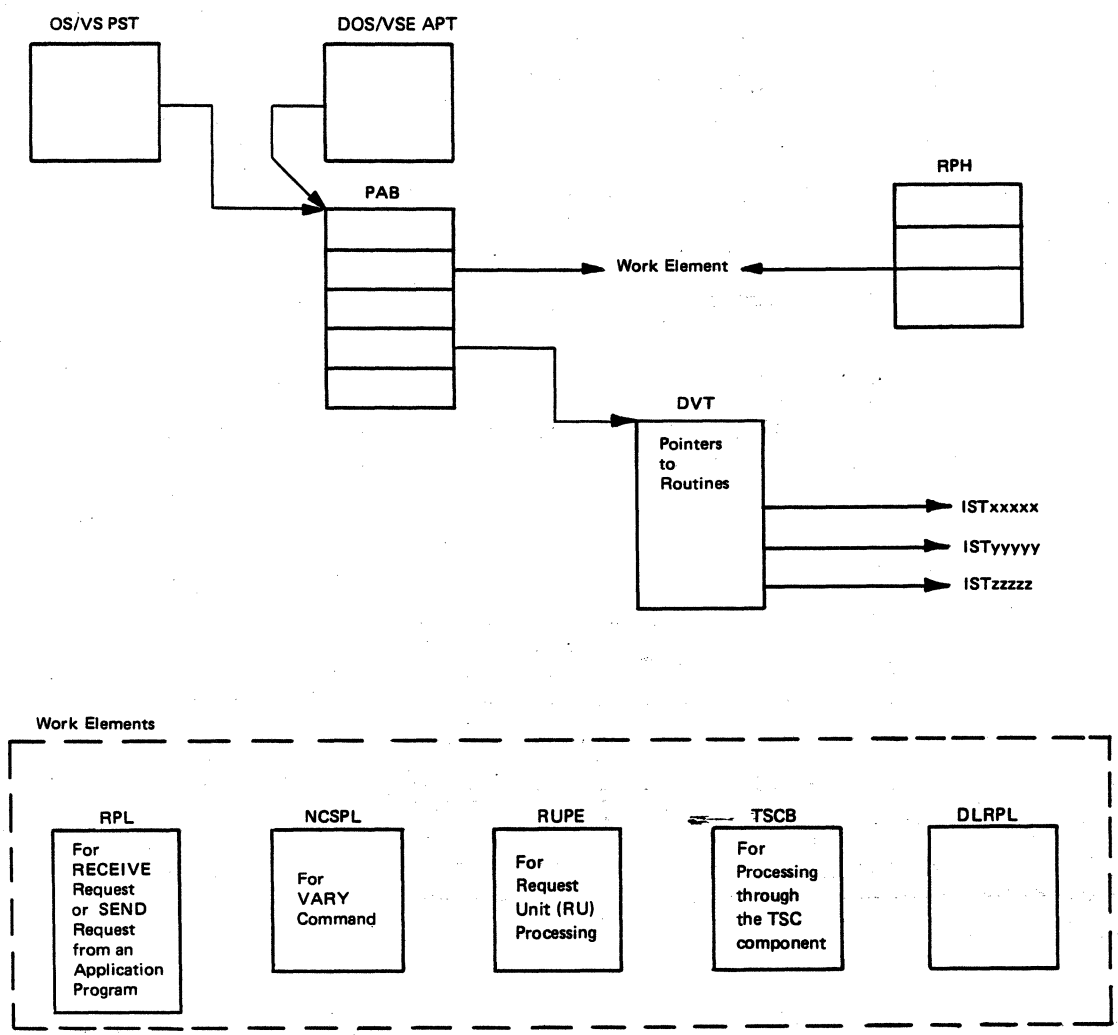


Figure 2-10. Control Blocks Used to Schedule a Process

For example, a request parameter list (RPL) could represent a RECEIVE request from an application program, a network configuration services parameter list (NCSPL) could represent a VARY command from the network operator, or a transmission subsystem component block (TSCB) could represent a SEND request from the SSCP to the TSC.

A work element is normally passed from one component to another by queuing it to a process anchor block (PAB). The process anchor block is a control block that represents a specific set of routines. In the process anchor block there is a pointer to a table that contains the addresses of these routines. This table is called the destination vector table (DVT). Consequently, the first step in preparing a work element for processing by a particular group of routines is to queue it to the PAB for that group.

After the work element is queued to the PAB, the PAB is scheduled to be dispatched. A PAB is scheduled by queuing it to the asynchronous process table (APT) or to a chain of PABs already queued.

There are several chains of PABs. Each chain is for a different kind of processing, and each has a different dispatching priority. When ACF/VTAM has processed a work element, the APT is searched for the first PAB in the highest priority PAB chain. The address of the work element is taken from the PAB and placed in the request parameter header (RPH). The RPH is a control block that contains the work element address and other data pertaining to the processing environment. The address of the RPH is put into register 1, and control is given to the first routine designated in the destination vector table. ACF/VTAM execution then continues until a work element is queued to another PAB for the next process. The queuing, scheduling, and dispatching of a work element are done by the process scheduling services (PSS) component of ACF/VTAM.

PABS USED BY MAJOR CONTROL BLOCKS

Figure 2-11 lists the standard PABs that are used in ACF/VTAM processing. An entry in the list consists of the major control block that contains the PAB, the name of the PAB, and the names of the work elements that can be associated with the PAB.

Control Block	PAB	Work Elements
ACDEB	ACDNEPAB (Receive Any PAB)	RPL
	ACDSSPAB (System Service PAB)	RPL
ATPX	APTXPAB (TPIO SEC PAB)	RPL
	APTXPABE (TPIO TSEXP PAB)	RPL
	APTXPABI (TPIO TSIP PAB)	RPL
	APTXPABN (TPIO TSNORM PAB)	RPL
ATCVT	ATCCSPAB (SSCP Configuration Services PAB)	RUPE
		NSCPL
		DLRPL
	ATCITPAB (Internal Trace DYPAB)	NCSPL
	ATCLUSRT (LUS Router DYPAB)	RUPE
	ATCNOSPB (NOS DYPAB)	RUPE
	ATCNSPAB (TSC No Session DYPAB)	TSCB
	ATCPOPAB (Program Operator DYPAB)	POWE
	ATCPUIOP (PU Services I/O DYPAB)	RUPE
		TSCB
	ATCPUPAB (PU Services DYPAB)	RUPE
	ATCPXPAB (SMS Dynamic Expansion DYPAB)	PCB
		BPCB
	ATCSMPAB (SSCP DYPAB)	RUPE
		NSCPL
		DLRPL
	ATCTCIPB (TOLTEP DYPAB)	TIE
		TSCB
	ATCTMRPB (Termination Task DYPAB)	PST
	ATCVDPAB (VARY Definition DYPAB)	NCSPL
FMCB	TSPTSIP (TSC Inbound PAB)	TSCB
		RPL
	TSPTSOP (TSC Outbound PAB)	TSCB
		RPL
ICNCB	ICNPCPAB (TSC Path Control Routing PAB)	TSCB
	ICNUFPAB (TSC Utility Function PAB)	none
LDNCB	LDNPCPAB (TSC Path Control Routing PAB)	TSCB
		RPL
	LDNUFPAB (TSC Utility Function PAB)	none
LUCB	LUCPAB (TSC Path Control Routing PAB)	TSCB
		LIE
UECB	UECPAB (User-Exit PAB)	RPL
		UECB

Figure 2-11. Major Control Block PABs

PART 2. COMPONENTS OF ACF/VTAM

Each chapter in this part describes the operation of an ACF/VTAM component. A component is a collection of routines that act together to perform a set of related functions. ACF/VTAM Logic, which is more detailed than this manual, is organized by component in the same order as Part 2 of this manual. Module-naming conventions (which follow) show the components, their chapter numbers in this manual, and their method-of-operation (MO) and program-organization (PO) subsection numbers in ACF/VTAM Logic.

Chapter 3. Initialization and Termination

Chapter 4. System Definition (SYSDEF)

Chapter 5. System Services Control Point (SSCP)

Chapter 6. Network Operator Command Facilities

Chapter 7. Network Inquiry

Chapter 8. OPEN/CLOSE

Chapter 9. Application Program Interface (API)

Chapter 10. Transmission Subsystem Component (TSC)

Chapter 11. Process Scheduling Services (PSS)

Chapter 12. Storage Management Services (SMS)

Chapter 13. Reliability, Availability, and Serviceability (RAS)

Chapter 14. Physical Unit Services (PUS)

Chapter 15 Logical Unit Services (LUS)

Chapter 16. Multisystem Networking Facility (MSNF)

Chapter 17. Encrypt/Decrypt Feature

Chapter 18. Teleprocessing Online Test Executive Program (TOLTEP)

MODULE-NAMING CONVENTIONS

The names of the ACF/VTAM modules point to the component to which the module belongs. Module-naming conventions are in the following form:

ISTxxyzz is the full module name, where:

IST identifies the module as an ACF/VTAM module.

xx identifies the component as follows:

AC	SSCP configuration services (activation)
AI	application program interface
AP	process scheduling services
CD	SSCP cross-domain resource manager

CF network operator command facilities or initialization
CP system services control point (SSCP)
DE SSCP configuration services (deactivation)
ES process scheduling services
IN initialization/termination or system services control point
LU logical unit services
MA SSCP maintenance services
MG SSCP management services
NO network operator services
OC OPEN/CLOSE
OR storage management services
PU physical unit services
RA reliability, availability, and serviceability
SD system definition
SQ network inquiry
ST network inquiry
TS transmission subsystem

y identifies system dependencies as follows:

A OS/VS1
C common to DOS/VSE, OS/VS1, and OS/VS2 MVS
E DOS/VSE
F common to OS/VS1 and OS/VS2 MVS
M OS/VS2 MVS

zz identifies the individual module

Figure P2-1 relates the ACF/VTAM components to the fourth and fifth characters of the module names.

CONVENTIONS USED IN METHOD-OF-OPERATION DIAGRAMS

The chapters in this part describe how functions are performed by the components of ACF/VTAM. Each component is illustrated by a method-of-operation (MO) diagram. These diagrams correspond to the overview diagrams in the more detailed ACF/VTAM Logic.

The method-of-operation diagrams are based on the input-process-output sequence. The input to the component (or subcomponent) is shown on the left side, the process steps are listed in the center, and the output is shown on the right side. Arrows relate data in the input and output sections to the appropriate processing steps, show the sequence of processing, and indicate fields that contain important addresses. Figure P2-2 explains the meanings of the arrows used in the diagrams.

Only the last five characters of control block names are shown. Fields within control blocks are indicated by name or by content. Field names usually start with the fourth, fifth, and sixth characters of the control block name. Only relevant fields are shown; if space is left before or after a field it means that the data area contains other fields that are not relevant as input or output for this function. A space does not indicate how many fields were omitted. The order of the fields shown in the input and output boxes does not necessarily correspond to the actual order of fields in the data area.

ACF/VTAM Components	Unique Module Letters (4th & 5th)	Chapter In This Manual	MO/PO Subsection in PLM
Initialization and Termination	IN, CF	3	1
System Definition	SD	4	2
System Service Control Point	IN, CP	5	3
Network Operator Command Facilities	CF	6	4
Network Inquiry	SQ, ST	7	5
OPEN/CLOSE	OC	8	6
Application Program Interface	AI	9	7
Transmission Subsystem	TS	10	8
Process Scheduling Services	AP, ES	11	9
Storage Management Services	OR	12	10
Reliability, Availability, and Serviceability	RA	13	11
Physical Unit Services	PU	14	12
Logical Unit Services	LU	15	13
Multisystem Networking Facility	CD, CP, IN, SD, SQ	16	Multisystem Networking Facility Logic
Encrypt/Decrypt Feature	CD, CP, IN, OC, SD, SQ, TS	17	Encrypt/Decrypt Feature Logic
TOLTEP		18	Teleprocessing Online Test Executive Program

Figure P2-1. ACF/VTAM Components and Identifying Module ID Characters

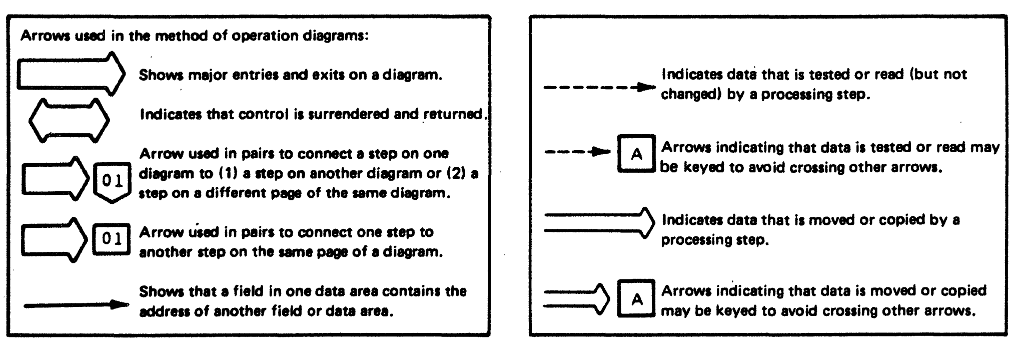


Figure P2-2. Arrows Used in the Method-of-Operation Diagrams

CHAPTER 3. INITIALIZATION AND TERMINATION

Initialization and termination services establish the ACF/VTAM operating environment; that is, they define the domain structure and identify system resources. When ACF/VTAM is to be terminated, these services stop I/O and session activities and close down the domain.

An overview of initialization and termination is shown in MO 1.

INITIALIZATION

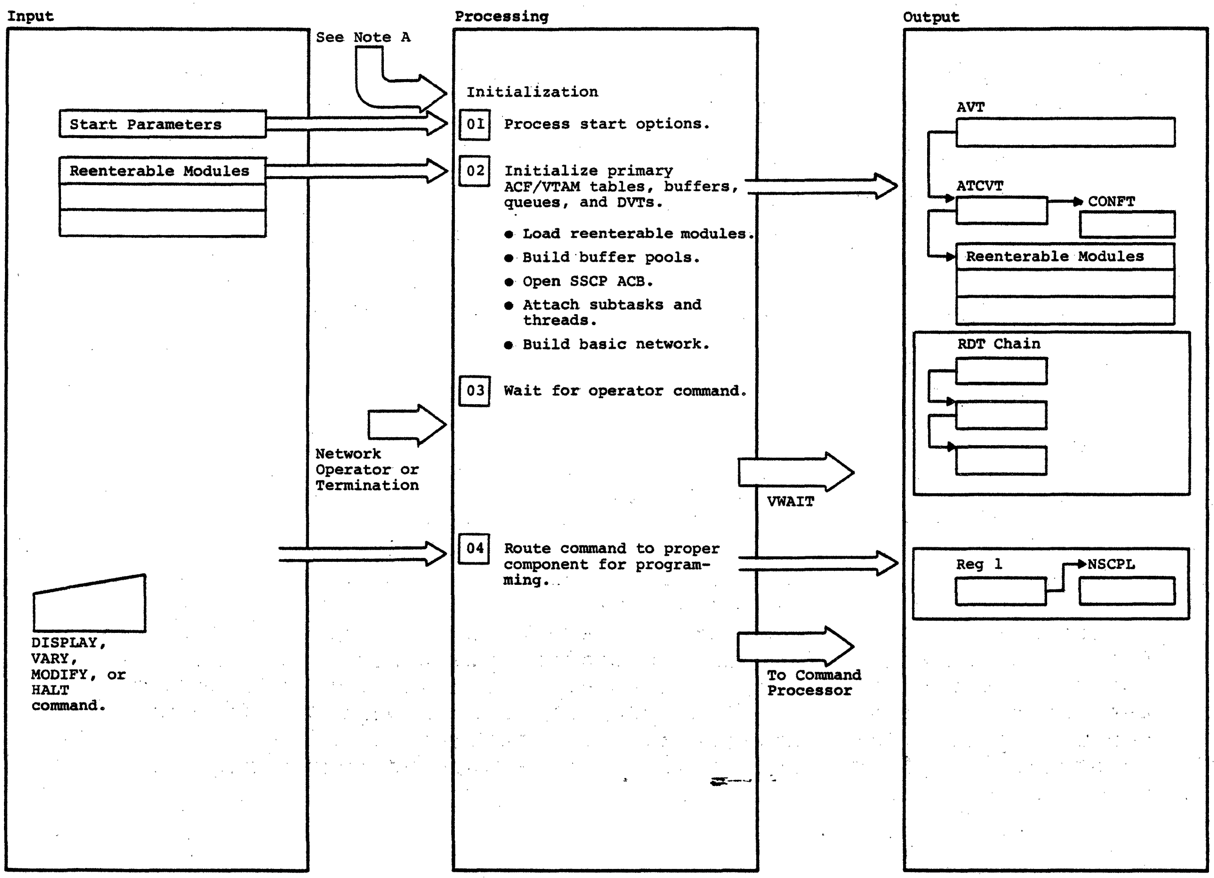
Initialization is a synchronous process that begins when the system operator issues an EXEC PROC= (DOS/VSE) or a START command (OS/VS) that names the ACF/VTAM procedure. The load module or phase that connects ACF/VTAM to the host operating system is also loaded by the host operating system as the result of a START VTAM (OS/VS) or an EXEC VTAM (DOS/VSE) command. The initialization modules then process the start options and network configuration parameters and load the other ACF/VTAM modules. Initialization then acquires buffer pools, builds the ACF/VTAM command processor queues, opens the ACB for the SSCP, attaches ACF/VTAM subtasks and threads, generates internal VARY ACT commands for each major node name specified in the configuration file and routes the commands to the SSCP (which processes them asynchronously), starts user-specified traces, and, in DOS/VSE, initializes operator-to-ACF/VTAM communication. The initialization subcomponent then continues to run after starting is completed, acting as a command router for all ACF/VTAM operator commands.

TERMINATION

Termination is a synchronous process that reverses the actions undertaken by the initialization subcomponent. Termination occurs as a result of a HALT command. The HALT command specifies whether ACF/VTAM is to be terminated in an orderly way (HALT), which allows application programs to end their sessions at their convenience and allows full operator command capability, or in an immediate manner (HALT QUICK), which disallows any communication between logical units and allows only limited operator command capability. In an orderly halt, application programs are allowed to close their ACB. Once all the application programs have closed their ACBs, immediate halt logic is entered automatically. During immediate halt processing, ACF/VTAM subtasks and threads are terminated, application programs with open ACBs are notified of a HALT QUICK and I/O processing for their sessions is stopped immediately, major nodes are deactivated, queues are dismantled, and storage areas are released. Once the application programs have closed their ACBs, operator-to-ACF/VTAM communication is terminated, and all ACF/VTAM phases are deleted.

In OS/VS systems, a HALT CANCEL command forces ACF/VTAM out of the system. This command causes the operating system's command scheduler, SVC 34, to schedule ACF/VTAM for abnormal termination.

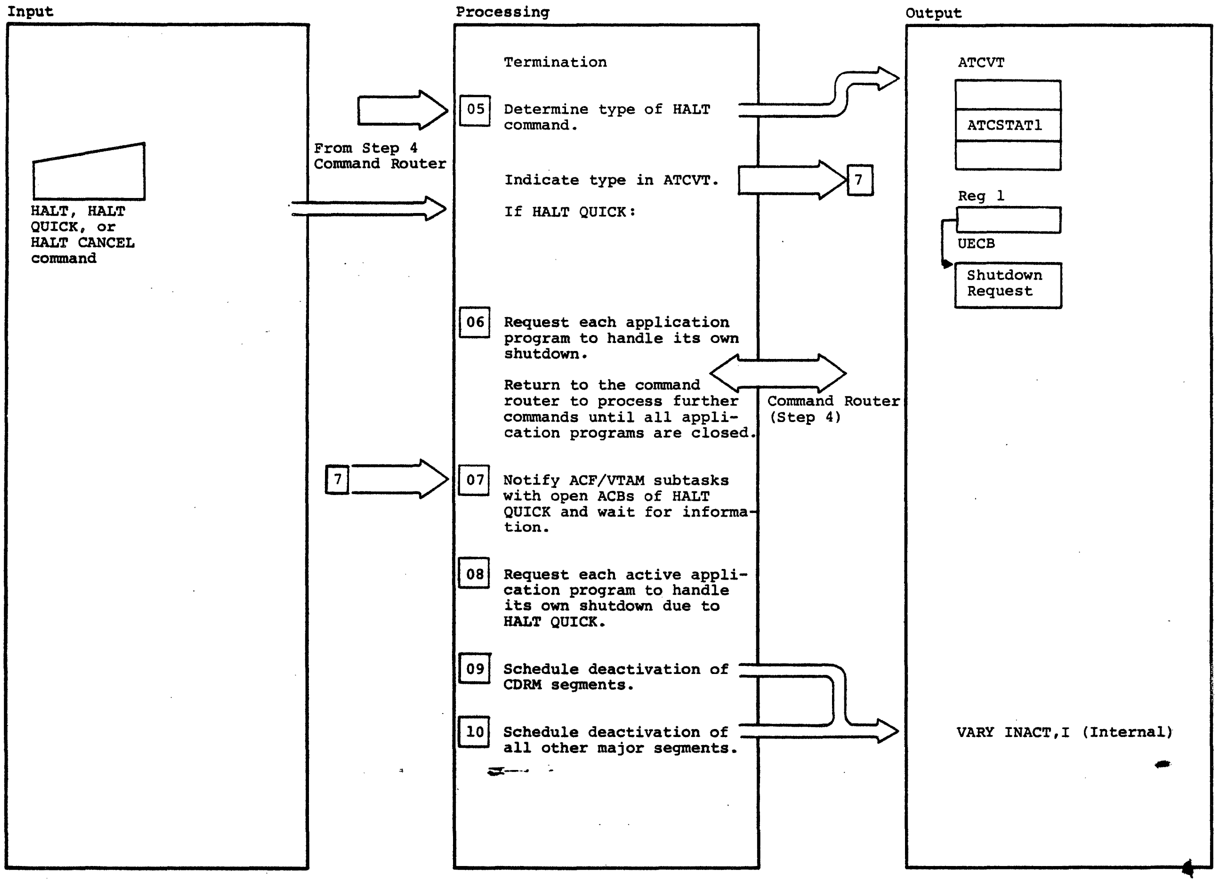
MO 1 (Part 1 of 3). Initialization and Termination



Notes	Routine	Label	Ref.
01 Start options are stored in B.ATCSTRxx in the source statement library, or they can be entered from the console by the network operator.			
04 DISPLAY command is handled by network operator command facilities.			MO 4
HALT or HALT QUICK command is handled by Termination (Step 5).			
VARY command is handled by SSCP.			MO 3
MODIFY command is handled by SSCP.			MO 3
A Entry is from system dispatcher (DOS/VSE), system master scheduler (OS/VS1), or system scheduler (OS/VS MVS).			
In OS/VS systems the ACF/VTAM partition or region size or address space for the system must be specified in the JCL before the START options are processed.			

Notes	Routine	Label	Ref.

MO 1 (Part 2 of 3). Initialization and Termination



Notes	Routine	Label	Ref.
05 The operator can enter either HALT or HALT QUICK to terminate ACF/VTAM. HALT means that application programs connected to ACF/VTAM are not forced to disconnect themselves, but application programs not using ACF/VTAM at the time the HALT is issued are denied access. Full command capability is allowed until either a HALT QUICK is issued or all the application programs close their ACBs. HALT QUICK means that the network is to be shut down immediately. Application programs not using ACF/VTAM are denied access. Existing I/O requests are processed, but new I/O requests and session requests are not allowed. Only a few commands, such as DISPLAY and VARY INACT, F, are allowed. HALT CANCEL (available in OS/VS1 and OS/VS2 MVS only) means immediate abnormal termination of the ACF/VTAM task. ACF/VTAM application programs are notified by TPEND exit or are abnormally terminated to force ACF/VTAM out of the system. Locally attached data communication devices are freed, and ACF/VTAM storage is returned to the system.			
09 Termination waits for the CDRMs to be deactivated by SSCP before continuing.			

Notes	Routine	Label	Ref.
10 Termination waits for these segments to be deactivated before continuing.			

MO 1 (Part 3 of 3). Initialization and Termination

Input

ATCVT — ISTVTLOD

Address of Modules

Processing

11 In OS/VS, notify termination subtask of HALT QUICK and wait for it to terminate.

12 Deactivate application program segments and ACF/VTAM threads (DOS/VSE) without ACBs or subtasks (OS/VS).

13 Notify system operator that ACF/VTAM is terminated.

14 Delete tables, queues, and loaded modules.

Return to System

Output

VTAM Terminated

Notes	Routine	Label	Ref.

Notes	Routine	Label	Ref.

CHAPTER 4. SYSTEM DEFINITION (SYSDEF)

The composition of the network is identified to ACF/VTAM through the ACF/VTAM definition deck. There is a group of statements for each set of:

- Application programs
- Lines and terminals attached to the NCP
- SNA terminal products attached over switched lines
- SNA terminal products attached over the channel
- Non-SNA terminals attached over the channel
- CDRMs and CDRSCs
- Path tables

Except for path tables, each group of statements represents a major node in the domain.

A resource definition table (RDT) segment is built for each major node and is composed of a header followed by a number of entries. Each entry represents a node within that major node. Consequently, each RDT segment is a collection of specific (minor) node entries that are related by being part of one major node. After the generation decks have been processed, ACF/VTAM has a control block representation of the data communications system. There is an RDT segment for each:

- Application program major node (with an entry for each application program defined)
- NCP major node (with an entry for each line and device connected to that NCP)
- Switched SNA major node (with an entry for each PU and LU defined)
- Local SNA major node (with entries for each local PU and LU defined)
- Local non-SNA major node (with entries for each local non-SNA terminal defined)
- CDRM major node (with an entry for each CDRM defined)
- CDRSC major node (with an entry for each CDRSC defined)

No RDT segment is built for a path table.

An overview of system definition operation is shown in MO 2.

CID HANDLING

System definition also handles CIDCTL macro instructions issued by other ACF/VTAM components to obtain information from the addressing control blocks.

The ACF/VTAM system address shown in Figure 4-1 is called a communication ID (CID). A CID has a 2-byte secondary network address and a

2-byte primary network address. The primary and secondary network addresses each contain two values. The first value (the major node ID) is an index into the major node table to find an entry for a specific node table. The second value (the specific node ID) is an index to the specific node table.

When an application program sends an I/O request or message to a specific node, the primary network address is used to find the node control block or LUCB; then the secondary network address is used to find the FMCB that is chained off the NCB or LUCB. The secondary field of the CID contains the destination node's network address, and the primary field contains the application program's network address. Thus, the CID is a system address that uniquely identifies both the destination node and the application program with which it is in session.

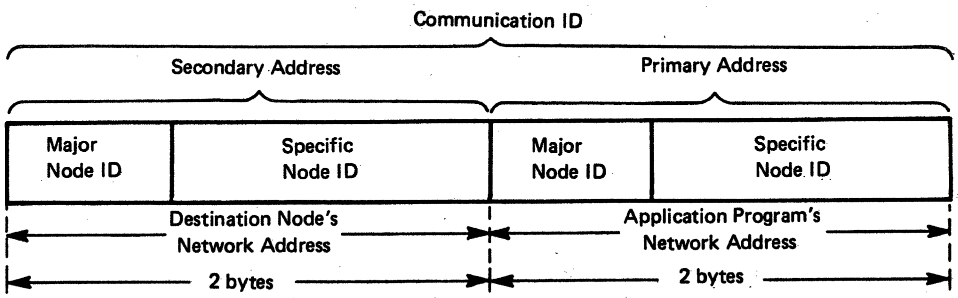
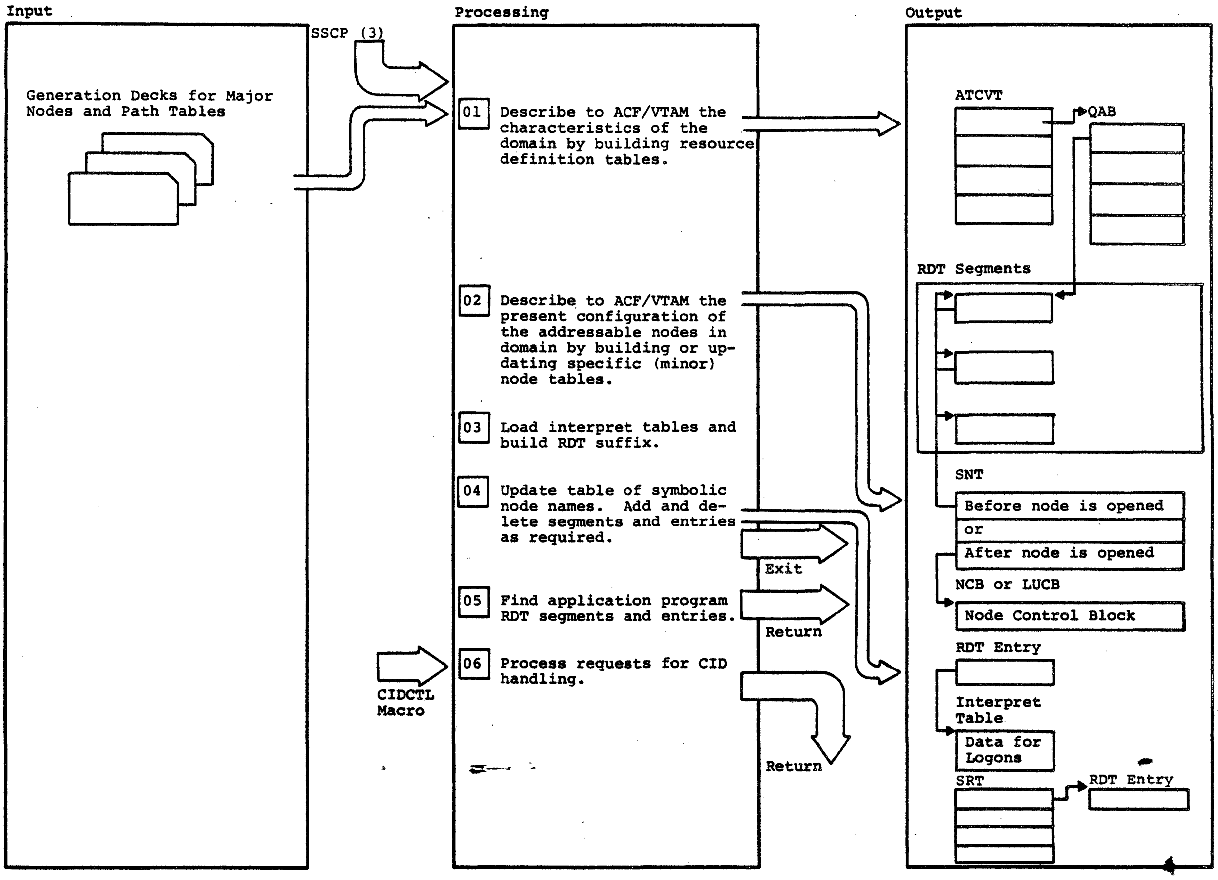


Figure 4-1. Format of a Communication ID

MO 2. System Definition



Notes	Routine	Label	Ref.

Notes	Routine	Label	Ref.

CHAPTER 5. SYSTEM SERVICES CONTROL POINT (SSCP)

The system services control point (SSCP) is a collection of routines that control the domain configuration by activating and deactivating nodes, participate in establishing and terminating sessions between nodes, and provide various measurement and maintenance services. The SSCP can be divided into five main subcomponents:

- Configuration services
- Session services
- Cross-domain resource manager (CDRM)
- Maintenance services
- Measurement services

An overview of SSCP operation is shown in MO 3.

CONTROLLING THE NETWORK CONFIGURATION

Once the initial network configuration has been defined, the SSCP routines handle requests from the network operator to activate and deactivate nodes. The operator can modify the network configuration to compensate for such changes as increased or decreased workload, hardware failure of one or more network components, or a change in application program throughput. After changing the configuration, the operator can display the status of the network or one of its components (see Chapter 4). The ACF/VTAM operator commands used to modify the network are:

- VARY, which activates and deactivates nodes
- MODIFY, which changes the operational characteristics of ACF/VTAM.

MONITORING SESSIONS

The SSCP routines also assist in initiating and terminating sessions between logical units in a network. Two types of sessions must be established before communication between logical units can take place. First, the SSCP establishes a session between itself and the device (such as a cluster control unit) with which it will communicate. This session is between the SSCP and the controlling portion of the device, known as the physical unit services (PUS). Second, the SSCP establishes a session with one or more of the logical units associated with the device.

When the SSCP has established sessions with one or more logical units, it is ready to handle and monitor communication between the logical units. The communication between the SSCP and a logical unit involves two types of data flow:

- Normal flow (inbound or outbound): In this type of flow, requests are sent sequentially, one after the other. A request that was sent before another request must arrive sooner, and, if a response is indicated for both, the response to the first request must be sent back before the response to the second request.

- **Expedited flow (inbound or outbound)**: In this type of flow, an important request (usually one that contains control indicators) is transmitted immediately, ahead of normal flow requests that are awaiting transmission. Only one expedited flow request can be sent at a time, and a response must be received to one expedited flow request before another can be sent.

These two types of data flow and the two types of sessions are shown in Figure 5-1.

Through the SSCP, a session is established between two logical units. The commands used for establishing and controlling sessions between the SSCP and logical units are:

Contact, which the SSCP sends to an NCP or PU services to request the activation of data link control contact with the station associated with the specified adjacent physical unit.

Activate Physical Unit, which the SSCP sends to establish a session between the SSCP and a physical unit (such as an NCP or cluster control unit).

Activate Logical Unit, which the SSCP sends to establish a session between the SSCP and a logical unit.

Initiate, which is sent from a logical unit to the SSCP to request that a session be established. This includes character-coded logons that the SSCP converts to Initiate Self commands.

Terminate, which is sent from a logical unit to the SSCP to end a session between logical units. This includes character-coded logoffs that the SSCP converts to Terminate Self commands.

Deactivate Logical Unit, which the SSCP sends to terminate the session between the SSCP and a logical unit.

Deactivate Physical Unit, which the SSCP sends to terminate the session between the SSCP and a physical unit.

When a node receives one of these commands, it returns a response, if requested, to indicate that the command has been received.

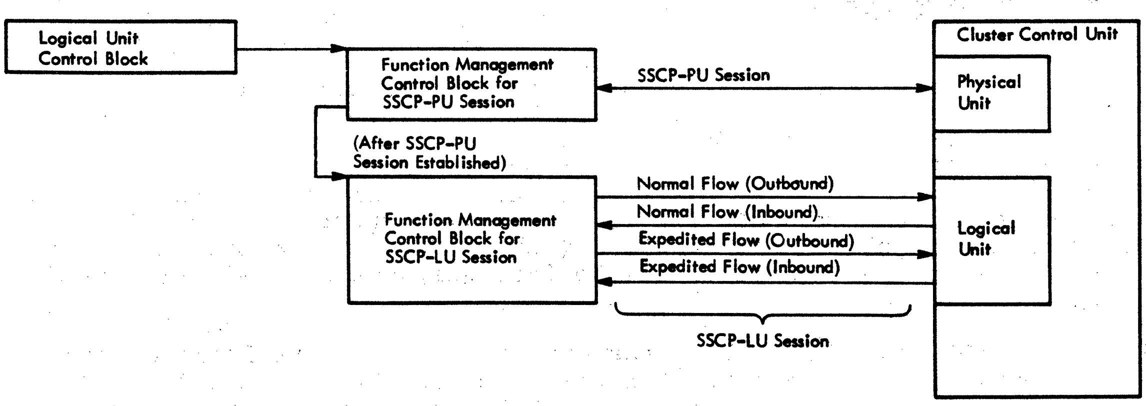


Figure 5-1. Establishing SSCP Sessions for Data Transmission

NETWORK FLOWS

Figures 5-2 through 5-21 illustrate the flow of requests and responses between the SSCP and logical and physical units to establish and terminate sessions. The following cases are illustrated:

- Activation of an NCP major node
- Activation of a cluster control unit
- Initiation of a session by a logical unit
- Initiation of a session by an application program issuing a SIMLOGON macro instruction
- Initiation of a session by an application program by issuing an OPNDST ACQUIRE macro instruction
- Initiation of a session between two application programs
- Establishment of a switched connection
- Sending a character-coded request to the SSCP
- Immediate termination of a session requested by a logical unit
- Termination of a session by an application program that is the primary end
- Termination of a session by an application program that is the secondary end

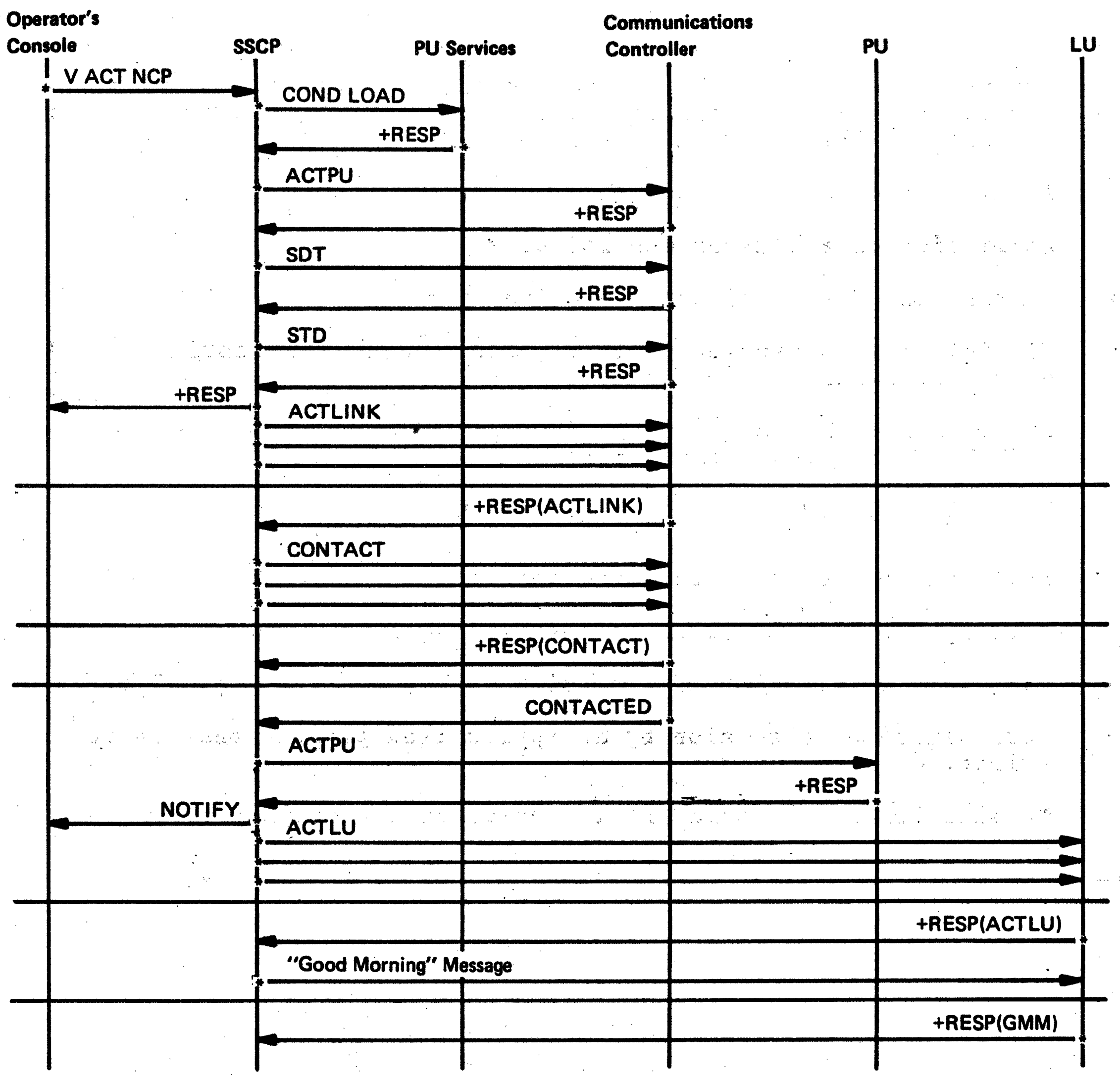


Figure 5-2. Activation of a Local Communications Controller

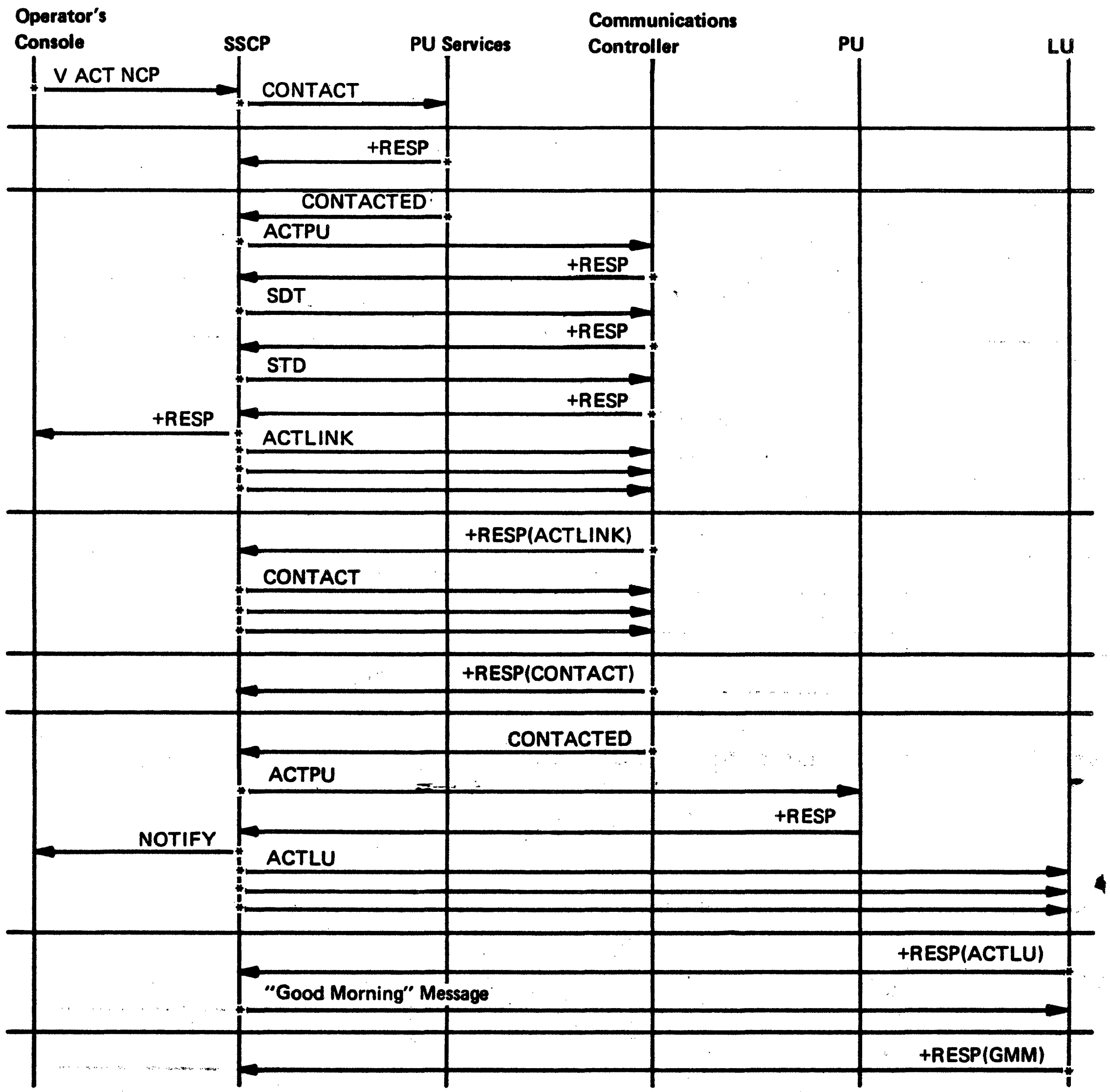


Figure 5-3. Activation of a Remote Communications Controller

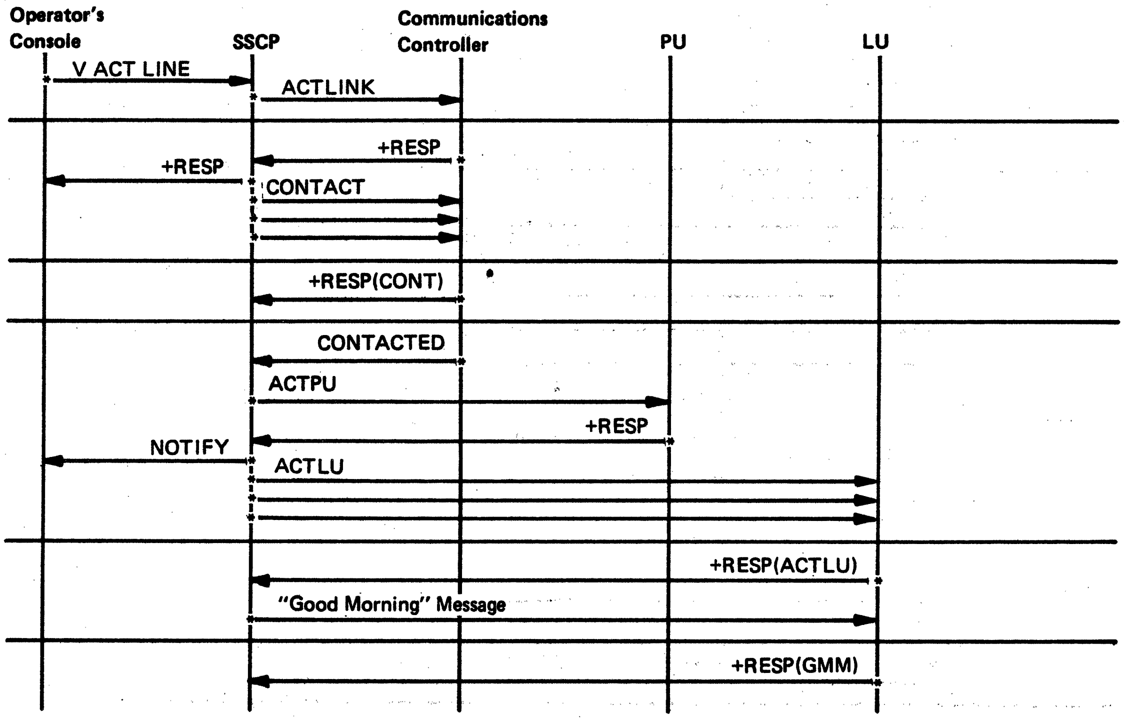


Figure 5-4. Activating a Line (ACTLINK)

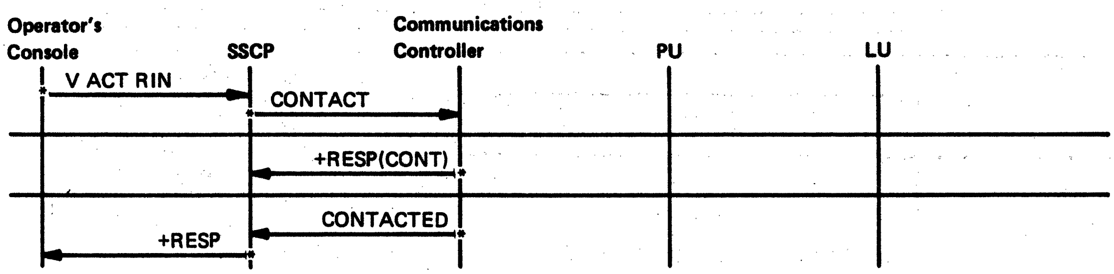


Figure 5-5. Activating a Cross-Subarea Link Station

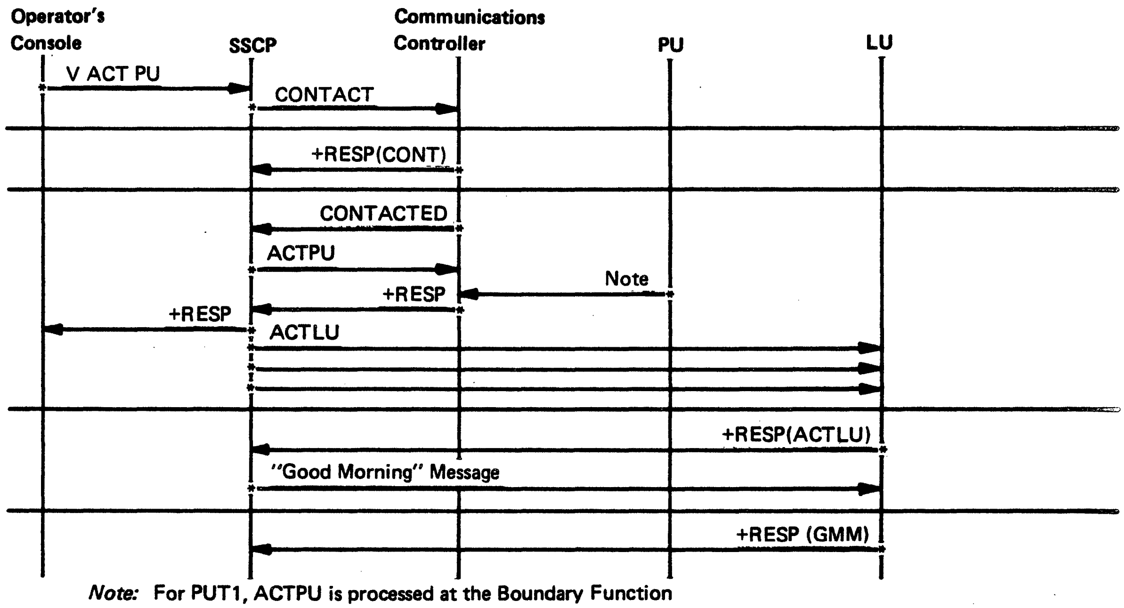


Figure 5-6. Activating a Physical Unit Type 1 or Type 2

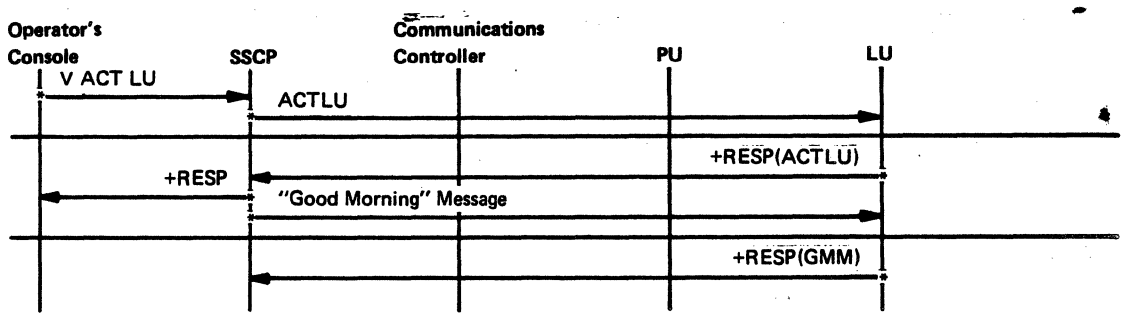


Figure 5-7. Activating a Logical Unit

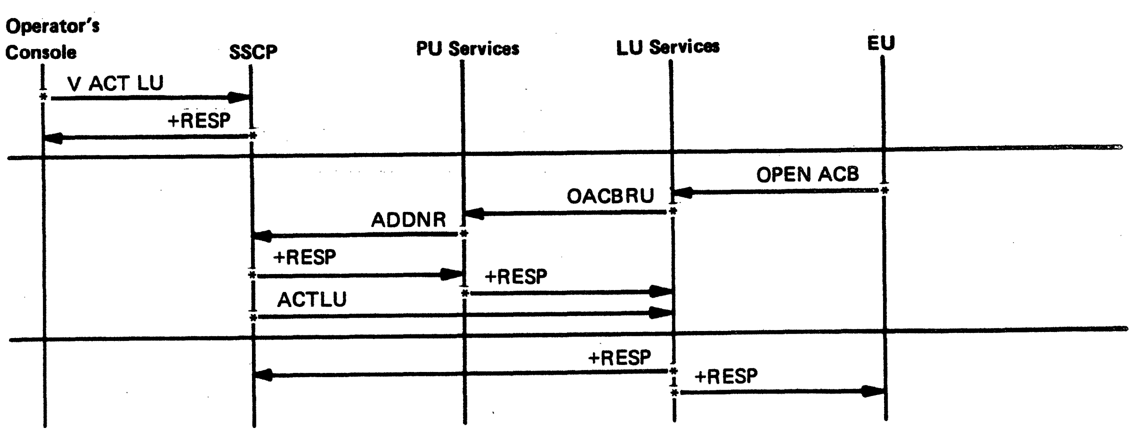


Figure 5-8. Activating an Application Program

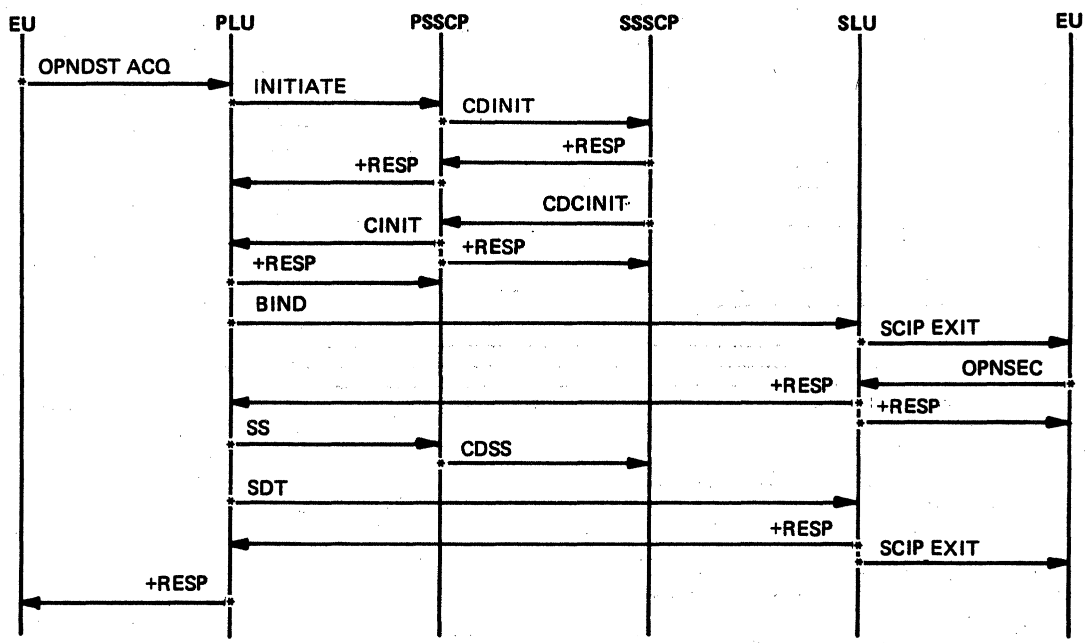


Figure 5-9. Primary Logical Unit Initiate (OPNDST Acquire)

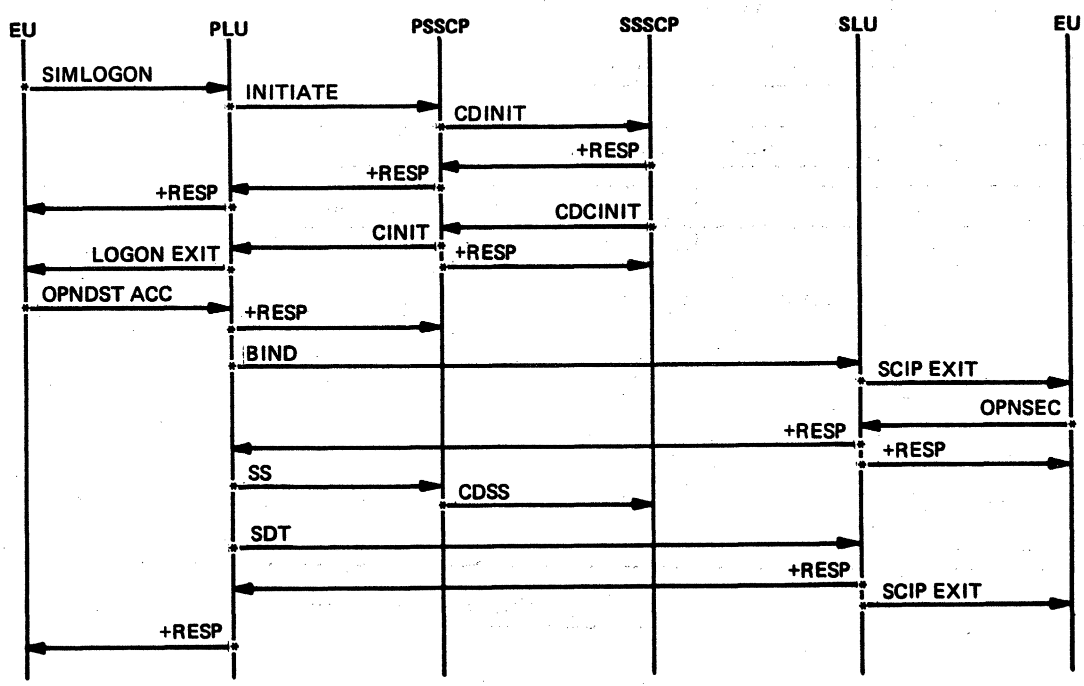


Figure 5-10. Primary Logical Unit Initiate (SIMLOGON)

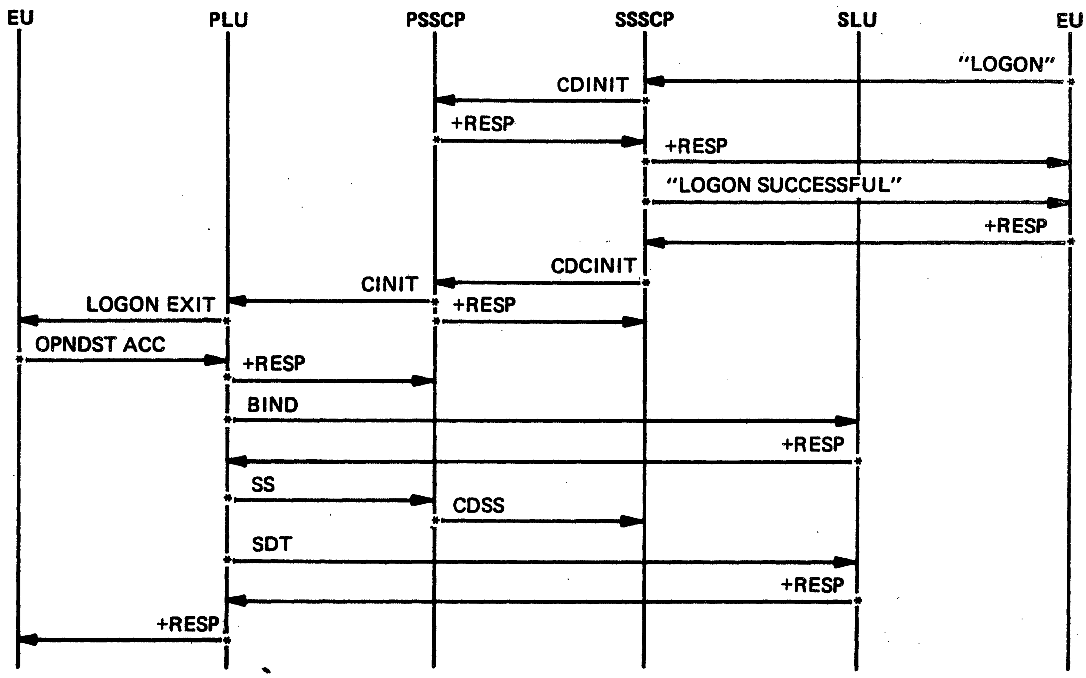


Figure 5-11. Secondary Logical Unit Initiate (LOGON)

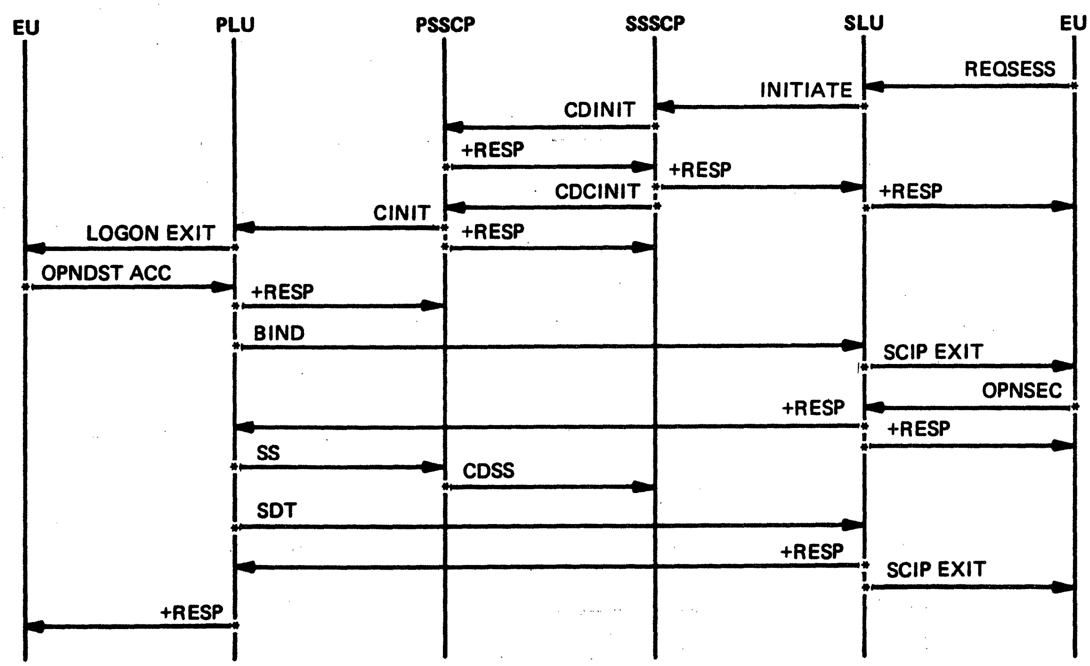


Figure 5-12. Secondary Logical Unit Initiate (REQSESS)

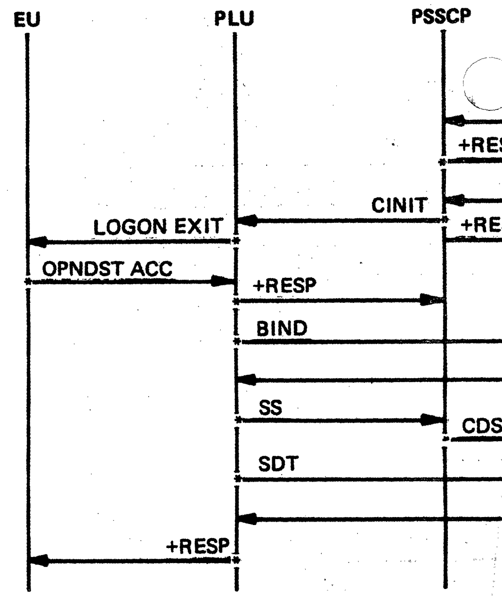


Figure 5-13. Secondary Logic

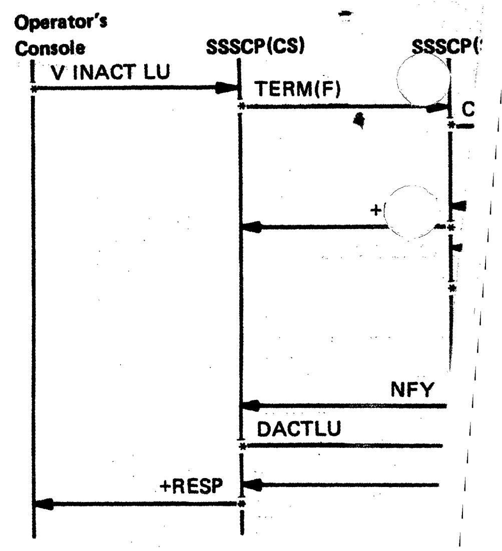


Figure 5-14. Inactivate I

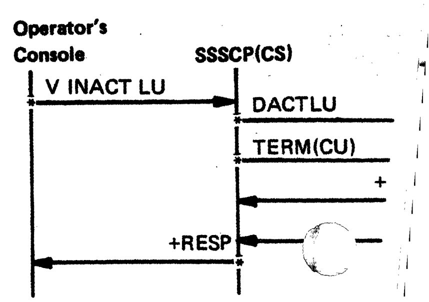
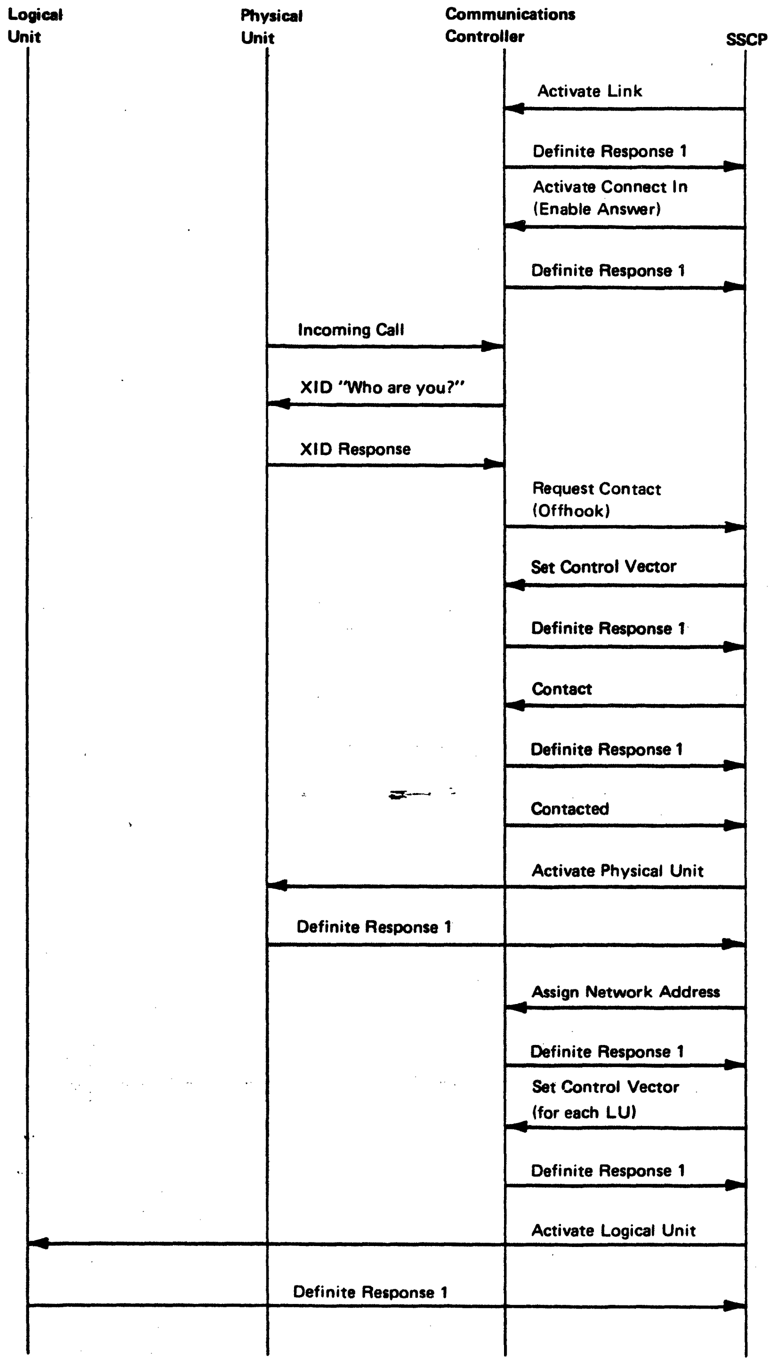
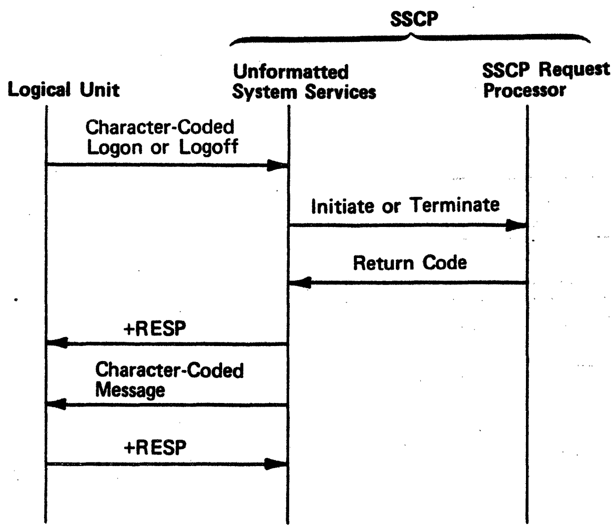


Figure 5-15. Inactiva



Note: To establish a switched connection the SSCP sends an Activate Link command to indicate the link is active. An Activate Connect In command is sent to enable the communications 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 communications controller sends an XID (exchange identification) and the physical unit responds with its ID (station address). The communications controller sends a Request Contact (Offhook) command to the SSCP. SSCP sends a Set Control Vector command containing address and pacing information to the communications controller. The standard activation sequence (Figure 5-3) then occurs.

Figure 5-16. Establishing a Switched Connection



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 command. The command is then passed to the SSCP request processor. If the return code indicates an unsuccessful transmission, unformatted system services converts the command into a form understandable to the terminal logical unit.

Figure 5-17. Sending an Unformatted Request to the SSCP

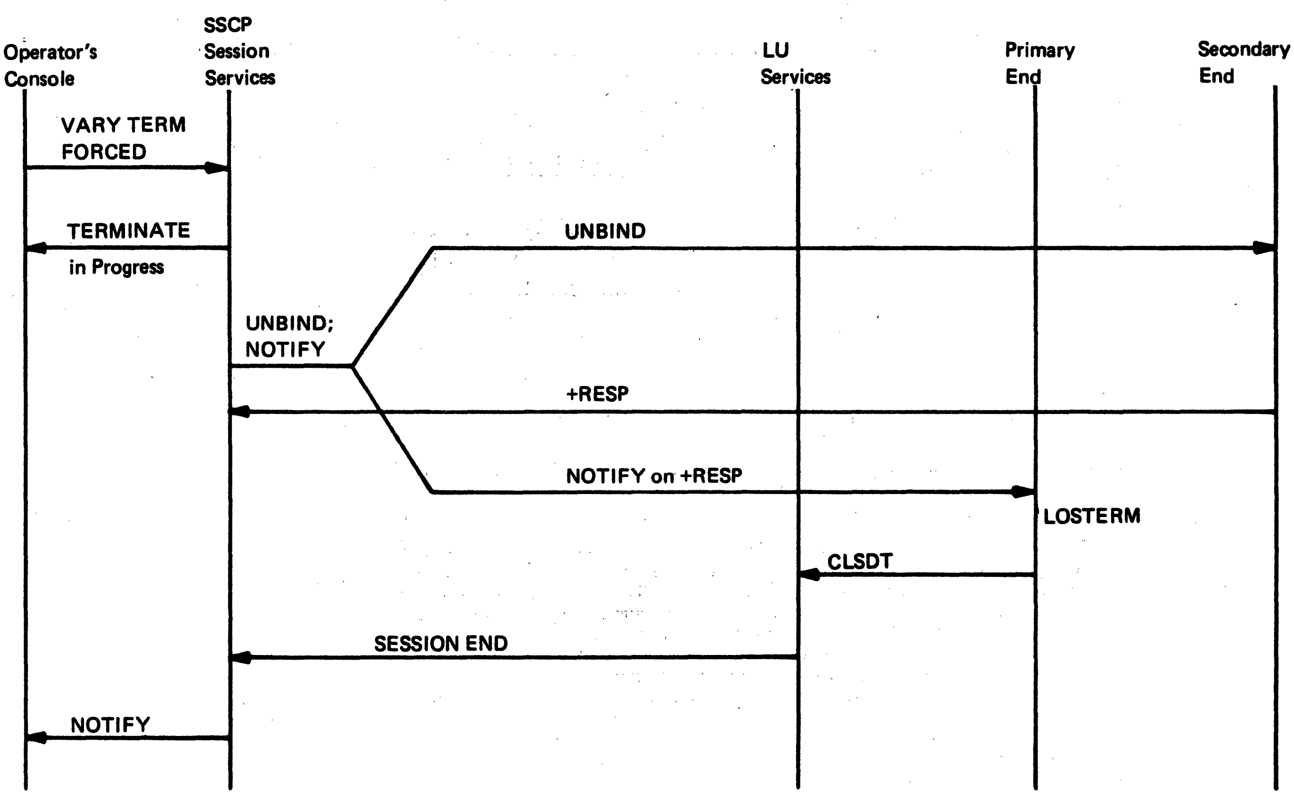


Figure 5-18. VARY TERM Forced

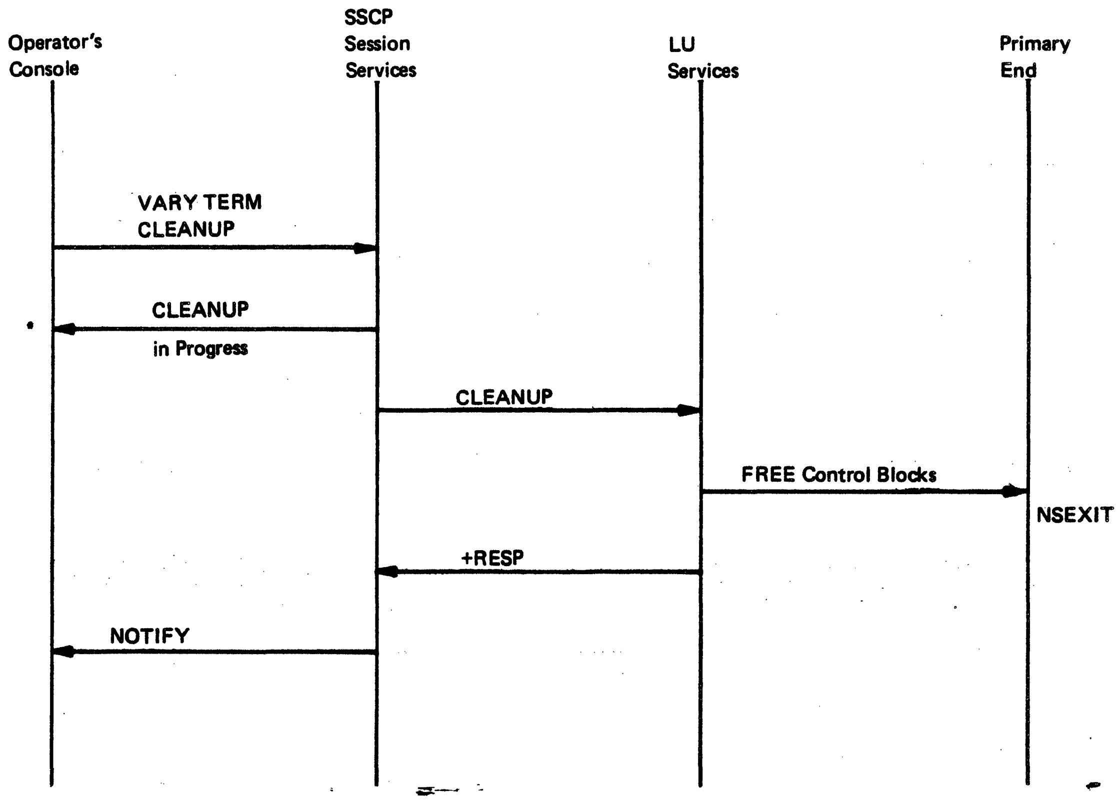


Figure 5-19. VARY TERM Cleanup

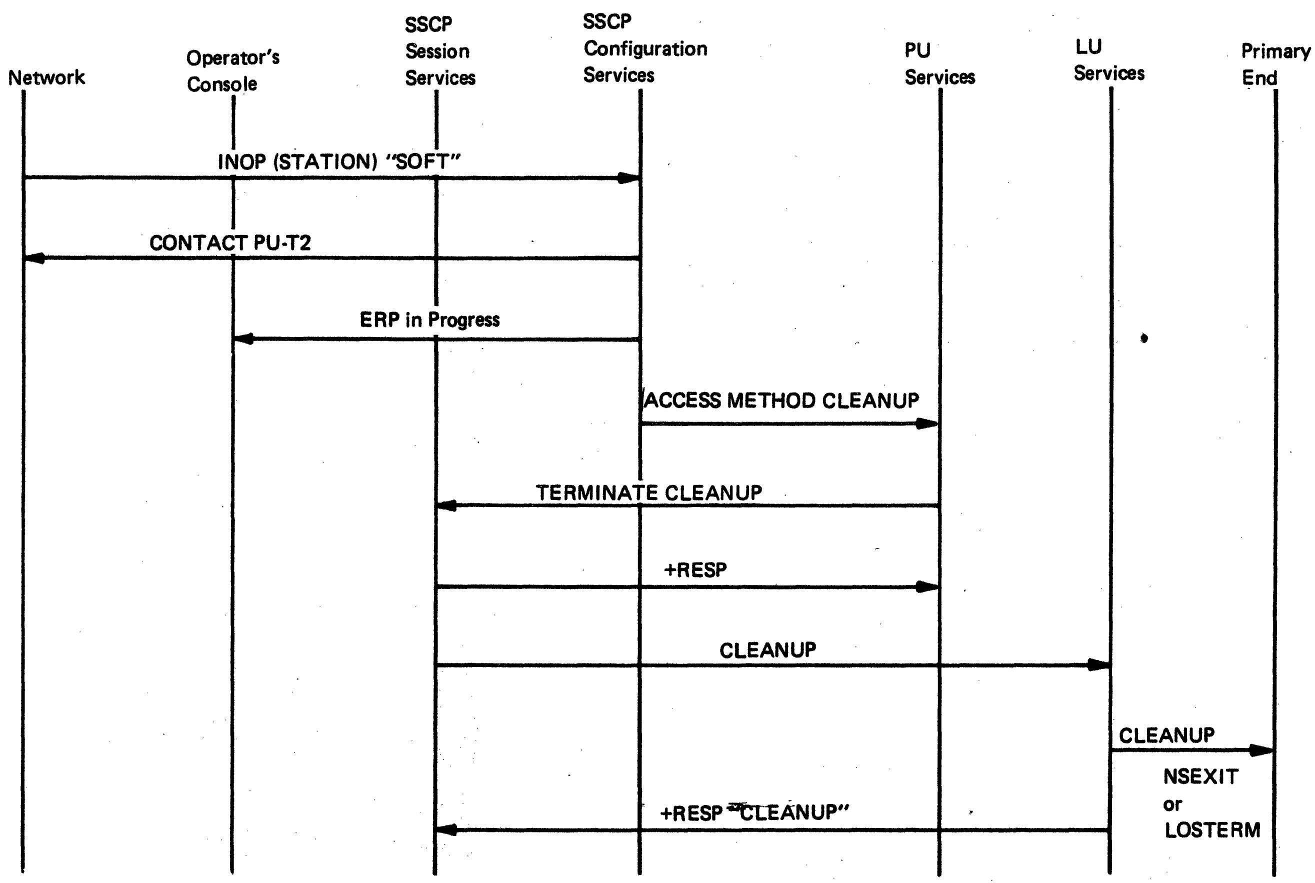


Figure 5-20. ERP Processing

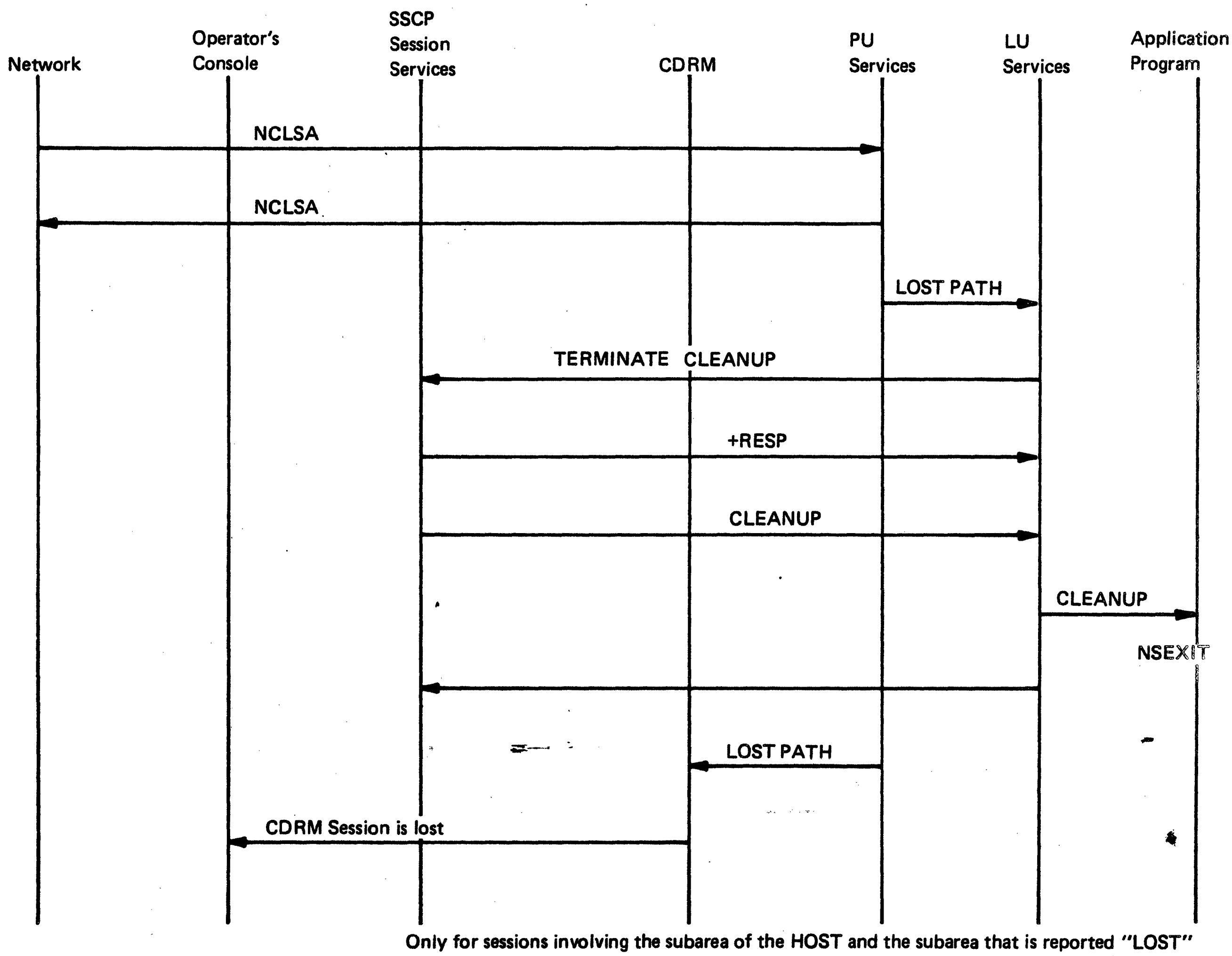


Figure 5-21. Lost Subarea Processing

CONFIGURATION SERVICES

Configuration services is the SSCP subcomponent that manages the configuration of the network. Configuration services records the status of each resource in the network. It updates these records should the status change as a result of requests received from network operators (such as VARY ACT or VARY DEACT commands) and as the result of requests and responses received from the resources themselves (such as INOP, REQ DISC, or ACTLU RUs). (See Figure 5-22.) Configuration services also supports miscellaneous commands that are used to modify the characteristics of the network resources (such as a VARY ANS command). (See Figure 5-23.)

The status of each resource is kept in its RDT entry in the form of a finite state machine (FSM). At any instant, the finite state machine is in a single state. The FSM shifts to a different state only as a result of a specific triggering event. The triggering event could be either an ACF/VTAM command or an internal SNA command. For example, if the current state of an FSM that represents a nonswitched logical unit is INACTIVE and the triggering event is a VARY ACT command for that logical unit, configuration services sends an ACTLU request and changes the state of the FSM to PEND ACTLU RESP. If the next triggering event is the arrival of a positive response to the ACTLU, configuration services changes the state of the FSM to ACTIVE.

MAINTENANCE SERVICES

Maintenance services is the SSCP subcomponent that handles terminal-user echo test processing. The terminal-user echo test allows a terminal user to verify that a path to ACF/VTAM is operational. The user can optionally provide test data to be sent to ACF/VTAM and specify how many times the data is to be returned. Should ACF/VTAM receive test data from a terminal, ACF/VTAM returns the test data, preceded by "IBMECHO", to the originating terminal the requested number of times.

MANAGEMENT SERVICES

Management services is the SSCP subcomponent that handles Forward RU processing. Forward RU processing is used when a communications network management (CNM) application program requests that the SSCP forward an RU to another network addressable unit.

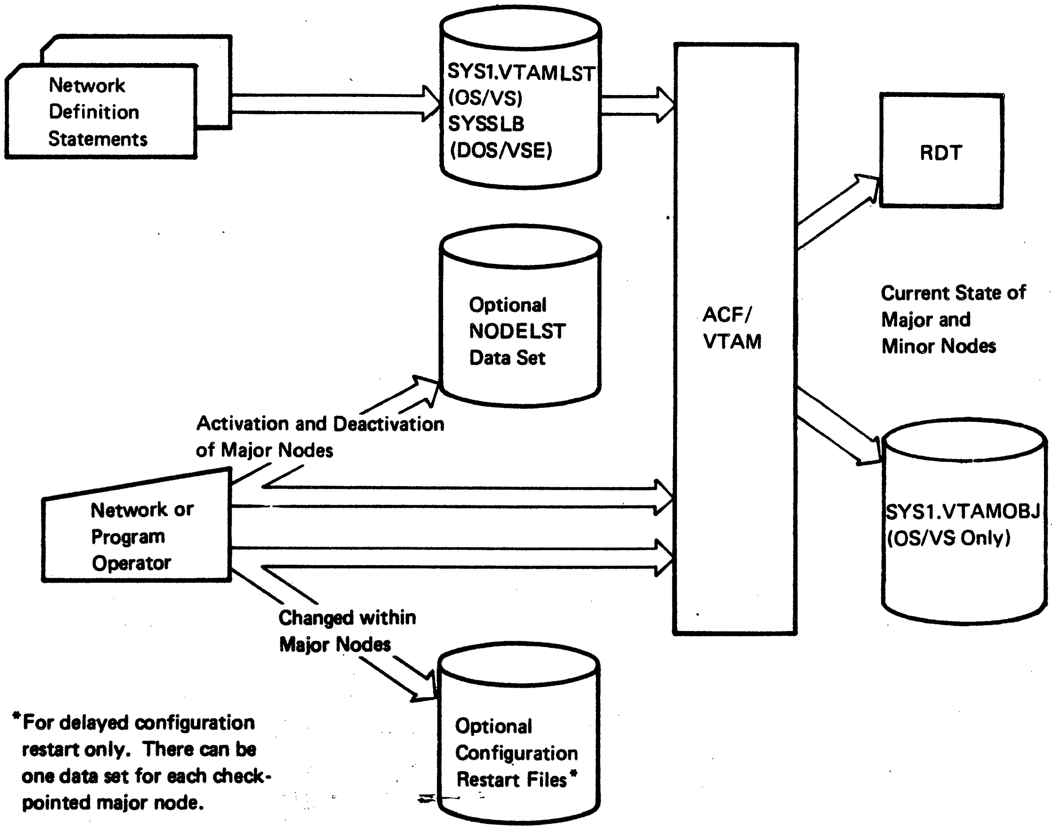


Figure 5-22. Recording Changes to the Network Configuration

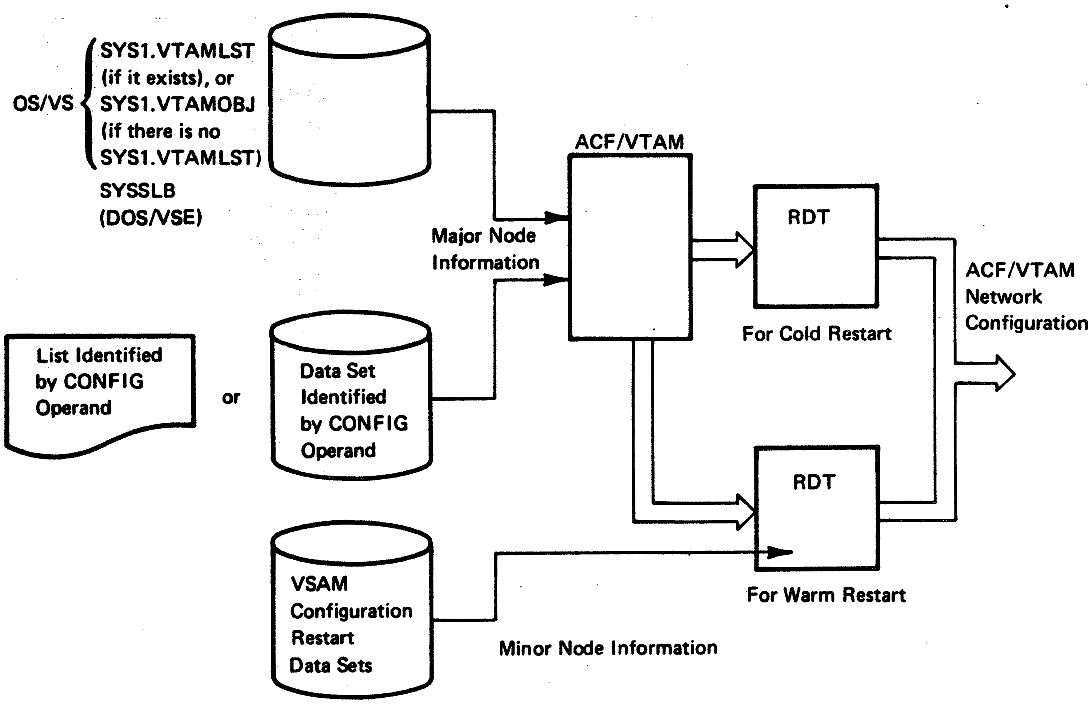
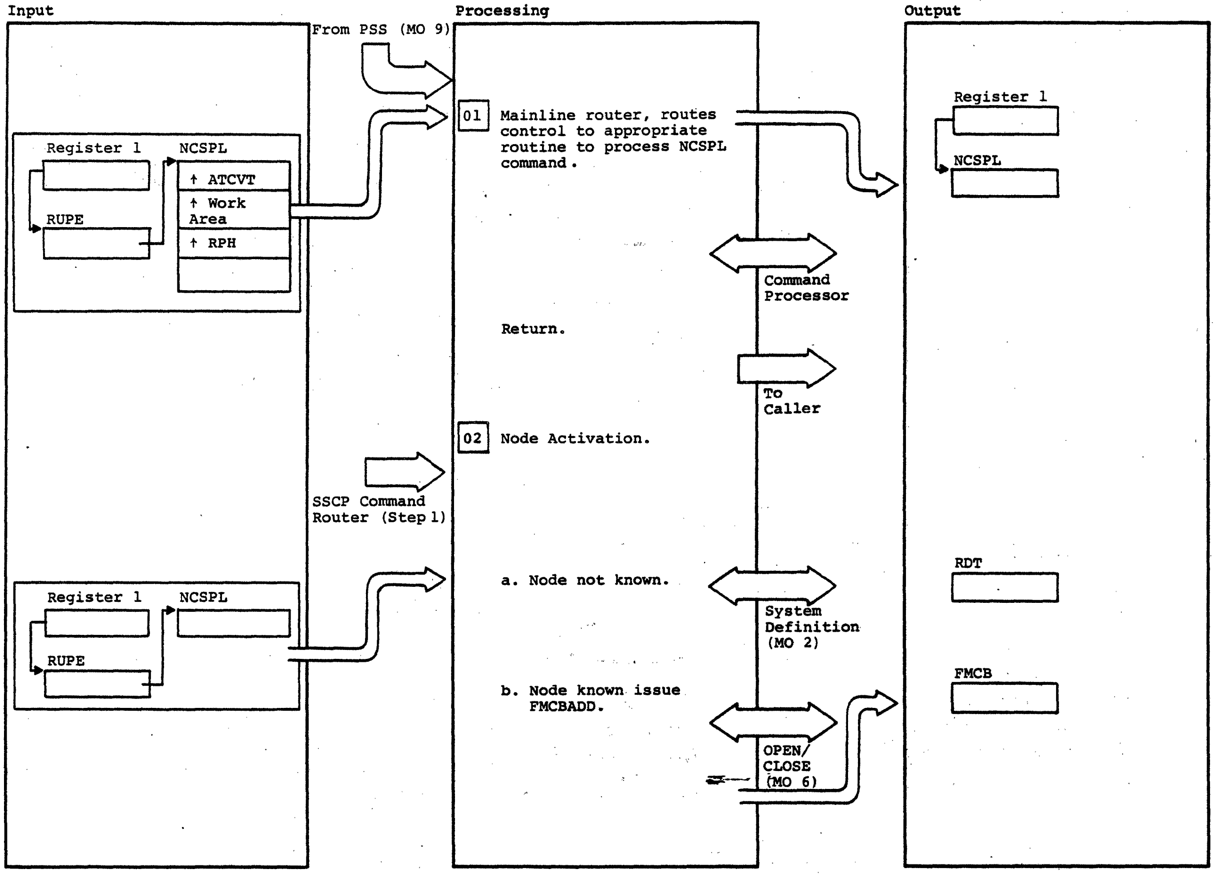


Figure 5-23. Delayed Configuration Restart

MO 3 (Part 1 of 6). System Services Control Point (SSCP)



Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.
01 SSCP processing is set up when the SSCP initial router is called by the ACF/VTAM task control routine to queue newly built NSCPLs to the VARY FE PAB. The SSCP mainline router receives control from PSS and either processes the NCSPL command internally or calls the appropriate routine to process the command.							

MO 3 (Part 2 of 6). System Services Control Point (SSCP)

Input

Processing

c. Activate major nodes and minor nodes.

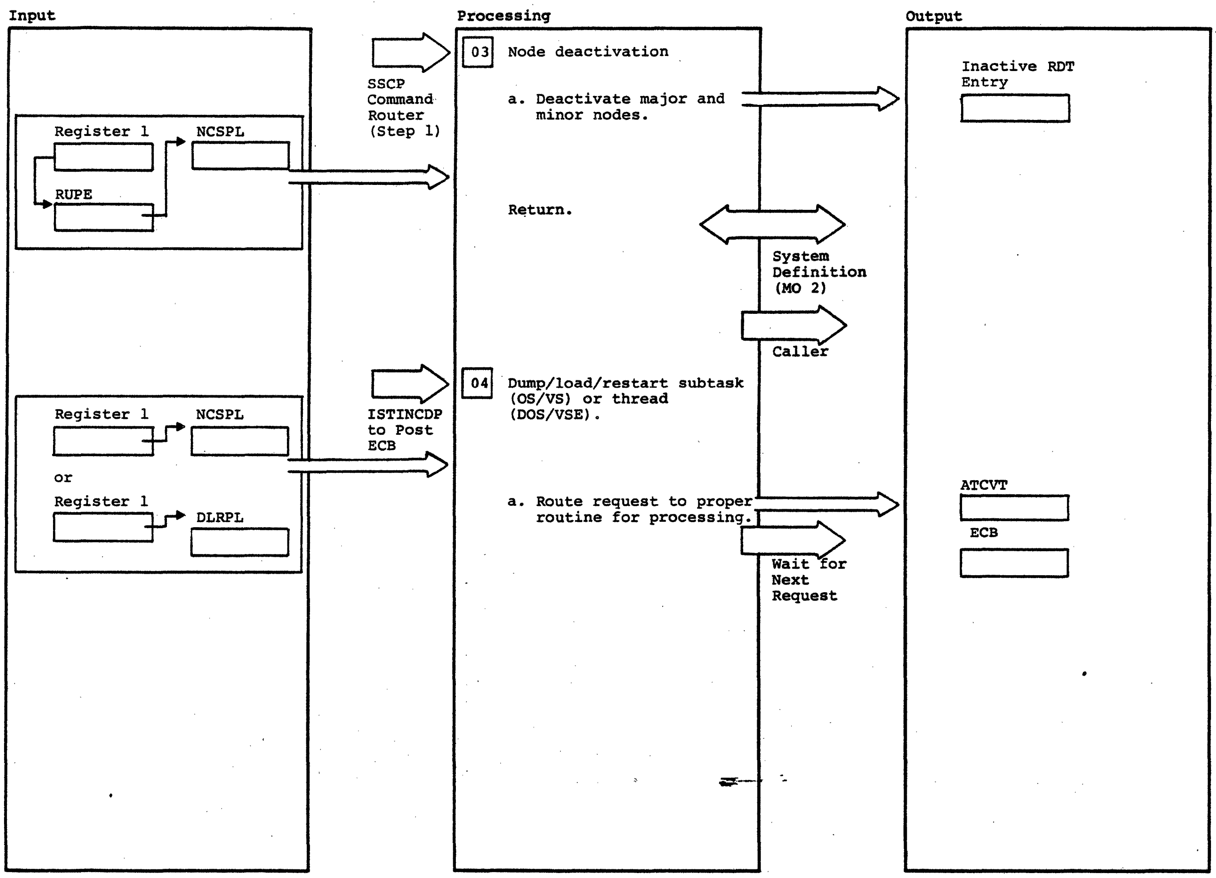
Return. → To Caller

Output

Active RDT entry → Node Control Block

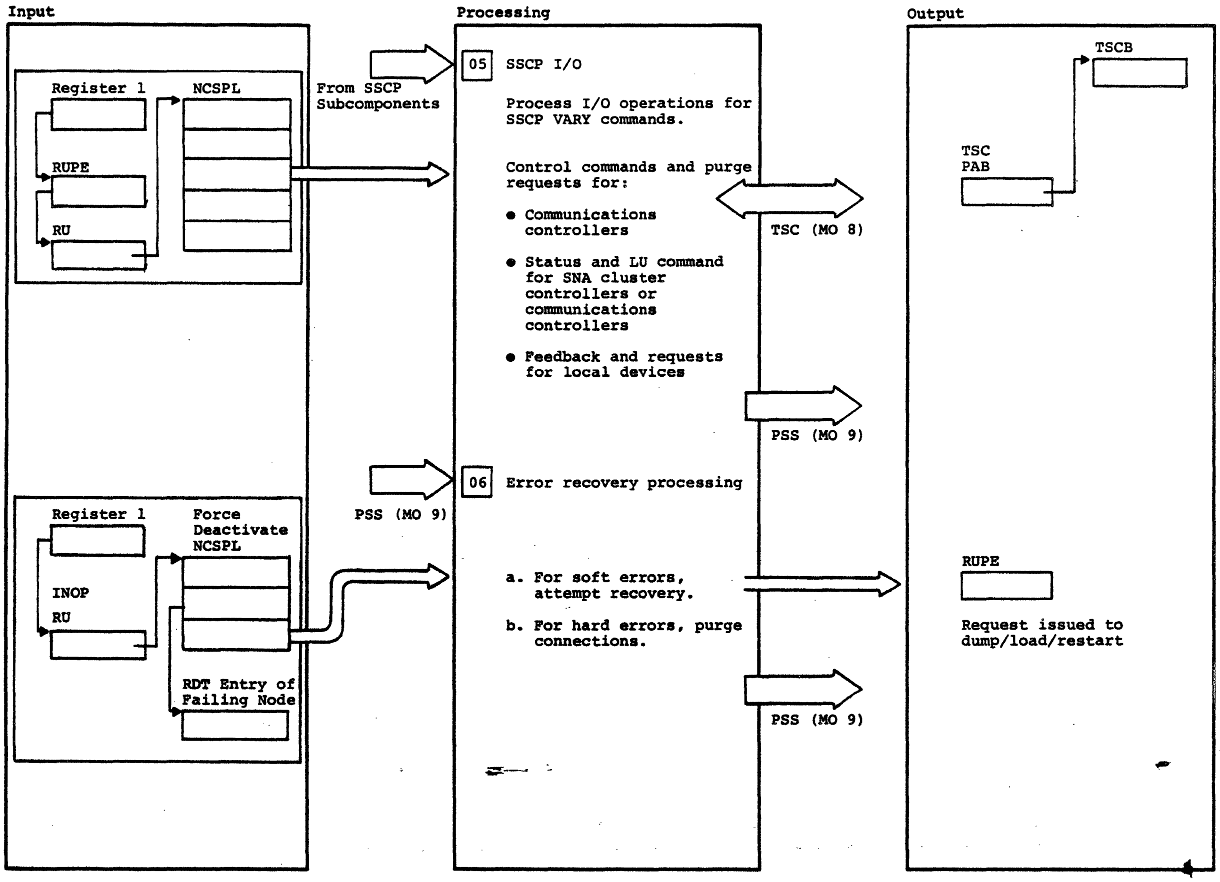
Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.
02 The activation subcomponent of SSCP receives control from the SSCP router and makes major and minor nodes known and available to ACF/VTAM. a. If the major node to be activated is not known, this subcomponent uses system definition to build an RDT segment for the major node. b. The activation subcomponent issues an FMCBADD macro instruction to build an FMCB on which to schedule any SSCP I/O for the node.							

MO 3 (Part 3 of 6). System Services Control Point (SSCP)



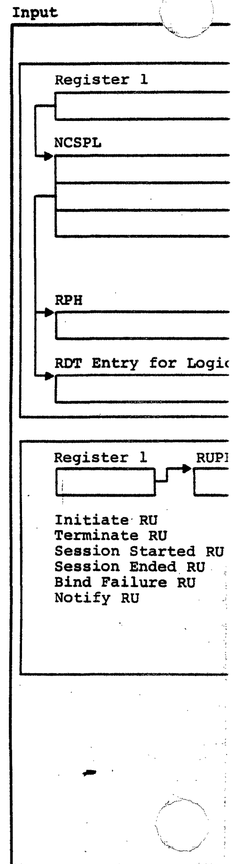
Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.
03 The deactivation subcomponent of SSCP receives control from the SSCP router to deactivate major and minor nodes and make them unknown to ACF/VTAM. Once the major node is inactive, SSCP calls system definition to release the RDT segment. a. For normal deactivation, it waits for the user session to end and then deactivates the node. For immediate or forced deactivation, it forces the end of the user session and then deactivates the node. 04 The dump/load/restart subtask receives control when a DLRPL or an NCSPL is queued to the D/L/R work queue and the D/L/R subtask is posted. a. Processing depends on the type of request. The function is performed by a call (DOS/VSE) or an ATTACH (OS/VS) to the appropriate D/L/R routines to handle the request. Processing includes: • Dump NCP • Load NCP • Local PU activation/ deactivation • Operator query • Checkpoint configuration							

MO 3 (Part 4 of 6). System Services Control Point (SSCP)



Notes	Routine	Label	Ref.
05 The I/O subcomponent of SSCP receives control from other SSCP subcomponents. It handles I/O processing for any nodes requiring I/O.			
06 The Configuration Services subcomponent of SSCP receives control from PSS. Entry is by TPQUE for an INOP, force deactivate, or when an I/O operation has been completed, for Configuration Services. a. If the error is recoverable, attempt recovery and issue soft purge to SSCP session with failing node. b. If the error is not recoverable, terminate all sessions and schedule a deactivate request for the node.			

Notes	Routine	Label	Ref.



Notes

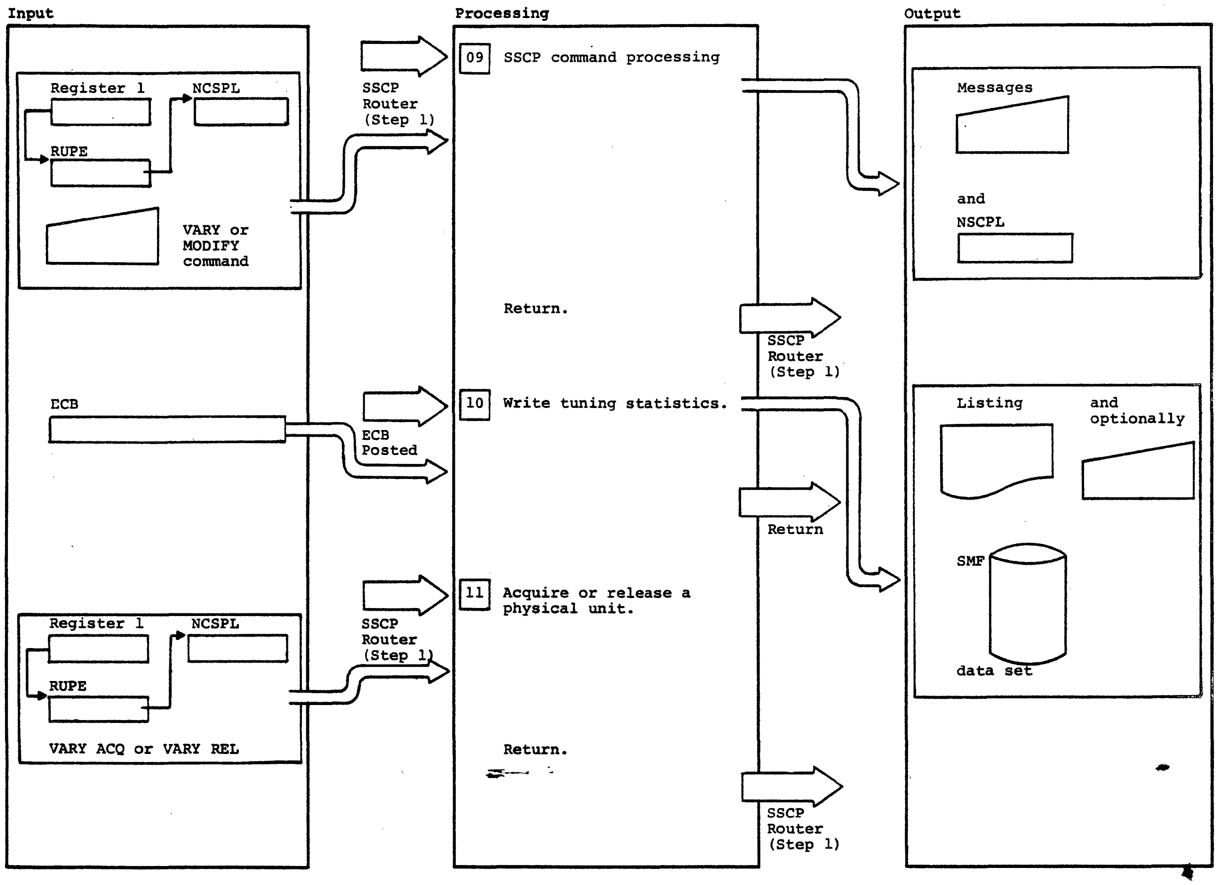
07 Configuration Se
control from the
a result of an O
the session moni
dial-start when
is received. Di
tions are create
connections are
processing routi

08 The session serv
processing subco
receives control
a RUPE being que
session monitor

a. For character
unformatted s
function mana
given control.

b. For field-for
the appropria
called to pro

MO 3 (Part 6 of 6). System Services Control Point (SSCP)



Notes	Routine	Label	Ref.
09 This SSCP subcomponent is a collection of independent command processors. Each processor handles a specific command as indicated in the NCSPL. Each command is an operator command or an internally generated SSCP command. Commands processed include: • VARY INOP • VARY PATH • MODIFY • VARY ANSWER All SSCP command processing routines are called by the SSCP router.			
10 The tuning statistics subcomponent of SSCP is a subtask of ACF/VTAM. When first entered, it sets up the operating environment and waits for the following ECBs to be posted: • Time ECB • ECB due to deactivation of a local programmable controller • ACF/VTAM termination ECB When one of the above ECBs is posted, this module gets control to format and write tuning statistics.			
11 This SSCP subcomponent provides switched-line backup for a non-switched line PU, through use of the ACQ and REL parameters of the VARY command.			

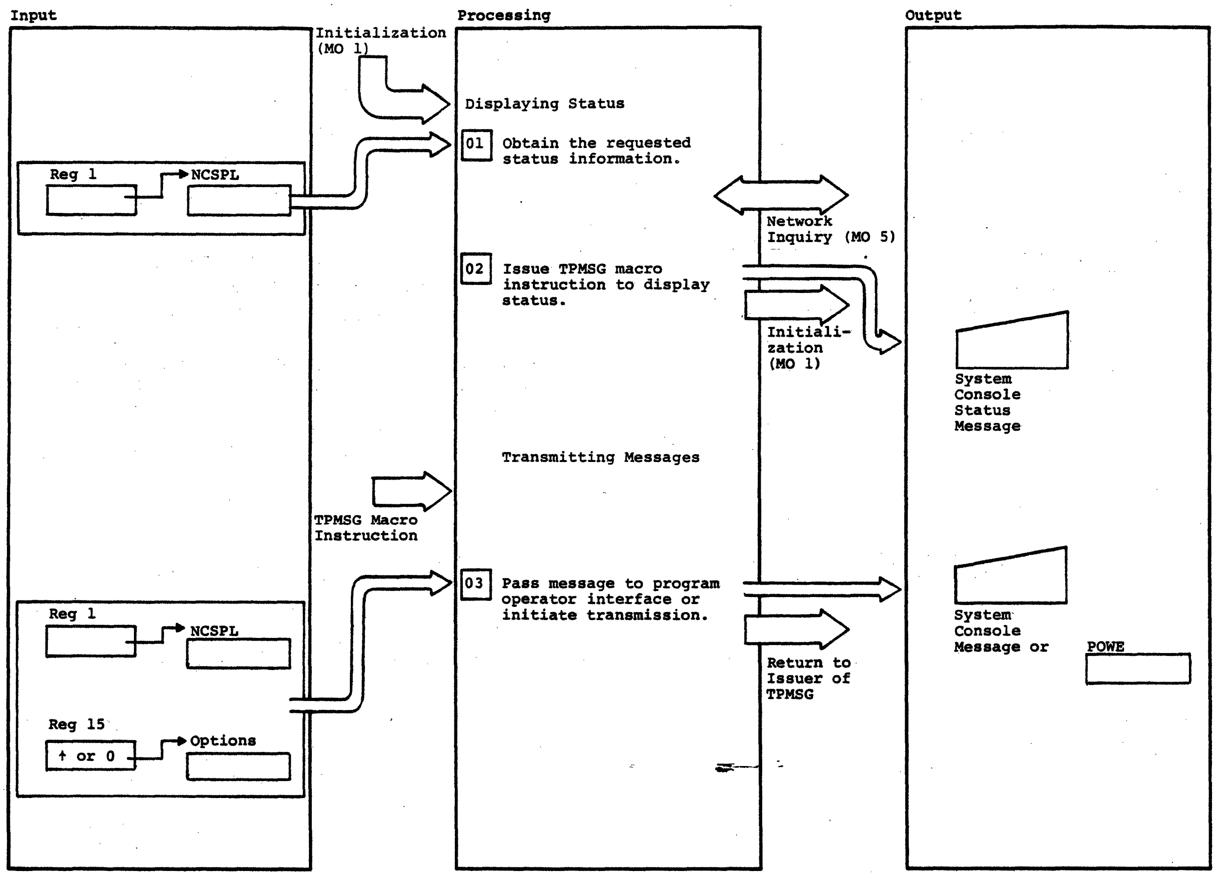
CHAPTER 6. NETWORK OPERATOR COMMAND FACILITIES

Network operator command facilities does the following:

- The display processor displays information requested with DISPLAY commands issued by the network operator.
- The TPMSG processor transmits operator messages to the network operator or routes them to application programs that have been designated to handle program operator functions.
- The program operator interface (POI) receives ACF/VTAM operator commands from authorized application programs (SENDCMD processing) and routes command responses and unsolicited ACF/VTAM operator messages to the authorized application programs (RCVCMD processing).
- Network operator services (NOS) provides support for ACF/VTAM network operator commands. NOS is entered with an unformatted command issued by the network operator (for example, DISPLAY NCPSTOR), or with a formatted RU from the SSCP (for example, DELIVER RECORD STORAGE to get the results of a DISPLAY NCPSTOR).

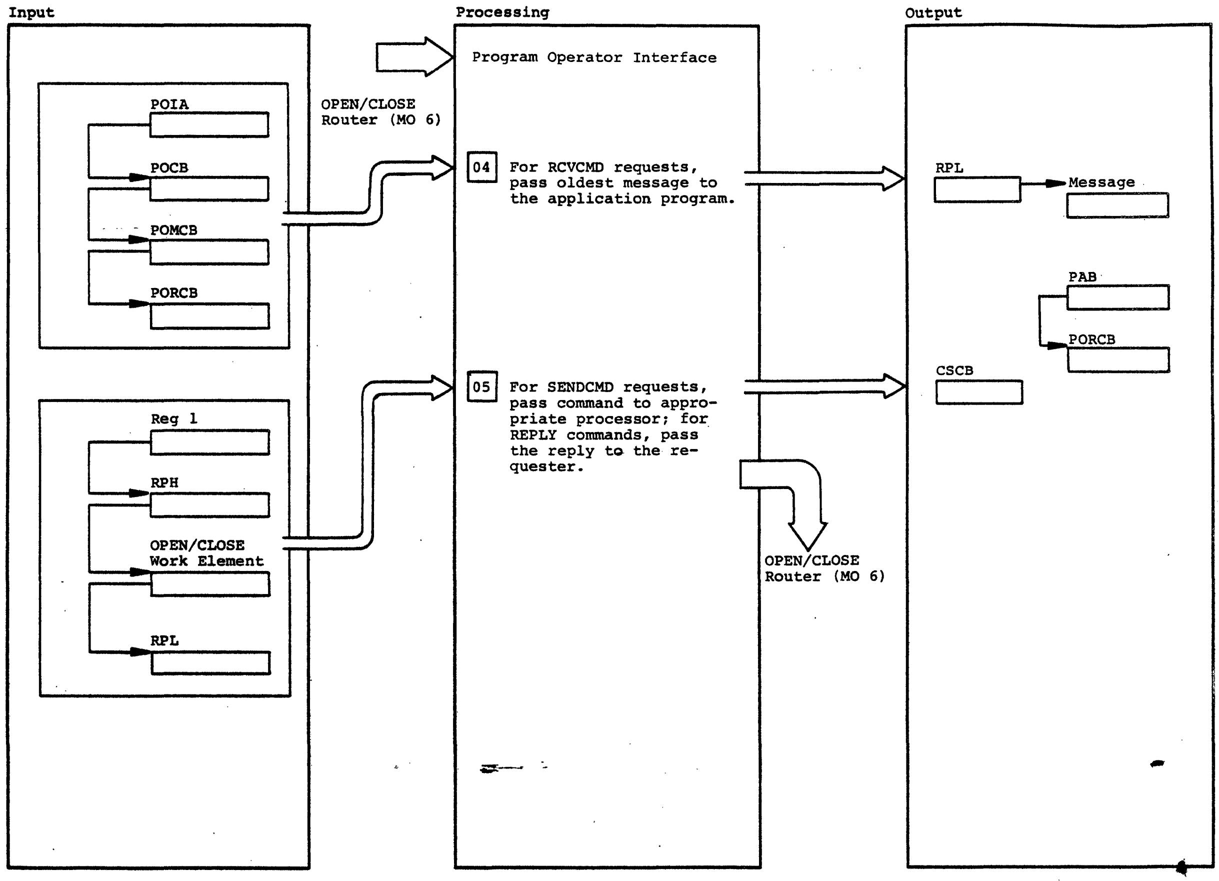
An overview of ACF/VTAM operator command facilities is shown in MO 4.

MO 4 (Part 1 of 3). Network Operator Command Facilities



Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.
02 Issued as many times as necessary to provide the requested information.							

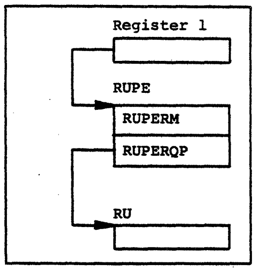
MO 4 (Part 2 of 3). Network Operator Command Facilities



Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.

MO 4 (Part 3 of 3). Network Operator Command Facilities

Input



Processing

06 Network Operator Services

a. Perform the function requested by the operator command, either directly or by requesting services of the SSCP.

b. Associate the RU received from the SSCP with the NOS procedure that initiated it, and complete the procedure.

→ OPEN/CLOSE Router (MO 6)

Output

Notes	Routine	Label	Ref.
06 Network Operator Services provides support for the network operator's interface to VTAM. Requests are received by NOS from the network operator, as unformatted commands, such as DISPLAY NCPSTOR, or from the SSCP, as formatted RUs, such as DELIVER. NOS is a Communication System Manager (CSM) LU. It uses its SSCP-LU session to request functions that it cannot perform directly. a. NOS understands what has to be done for each operator command. Where possible, it performs the function directly. Otherwise, it sends RUs to the SSCP to request the desired function. These RUs can be either direct requests for SSCP action or FORWARDS of requests to be sent by the SSCP to a NAU. b. RUs are sent by the SSCP to the NOS complete procedure requested on behalf of an operator command. For example, DISPLAY NCPSTOR causes NOS to forward a DISPSTOR RU to the SSCP. When the NCP replies with a RECORD STORAGE RU, it is delivered to NOS, which formats the messages for the display.			

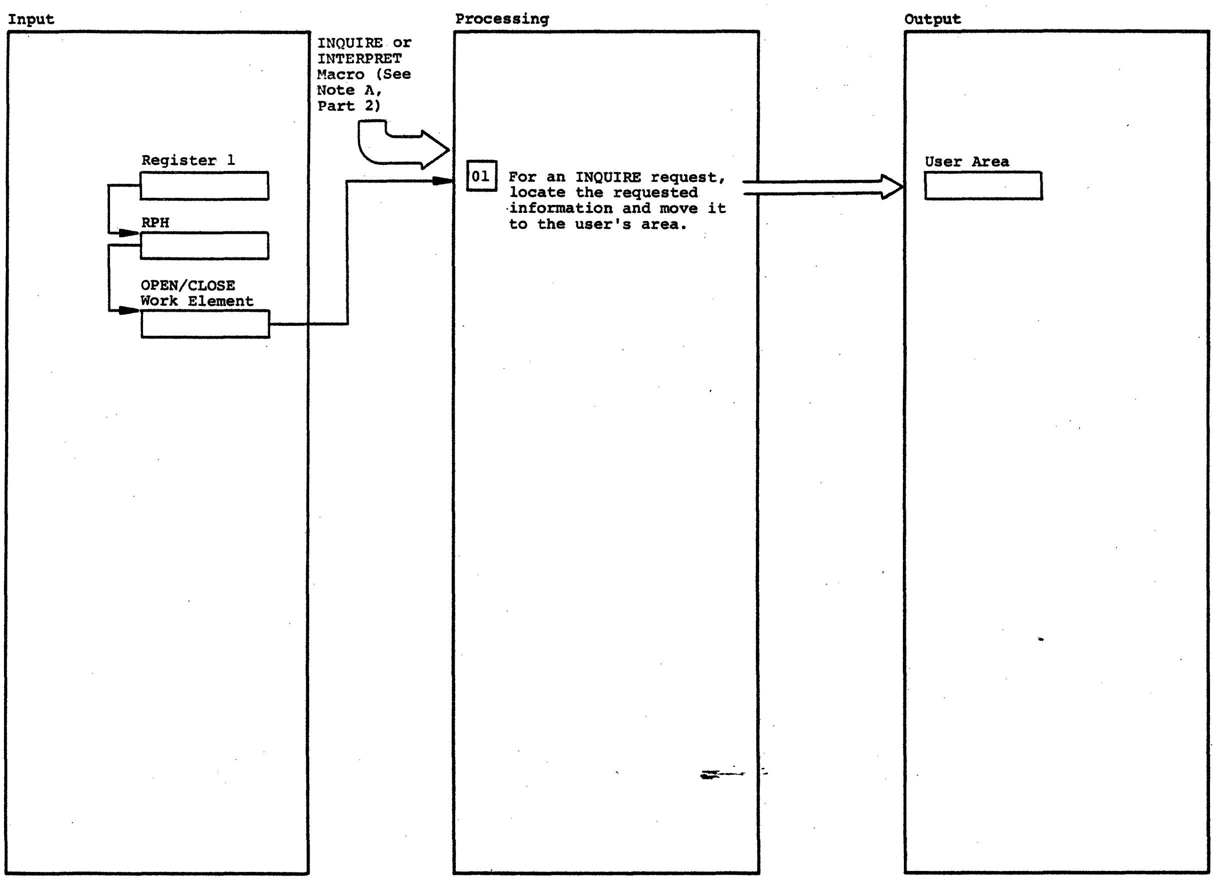
Notes	Routine	Label	Ref.

CHAPTER 7. NETWORK INQUIRY

This component contains the macro instructions that are issued by the DISPLAY command processor to obtain all the information that can be requested by a DISPLAY command.

An overview of network inquiry operation is shown in MO 5.

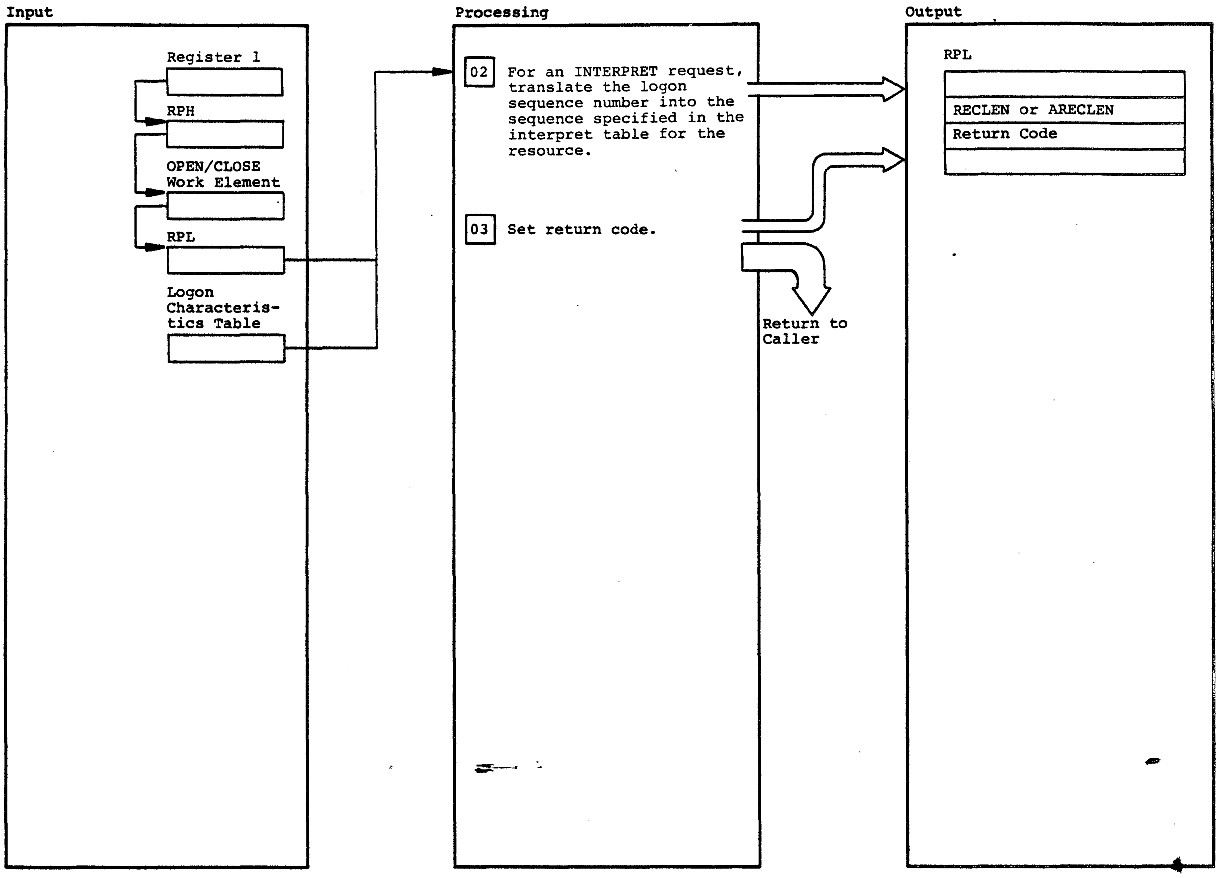
MO 5 (Part 1 of 2). Network Inquiry



Notes	Routine	Label	Ref.
01 Types of data that can be requested by INQUIRE and the source of information are as follows:			
• Logon date - from the location obtained from LOGON SIB.			
• Device characteristics - from the node's RDT entry.			
• Counts - by counting SIBs queued to the application program's RDTE.			
• Terminal - by search of RDT entries for the specified node.			
• Application program status - from the application program RDT entry.			
• Display of a resource - from the RDT entry, node control block, FMCB and/or SIB queue, depending on the display requested.			
• Communications controller name - from the RDT entry.			
• CID translation - if CID supplied, from the RDT; if symbolic name supplied, from the FMCB.			
• Top request on LOGON application program's logon queue - from the SIB queue and RDT entry.			
• Session parameters - from the logon mode table (pointed to by the RDT entry or ATCVT), if the application program specifies NIBLMODE, or from the logon data buffers.			

Notes	Routine	Label	Ref.
• Session key - from the FMCB.			
Network inquiry processing for domain display requests:			
• DISPLAY MAJNODES - from the RDTE for each communications controller, local SNA major node, local non-SNA major node, application program RDT segment, CDRM RDT segment, CDRSC RDT segment, and switched SNA major node in the domain.			
• DISPLAY LINES - from the RDTE for each line off each active communications controller in the domain and its associated RDTE.			
• DISPLAY TERMS/DISPLAY CLSTR - from the RDTE for each LU or terminal in the domain and all their associated higher-level node RDTES.			
• DISPLAY APPLS - from the application program RDT segments in the domain.			
• DISPLAY BFRUSE - from the buffer pool control blocks for each ACF/VTAM buffer pool.			
• DISPLAY CDRMS/DISPLAY CDRSCS - from the CDRM or CDRSC RDT segments respectively.			
• DISPLAY PENDING - from the RDTE for each node in the system that has an outstanding SSCP request.			
• DISPLAY PATHTAB - from the MNT entries.			

MO 5 (Part 2 of 2). Network Inquiry



Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.
02 The logon sequence is specified by the AREA and RELCEN fields of the RPL. The sequence in the interpret table and its length are returned by the AAREA and ARECLEN fields of the RPL.							
03 If the major return code is 0 (successful) the RPL field RECLEN (for INQUIRE requests) or ARECLEN (for INTERPRET requests) contains the length of valid data. If the major return code is 8 (error) and the reason code is 2B (area too small), the RPL field RECLEN or ARECLEN gives the length needed to hold the requested data.							
A Entry is from LU services routing after an INQUIRE or INTERPRET macro is issued in an application program or in network operator command handling.			MO 4				

CHAPTER 8. OPEN/CLOSE

OPEN/CLOSE comprises four function groups;

- OPEN and CLOSE access method control block
- System services routing
- FMCB handling
- SRT handling

An overview of OPEN/CLOSE operation is shown in MO 6.

OPEN/CLOSE ACCESS METHOD CONTROL BLOCK (ACB)

The OPEN/CLOSE ACB receives control from the system to process the opening or closing of a ACP/VTAM ACB. An OPEN or CLOSE ACB AMRU (access method RU) is built and passed to physical unit services to perform the physical unit services portion of the OPEN/CLOSE ACB (Figures 8-1 and 8-2).

SYSTEM SERVICES ROUTING

This function handles the end user requests and responses by setting up the proper interface and invoking the appropriate processor. SENDCMD, RECVCMD, and INQUIRE DISPLAY requests are passed to their respective processors. All other requests are passed in the form of RUPEs to the LUS manager. Responses are also passed to LUS.

FMCB HANDLING

The FMCB handler provides a macro interface (FMCBADD and FMCBDEL) to build and delete FMCBs. FMCBADD invokes procedures to build the EPT, DVTs, and FMCB. FMCBDEL releases what FMCBADD builds.

SRT HANDLING

SRT handling provides a macro interface for adding SRT entries and for finding and optionally deleting SRT entries for specific symbolic names.

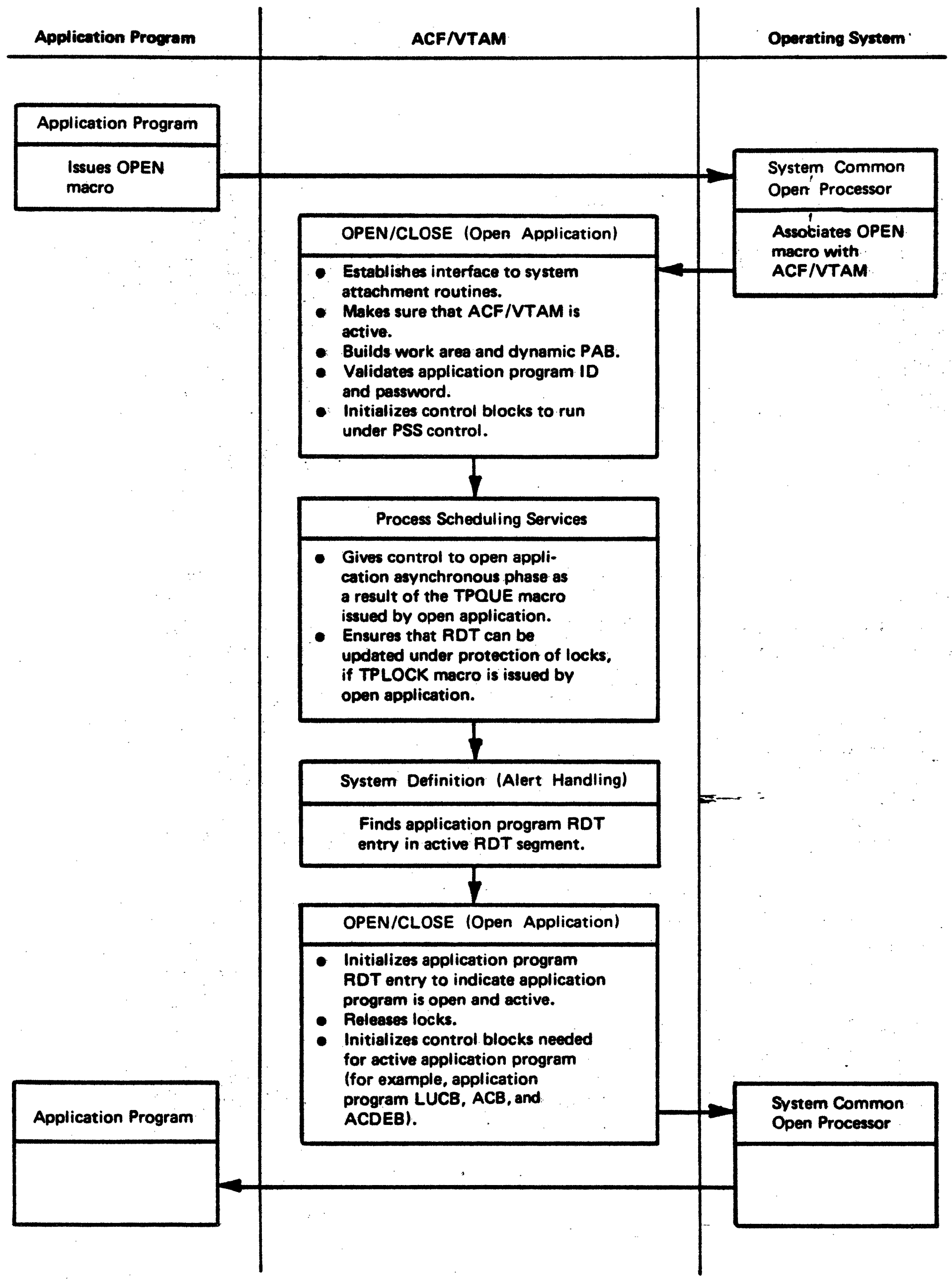


Figure 8-1. Processing an OPEN Macro Instruction

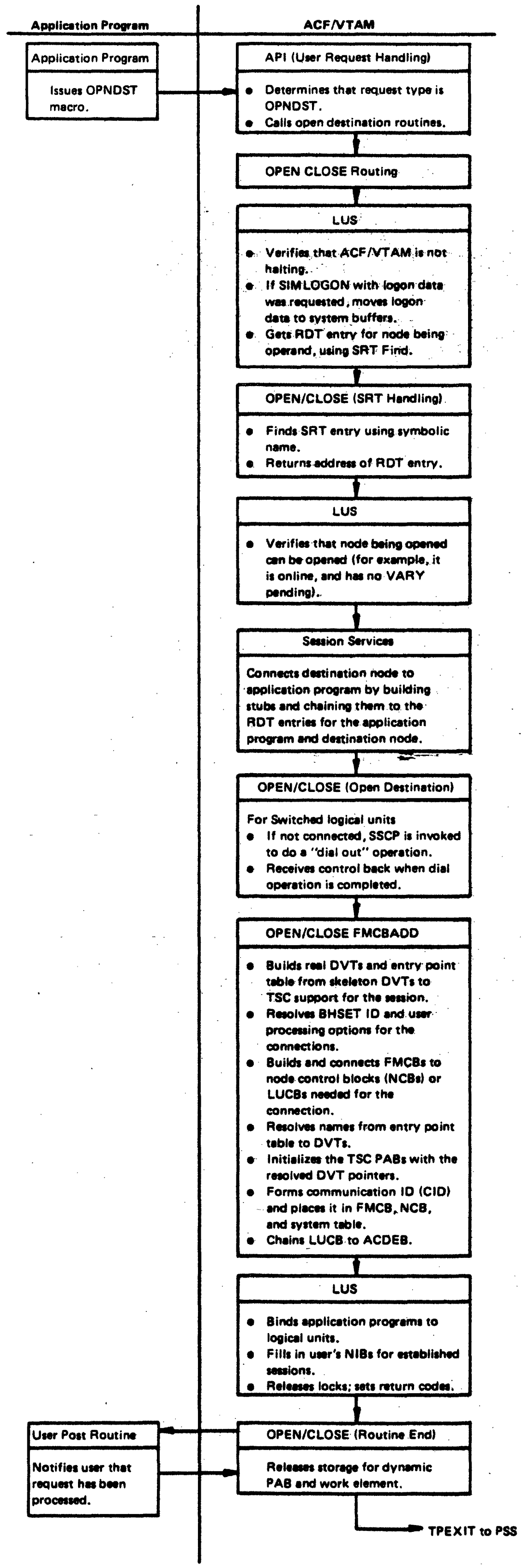
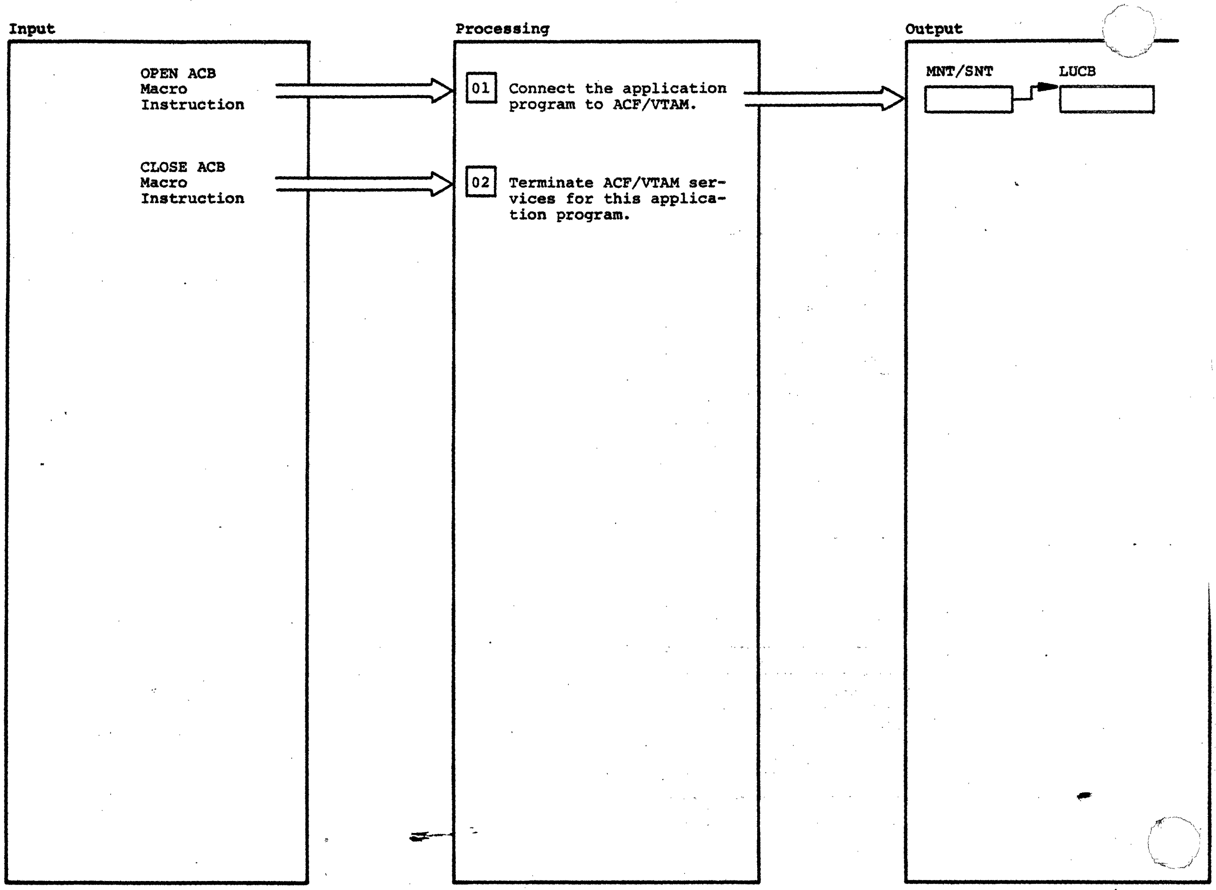


Figure 8-2. Processing an OPNDST Macro Instruction

MO 6. OPEN/CLOSE



Notes	Routine	Label	Ref.

Notes	Routine	Label	Ref.

CHAPTER 9. APPLICATION PROGRAM INTERFACE (API)

The collection of ACF/VTAM routines that initially handle Input/Output (I/O) requests from application programs is called the application program interface (API).

When the API receives a request from an application program (in the form of a macro instruction), it interprets the macro instruction and passes it (as an RPL) to TSC or OPEN/CLOSE. After the request has been transmitted to the destination node, the API receives feedback information from TSC or OPEN/CLOSE and checks that the operation has been completed successfully. The API then returns the feedback information to the application program.

An overview of API operation is shown in MO 7. API synchronous processing is shown in Figure 9-1. API asynchronous processing is shown in Figure 9-2.

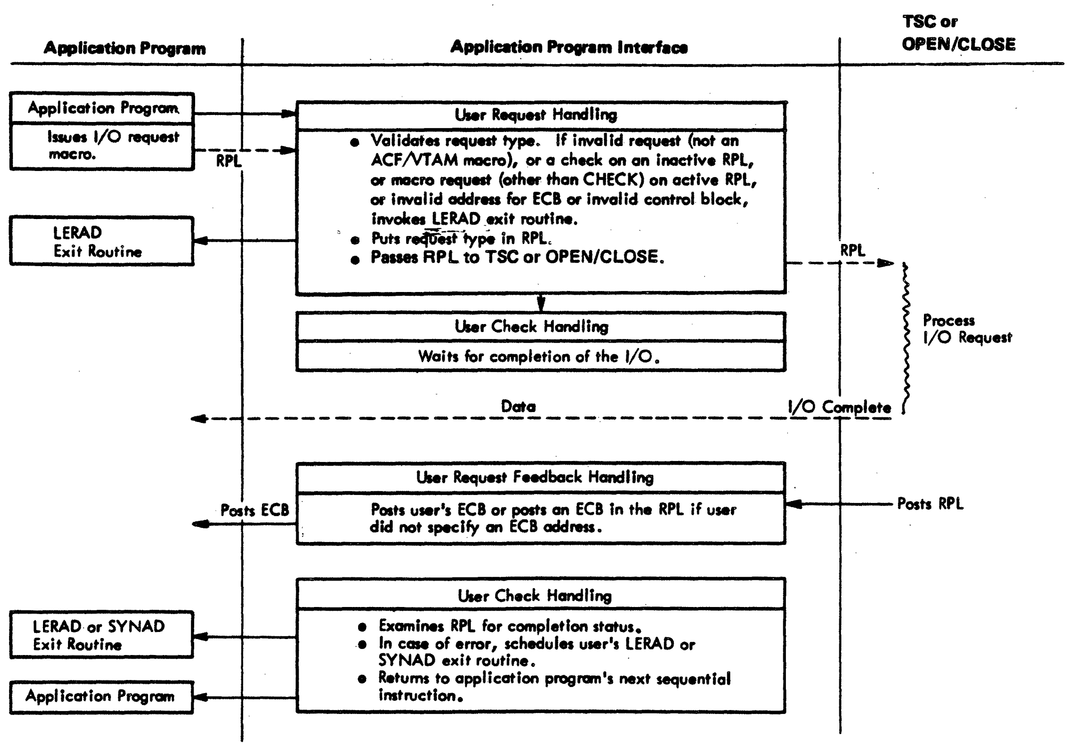


Figure 9-1. API Synchronous Processing

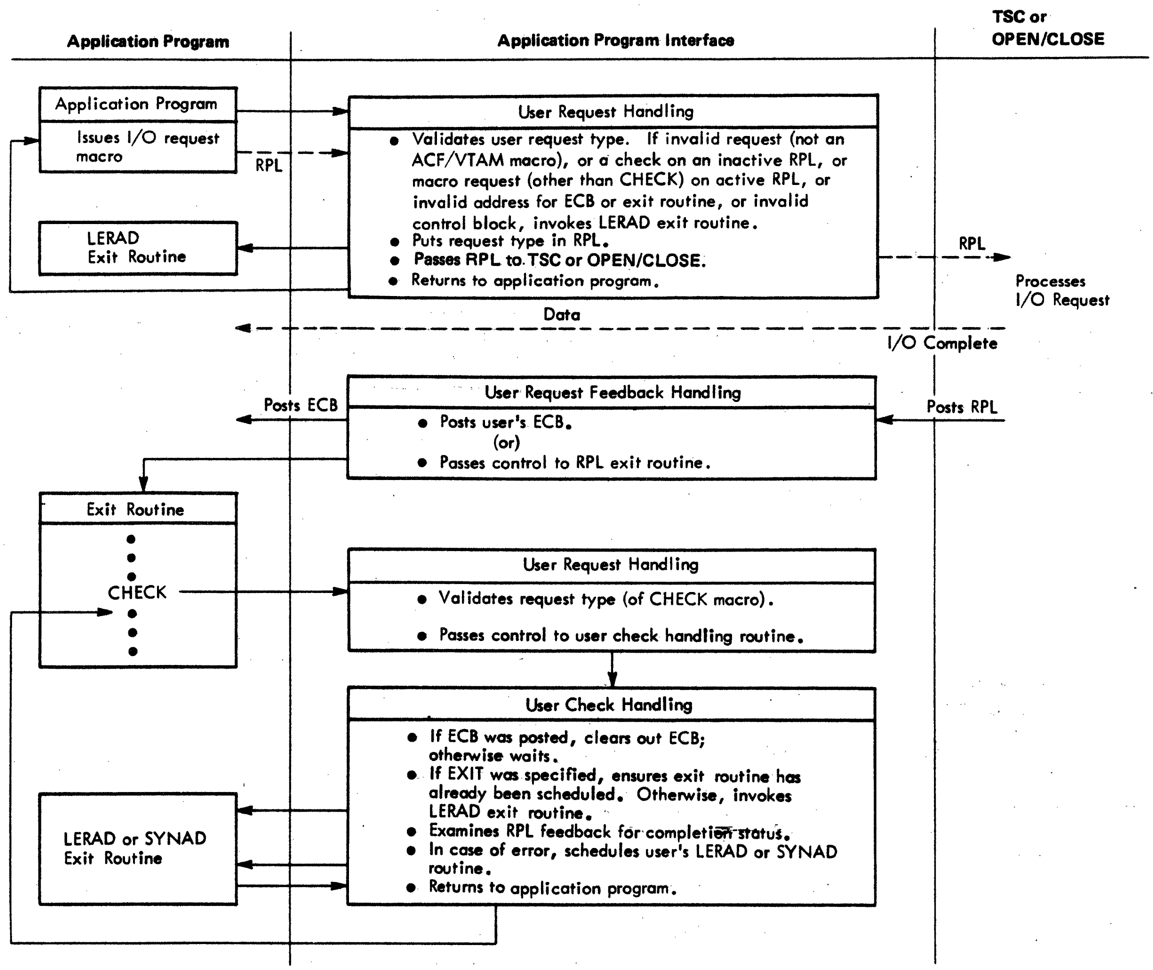


Figure 9-2. API Asynchronous Processing

EXIT ROUTINE FACILITIES

Exit routines are special interfaces between an application program and ACF/VTAM. These routines enable the application program to receive information from ACF/VTAM and respond to it. Exit routines do such things as receive logon requests (LOGON exit routine), notify an application program that contact with the other end of a session has been lost (LOSTERM exit routine), or notify the application program that ACF/VTAM's services are no longer available (TPEND exit routine). These exit routines are coded as part of the program they serve, but operate asynchronously; that is, they are driven by events that interrupt and occur in parallel with main line application programs. These routines enable an application program to order that a certain task be done and then continue with its own processing while the exit routine is handling the specific task.

EXLST EXIT ROUTINES

The application program can use the EXLST macro instruction to identify a variety of exit routines that ACF/VTAM schedules when particular events occur:

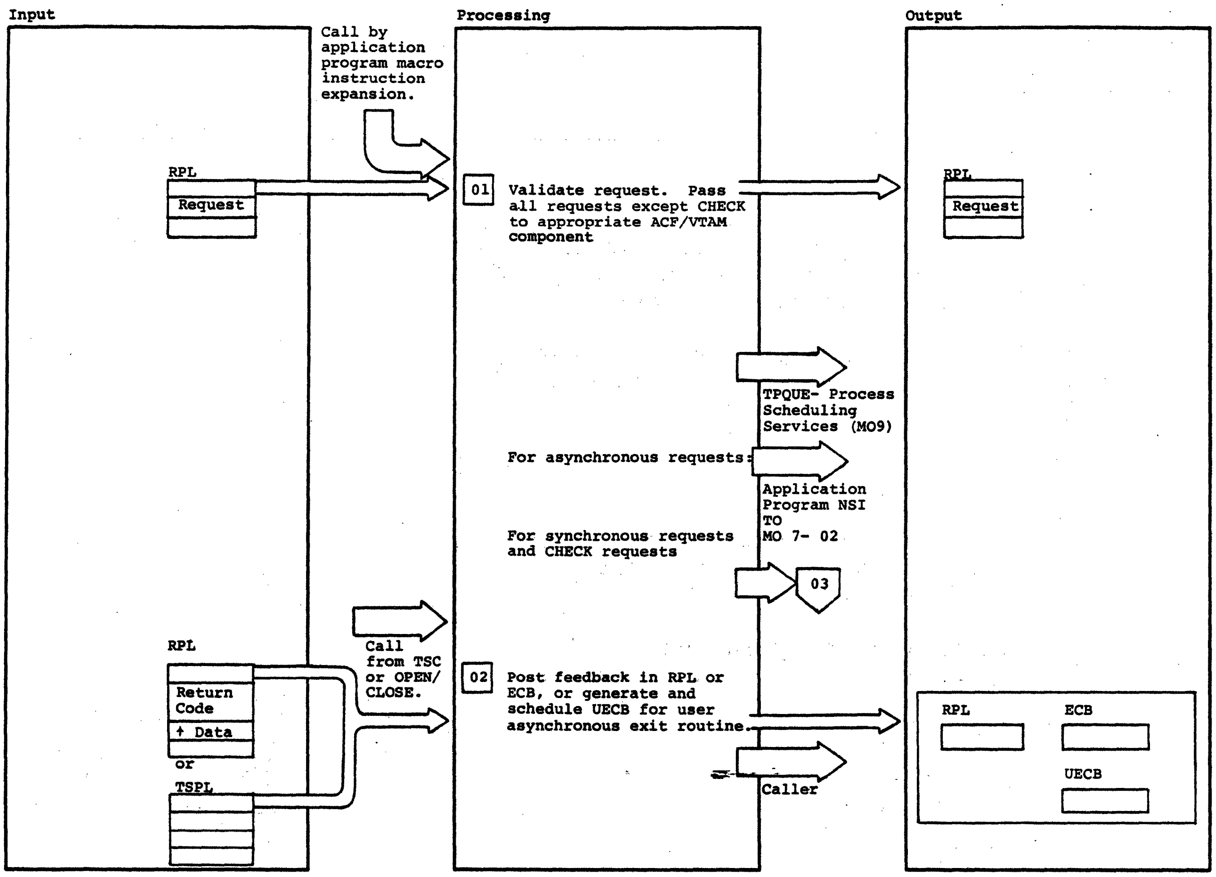
Event	Exit Routine
A logon has been received from or has been entered on behalf of a logical unit, and the logon is waiting to be processed.	LOGON
A session with a logical unit has been temporarily interrupted or permanently lost, the logical unit has requested that the session be terminated, or an event has occurred that might affect future operation of the session.	LOSTERM
A session with a logical unit has been broken because of a session outage, or a session that is partially established cannot be fully established.	NSEXIT
A logical unit already in session is wanted by another application program.	RELREQ
ACF/VTAM is being shut down.	TPEND
The application program has made an invalid request.	LERAD
A physical error has occurred during a session establishment or I/O operation.	SYNAD
A special type of input has arrived from a logical unit (the types of input are discussed later).	DFASY RESP SCIP

When one of these events occurs, the execution of the application program is interrupted, and the appropriate exit routine is given control. If another event occurs while the exit routine is being executed, the next exit routine is not invoked until the first has completed execution (this applies as well for RPL exit routines).

Unlike accounting, authorization, and logon-interpret exit routines, which are included as part of the system during ACF/VTAM definition, the EXLST exit routines are included as part of the application program. The addresses of these routines are placed in an EXLST control block by the application program.

Each exit routine is scheduled by changing the program information block (PIB) save area address. Any processing in the routine that places the routine in a wait state should be used with caution, because the application program's entire task waits while the exit routine waits. Exit routines other than the LERAD and SYNAD exit routines must be reentera-ble only if two or more application programs share the same exit routine.

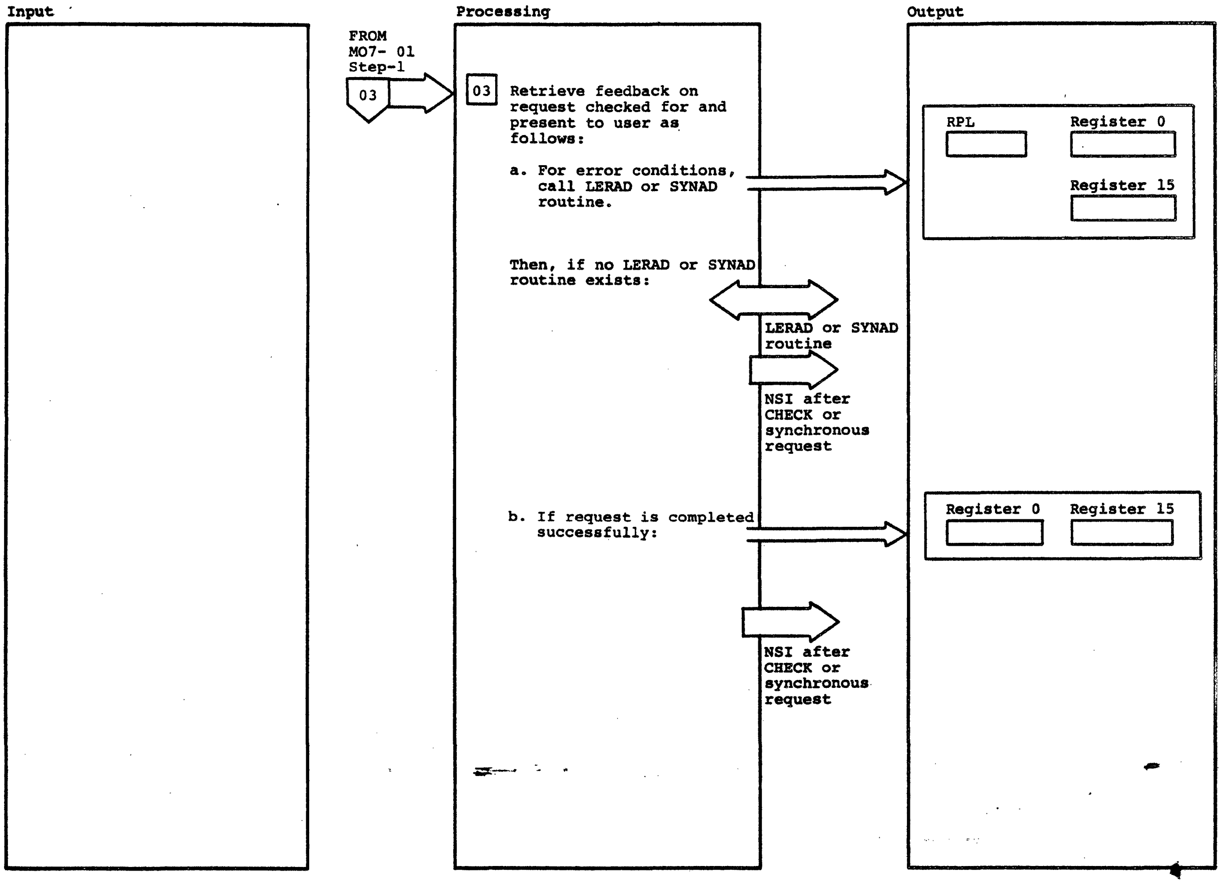
MO 7 (Part 1 of 2). Application Program Interface (API)



Notes	Routine	Label	Ref.
01 Valid input request codes are: x'02' CHECK (OS/VS only) x'14' CHECK (DOS/VSE only) x'15' SETLOGON x'16' SIMLOGON x'17' OPNDST x'1A' INQUIRE x'1B' INTERPRET x'1F' CLSDST x'22' SEND x'23' RECEIVE x'24' RESETSR x'25' SESSIONC x'27' SENDCMD x'28' REVCMD x'29' REQSESS x'2A' OPNSEC x'2C' TERMSESS			

Notes	Routine	Label	Ref.

MO 7 (Part 2 of 2). Application Program Interface (API)



Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.

CHAPTER 10. TRANSMISSION SUBSYSTEM COMPONENT (TSC)

The transmission subsystem component (TSC) provides a mechanism for information exchange between network addressable units (NAUs). The NAUs supported by ACF/VTAM are logical units, physical units, and the system services control point (SSCP). A request for data transfer can be passed to the TSC from the application program interface (API), LUS, the SSCP, PUS, or the system I/O supervisor (IOS). Data can be routed through the TSC to an application program (by the API or LUS), the SSCP, PUS, or a communication device (by IOS).

The TSC consists of nine functional groups, as follows:

1. ACF/VTAM environment routes control to the proper part of the TSC and performs TSC control block initialization.

2. Presentation services checks the validity of user requests and converts requests and responses from the form used by the TSC into the form used by outside components.

3. Function management data function interpreter builds message units suitable for the type of logical unit to which the message is destined. See Chapter 1.

4. Data flow control validates the data flow protocols.

5. Transmission control builds the transmission header of the path information unit (PIU), which controls the flow of data through the network, and does SNA state checks.

6. Path control selects the path and routes information (within TSC) to the next node in the path. It also does the boundary function.

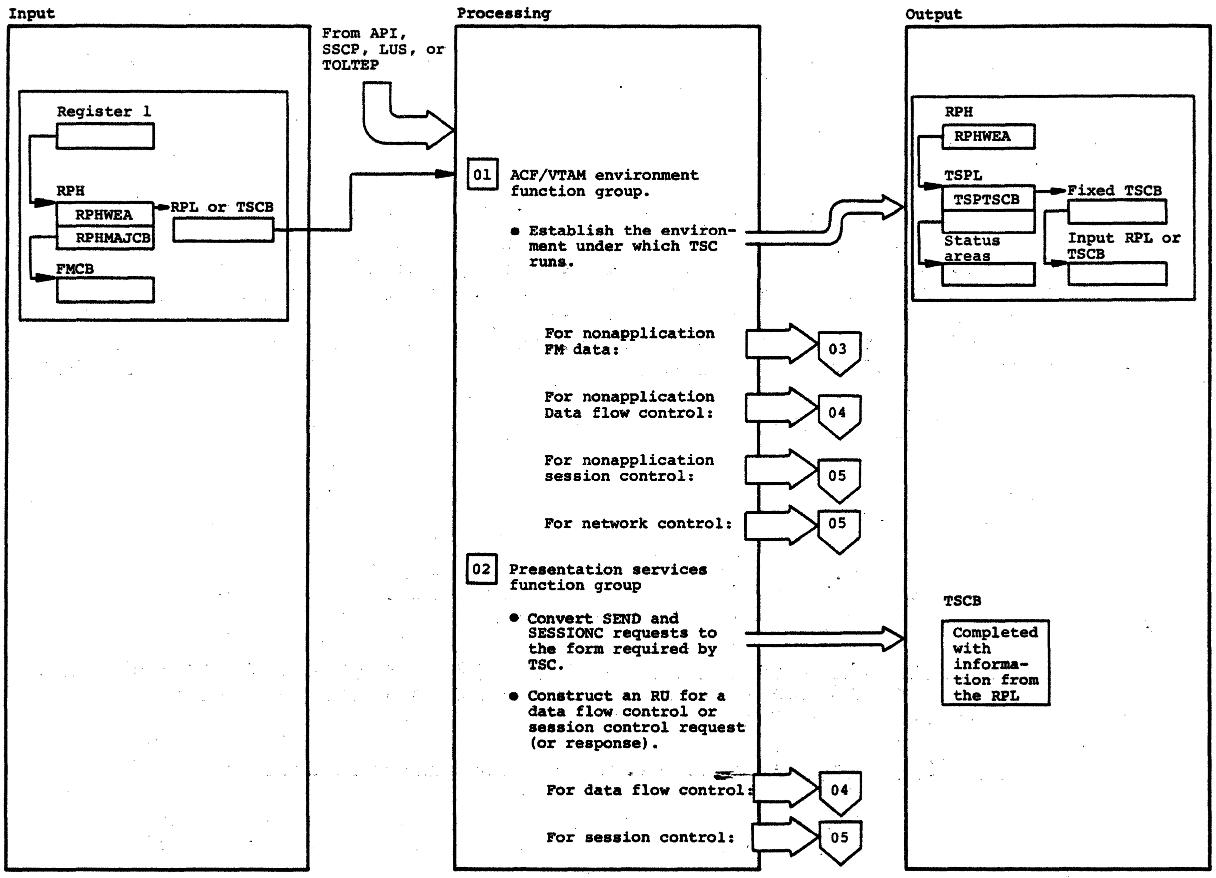
7. Boundary and transform functions establish sessions that flow through the boundary node, handles pacing between the primary logical unit and the boundary mode, handles pacing between the boundary mode and the secondary logical unit, maintains the state of all the sessions flowing through the boundary node, and provides support for non-SNA terminals.

8. Data link control interfaces with the system I/O supervisor, which controls the physical I/O paths to communication devices.

9. Presentation services converts input data to the format required by the other components, interfaces between TSC and the application program interface (API), and provides for transmitting data from TSC to the SSCP and TOLTEP.

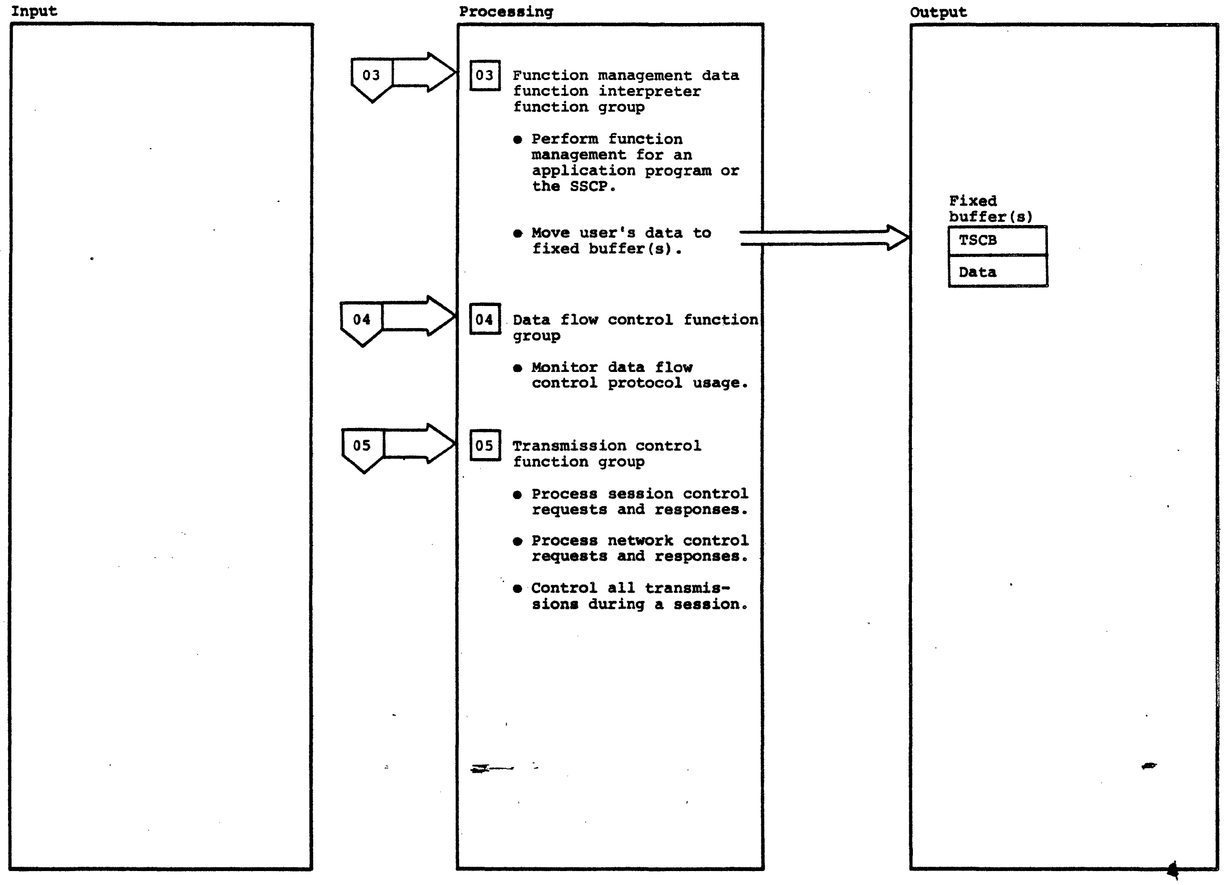
An overview of the operation of TSC is shown in MO 8.

MO 8 (Part 1 of 5). Transmission Subsystem Component (TSC)



Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.
01 The ACF/VTAM environment function group sets up the environment used within TSC. It establishes a standard TSC parameter list, and routes control to the proper section of TSC to process the particular request. Having its own parameter list allows TSC to remain independent of changes to the major ACF/VTAM I/O control blocks (like the FMCB). The TSC status areas are actually contained within the control block structure under which TSC is running. However, only the environment function group has to change if one of these control blocks is redefined.			8.1				
02 Input that originates from an application program is passed to TSC in an RPL. Input from other ACF/VTAM subcomponents is passed to TSC in a transmission subsystem control block (TSCB). The presentation services function group provides the interface between the TSC and the application program interface (API). This includes converting API requests (RPLs) into a format that is acceptable to the rest of the TSC (TSCBs).			8.2.1 8.2.2 8.2.3 8.2.4				

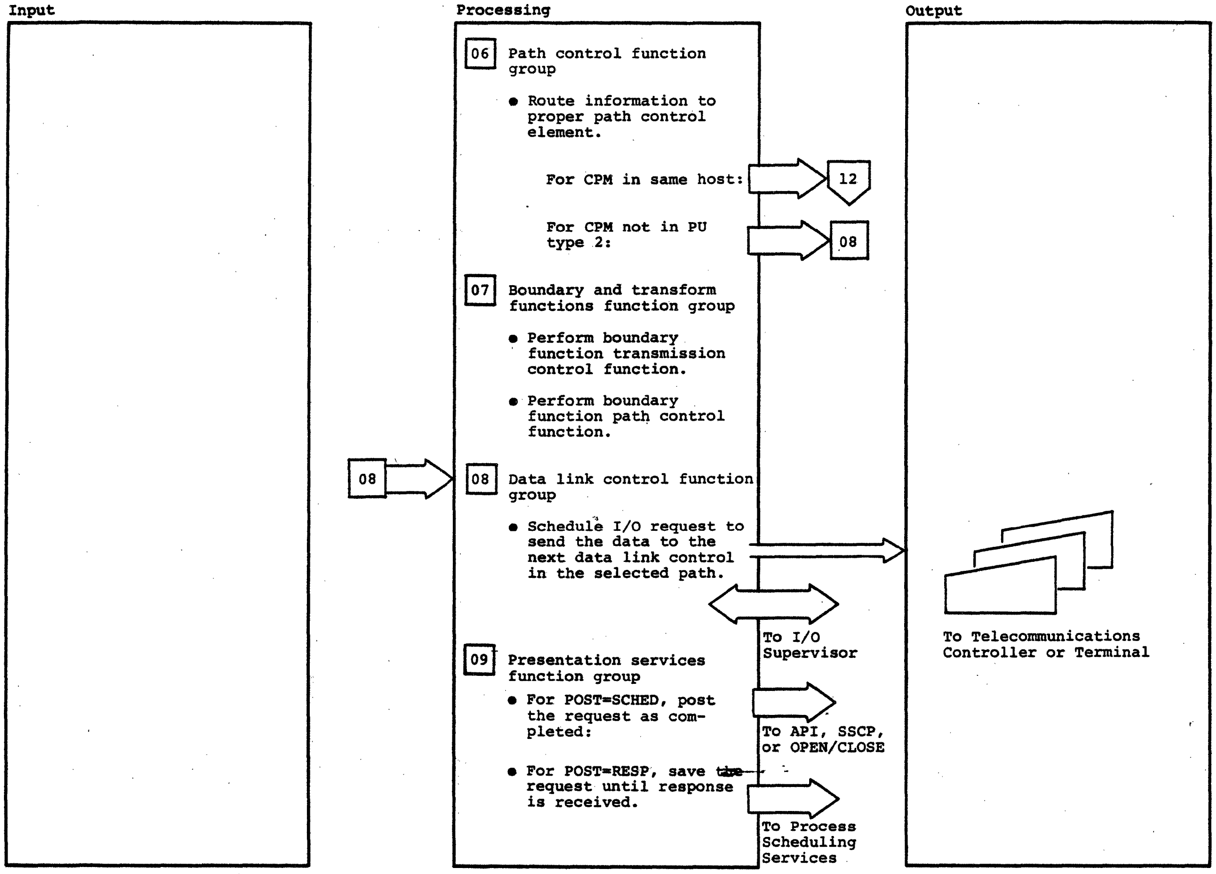
MO 8 (Part 2 of 5). Transmission Subsystem Component (TSC)



Notes	Routine	Label	Ref.
03 The FM data function interpreter function group provides function management for an application program. It also provides function management for the SSCP. This includes processing for selected network services requests.			8.3
04 The data flow control (DFC) function group provides the various data flow control protocols available to NAUs. The DFC function group monitors all information flowing from (and to) an application program.			8.4
05 The transmission control (TC) function group controls all data flowing through the telecommunications network. TC provides a set of well-defined data flows, control functions, and protocols which enable a network addressable unit (NAU) to communicate through the network to another NAU. A host application program NAU is capable of supporting multiple sessions with other NAUs in the network. TC uniquely identifies and maintains the integrity of each session using pairs of network addresses. One network address identifies each NAU; a pair of network addresses, therefore, uniquely identifies			8.5

Notes	Routine	Label	Ref.
the session that is established between the two NAUs. Transmission control consists of three sections: • Session control provides the NAU with functions for controlling the operation of its sessions. • Network control provides transmission control and path control with administrative functions. • The connection point manager (CPM) controls the transmission of requests and responses during the session, including those generated by session control and network control.			

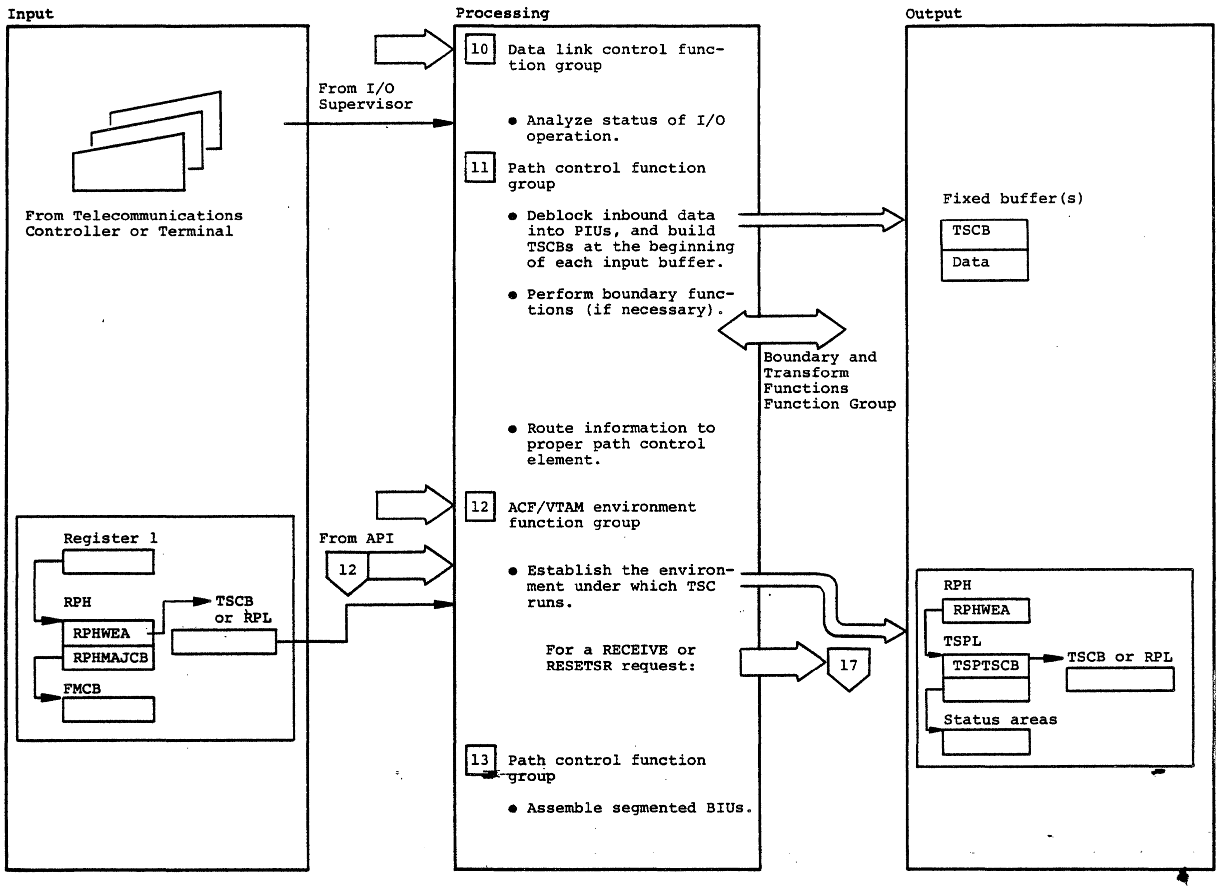
MO 8 (Part 3 of 5). Transmission Subsystem Component (TSC)



Notes	Routine	Label	Ref.
06 The path control (PC) function group routes information from a sending connection point manager (CPM) to a receiving CPM. Path control selects the communications path, and all communication between CPMs is kept path-independent. Path control allows communications between CPMs that reside in the same host (application program to application program), between CPMs that reside in the same domain (application program to logical unit in the same domain), and between CPMs that reside in different domains (application program to logical unit or application program in a different domain). The path selected by path control may consist of several interconnecting links and associated nodes. The transmission header (TH) is used to pass control information from the sending path control to the receiving path control, and to all intermediate path controls that are in the selected path.			8.6
07 The boundary and transform functions function group performs the boundary functions for a locally attached cluster controller (PU type 2).			8.8

Notes	Routine	Label	Ref.
Boundary function transmission control consists of the following sections: • Boundary function common session control (BF.CSC) establishes sessions that flow through the boundary node. • Boundary function connection point manager (BF.CPM) handles pacing between the PLU and the boundary node, and between the boundary node and the SLU. • Boundary function session control function interpreter (BF.SC.FI) maintains the state of all sessions flowing through the boundary node. This function group also provides support for non-SNA display terminals by converting all outbound SNA requests into the appropriate series of FIDO BTUs.			
08 The data link control (DLC) function group manages the communications links. These include the data links to locally attached communications controllers, locally attached cluster controllers, and locally attached display terminals. DLC provides the interface between the host operating system I/O supervisor and path control. It constructs and schedules channel programs.			8.7.1 8.7.2

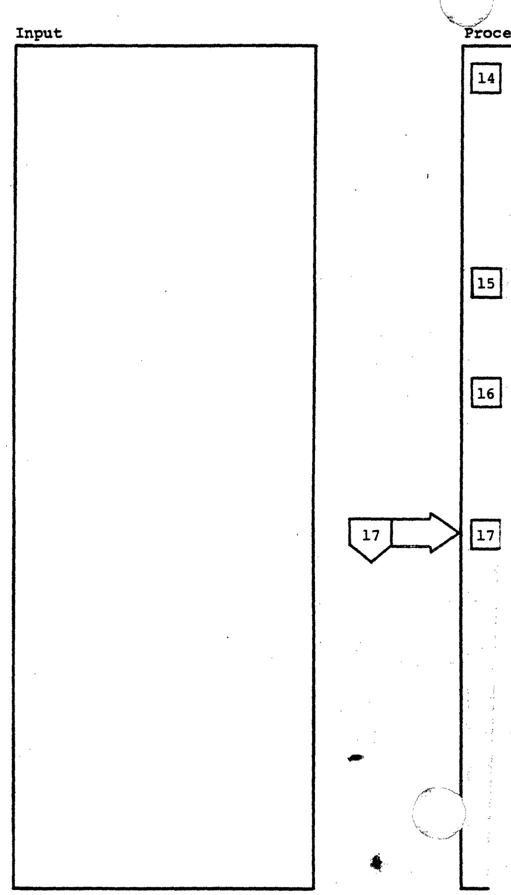
MO 8 (Part 4 of 5). Transmission Subsystem Component (TSC)



Notes	Routine	Label	Ref.
10 The data link control (DLC) function group manages the communications links. These include the data links to locally attached communications controllers, locally attached cluster controllers, and locally attached display terminals. DLC provides the interface between the host operating system I/O supervisor and path control. It analyzes the status whenever a channel program ends.			8.7.1 8.7.2
11 The path control (PC) function group routes information from a sending connection point manager (CPM) to a receiving CPM. Path control selects the communications path and all communication between CPMs is kept path-independent.			8.6

Notes	Routine	Label	Ref.

MO 8 (Part 5 of 5). Transmission Sub[illegible]ten



Notes	Routine
14 Transmission control consists of three sections: • Session control provides the NAU with functions for controlling the operation of its sessions. • Network control provides transmission control and path control with administrative functions. • The connection point manager (CPM) controls the transmission of requests and responses during the session, including those generated by session control and network control. 15 The data flow control (DFC) function group provides the various data flow control protocols available to NAUs. The DFC function group monitors all information flowing from (and to) an application program. 16 The FM data function interpreter function group provides function management for an application program. It also provides function management for the SSCP. This includes processing for selected network services requests.	

CHAPTER 11. PROCESS SCHEDULING SERVICES (PSS)

A set of routines called process scheduling services (PSS) provides queuing, scheduling, and dispatching services to allow ACF/VTAM processes to control the flow of requests. They also provide locking services and establish a recovery environment.

An overview of PSS operation is shown in MO 9.

PSS MACRO INSTRUCTIONS

A set of macro instructions is used to invoke and control PSS operations. These macro instructions are used only in ACF/VTAM routines; they are not available for use in application programs. The PSS macro instructions are:

- TPQUE: Adds a work element to a queue or a PAB and optionally schedules the process represented by the PAB to be dispatched.
- TPDEQ: Removes a work element from a queue or a PAB.
- TPSCHED: Schedules for execution a process represented by a PAB.
- TPEXIT: Indicates that a process has terminated and the PAB can be rescheduled.
- TPESC: Transfers control to the next routine indicated by the DVT.
- TPWAIT: Suspends execution of the issuing process.
- TPPOST: Schedules a process to resume execution at the instruction after the TPWAIT.
- TPLOCK: Obtains shared or exclusive ownership of a lock or queues the request if a lock cannot be obtained immediately.
- TPUNLOCK: Releases ownership of a lock and, if the lock is now free, processes any queued lock requests.
- TPDVTS: Changes the list of processes located by the DVT.
- TPFEL: Locates the oldest work element on a queue.
- ACTAP: Schedules PSS to receive control asynchronously.
- APSINIT: Allocates and initializes PSS control blocks.
- APSTERM: Releases PSS control blocks when they are no longer needed.
- TPIO: Used by API to inform ACF/VTAM of an application program request.
- TPSTAE: Creates, changes, or removes a STAE recovery environment (OS/VS1).

APPLICATION PROGRAM INTERFACE WITH PSS

Since ACF/VTAM routines run in supervisor state and under the ACF/VTAM protection key, an SVC interface is required for the application program interface to initiate a request. This SVC is made system-independent by its inclusion within the expansion of the TPIO macro instruction.

The TPIO macro instruction causes an SVC state and protection key change. Within the SVC routine, a TPQUE and TPSCHED are executed to initiate the processing required for the request.

EXECUTION SEQUENCE CONTROL

The execution sequence control routine provides linkage to a set of sequential processes. These processes are pointed to by entries in the destination vector table (DVT). Each DVT has a name that is located in the DVT entry point table (EPT). A DVT entry represents a routine, and many DVT entries make up a level of the system. The entry point table and corresponding DVT entries are constructed during the binding process from a skeleton DVT, and they determine the sequence in which routines process requests flowing through ACF/VTAM.

Functions of Execution Sequence Control

The operation of the execution sequence control routine is governed either by parameters received when it is called or by information in the entry point table or in the real DVT.

The execution sequence control routine transfers control in one of two ways, depending on the type of call used to enter the sequence control routine:

Next Process: Control is given to the routine whose address is in the DVT entry pointed to by the RPH. The pointer in the RPH is then updated to indicate the next DVT entry, that is, the entry for the next routine that is to be executed in processing this DVT.

Named DVT: When entry is made with this type of call, the name passed is resolved into a DVT pointer through the entry point table. Control is then passed to the first entry in that DVT. A pointer to the next entry in the DVT is saved in the RPH.

Process Scheduling and Dispatching

PSS routines schedule the execution of other ACF/VTAM routines that are pointed to by the DVT, thus enabling ACF/VTAM to schedule its operation independently of the operating sytem dispatcher. See "Scheduling an ACF/VTAM Process" in Chapter 2.

PSS scheduling and dispatching routines are system-dependent. For a description of their operation in each system, see ACF/VTAM Logic.

ACF/VTAM LOCKS

Since ACF/VTAM offers a high degree of parallel and asynchronous operation, special attention must be paid to serializing ACF/VTAM's internal use of shared resources. The mechanism used to accomplish this serialization, the lock structure, is an internal function and is not visible to the user. The locking method used is to set locks on all the code that causes the updates (rather than setting locks individually on each resource to be updated). This method prevents lockout when waiting for several resources (that is, a task causing itself to be locked out of a resource while it waits for it). For a description of the locations and levels of ACF/VTAM locks, see ACF/VTAM Diagnostics.

Classes of Lock Holders

There are two classes of lock holders:

Shared: When a lock is obtained as shared, the obtainer can only read the resource. Any number of programs can hold a shared lock on a resource at the same time.

Exclusive: When a lock is obtained as exclusive, no other program can obtain the lock (either shared or exclusive) on the resource until the exclusive lock holder releases it. To change or delete a resource, a program must hold an exclusive lock on it.

Access to Locks

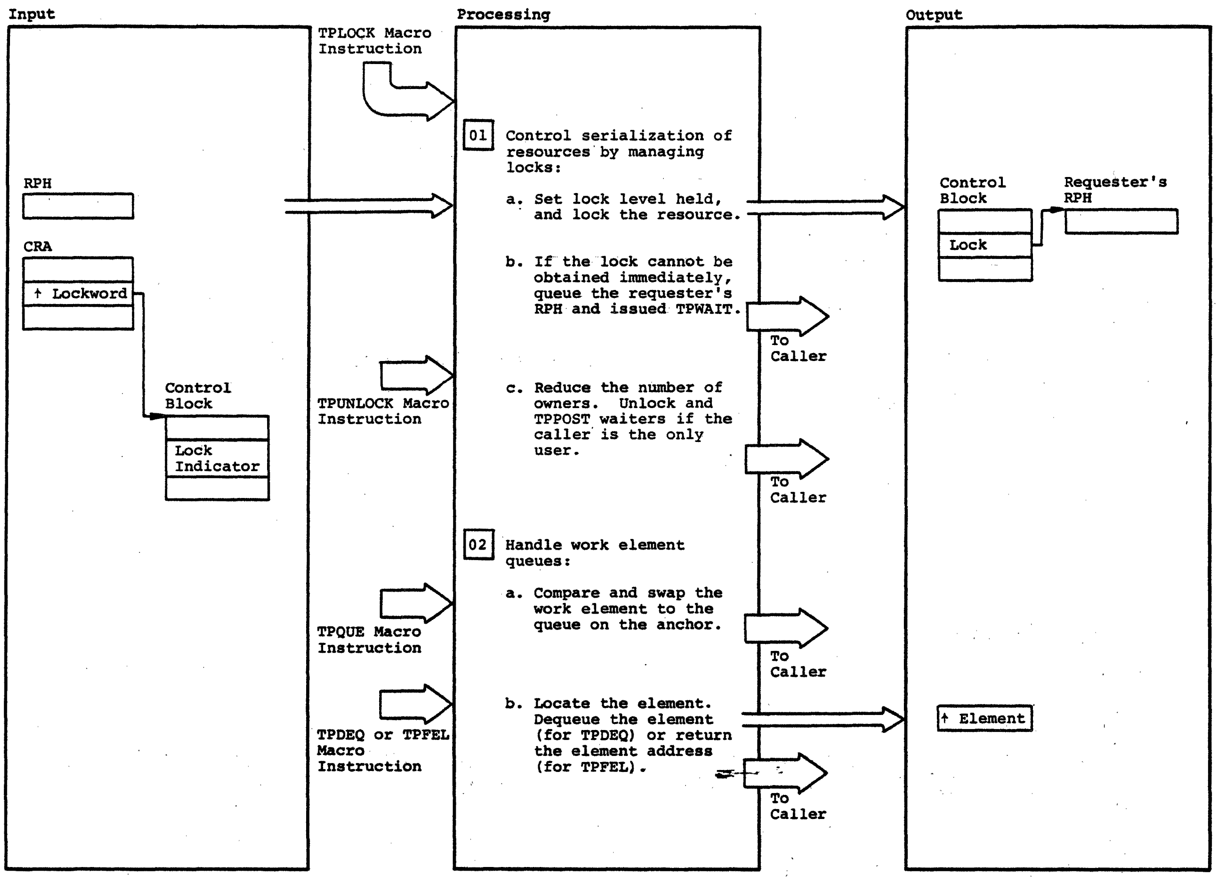
Access is controlled as follows:

- A request to share a lock that is free or already shared (with no outstanding exclusive requests) is honored immediately.
- A request for exclusive control of a lock that is already held is queued until all programs sharing the lock release it. The request is then honored.
- Any request for a lock that is held as exclusive or has an exclusive request outstanding is queued until the exclusive use is completed.
- Wait requests are handled in a first-in, first-out order.

Locking Hierarchies

All ACF/VTAM locks are allocated according to a predetermined hierarchy. This means that a program holding one lock cannot obtain a lock that is higher in the hierarchy without first releasing the lower one. Strict enforcement of this hierarchy prevents any possibility of a deadlock, in which two programs each hold a lock and are queued for the lock held by the other program.

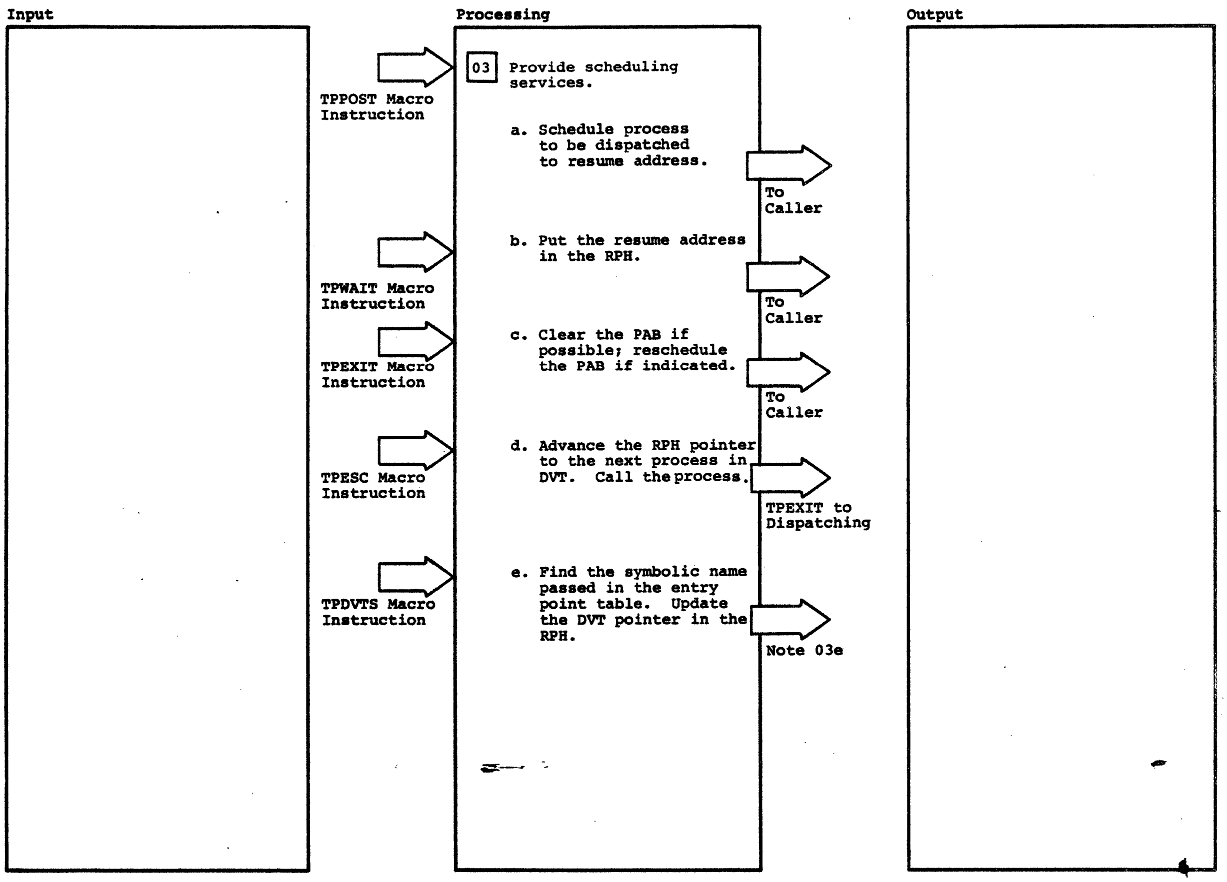
MO 9 (Part 1 of 3). Process Scheduling Services (PSS)



Notes	Routine	Label	Ref.

Notes	Routine	Label	Ref.

MO 9 (Part 2 of 3). Process Scheduling Services (PSS)



Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.
03 e. If the TPDVTS macro instruction specifies "no return," a TPESC macro instruction is issued to advance the RPH pointer to the next process in the DVT. Otherwise, return is made to the caller.							

MO 9 (Part 3 of 3). Process Scheduling Services (PSS)

Input		Processing		Output
	TPIO Macro Instruction	**04** Schedule asynchronous processes. a. Locate the PAB address. Issue an SVC to perform the processing.	Next Sequential Instruction	
	TPQUE Macro Instruction	b. Compare and swap the work element to the PAB. If reset, queue the work element to the reset queue.	To Caller	
	TPSCHED Macro Instruction	c. Schedule the PAB to be dispatched at the first routine pointed to by the DVT.	To Caller	
	From System Dispatcher	**05** Dispatch process: Select work to dispatch; dispatch the PAB.	See Note 05	

Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.
05 Work to dispatch can be: • Posted Process • Scheduled PAB • User exit For PABs, locate an RPH to use. If continuing after TPEXIIT, reuse the RPH. The next choice is the preallocated RPH in the APT or PST. If neither is available, issue REQSTORE for LPBUF. User exit address must be validated; it comes from the RPL or ACB. Return from the dispatched process is by TPEXIT or TPWAIT for ACF/VTAM processes, or on register 14 for user exits.							

CHAPTER 12. STORAGE MANAGEMENT SERVICES (SMS)

Storage management services provide storage services to all ACF/VTAM components by allocating or deallocating buffers. These services are provided only for ACF/VTAM components; they are not available to application programs.

An overview of storage management services is shown in MO 10.

To obtain the services, ACF/VTAM components use internal macro instructions (described below). The macro instructions and return codes are the same for all components. In addition, the macro and return codes are the same for components in the DOS/VSE and OS/VS versions of ACF/VTAM. The interface to ACF/VTAM routines is common across all operating systems; however, the internal coding of the storage management service routines is different because of differences in the system. For example, in DOS/VSE, buffer pools are constructed during ACF/VTAM initialization by issuing the GETVIS macro; in OS/VS, the GETMAIN macro is issued to get storage.

Storage management services provide all ACF/VTAM components with fixed-length buffer pools. Separate pools are maintained for fixed-storage and pageable-storage buffers. In addition to the fixed-length buffer services, the GETSTOR or VTALLOC and FREESTOR or VTFREE macro instructions provide a limited "contiguous storage area" service.

A buffer-pool directory, pointed to by the ATCVT, contains an entry for each pool. The entry points to a chain of the buffers assigned to that pool and indicates whether the buffers are available for allocation.

EXPANDING AND CONTRACTING BUFFER POOLS

Each ACF/VTAM fixed-length buffer pool can be expanded when the demand for buffers from the pool exceeds the current supply of available buffers. Conversely, the expanded portions of an ACF/VTAM buffer pool are freed when the extra buffers are no longer needed.

Dynamic expansion of buffer pools can be controlled by ACF/VTAM start options. These start options can specify an expansion threshold parameter and an expansion size parameter. The pool expansion threshold parameter is used to determine when to expand a buffer pool. When the number of available buffers in a pool reaches or drops below its expansion threshold value, the pool is scheduled for expansion. The expansion size parameter indicates the number of buffers by which a pool is to be increased whenever expansion is necessary. Space for pool expansions is obtained in page-size increments, the amount obtained being the number of pages needed to supply the specified number of buffers. Storage management services calculates a contraction threshold value for each buffer pool. When the number of available buffers in a pool reaches or exceeds this threshold, extents that are not in use are scheduled to be freed. The contraction threshold value equals the expansion threshold value, plus the number of buffers in two extents.

STORAGE MANAGEMENT MACRO INSTRUCTIONS

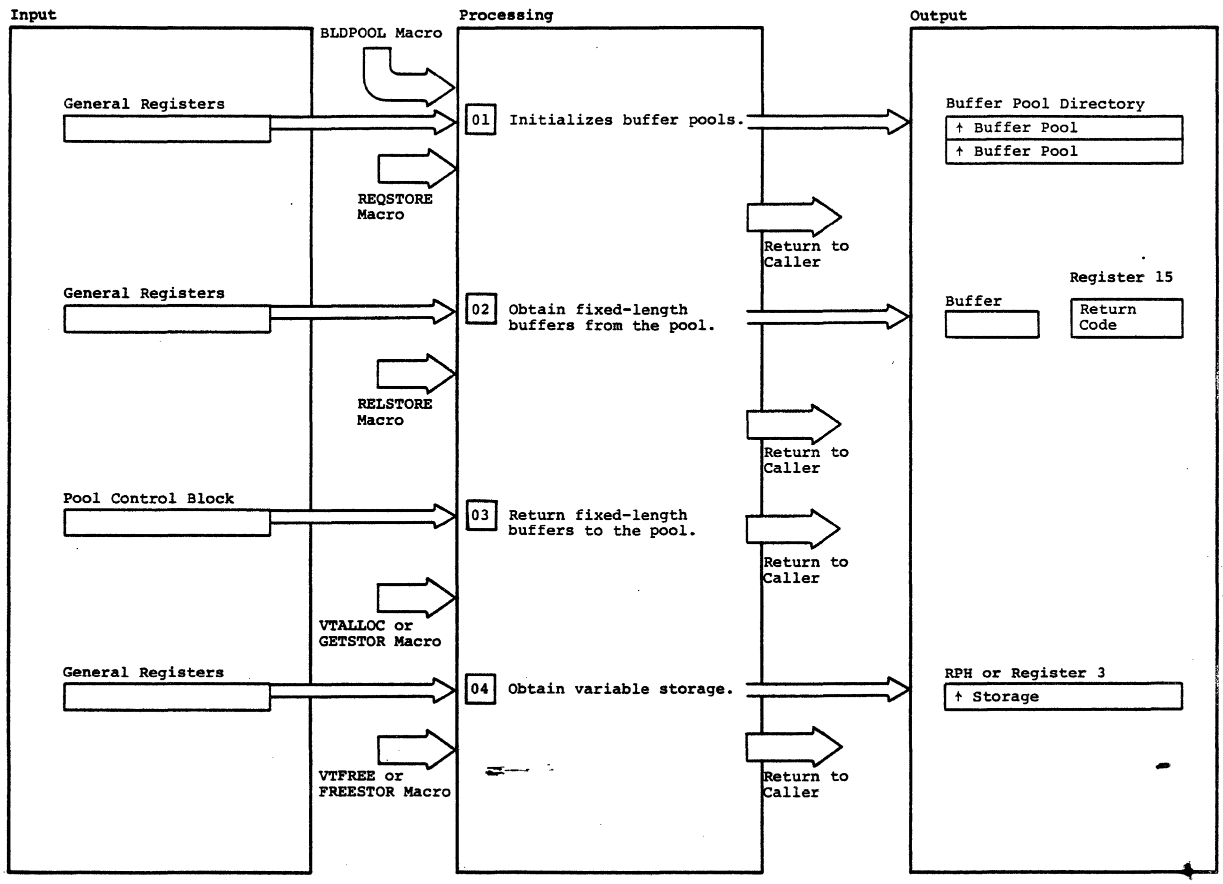
There are two pool service macro instructions: BLDPOOL and DELPOOL. BLDPOOL enables an ACF/VTAM routine to build a buffer pool, while the

DELPOOL macro instruction enables a routine to delete it. Because only shared buffer pools (in which space is allocated for more than one type of buffer) are supported, these macro instructions are issued only in the initialization and termination routines. One BLDPOOL macro instruction is included for each shared buffer pool in the initialization routine, and one corresponding DELPOOL macro instruction is included in the termination routine.

Buffer service provides two macro instructions: REQSTORE and RELSTORE. These macro instructions enable ACF/VTAM routines to request or release fixed-length buffers from or to predefined buffer pools.

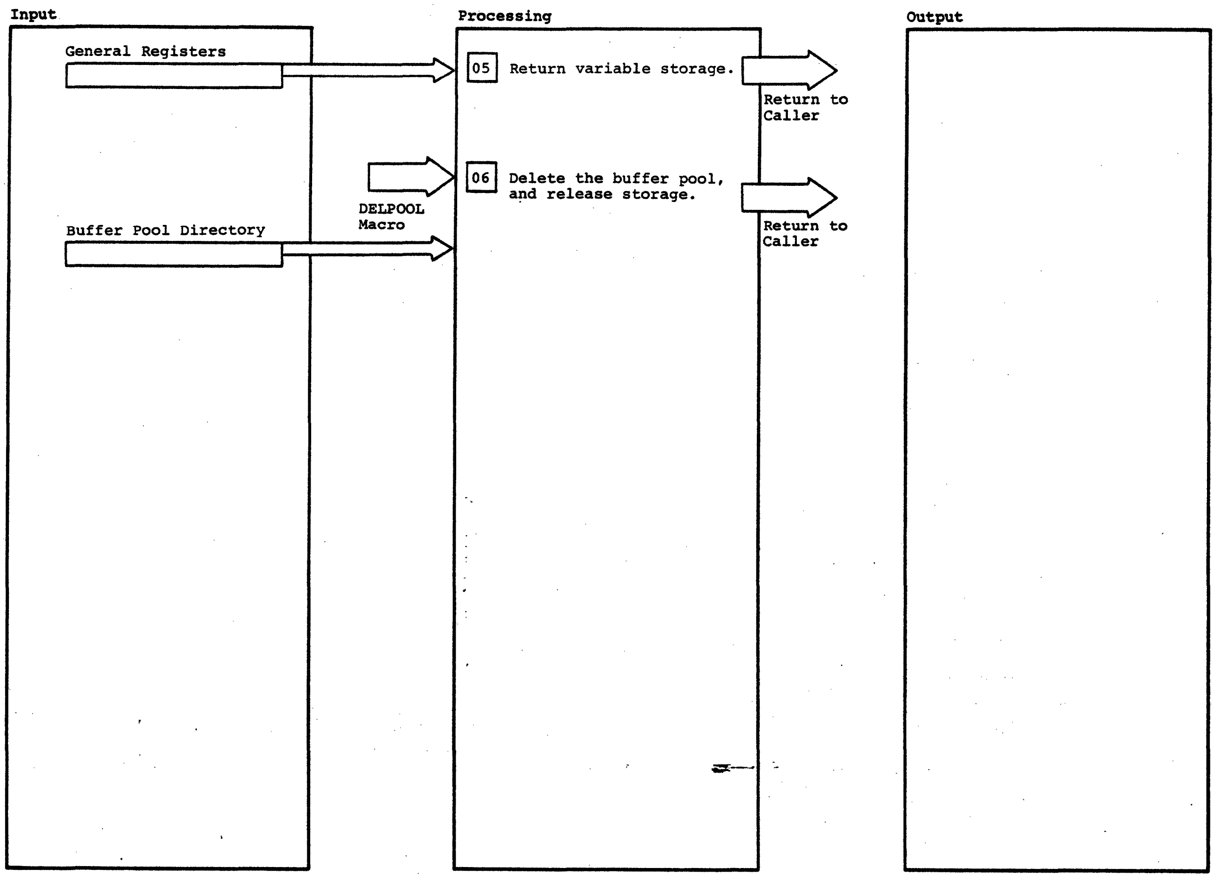
Variable-length storage areas can be acquired and released with the GETSTOR or VTALLOC and FREESTOR or VTFREE macro instructions.

MO 10 (Part 1 of 2). Storage Management Services (SMS)



Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.

MO 10 (Part 2 of 2). Storage Management Services (SMS)



Notes	Routine	Label	Ref.

Notes	Routine	Label	Ref.

CHAPTER 13. RELIABILITY, AVAILABILITY, AND SERVICEABILITY

ACF/VTAM has two types of reliability, availability, and serviceability (RAS) aids, which enhance the usability of ACF/VTAM:

- Serviceability aids, which assist in problem determination
- Reliability and availability aids, which enhance ACF/VTAM's operation and performance

An overview of the operation of RAS routines is shown in MO 11.

SERVICEABILITY AIDS

Internal Trace

The internal trace keeps a record of ACF/VTAM resource usage and flow of control. The trace is kept in the form of predefined 32-byte records that can be read in a dump of ACF/VTAM storage or, in OS/VS, in the output of the system general trace facility.

Tracing Facility

In addition to the internal trace, ACF/VTAM provides internal traces of buffers, buffer pools, I/O operations, and lines. Information on how to use these traces in each operating system is given in *ACF/VTAM Diagnostics*.

ACF/VTAM provides four types of traces: (1) buffer (specified in the start procedure and in MODIFY operator commands as BUF), (2) line (specified as LINE), (3) I/O (IO), and (4) buffer pool usage (SMS). This function takes place in three stages:

- Establishing and maintaining a trace environment in ACF/VTAM in response to operator commands.
- Generating and writing trace records. In OS/VS, this is done in conjunction with the system general trace facility.
- Formatting and printing trace information. In OS/VS, ACF/VTAM provides formatting for the system HMDPRDMP service. In DOS/VSE, ACF/VTAM both formats and prints the trace information.

Buffer Trace: This trace provides a trace of buffer contents for any of these types of addressable nodes: terminal component, logical unit (which can be a terminal), physical unit (cluster control unit), or NCP. A buffer trace is a means of identifying all changes to data caused by ACF/VTAM. As a message coming from or going to an application program enters ACF/VTAM control, it is placed in a trace record and written to auxiliary storage. Included in the record are the node name of the application program, the resource ID of the logical unit, and message data.

NCP Line Trace: Operating parameters of a line are traced by the NCP line trace whenever a specified interruption level occurs. One line can be traced at a time. If more than one line trace request is made, additional requests are rejected until the preceding trace has ended.

I/O Trace: This trace collects information for a local or remote NCP or for remote devices attached to the NCP. The data collected is variable; it depends on the type of node being traced.

Buffer Usage Trace: The ACF/VTAM buffer usage trace aids the user in determining a reasonable size for each of the 11 buffer pools. Periodically, a trace record is created that identifies, for each pool, the number of elements in the pool, the number of currently unallocated elements, the maximum number of elements that were concurrently in use since the last trace record was written, a maximum number of concurrently queued requests for buffers since the last trace record was written, the maximum number of elements in the pool since the last trace record was written, and the number of times the pool was expanded since the last trace record was written.

Formatted Dump

ACF/VTAM's formatted dump interacts with two different OS/VS components: ABDUMP and PRDMP. ABDUMP operates as part of the operating system's abnormal termination (ABEND) procedure; ABDUMP selects the storage to be dumped and dumps it. PRDMP formats and prints the dump.

Teleprocessing Online Test Executive Program (TOLTEP)

TOLTEP controls the selection, loading, and execution of online tests (OLTs) within an ACF/VTAM system. An OLT is a specific device test designed to help diagnose a hardware problem or to verify the usability of a control unit or terminal. For more information about TOLTEP, see Chapter 18.

Hardware Error Recording

Error recording is a part of the error recovery procedures of the configuration services subcomponent. The error data is placed in the system error record file (SYSREC) for subsequent editing and printing. Error recording takes place under these conditions:

- Permanent error: Errors that are either unrecoverable or errors from which ACF/VTAM has failed to recover
- Counter overflow: Written whenever the SIO (Start I/O) counter, the temporary error counter, or one of the device statistics table counters is about to overflow
- End-of-day: Written whenever the terminal or line is brought offline (with a VARY INACT command)

Patch Area

A patch area is supplied as a separate phase (DOS/VSE) or load module (OS/VS). This area is used to insert short service programs or fixes. This patch area is a separate ACF/VTAM module whose size can be varied by the user with the use of link-edit facilities.

Mapping of CSECTs, Control Blocks, and Buffers

Each ACF/VTAM CSECT contains information indicating the CSECT's name and its change level. The first byte of certain control blocks and buffers contains an identifier that is set when the control block or buffer is initialized.

RELIABILITY AND AVAILABILITY AIDS

Hardware Error Recovery Procedures

When an I/O error interruption occurs, device-dependent and control-unit-dependent error recovery procedures (ERPs) are invoked. The ERPs examine the channel status word and the sense data to determine the type of error and the action required to attempt recovery from the error. If the ERP cannot correct the error (that is, if the error is permanent), the ERP sends a message to the operator and records the error in SYSREC. If the ERP corrects the error (that is, if the error is temporary), no operator message is issued, and the temporary error counter is incremented.

Software Error Handling in OS/VS

ACF/VTAM handles three types of software errors:

- ACF/VTAM errors that occcur during application program processing
- ACF/VTAM errors that occur during task processing
- Errors resulting from application program requests for ACF/VTAM services

For these types of errors, ACF/VTAM performs cleanup and recovery processing, dumps and records error condition information, and makes the appropriate entries in the ACF/VTAM audit trail.

Process Recovery and Abend handling (OS/VS)

These facilities handle the initial recovery on an abnormal termination and dispatch recovery routines. If recovery is not attempted, or if it fails, abend handling ensures that resources allocated to ACF/VTAM are cleaned up.

Commands and Messages

ACF/VTAM commands enable the operator to define the network's configuration and display its status. By using the commands, the operator can activate and deactivate nodes and change their attributes while the system is running. The command processing routines are supported by error messages and warning messages that notify the operator of the need for reconfiguration and report problems that occur during reconfiguration.

Testing the Logical Unit

The logical unit may be tested by either the Echo Test or the Link Level 2 Test.

Echo Test: A logical unit to SSCP Echo Test allows a logical unit to test the connection between itself and the SSCP. Echo Test uses the Unformatted System Services (USS) function of the SSCP and provides the test for all logical units that support USS within the SSCP's domain.

Echo test is initiated by the terminal operator and assures the operator that the communication link to the host system's SSCP is operable; however, the terminal's path to an application program in a host system is not verified by Echo Test because of the possibility of a different path's being used for logical unit to application program sessions.

Link Level 2 Test: The Link Level 2 Test provides the capability of testing a single or multiple secondary terminal and, at the same time, allowing other sessions on the same link to continue operations. The secondary terminal under test is dedicated to the test and is not available to network users until the test has been terminated.

Error Detection and Indication

All requests from the network operator and application programs are initially tested for errors. If an error is detected when the request is made, ACF/VTAM notifies the requester and rejects the request. Any processing done on behalf of the rejected request is cleaned up. Later, if an error condition is encountered during the processing of a request, ACF/VTAM notifies the requester and stops processing the request. If possible, ACF/VTAM cleans up the processing that was done on behalf of the request.

Storage Reliability

All ACF/VTAM routines, buffers, and control blocks (except user-defined ACBs, NIBs, and RPLs) reside in the ACF/VTAM partition or address space, which has its own storage protection key. The storage protection key prevents unauthorized user programs from using the ACF/VTAM partition or address space. In DOS/VSE systems, all ACF/VTAM routines, control blocks, and buffers reside in the ACF/VTAM partition, which has its own storage protection key. In OS/VS systems, they are placed in the ACF/VTAM partition (OS/VS1), in address space (OS/VS MVS), or in protected system areas.

ACF/VTAM's storage management routines enable storage requests to be queued if storage is not available at the time of the request. This queuing ensures that, once application program requests (such as SEND

and RECEIVE) are initially accepted, they a
rejected because storage is not immediately

Locks

ACF/VTAM has its own locking structure to ensu
require serial access are not used by more tha
a time. A locking hierarchy is established to
situations and to reduce the amount of waiting
are described in Chapter 14 and in ACF/VTAM Dia

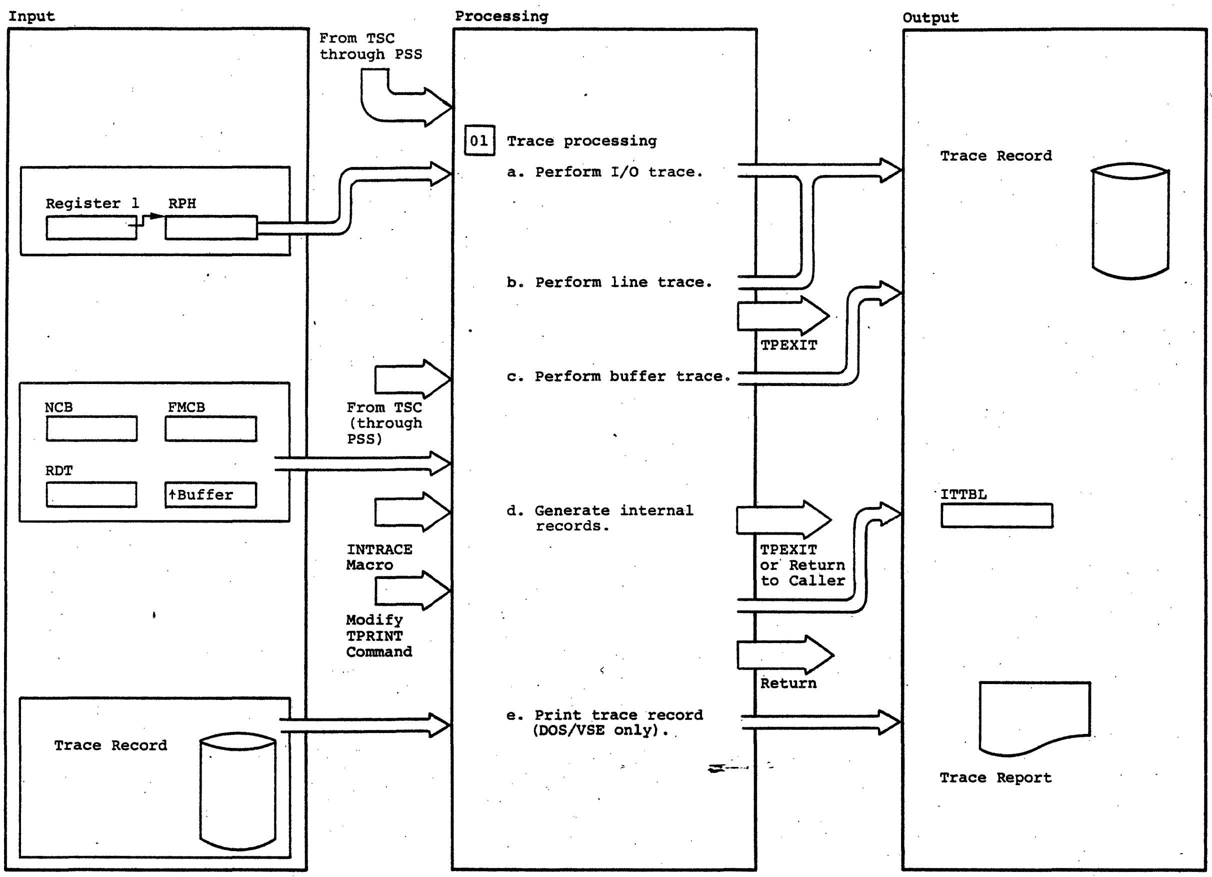
Sequence Numbers

ACF/VTAM assigns sequence numbers to I/O requests
numbers are maintained for inbound and outbound re
programs can use these sequence numbers to correla
requests and to assist in analyzing the trace reco

ACF/VTAM SLOWDOWN PROCESSING

ACF/VTAM requires a certain minimum number of buffers
the required number of buffers is not available, ACF/
slowdown mode, whether ACF/VTAM has data to read or n
program requests are not rejected; they are queued and
after slowdown mode has ended. When enough buffers be
for reads, ACF/VTAM ends the slowdown mode.

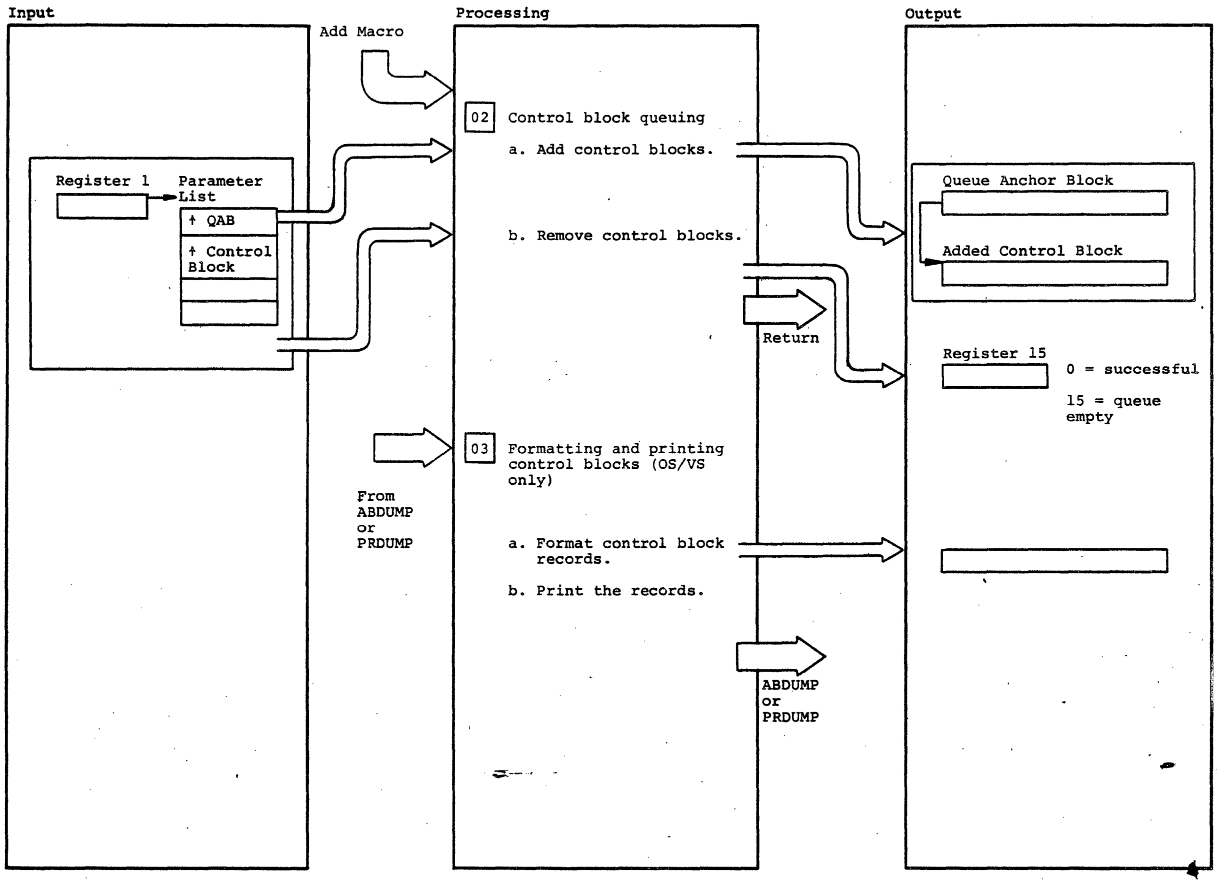
MO 11 (Part 1 of 4). Reliability, Availability, and Serviceability (RAS)



Notes	Routine	Label	Ref.
01 d. In DOS/VSE, the trace records can be printed either by the TPRINT utility in its own partition or as a subtask. In OS/VS the general trace facility and the HMDPRDMP utility are used to format and print the trace records.			

Notes	Routine	Label	Ref.

MO 11 (Part 2 of 4). Reliability, Availability, and Serviceability (RAS)



Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.

MO 11 (Part 3 of 4). Reliability, Availability, and Serviceability (RAS)

Input

Processing

Output

System Recovery/ Termination Manager →

04 Process recovery (OS/VS only)

a. If a retry is possible, set indicators for the system return control to the module in error.

b. Otherwise, set indicators for the system to continue the abend (resulting in the ACF/VTAM resource recovery manager's getting control).

→ System Recovery Termination Manager

System Recovery Termination Manager →

05 Resource recovery (OS/VS only)

a. Clean up ACF/VTAM resources.

b. Schedule the ACF/VTAM termination subtask PAB.

→ TPQUE Process Scheduling Services (MO 9)

Notes	Routine	Label	Ref.

Notes	Routine	Label	Ref.

MO 11 (Part 4 of 4). Reliability, Availability, and Serviceability (RAS)

Input

Processing

From ACF/VTAM Resource Recovery Manager through TPQUE to PSS

06 ACF/VTAM termination subtask (OS/VS only)

→ System Recovery Termination Manager

Clean up resource being used by the application program that is in error.

→ TPEXIT-Process Scheduling Services

Output

Notes	Routine	Label	Ref.

Notes	Routine	Label	Ref.

CHAPTER 14. PHYSICAL UNIT SERVICES (PUS)

Physical unit services (PUS) is used to control network resources local to the host, for example:

- Activate and deactivate local resources at SSCPs direction.
- Process inoperative conditions for local resources.
- Notify half-sessions within the host subarea of session failure on receipt of Lost Subareas RUs.

An overview of the operation of PUS is shown in MO 12.

ROUTING A REQUEST OR RESPONSE

Physical unit services is dispatched on a single PAB, with its input being a request/response unit processing element (RUPE).

PROCESSING ACCESS METHOD CLEANUP REQUESTS

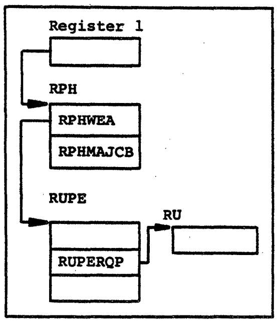
SSCP sends an Access Method Cleanup request to PUS when a line or physical unit fails. PUS notifies the SSCP of the session reset of logical units (for the LUS that have sessions with application programs) by issuing a Terminate Other Force Cleanup request. SSCP also sends Cleanup requests to PUS when a Cleanup request results from a forced deactivate command for a logical unit. PUS notifies any application program that is a primary end of a session involving a deactivated logical unit. It does this by issuing a Terminate Other Orderly (Cleanup) request to the session services subcomponent of SSCP. The session services subcomponent terminates the sessions.

PROCESSING LOST SUBAREA REQUESTS

Lost Subarea requests can be received by PUS from other subarea nodes. PUS responds to the Lost Subarea request, if necessary, and calls system definition, passing the network address of the subarea that is lost, or which sent the Lost Subarea request. System definition returns a list of all the subareas that this domain might try to reach through the lost subarea. The list is made up from the matches found between the list received from the adjacent subarea and the list received from system definition. Each logical unit that has a session with a logical unit in this lost subarea list (including the CDRM) is notified that the session is lost. This is done by sending to logical unit services a Lost Path request for each active or pending active LU-LU session on the list. For CDRM-to-CDRM sessions, the Lost Path request is sent directly to the CDRM, and an NSLSA is sent to the SSCP.

MO 12. Physical Unit Services (PUS)

Input



Processing

PUS PAB - ATCPUPAB

01 Route the RU.

a. If a request, route the request to the appropriate request processor.

b. If a response, route the response to the appropriate response processor.

c. Reject unsupported RUs.

To PSS (MO 9)

Output

Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.
01 The Physical Unit Services router determines what the input Request/Response Unit (RU) is and routes to the appropriate processing routine. All unsupported RUs are rejected with "function not supported."	ISTPUCRT						

CHAPTER 15. LOGICAL UNIT SERVICES (LUS)

Logical unit services (LUS) provides support for logical units in this domain by:

- Establishing and terminating sessions between an application program and other logical units. This processing is performed by the open-destination and close-destination routines.
- Providing the logical unit function for terminal systems that do not provide their own LU function.
- Handling and processing end user requests, LU-LU session requests, function management requests, and unformatted requests. End user requests and LU-LU session requests are routed to the LUS manager for processing. Function management RUs, which flow on the SSCP-to-LU session, are sent to the function interpreter for processing. Unformatted requests are passed on to the operator.
- Handling Lost Path, Cleanup, and Notify request units (RUs). These RUs are sent to LU services and then forwarded to the application programs affected. The RUs are used to notify the application programs of path errors so that sessions can be terminated in an orderly manner.

This component also handles INQUIRE and INTRPRET macro instructions issued in application programs or by ACF/VTAM itself. These macro instructions are issued to obtain data from control blocks that are otherwise not available to application programs. INQUIRE macro instructions can be issued to obtain information about the status of application programs, devices, and logons.

An overview of the operation of LUS is shown in MO 13.

LUS MANAGER

The LUS manager represents the interface from the end user to LUS. End User requests in the form of access method requests (AMRUs) are passed to the LUS manager by the LUS router. End user requests processed by the LUS manager include OPNDST, OPNSEC, SIMLOGON, REQSESS, TERMSESS, CLSSEC, and CLSDST.

FUNCTION INTERPRETER

The SSCP-LU secondary half-session represents the interface from the SSCP to LUS. Requests are received from the SSCP by the LUS router and passed to the function interpreter for processing. The function interpreter validates RUs received and passes them on to the appropriate finite state machine for processing. The finite state machines will pass RUs to the LUS manager for processing, if necessary. RUs processed by the function interpreter include INITIATE, CINIT, NSPE, NOTIFY, SESSION STARTED, BIND FAILURE, UNBIND FAILURE, SESSION ENDED, CTERM, and TERMINATE requests.

OPEN DESTINATION PROCESSING

The processing of an OPNDST macro instruction is shown in Figure 8-2. The two aspects of open-destination processing are:

- Establishing a session between two logical units (for example, between two application programs)

- Determining which routines will be used to prepare requests for and process requests from the type of logical unit for which the session is to be established (through construction of a real DVT)

In processing an open-destination request, except for application programs, the open-destination routines must ensure that the desired logical unit has not been previously allocated.

To establish the session, the open-destination routines use the session services component to build a session information block (SIB). The SIB represents the status of a session request and indicates one of three possible states for the request: session request queued, session request pending, and session active. If the destination node is available for a session, session services indicates in the SIB that the session is active. If a SIMLOGON macro instruction is issued and the destination node is not available for a session, session services indicates in the SIB that the session request is to be queued until the destination node becomes available. Then, when the destination node becomes available for a session and an OPNDST ACCEPT macro instruction is issued, the SIB is converted to the session pending state. Then, the open destination routines build a logical unit control block (LUCB), and a function management control block (FMCB) to represent the session. The FMCB is chained to the LUCB, which, in turn, is chained to the ACDEB for the application program. After the destination node has accepted the session and LU services has issued the Session Started command, the SIB is marked session active.

The second aspect of open-destination processing is determining which routines shall process requests going to and being received from the terminal. The application program provides the open-destination routine with the identification of the skeleton DVT, which symbolically identifies the processes required to support the session. Real DVTs are constructed from the skeleton DVTs and are then chained to the FMCB for the appropriate half-session. This processing is performed by issuing the FMCBADD macro. The FMCB build routines are part of OPEN/CLOSE processing.

If the destination node is an application program, a Bind is sent to the application program's SCIP exit routine, after which the secondary end issues either a SESSIONC macro instruction to reject the session or an OPNSEC macro instruction to establish the session. See "OPNSEC Processing" below.

If the destination node is a device-type logical unit, the application program must be bound to the logical unit before open-destination processing is completed. The open-destination routines send the Bind command to the logical unit. If a positive response is received, a Start Data Traffic command is sent (if required), and open-destination processing is completed. If a negative response is received, processing is terminated.

The open-destination routines construct a communication ID for the session, add this ID to the CID table, and return the ID to the application program. The application program then uses the communication ID to communicate with the terminal.

An application program can request that it become the primary end of a session by issuing an OPNDST macro instruction or it can request that it become the secondary end of a session by issuing an REQSESS macro instruction (see "REQSESS Processing" below). A device-type logical unit can only make such a request indirectly by issuing an Initiate command. When the system services control point (SSCP) receives the Initiate, it invokes session services to build an SIB. After the SIB is built, the application program can accept logons from other logical units. When the application program issues an OPNDST ACCEPT macro instruction, the SIB is converted to indicate a pending session.

REQSESS PROCESSING

The REQSESS macro instruction is used by an application program to indicate that it wishes to be the secondary end of a session. When a REQSESS macro instruction is executed, an Initiate RU is sent to SSCP, which causes entry into the LOGON exit routine of the application program that is to be the primary end of the session. If the primary end accepts the session, it responds by issuing an OPNDST macro instruction, which results in a Bind command's being sent to the secondary end's SCIP exit routine. The SCIP exit routine receives control, and, if the secondary end accepts the session, it issues an OPNSEC macro instruction to start the session. If the secondary end does not accept the session, it issues a SESSIONC (-RSP(BIND)) macro instruction. If the primary end rejects the session, it issues a CLSDST macro instruction.

OPNSEC PROCESSING

The OPNSEC macro instruction sets up an application program's control block structure so that it can be the secondary end of a session. OPNSEC processing starts after a Bind command is passed to the secondary end's SCIP exit routine as a result of an OPNDST macro instruction issued by the primary end. The SCIP exit routine determines whether the secondary end can accept the session. If a session can be established, the OPNSEC macro instruction is executed and a positive response to the Bind is sent to the primary end. The primary end's OPNDST processor sends a Session Started RU to notify the SSCP that a session has been established.

CLOSE DESTINATION PROCESSING

The CLSDST macro instruction terminates sessions between nodes and releases the resources for future sessions. The nodes are returned to the state they were in prior to OPNDST processing. The session services component is used to terminate the sessions between primary and secondary ends of sessions. The control blocks that defined the session are released or reset. Support functions in the host operating system are released. The environment is then as it was before open-destination processing. A new session can then be established if an outstanding session request is present.

An application program acting as the primary end of a session terminates that session by issuing either a CLSDST or a CLOSE macro instruction. Any control blocks associated with the secondary end of the session are released. The FMCB for the secondary end of the session is dequeued from the primary end's LUCB. The secondary end's session control blocks are cleaned up to allow it to have future sessions.

There are two forms of the CLSDST macro instruction:

- CLSDST with the RELEASE option releases the allocated node and frees the support modules and control blocks.
- CLSDST with the PASS option is used by an application program to pass a node to another application program.

TERMSESS PROCESSING

The TERMSESS macro instruction is used to terminate a session and can be issued only by the application program acting as the secondary end of the session to be terminated. When the TERMSESS macro instruction is executed, a Terminate RU is sent to the SSCP, which causes entry into the primary end's LOSTERM exit routine. If it is an unconditional TERMSESS request, the SSCP sends an Unbind command to the secondary end. If it is a conditional TERMSESS request, the primary end then issues a CLSDST macro instruction, which causes an Unbind command to be sent to the secondary end, and the break session processor of LUS to be invoked to clean up the secondary end's session control blocks to allow it to have future sessions.

INQUIRE PROCESSING

The INQUIRE macro instruction is issued by the application program to obtain specific types of information. Depending upon the type of INQUIRE specified, information is retrieved and passed back to the user. Types of information provided are:

- User data from a CINIT for a pending session (LOGONMSG)
- Device characteristics for a specific device (DEVCHAR)
- Counts of active sessions and of queued CINITs for this application (COUNTS)
- Logical unit name and the session ID for the oldest queued CINIT (TOPLOGON)
- CID corresponding to symbolic name or symbolic name corresponding to the CID (CIDXLATE)
- NIB list for the specified resource and all associated resources (TERMS)
- Application or CDRSC status for a particular resource (APPSTAT)
- Session parameters for a particular session (SESSPARM)
- Session key and seed for a particular session (SESSKEY)

The information for a particular request is built directly in an area provided by the issuer of the INQUIRE macro instruction. Because of the dynamic nature of some of the information that can be requested, the caller cannot always know in advance how large a work area is needed to contain all the requested information. Therefore, LUS places the requested data in the user's area if the user's area is large enough and provides feedback information to the user about the amount of data supplied. Thus, if the amount of data given to the user is less than or equal to the size of the work area provided, all the information is transferred. However, if the amount of requested information is greater than the size of the area provided, the user can recognize the need to reissue the request and provide a work area large enough to contain all the requested data.

INTRPRET PROCESSING

The INTRPRET macro instruction is issued by an application program to translate an input character string, using the interpret table associated with the resource, into an application name. An INTERPRET request results in an inspection of the interpret table associated with the specific resource, and the interpret table sequence is returned in the user-specified area.

PROCESSING LOST PATH RUS

Lost Path RUs are sent to LUS by PUS. LUS converts the Lost Path RUs to Terminate Other Orderly Cleanup RUs and queues them to the session services PAB. The session services subcomponent of SSCP terminates the session and sends a Cleanup or Notify request to LUS.

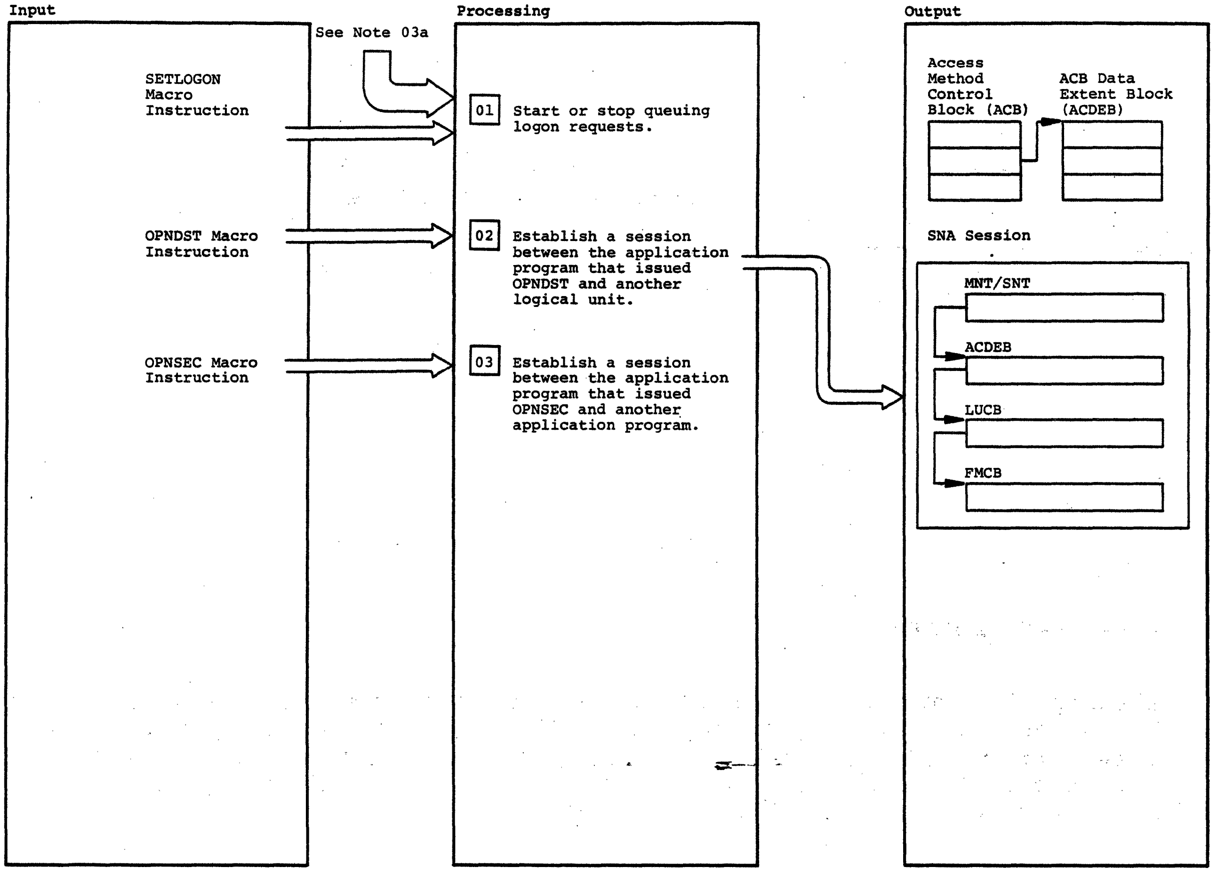
PROCESSING CLEANUP RUS

Cleanup RUs are sent to LUS by the session services subcomponent of SSCP. If the application program has an NSEXIT exit routine, LU services frees control blocks such as the FMCB, DVT, and EPT, and then schedules the NSEXIT exit routine. Otherwise, the application program is notified through its LOSTERM exit routine.

PROCESSING NOTIFY RUS

Notify RUs are sent to LUS by session services. Each Notify RU is handled by building UECB and VRPL control blocks and scheduling the application program's NSEXIT routine.

MO 13 (Part 1 of 4). Logical Unit Services (LUS)



Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.
02 Issuer of OPNDST is primary end of session.							
03 Issuer of OPNSEC is secondary end of session.							
a. Logical unit services receives control from either OPEN/CLOSE PUs or the SSCP.							

MO 13 (Part 2 of 4). Logical Unit Services (LUS)

Input

REQSESS Macro Instruction

SIMLOGON Macro Instruction

Processing

04 Request the establishment of a session between the application program that issued REQSESS and another application program.

05 Create an SIB and, optionally, drive the application progam's LOGON exit routine or complete an outstanding OPNDST ACCEPT request.

Output

RDT (Primary)

SIB

RDT (Secondary)

Notes	Routine	Label	Ref.
04 Issuer of REQSESS will be secondary end of session if session is established.			

MO 13 (Part 3 of 4). Logical Unit Services (LUS)

Input

TERMSESS Macro Instruction

CLSDST Macro Instruction

Close secondary end application program session processing

Processing

06 Request the termination of an application-program-to-application-program session on behalf of the secondary end.

07 Terminate session between the application program that issued CLSDST and another logical unit.

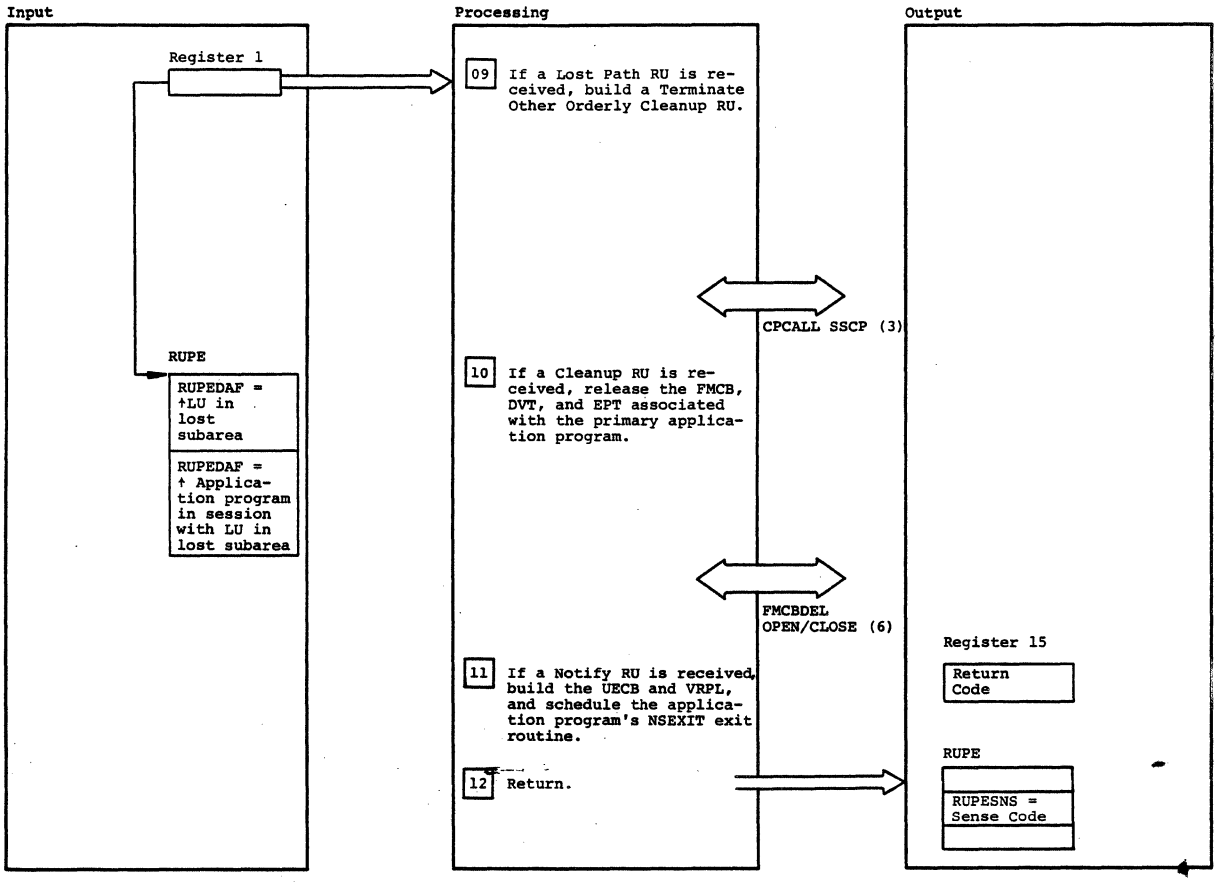
08 Clean up secondary end application program's control blocks to allow future sessions.

Output

Notes	Routine	Label	Ref.
07 Issuer of CLSDST must be primary end.			

Notes	Routine	Label	Ref.

MO 13 (Part 4 of 4). Logical Unit Services (LUS)



Notes	Routine	Label	Ref.	Notes	Routine	Label	Ref.
10 If the application program has an NSEXIT routine, LUS releases the control blocks associated with the session before calling the NSEXIT exit routine. If the NSEXIT routine is not coded, the application program is notified through its LOSTERM exit routine.							

CHAPTER 16. MULTISYSTEM NETWORKING FACILITY (MSNF)

The Multisystem Networking Facility (MSNF) is the ACF/VTAM feature that makes it possible to have a data communication network with more than one host system and more than one domain. With this feature, Logical Units (LUs) can communicate between domains. LUs are application programs or logical unit devices (including BSC 3270 terminals defined as PU=YES). Local 3705 Communications Controllers are used to connect domains. The cross-domain resource manager(CDRM) is the major component of ACF/VTAM for the Multisystem Networking Facility. Other functions for the Multisystem Networking Facility are provided by modules that are extensions of the system definition (SYSDEF), system services control point (SSCP), and network inquiry components of ACF/VTAM.

An overview of the operation of these components and their detailed descriptions can be found in the publication ACF/VTAM Logic: Multisystem Networking Facility.

CHAPTER 17. ENCRYPT/DECRYPT FEATURE

The ACF/VTAM Encrypt/Decrypt Feature improves data security for communications between logical units that are capable of cryptography. The Encrypt/Decrypt Feature is available in ACF/VTAM for OS/VS1 and OS/VS2 MVS. It requires that the IBM Programmed Cryptographic Facility be active in the host processor and that cryptographic keys be defined in the cryptographic key data set (CKDS) as described in *IBM Programmed Cryptographic Facility Installation Reference Manual*.

The Encrypt/Decrypt Feature supports single-domain cryptographic sessions and, in conjunction with the ACF/VTAM Multisystem Networking Facility, it supports cross-domain cryptographic sessions.

PACKAGING OF THE ENCRYPT/DECRYPT FEATURE

The Encrypt/Decrypt Feature consists of extensions to the initialization, system definition, SSCP, network inquiry, OPEN/CLOSE, TSC, and CDRM components of ACF/VTAM. Systems without the Encrypt/Decrypt Feature contain base modules in place of the Encrypt/Decrypt Feature modules. The base modules reject requests for cryptographic services and issue error messages stating that this feature is not supported in this host system. Descriptions of the base modules are included in the publications *ACF/VTAM Logic* and *ACF/VTAM Multisystem Networking Facility Logic*. When the Encrypt/Decrypt Feature is installed, the base modules are replaced by the feature modules, which are described in *ACF/VTAM Encrypt/Decrypt Feature Logic*.

CHAPTER 18. TELEPROCESSING ONLINE TEST EXECUTIVE PROGRAM (TOLTEP)

The Teleprocessing Online Test Executive Program (TOLTEP) schedules, controls, and supports the online test programs (OLTs). The OLTs exercise and evaluate the hardware capabilities of the various IBM devices supported by ACF/VTAM. TOLTEP is attached by ACF/VTAM when ACF/VTAM is started and resides in the ACF/VTAM partition or address space.

More than one user can use TOLTEP concurrently under one ACF/VTAM task. TOLTEP runs concurrently with other tasks in the system.

Refer to ACF/VTAM Diagnostic Techniques for information on how to run TOLTEP and the devices that TOLTEP supports.

SYSTEM RELATIONSHIP

Figure 18-1 shows the relationship of TOLTEP to ACF/VTAM,the operating system that selected ACF/VTAM, the OLTs, the control terminal, and the terminals being tested.

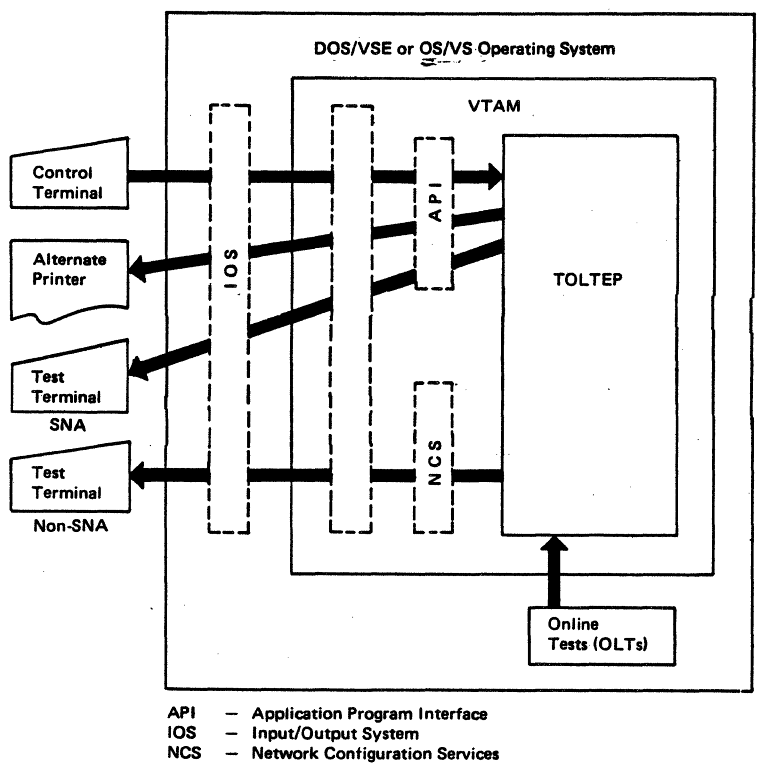


Figure 18-1. TOLTEP/System Relationship

TOLTEP/API

TOLTEP communicates with the control terminal, the alternate printer, and SNA test terminals by using the application program interface (API) in the same manner as other application programs. TOLTEP issues API macro instructions such as OPEN, OPNDST, INQUIRE, SEND, RECEIVE, CLSDST, and CLOSE.

TOLTEP/OLTS

The OLT test section communicates with TOLTEP by using the TOLTEP macro processing routines (described in ACF/VTAM Logic: Base System). The OLT calls TOLTEP routines such as EXIO (to execute I/O) and DPRINT (to print diagnostic messages). This operation is the same as other online test executive programs.

TEST PARAMETERS

The test run definition (a response to ENTER-DEV/TEST/OPT/) is the primary input to TOLTEP. The TOLTEP user selects the devices to test, the tests to execute, and the options affecting the way TOLTEP handles the testing process. The TOLTEP user enters the test run definition from the TOLTEP control terminal. TOLTEP also provides the following verbs for other, non-testing operations.

- CT=term allows the user to specify another terminal as the TOLTEP control terminal.
- DUMP allows the user to dump a specified portion of TOLTEP virtual storage.
- PROMPT allows the user to request examples of correct replies to the ENTER-DEV/TEST/OPT message.
- CANCEL allows the user to terminate a TOLTEP testing session from the control terminal.
- TALK allows the OLT to communicate with the control terminal operator.
- UPDATE allows the user to exhibit, add, change, copy, or delete a configuration data set (for OS/VS only).

MESSAGES

TOLTEP issues messages during the testing that inform the TOLTEP user of the status of the testing. These messages and appropriate replies, if any, are described in ACF/VTAM TOLTEP. TOLTEP does not require the reply character and the message ID but they are required by OS/VS if the system console is being used as the TOLTEP control terminal. They are not required by TOLTEP in DOS/VSE or OS/VS when the TOLTEP control terminal is other than the system console.

The OS/VS message prefix of ITA is used for TOLTEP messages. The DOS/VSE message prefix is F. The message ID and text are the same for all systems.

TOLTEP DATA AREAS

The formats of the resident common area and the test work area are provided in the *ACF/VTAM Data Areas*.

RESIDENT COMMON AREA (RCA)

The resident common area is initialized when TOLTEP is attached by ACF/VTAM. This area serves as an anchor block for all TOLTEP modules and contains:

- The TOLTEP ACB (access method control block)
- Pointers to the chain of test work areas (TWAs)
- Pointers to several TOLTEP routines
- Control information
- The TOLTEP RPL (request parameter list)
- A work area

TEST WORK AREA (TWA)

For each user that starts TOLTEP, the test initialization routine creates a test work area for that particular user. The test work area is associated with the user and contains several subtables and fields that allow the user to control testing. The subtables in the test work area are:

Section Control Table (SCT)

The section control table contains:

- Flags associated with the test section and the OPT response to the ENTER-DEV/TEST/OPT message.
- A copy of the first 20 bytes of the CDS for the primary device (the device currently being tested as specified in the DEV response to the ENTER-DEV/TEST/OPT message).

Connection Description Area (C/D)

The connection description area contains:

- Chain pointers to other test work areas
- Pointers to areas within the OLT area
- TOLTEP processing condition flags

Device Entry Table (DET)

The device entry table is filled in by the device decode routine. There are 18 entries in the table: the first is for the control terminal, the second is for the alternate printer, and the remaining 16 (the maximum number of devices that may be specified) are for the devices specified in the DEV response to the ENTER-DEV/TEST/OPT message. Each entry contains:

- Device information
- Flags indicating TOLTEP's use of the device

Section List Table (SLT)

The section list table is filled in by the test decode routine. This table contains:

- The "family name" of the test section (for example, T3700)
- Flags indicating testing status
- The IDs and flags associated with each test section specified (repeated 26 times)
- Pointers and values used in selecting the test sections

General Work Area (GWA)

The last subtable in the test work area is the general work area, which contains:

- Save areas for individual TOLTEP routines that have calls to other routines in response to an OLT request
- Buffers for messages from TOLTEP
- Work areas
- Request Parameter Lists (RPLs) for the control terminal and alternate printer.
- Various other fields used by TOLTEP routines during testing

ABBREVIATIONS

ABEND	Abnormal end of task
ACB	Access method control block
ACDEB	ACF/VTAM data extent block
ACF	Advanced Communications Function
ACF/VTAM	Advanced Communications Function for VTAM
ACTLU	Activate Logical Unit
ACTPU	Activate Physical Unit
API	Application program interface
APPL	Application (program)
APT	Asynchronous processing table
ASCB	Address space control block
ASXB	Address space extension block
ATCVT	ACF/VTAM communication vector table
AVT	Address vector table
BFT	Boundary function table
BHSET	Block handler routines
CA	Channel adapter
CCW	Channel command word
CDRM	Cross-domain resource manager
CDRSC	Cross-domain resource
CID	Communications identifier
CIDCTL	CID control
CLSDST	Close destination
CONFIG	Configuration
CONFT	ACF/VTAM configuration table
CRA	Component recovery area
CRR	Component recovery record
CSM	Communication system manager
CSMI	Communication system manager interface
CSW	Channel status word
CVT	Communications vector table
C/D	Connection Description (TOLTEP)
DACTLU	Deactivate Logical Unit
DACTPU	Deactivate Physical Unit
DAF	Destination address field
DET	Device entry table
DEV	Device
DLRPL	Dump/load/restart parameter list
DVT	Destination vector table
ECB	Event control block
ERP	Error recovery procedure
FID0	Format identifier 0 (PIU)
FID1	Format identifier 1 (PIU)
FID2	Format identifier 2 (PIU)
FIFO	First-in first-out
FMCB	Function management control block
ID	Identification
ICNCB	Intelligent controller node control block
IRB	Interruption request block
I/O	Input/output
LDNCB	Local device node control block
LIFO	Last-in first-out
LQAB	Lock queue anchor block
LU	Logical unit

LUS	Logical unit services
LUCB	Logical unit control block
MNT	Major node table
NAB	Network address block
NCB	Node control block
NCSPL	Network configuration services parameter list
NIB	Node identification block
OLT	Online test
OPNDST	Open destination
OPT	Option
PAB	Process anchor block
PIU	Path information unit
PLM	Program logic manual
PSS	Process scheduling services
PST	Process scheduling table
PU	Physical unit
PUS	Physical unit services
QAB	Queue anchor block
RAS	Reliability, availability, and serviceability
RCA	Resident common area
RDT	Resource definition table
RDTE	Resource definition table entry
RH	Request/response header
RPH	Request processing header
RPL	Request parameter list
RU	Request/response unit
RUPE	Request/response unit processing element
SAF	Source address field
SCT	Section control table
SDT	Start Data Traffic
SDVT	Skeleton destination vector table
SIB	Session information block
SIO	Start I/O
SLT	Section list table
SMS	Storage management services
SNA	Systems Network Architecture
SNT	Specific (minor) node table
SRB	Service request block (OS/VS2 only)
SRT	Symbol resolution table
SRTD	Symbol resolution table directory
SRTE	Symbol resolution table entry
SSCP	System services control point
STSN	Set and Test Sequence Numbers
SYSDEF	System definition
TIE	TOLTEP interface element
TRAC	Trace record
TRCPL	Trace parameter list
TWA	Test work area
TOLTEP	Teleprocessing Online Test Executive Program
TSC	Transmission subsystem component
TSCB	Transmission subsystem control block
TSPL	Transmission subsystem parameter list

UECB	User exit control block
VTAM	Virtual Telecommunications Access Method
WRE	Waiting request element
XID	Exchange identification

Glossary

This glossary defines terms and abbreviations that are important in this book. It does not include terms previously established for IBM operating systems and IBM products used with ACF/VTAM. Additional terms can be found by referring to the index, to prerequisite and corequisite books, and to the *IBM Data Processing Glossary,* GC20-1699.

This glossary includes definitions developed by the American National Standards Institute (ANSI) and the International Organization for Standardization (ISO). This material is reproduced from the *American National Dictionary for Information Processing,* copyright 1977 by the Computer and Business Equipment Manufacturers Association, copies of which may be purchased from the American National Standards Institute, 1430 Broadway, New York, New York 10018.

A complete commentary taken from ANSI is identified by an asterisk that appears between the term and the beginning of the commentary; a definition taken from ANSI is identified by an asterisk after the item number for that definition.

The symbol *ISO* at the beginning of a definition indicates that it has been discussed and agreed upon at meetings of the International Organization for Standardization Technical Committee 97/Subcommittee 1 (Data Processing), and has also been approved by ANSI.

The symbol *SC1* at the beginning of a definition indicates that it is reprinted from an early working document of ISO Technical Committee 97/Subcommittee 1 and that final agreement has not yet been reached among its participating members.

A

ACB. Access method control block.

ACB name. (1) The name of an ACB macro instruction. (2) A name that can be specified in the ACBNAME parameter of an APPL statement. This name allows an ACF/VTAM application program that is used in more than one domain to specify the same application program identification (pointed to by the APPLID parameter of the program's ACB statement) in each copy. ACF/VTAM knows the program by both its ACB name and its network name (the name of the APPL statement). Program users within the domain can request logon using the ACB name or the network name; program users in other domains must use the network name (which must be unique in the network). Contrast with *network name.*

accept. In ACF/VTAM, to establish a session between a logical unit and a primary application program as the result of a logon. The logon may be originated by the logical unit, the terminal operator, the network operator, another primary application program, or ACF/VTAM. Contrast with *acquire (1).*

access method control block (ACB). A control block that links an application program to VSAM or ACF/VTAM.

accounting exit routine. In ACF/VTAM, an optional, user-written routine that collects statistics about sessions in the communication network.

ACF. Advanced Communications Function.

ACF/NCP. Advanced Communications Function for the Network Control Program.

ACF/TCAM. Advanced Communications Function for the Telecommunications Access Method.

ACF/VTAM. Advanced Communications Function for the Virtual Telecommunications Access Method.

ACF/VTAM application program. A program that has opened an ACB to identify itself to ACF/VTAM and can now issue ACF/VTAM macro instructions.

ACF/VTAM definition. The process of defining the communication network to ACF/VTAM (which is called "network definition") and modifying IBM-defined characteristics to suit the needs of the user.

ACF/VTAM definition library. The DOS/VSE files or OS/VS data sets that contain the definition statements and start options filed during ACF/VTAM definition.

acquire. (1) In relation to an ACF/VTAM application program, to establish a session between a logical unit and an application program in the absence of a logon. The session is established at the primary application program's initiative. Contrast with *accept.* (2) In relation to ACF/VTAM resource control, to take over resources (communications controllers or physical units) that were formerly controlled by a data communication access method in another domain, or to assume control of resources that were controlled by this domain but released. Contrast with *release.* See also *resource takeover.*

active. Pertaining to a major node that has been made known to ACF/VTAM or pertaining to a minor node that is in session or available for a session. Contrast with *inactive.*

adjacent domain. A domain that is physically connected to another domain by a single cross-domain link or by a shared local communications controller.

adjacent node. A node that is physically connected to another node by a single data link.

Advanced Communications Function (ACF). A group of program products for users of DOS/VSE and OS/VS that can provide improved single-domain and, optionally, multiple-domain data communication capability.

Advanced Communication Functions for the Network Control Program (ACF/NCP). A program product that provides communications controller support for single-domain and multiple-domain data communication.

Advanced Communications Function for the Telecommunications Access Method (ACF/TCAM). A program product that provides single-domain data communication capability, and, optionally, multiple-domain capability.

Advanced Communications Function for the Virtual Telecommunications Access Method (ACF/VTAM). A program product that provides single-domain data communication capability and, optionally, multiple-domain capability.

any-mode. In ACF/VTAM: (1) The form of a receive request that obtains data from one unspecified terminal. (2) The form of request that establishes a session with an unspecified logical unit that has logged on. (3) Contrast with *specific-mode.* See also *continue-any mode.*

application layer. In SNA, the functional layer of each individual session in which the end user's application program is executed. See also *function management, transmission subsystem.*

application program identification. The symbolic name by which an application program is identified to ACF/VTAM. It is specified in the APPLID parameter of the ACB macro instruction. It corresponds to the ACBNAME parameter in the APPL statement or, if the ACBNAME is defaulted, to the name of the APPL statement.

application program major node. In ACF/VTAM, a member (OS/VS) or book (DOS/VSE) of the ACF/VTAM definition library that contains one or more APPL statements, each representing an application program.

APPLID routine. Synonym for *logon-interpret routine.*

asynchronous operation. In ACF/VTAM, an operation such as a request for session establishment or data transfer in which the application program is allowed to continue execution while ACF/VTAM performs the operation. ACF/VTAM interrupts the program as soon as the operation is completed.

asynchronous request. In ACF/VTAM, a request for an asynchronous operation.

authorization exit routine. In ACF/VTAM, an optional, user-written routine that approves or disapproves requests for session establishment or session termination.

authorized path. In ACF/VTAM for OS/VS2 MVS, a facility that enables an authorized application program to specify that a data transfer or related operation be carried out in a faster manner than usual.

automatic logon. A process by which ACF/VTAM creates a logon for a logical unit to a designated application program whenever the logical unit is not in session with or queued for a session with another program. Specifications for the automatic logon can be made when the logical unit is defined or can be made by the network operator in the VARY NET,LOGON command. See also *controlling application program.*

available. In ACF/VTAM: (1) Pertaining to a logical unit that supports only one session, is active, and is not in session with or queued for a session with an application program. (2) Pertaining to an exit routine that has been specified by an application program and that is not being executed.

B

basic information unit (BIU). In SNA, the unit of data and control information that is passed between connection point managers. It consists of a request/response header (RH) followed by a request/response unit (RU).

basic mode. In ACF/VTAM Release 1 and in VTAM, a mode of data transfer in which the application program can communicate with non-SNA terminals. Contrast with *record mode.*

basic transmission unit (BTU). (1) In the network control program, the unit of exchange between the host processor and the communications controller. It consists of control information and may also include data. The control information consists of a basic transmission header (BTH) and a basic device unit (BDU). All data transferred between the processing unit and the communications controller is preceded by a BTU. (2) In SNA, the unit of data and control information passed between path control components. The BTU can consist of one or more path information units (PIUs), depending on whether blocking is done by the path control that builds the BTU.

boundary function. In SNA: (1) A general term used for any one of several capabilities of a communications controller node: (a) transforming the network address form to a local address form, and vice versa, for attached terminals or cluster control unit nodes; (b) performing physical unit services and sequence numbering for attached, low-function terminals within its subarea; and (c) providing pacing of the data flows for secondary LUs within a subarea. (2) The programming component and functional structure that performs the above capabilities.

boundary node. In SNA, a network node that performs boundary functions. See also *boundary function.*

bracket. In ACF/VTAM, an uninterruptible unit of work, consisting of one or more chains of request units and their responses, exchanged between two logical units. Examples are data base inquiries/responses, update transactions, remote job entry output sequences to work stations, and similar applications.

bracket protocol. In SNA, a data flow control protocol in which exchanges between logical units (LUs) are achieved through the use of brackets, with one LU designated at session initiation as the first speaker, and the other LU as the bidder. The bracket protocol involves bracket initiation and termination rules.

C

cancel closedown. A closedown in which ACF/VTAM is abnormally terminated as the result of an operator command.

CDRM. Cross-domain resource manager.

CDRSC. Cross-domain resource.

change-direction protocol. In ACF/VTAM, a method of communication in which the sender stops sending on its own initiative, signals this fact to the receiver, and prepares to receive.

character-coded. In ACF/VTAM, pertaining to a logon or logoff command usually entered by a terminal operator from a keyboard and sent by a logical unit in character form. The character-coded command must be in the syntax defined in the installation's unformatted system services definition table. Contrast with *field-formatted.*

checkpoint. A point at which the information about the status of a data communication system is recorded. The system can then be reconstructed to its status at or near the time of failure.

CID. Communication identifier.

ciphertext. In ACF/VTAM, synonym for *enciphered data.*

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 also *cancel closedown, orderly closedown,* and *quick closedown.*

cluster control unit. A device that can control the input/output operations of more than one device. A remote cluster control unit is attached to a host processor only through a communications controller. A local cluster control unit is attached through a channel. A cluster control unit 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 2972 Station Control Unit. See also *communications controller* and *SDLC cluster controller.*

cluster controller. See *cluster control unit* and *SDLC cluster controller.*

command. (1) A request from a terminal for the performance of an operation or the execution of a particular program. (2) In SNA, a request unit initiating an action or beginning a protocol; it is used in contrast with reply, which is a request unit (not a response) that is sent in reaction to a command. For example: Quiesce (a data flow control request), is a command, while Quiesce Complete is the reply. (3) In SNA, a data flow control or session control request that may be sent or received by an application program.

common network. In SNA, the network consisting of path control and data link control elements that routes and moves path information units between any two transmission control elements.

communication control character. *(ISO) Synonym for *transmission control character.*

communication control unit. A communication device that controls the transmission of data over lines in a telecommunication network. Communication control units include transmission control units and communications controllers.

communication controller. A type of communication control unit whose operations are controlled by one or more programs stored and executed in the unit. Examples are the IBM 3704 and 3705 Communications Controllers.

communication identifier (CID). In ACF/VTAM, a key for locating the control blocks that represent an active session. The key is created during the session-establishment procedure and deleted when the session ends.

communication line. Any physical link, such as a wire or a telephone circuit, that connects one or more remote terminals to a communication control unit, or connects one communication control unit with another. Contrast with *data link.*

configuration restart. In ACF/VTAM, the facility for immediate recovery after a failure in the NCP or communication controller or after a loss of contact with a physical unit or logical unit, or for delayed recovery after a failure or deactivation of a major node, ACF/VTAM, or the host processor. Recovery may include reloading the NCP or restoring the network by means of a checkpoint. Restarting by means of checkpoint data requires the user to specify one or more VSAM data sets in which ACF/VTAM keeps a record of changes to initial configuration data.

connection. See *physical connection.*

connection point manager (CP manager). In SNA, one of the three components of transmission control; it provides a common mechanism by which session control, network control, and network addressable units communicate with their corresponding elements through the common network. The unit of information handled by the connection point manager is a request/response unit (RU). The unit of control information built by the sending connection point manager and interpreted by the receiving connection point manager is a request/response header (RH). See also *session control, network control.*

continue-any mode. In ACF/VTAM, a state into which a logical unit is placed that allows its input to satisfy an input request issued in any-mode. While this state exists, input from the logical unit can also satisfy input requests issued in specific-mode. Contrast with *continue-specific mode.*

continue-specific mode. In ACF/VTAM, a state into which a logical unit is placed that allows its input to satisfy only input requests issued in specific-mode.

controlling application program. An application program to which a logical unit (other than a secondary application program) is automatically logged on whenever the terminal is active and available. See also *automatic logon.*

converted command. An intermediate form of a character-coded logon or logoff command produced by ACF/VTAM through use of an unformatted system services definition table. The format of a converted logon or logoff 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 *character-coded.*

cross keys. In ACF/VTAM, see *cross-domain keys.*

cross-domain. Pertaining to control or resources involving more than one domain.

cross-domain keys. A pair of cryptographic keys used at a host processor to encipher the cryptographic session key that is sent to another host processor and to decipher the cryptographic session key that is sent from another host processor during cross-domain cryptographic session establishment.

cross-domain link. A data communication line physically connecting two domains. See also *local-to-local link.*

cross-domain resource (CDRSC). A resource owned by another domain but known in this domain by name and associated cross-domain resource manager.

cross-domain resource manager (CDRM). The portion of the system services control point (SSCP) that controls cross-domain sessions.

cross-domain session. A session between network addressable units in different domains

CRV. Cryptographic verification.

cryptographic. Pertaining to the transformation of data to conceal its meaning.

cryptograhic algorithm. A set of rules that specify the mathematical steps required to encipher and decipher data.

cryptographic key. A 64-bit value (containing 56 independent bits and 8 parity bits) that functions with the DES algorithm in determining the output of the DES algorithm. See also *cross-domain keys, cryptographic session key, host master key,* and *secondary logical unit key.*

cryptographic session. A session in which data may be enciphered before it is transmitted and deciphered after it is received. See also *required cryptographic session* and *selective cryptographic session.* Contrast with *clear session.*

cryptographic session key. In ACF/VTAM, a data encrypting key used to encipher and decipher data transmitted in a cryptographic session.

cryptographic verification (CRV) request. A request unit used to return an initial chaining value to the secondary logical unit to verify that the secondary logical unit is using the same cryptographic session key.

D

data communication. *The transmission and reception of data.

data encrypting key. A cryptographic key used to encipher and decipher data transmitted in a cryptographic session. See also *cryptogrpahic session key.* Contrast with *key encrypting key.*

Data Encryption Standard (DES) algorithm. A cryptographic algorithm designed to encipher and decipher 8-byte blocks of data using a 64-bit cryptographic key, as specified in the *Federal Information Processing Standard Publication 46,* January 15, 1977.

data flow. In SNA, any of several flows in a given session, characterized as either primary-to-secondary or secondary-to-primary, each of which may be normal or expedited.

data flow control. In SNA, a set of protocols and control functions within function management used to assist in controlling the flow of requests and responses within a session. Contrast with *function management data (FMD) services.*

data flow control protocol. In SNA, the sequencing rules for requests and responses by which network addressable units in a communication network coordinate and control data transfer and other operations. For example, see *bracket protocol.*

data link. (1) (SC1) An assembly of those parts of two data terminal equipments that are controlled by a link protocol, together with their interconnecting data circuit, that enables data to be transferred from a data source to a data sink. (2) The communication channel, modem, and communication controls of all stations connected to the communication channel, used in the transmission of information between two or more stations. (3) The physical connection and the connection protocols between the host and communication controller nodes via the host data channel. (4) Contrast with *communication line.*

Note:. *A communication line is the physical medium; for example, a telephone wire, a microwave beam. A data link includes the physical medium of transmission, the protocol, and associated communication devices and programs–it is both logical and physical.*

data link control (DLC). In SNA, one of the constituent parts of the transmission subsystem, and one of two constituent parts of the common network. It initiates, controls, checks, and terminates the data transfer over a data link between two nodes. Two distinct DLCs are defined in SNA: the DLC for the System/370 data channel, and SDLC for serial-by-bit data links. See also *path control* and *transmission control.*

data link control protocol. A set of rules used by two nodes on a data link to accomplish an orderly exchange of information. Synonymous with *line discipline.*

data transfer. In data communication, the sending of data from one point in a communication network and the receiving of the data at another point in the network.

data transmission. The sending of data from one point in a communication network for reception elsewhere.

decipher. To convert enciphered data into clear data. Contrast with *encipher.*

definite response. In SNA, a form of response requested in the request header for a request unit; the receiver is requested to return a response whether positive or negative. Contrast with *exception response* and *no response.*

definition statement. In ACF/VTAM, the means of describing an element of the communication network.

DES. Data Encryption Standard.

device control character. (ISO) A control character used for the control of ancillary devices associated with a data processing system or data communication system, for example, for switching such devices on or off.

device-type logical unit. A logical unit that represents something other than an ACF/VTAM application program (such as a logical unit for a 3270 terminal or a logical unit for a 3790 application program).

disconnection. The termination of a physical connection.

DLC. Data link control.

domain. In a data communication system, the portion of the total network that is controlled by the SSCP in one data communication access method.

domain operator. In a multiple-domain network, the person or program that controls the operation of the resources controlled by one access method. Contrast with *network operator (3).*

dormant state. In ACF/VTAM, when processing a VARY NET,INACT,I command, the stage at which all I/O for the node has been completed. At this stage, the node's session has ended, the node may be physically disconnected as well.

DRDS. Dynamic reconfiguration data set.

duplex. *(1) (ISO) In data communication, pertaining to a simultaneous two-way independent transmission in both directions. Synonomous with *full duplex.* (2) Contrast with *half duplex.*

dynamic reconfiguration. In ACF/VTAM, the process of changing the network configuration (SNA clusters and terminals) associated with an NCP boundary node, without regenerating the NCP's configuration tables.

dynamic reconfiguration data set (DRDS). In ACF/VTAM, a data set used for storing definition data that can be applied to a generated NCP configuration at the operator's request.

E

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.*

encipher. (1) To scramble data or convert it, prior to transmission, to a secret code that masks the meaning of the data to any unauthorized recipient. (2) In ACF/VTAM, to convert clear data into enciphered data. Contrast with *decipher.*

enciphered data. Data that is intended to be illegible to all except those who legitimately possess the means to reproduce the clear data. Synonymous with *ciphertext.*

end user. The ultimate source or destination of information flowing through a system. An end user may be an application program, an operator (such as a terminal user or a network operator/administrator), or a data medium (such as cards or tapes).

exception message. See *exception request.*

exception request. In communicating with a logical unit, a request unit that indicates an unusual condition such as a sequence number being skipped. When ACF/VTAM detects such a condition, it notifies the application program. ACF/VTAM or the application program provides sense information which is included in the response that is sent to the logical unit.

exception response. (1) In SNA, a response requested in the RH for a request unit; the receiver is requested to return a response only if it is negative. Contrast with *definite response.* (2) Synonym for *negative response.*

exit list (EXLST). A control block that contains the addresses of routines that receive control when specified events occur during execution; for example, routines that handle logon processing or I/O errors.

exit routine. Any of several types of special-purpose user-written routines. See *accounting exit routine, authorization exit routine, EXLST exit routine, logon-interpret routine,* and *RPL exit routine.*

EXLST exit routine. A type of user-written routine whose address has been placed in an exit list (EXLST) control block. See also *RPL exit routine.*

expedited flow. In SNA, a data flow designation indicated in the transmission header (TH). It is independent of and controls the normal flow. Both the primary-to-secondary and secondary-to-primary flows are split into normal and expedited flows. Requests and responses on a given (normal or expedited) flow are processed sequentially within the path, but the expedited flow traffic may be moved ahead of the normal flow traffic within the path. Contrast with *normal flow.*

external domain. A domain controlled by a different system services control point (SSCP).

F

FD. Full duplex. See *duplex.*

FDX. Full duplex. See *duplex.*

FID. Format identification (field).

field-formatted. Pertaining to a logon or logoff command that is encoded into fields, each having a specified format such as binary codes, bit-significant flags, and symbolic names. Contrast with *character-coded.*

format identification (FID) field. In SNA, a field in a transmission header (TH) that defines the subsequent format of the header and the type of TH fields involved with a transmission.

formatted system services (FSS). A portion of ACF/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.*

full duplex (FD, FDX). *Synonym for *duplex.*

function management (FM). (1) In SNA, the layer of functional capability between the application layer and the transmission subsystem. It includes data flow control and function management data (FMD) services. See also *application layer, transmission subsystem.* (2) In ACF/VTAM, the insertion of control information within request units so that the request units sent to a particular type of terminal are in the required format and so that request units received from that type of terminal are handled properly.

function management data (FMD) services. In SNA, the component of function management responsible for request/response units marked as "function management data." This includes presentation services and logical unit services (within the logical unit), physical unit services (within the physical unit), and network services (within the system services control point). Contrast with *data flow control.*

H

half duplex. *(1) In In data communication, pertaining to an alternate, one way at a time, independent transmission. (2) Contrast with *duplex.*

host computer. (1) The primary or controlling computer in a multiple computer operation. (2) A computer used to prepare programs for use on another computer or on another data processing system; for example, a computer used to compile, link-edit, or test programs to be used on another system. (3) In a data processing system that includes ACF/VTAM or ACF/TCAM, the computer in which ACF/VTAM or ACF/TCAM resides.

host master key. (1) A cryptographic key used to encipher operational keys that will be used at the host processor. decipher cryptographic session keys.

host processor. In a data communication system, the processing unit in which the data communication access method resides.

I

ICV. Initial chaining value.

inactive. In ACF/VTAM, pertaining to a major node that has not been made known to ACF/VTAM and is unavailable for use, or pertaining to a minor node that is not in session with nor available for a session with an application program. Contrast with *active.*

initial chaining value (ICV). An 8-byte random number used to verify that both ends of a cryptographic session have the same cryptographic session key. It is generated by the secondary end of the session upon receipt of a Bind request for a cryptographic session and is returned to the primary end of the session in the Bind response, enciphered under the cryptographic session key received in the Bind request. The initial chaining value is also used as input to the CIPHER macro to encipher or decipher data in a cryptographic session.

intensive mode error recording. The recording of information by ACF/NCP/VS about temporary errors on SDLC links. In addition to recording the initial error status that starts an error recovery process and the final error status that causes a permanent error record to be generated, intensive mode error recording causes each temporary error between the initial error and final error to be recorded in a RECMS RU and sent to the SSCP that requested intensive mode error recording.

intermediate node. In SNA, a physical unit that is capable of routing path information units to another subarea.

interpret table. In ACF/VTAM, a user-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.

K

key encrypting key. A cryptographic key used to encipher and decipher other cryptographic keys. Contrast with *data encrypting key.*

L

line. See *communication line.*

line control. The scheme of operating procedures and control signals by which a data link is controlled. For example, synchronous data link control (SDLC).

line group. One or more communication lines of the same type, that can be activated and deactivated as a unit.

link level 2 test. In ACF/VTAM, a test of a dedicated secondary station on a shared link with a shared communications controller.

local. (1) Pertaining to the attachment of devices directly by I/O channels to a host computer. Contrast with *remote.* (2) In data communication, pertaining to devices that are attached to a controlling unit by cables, rather than by data links.

local address. In SNA, an address transformed to or from a network address by the boundary function (for example, in a communications controller node) for use by a cluster control unit or terminal node. See also *network address, boundary function.*

local NCP. An NCP that is channel-attached to a host processor. Contrast with *remote NCP.*

local non-SNA major node. In ACF/VTAM, a major node whose minor nodes are locally-attached non-SNA terminals.

local SNA major node. In ACF/VTAM, a major node whose minor nodes are locally-attached physical and logical units.

local-to-local link. A data communication link between two local communications controllers. The link can be either a cross-domain link (communications controllers in different domains) or it can exist within a domain between local communications controllers controlled by the same system services control point.

local 3270 major node. See *local non-SNA major node.*

logical error. In ACF/VTAM, an error condition that results from an invalid request; a program logic error.

logical unit. In SNA, one of three types of network addressable units (NAUs). It is the port through which an end user accesses function management in order to communicate with another end user. It is also the port through which the end user accesses the services provided by the system services control point (SSCP). It must be capable of supporting at least two sessions–one with the SSCP, and one with another logical unit. It may be capable of supporting many sessions with other logical units. ACF/VTAM application programs must communicate with logical units in record mode. See also *physical unit, system services control point.*

log off. In ACF/VTAM, to request that a session be terminated.

logoff. In ACF/VTAM, a request that a session be terminated.

log on. In ACF/VTAM, to request that a session be established between an application program and a logical unit.

logon. In ACF/VTAM, a request that a session be established between an application program and a logical unit. See also *automatic logon* and *simulated logon.*

logon data. In ACF/VTAM: (1) The data portion of a field-formatted or character-coded logon from a terminal. (2) The entire logon sequence or message from a non-SNA terminal.

logon-interpret routine. In ACF/VTAM, a user-written exit routine, associated with a logon-interpret table entry, that translates logon data. Synonymous with *APPLID routine.*

logon message. Synonym for *logon data.*

logon mode. 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 ACF/VTAM, a set of entries for one or more logon modes. Each logon mode is associated with a logon mode name.

LU. Logical unit.

LU-LU session. In SNA, a session between two logical units in the network. It provides communication between two end users, each associated with one of the logical units.

M

major node. In ACF/VTAM, a set of minor nodes that can be activated and deactivated as a group. See also *minor node.*

message. (1) *An arbitrary amount of information whose beginning and end are defined or implied. (2) (SC1) A group of characters and control bit sequences transferred as an entity. (3) A combination of characters and symbols transmitted from one point to another.

minor node. In ACF/VTAM, a uniquely-defined resource within a major node that can be activated or deactivated by the VARY command. Synonymous with *specific node.* See also *major node.*

modem. (1) (SCI) A functional unit that modulates and demodulates signals. One of the functions of a modem is to enable digital data to be transmitted over analog transmission facilities. (2) *(modulator-demodulator) A device that modulates and demodulates signals transmitted over data communication facilities.

multiple-channel-attached communication controller. A communication controller that is channel-attached to more than one host computer.

multipoint line. A line or circuit interconnecting several stations. Contrast with *point-to-point line.*

Multisystem Networking Facility. In ACF, a feature that supports communication among multiple host processors operating with DOS/VSE, OS/VS1, and OS/VS2.

multithread application program. An ACF/VTAM application program that processes many requests from many logical units concurrently. Contrast with *single-thread application program.*

N

NAU. Network addressable unit.

NCP. Network control program.

NCP major node. In ACF/VTAM, a set of minor nodes representing devices (lines, terminals, and cluster control units) controlled by a network control program.

negative response. A response indicating that a request did not arrive successfully or was not processed successfully by the receiver in a session. Synonymous with *exception response.* Contrast with *positive response.*

negative response to polling limit. For a start-stop or BSC terminal, the maximum number of consecutive negative responses to polling that the communications controller accepts before suspending polling operations.

negotiable bind. In SNA, a session-establishment protocol that allows the secondary end of the session to accept, reject, or change the session parameters.

network. (1) (SC1) The assembly of equipment through which connections are made between terminal installations. (2) In data communication, a configuration in which two or more locations are physically connected for the purpose of exchanging data.

network address. In SNA, the address, consisting of subarea and element subfields, that uniquely identifies a link or the location of a network addressable unit. The conversion from a local address to a network address, or vice versa, is accomplished as part of the boundary function in the node attached to a cluster control unit or a terminal. See *local address.* See also *network name.*

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 in the transmission subsystem. Each NAU has a network address that represents it to the transmission subsystem. The transmission subsystem and the NAUs collectively constitute the communication system. See also *network name, network address.*

network control (NC). In SNA, a transmission control component that permits logically adjacent connection point managers to communicate through the common network, using sessions established for other purposes and thereby avoiding special session establishment. See also *connection point manager, session control.*

network control mode. The functions of a network control program that enable it to direct a communications controller to perform activities such as polling, device addressing, dialing, and answering. See also *emulation mode.*

network control program (NCP). A program, generated by the user from a library of IBM-supplied modules, that controls the operation of a communications controller.

network control program generation. The process, performed in a host system, of assembling and link-editing a macro instruction program to produce a network control program.

network definition. In ACF/VTAM, the process of defining the identities and characteristics of each node in the network and the arrangement of the nodes. Network definition is part of ACF/VTAM definition.

network name. In SNA, the symbolic identifier by which a network addressable unit or a data link is referred to by end users. See also *network address.* In a multiple-domain network, the name of the APPL statement is the network name and must be unique across domains. Contrast with *ACB name.*

network operator. (1) A person responsible for controlling the operation of a communication network. (2) An ACF/VTAM application program authorized to issue network operator commands. (3) In a multiple-domain network, the person or program that controls all the domains in the network. Contrast with *domain operator.*

network operator command. A command used to monitor or control the communication network.

network operator console. A system console or terminal in the network from which a network operator controls a communication network.

network operator logon. A logon requested on behalf of a terminal by means of a network operator command.

NIB. Node initialization block.

NIB list. A series of contiguous node initialization blocks.

no response. In SNA, an indication in the RH for a request unit that no response is to be returned to the request, whether or not it is received and processed successfully. Contrast with *definite response* and *exception response.*

node. (1) In SNA, a junction point in a network, represented by a physical unit. A node contains network addressable units. (2) In ACF/VTAM, a point in a communication network defined by a symbolic name. See also *major node* and *minor node.*

node initialization block (NIB). In ACF/VTAM, a control block associated with a particular node that contains information used by the application program to identify a node and indicate how communication requests directed at the node are to be implemented.

node name. In ACF/VTAM, the symbolic name assigned to a specific major or minor node during network definition.

non-SNA terminal. A terminal that does not use SNA protocols.

normal flow. In SNA, a data flow designation indicated in the transmission header (TH). Both the primary-to-secondary and secondary-to-primary flow are split into normal and expedited flows. The expedited flow is independent of and used to control the normal flow. Requests and responses on a given flow (normal or expedited) are usually processed sequentially within the path, but the expedited flow traffic may be moved ahead of the normal flow traffic within the path. Contrast with *expedited flow.*

O

orderly closedown. The orderly deactivation of ACF/VTAM and its domain. An orderly closedown does not take effect until all application programs have closed their ACBs. Until then, all data transfer operations continue. Contrast with *cancel closedown* and *quick closedown.*

P

pacing. (1) In data communication, a technique by which a receiving station controls the rate of transmission of a sending station to prevent overrun. (2) In SNA, a mechanism that permits a receiving connection point (CP) manager 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 either the receiver or the network can process them.

parallel session. In ACF/VTAM, two or more simultaneously active sessions between the same two logical units. Each session can employ a different function management protocol, a different transmission services protocol, and have independent session control.

partitioned emulation programming (PEP). A feature of the network control program, versions 2 and later, that allows a local IBM 3704 or 3705 controller to operate as an IBM 2701, 2702, or 2703 control unit (or any combination of the three) for certain data links, while performing network control functions for other links in the network.

path. (1) In ACF/VTAM, the intervening nodes and data links connecting a terminal and an application program in the host processor. (2) In defining a switched major node, a potential dial-out port that can be used to reach a physical unit. (3) In defining path tables, a route through an adjacent subarea to one or more destination subareas. (4) In SNA, the series of nodes, data links, and common network components (path control and data link control) that form the complete route traversed by the information exchanged between two network addressable units in session.

path control (PC). In SNA, one of the components of the transmission subsystem, and one of two components of the common network. It is responsible for managing the sharing of data link resources of the common network and for routing basic information units (BIUs) through it. It is aware of the location of NAUs in the network and of the paths between them. It maps the BIUs, handled by transmission control, into path information units (PIUs), and then into basic transmission units (BTUs) that are passed between path control and data link control. The unit of control information built by the sending path control component and interpreted by the receiving path control component is a transmission header (TH). See also *data link control, transmission control.*

path information unit (PIU). In SNA, the unit of transmission consisting of a transmission header (TH) and either a basic information unit (BIU) or a BIU segment.

path table. A table whose entries contain path information for a network node.

PEP. Partitioned emulation programming.

physical connection. A point-to-point connection, multipoint connection, or switched connection.

physical unit (PU). In SNA, one of the types of network addressable units; a PU is associated with each node that has been defined to a system services control point (SSCP). The SSCP establishes a session with the physical unit as part of the activation process. See also *logical unit, system services control point.*

PIU. Path information unit.

plaintext. Synonym for *clear data.*

point-to-point line. A data link that connects a single remote station to the computer; it may be either switched or nonswitched. Contrast with *multipoint line.*

positive response. A response that indicates a request was received and processed successfully. Contrast with *negative response.*

primary application program. An application program acting as the primary end of a session.

primary end of a session. The end of a session that has more control of the session through the use of primary protocols. Contrast with *secondary end of a session.*

program operator. An ACF/VTAM application program that is authorized to issue network operator commands and receive ACF/VTAM network operator awareness messages. See also *solicited messages* and *unsolicited messages.*

protocol. In SNA, the sequencing rules for requests and responses by which network addressable units in a communication network coordinate and control data transfer operations and other operations. See also *bracket protocol.*

PU. Physical unit.

Q

queued for a session. In ACF/VTAM, the state of a logical unit that has logged on to an application program but has not yet been accepted by that application program.

quick closedown. In ACF/VTAM, a closedown in which current data-transfer operations are completed, while session-establishment and new data-transfer requests are canceled. Contrast with *cancel closedown* and *orderly closedown.*

quiesce protocol. In ACF/VTAM, a method of communicating in one direction at a time. Either the application program or the logical unit assumes the exclusive right to send normal-flow requests, and the other node refrains from sending such requests. When the sender wants to receive, it releases the other node from its quiesced state.

R

RDT. Resource definition table.

record mode. In ACF/VTAM, the mode of data transfer in which the application program can communicate with logical units. Contrast with *basic mode.*

release. In ACF/VTAM resource control, to relinquish control of resources (communications controllers or physical units). See also *resource takeover.* Contrast with *acquire (2).*

remote. In ACF/VTAM, pertaining to devices that are physically connected by a communications line. Contrast with *local.*

remote NCP. An NCP that is not attached directly through a channel, but is attached through a data link to a local NCP that is channel-attached. Contrast with *local NCP* and *peer NCP.*

request header. See *request/response header.*

request parameter list (RPL). In ACF/VTAM, a control block that contains the parameters necessary for processing a request for data transfer, for establishing or ending a session, or for some other operation.

request/response header (RH). In SNA, a control field, attached to a request/response unit (RU), that specifies the type of RU being transmitted–request or response–and contains control information associated with that RU. See also *request/response unit.*

request/response unit (RU). In SNA, the basic unit of information entering and exiting the transmission subsystem. It may contain data, acknowledgment of data, commands that control the flow of data through the network, or responses to commands.

request unit. See *request/response unit.*

required cryptographic session. A cryptographic session in which all outbound data is enciphered and all inbound data is deciphered. Contrast with *selective cryptographic session* and *clear session.*

resource definition table (RDT). In ACF/VTAM, a table that describes the characteristics of each node available to ACF/VTAM and associates each node with an address, CID, or session ID.

resource takeover. In ACF/VTAM, the action of a network operator to transfer control of resources from one domain to another. See also *acquire (2)* and *release.*

responded output. In ACF/VTAM, a type of output request that is completed when a response is returned. Contrast with *scheduled output.*

response. See *request/response unit.*

response header. See *request/response header.*

response unit. In SNA, the request/response unit following a response header; it is sent in response to a request unit. Synonymous with *response.* See also *request/response unit.*

RH. Request/response header.

RPL. Request parameter list.

RPL-based macro instruction. In ACF/VTAM, a macro instruction whose parameters are specified by the user in a request parameter list.

RPL exit routine. In ACF/VTAM, a user-written routine whose address has been placed in the EXIT field of a request parameter list. ACF/VTAM invokes the routine to indicate that an asynchronous request has been completed. See also *EXLST exit routine.*

RU. Request/response unit.

S

scheduled output. In ACF/VTAM, a type of output request that is completed, as far as the application program is concerned, when the program's output data area is free. Contrast with *responded output.*

SDLC. Synchronous data link control.

SDLC cluster controller. A cluster control unit for a data communication subsystem.

secondary application program. An application program acting as the secondary end of a session.

secondary end of a session. That end of a session that has less control of the session and uses secondary protocols. For example, a terminal, or a secondary application program. Contrast with *primary application program.*

secondary logical unit (SLU) key. A primary key encrypting key used to protect a cryptographic session key during its transmission to the secondary end of the session.

selective cryptographic session. A cryptographic session in which an authorized application program is permitted to specify the enciphering of certain request units.

sequence number. A numerical identifier assigned by ACF/VTAM to each request/response unit exchanged between two nodes.

session. (1) The period of time during which a user of a terminal can communicate with an interactive system; usually, the elapsed time between logon and logoff. (2) The period of time during which programs or devices can communicate with each other. (3) In SNA, a logical connection, established between two network addressable units (NAUs) to allow them to communicate. The session is uniquely identified by a pair of network addresses, identifying the origin and destination NAUs of any transmissions exchanged during the session. See *LU-LU session, SSCP-LU session, SSCP-PU session.*

specific node. See *minor node.*

session control. In SNA, one of the components of transmission control. It is responsible for allocating resources necessary for a session, for purging data flowing in a session if an unrecoverable error occurs, and for resynchronizing the data flow after such an error.

session correlation value. In ACF/VTAM, when parallel sessions are activated, the value returned in the USER field in the NIB specified on a SIMLOGON or REQSESS macro instruction. It is used to correlate RUs and sessions.

session limit. (1) In the network control program, the maximum number of concurrent line-scheduling sessions on a non-SDLC, multipoint line. (2) In SNA, the maximum number of simultaneous sessions a particular network addressable unit can support.

session parameters. The communication protocols that govern a session between a logical unit and an application program or between two application programs. See also *logon mode.*

share limit. The limit of the number of SSCPs that can simultaneously share a resource.

shared. Pertaining to the availability of a resource to more than one use at the same time.

simulated logon. A logon generated for a terminal by ACF/VTAM at the primary application program's request. The primary application program accepts or rejects the terminal as if it had logged on.

single-channel-attached communication controller. A communication controller that is channel-attached to only one host computer.

single-thread application program. An ACF/VTAM application program that processes requests from terminals one at a time. Such a program usually requests synchronous operations from ACF/VTAM, waiting until each operation is completed before proceeding. Contrast with *multithread application program.*

SLU. Secondary logical unit.

SNA. Systems network architecture.

SNA terminal. A terminal that is compatible with systems network architecture.

SNBU. Switched network backup.

solicited message. A response from ACF/VTAM to a network operator command entered by a program operator. Contrast with *unsolicited message.*

specific-mode. In ACF/VTAM: (1) The form of request that obtains data from one specific terminal. (2) The form of request that establishes a session with a specific terminal that has logged on. (3) Contrast with *any-mode.* See also *continue-specific mode.*

SSCP. System services control point.

SSCP ID. An identifying number associated with an SSCP (that must be unique in a multidomain system) that enables a device (especially a dial-in device) to identify an SSCP at a particular location and enables another SSCP to identify this SSCP when establishing a session with it.

SSCP-LU session. In SNA, a session between a logical unit (LU) and the system services control point (SSCP). It is used to support logical unit-related control and use of the communication system. Each logical unit in the network participates in a session with the SSCP that provides services for that logical unit.

SSCP-PU session. In SNA, a session between a physical unit (PU) and the system services control point (SSCP) that is used to control the physical configuration and to control an individual node. Each physical unit in the network must participate in a session with the SSCP that provides services for that physical unit.

start options. In ACF/VTAM, the user-specified or IBM-supplied options that determine certain conditions that are to exist during the time an ACF/VTAM system is operating. These options include: the size of ACF/VTAM buffer pools, which major and minor nodes are to be traced by the ACF/VTAM trace facility, and which major nodes are to be made initially active. Start options can be predefined or specified by the network operator when ACF/VTAM is started.

subarea. A group of network addressable units that have the same subarea ID.

subarea ID. A subfield of network address.

switched network backup (SNBU). An optional facility that allows a user to specify, for certain types of stations, a switched line to be used as an alternate path (backup) if the primary line becomes unavailable or unusable.

switched SNA major node. In ACF/VTAM, a major node whose minor nodes are physical and logical units attached by switched SDLC links.

synchronous data link control (SDLC). A discipline for managing synchronous, transparent, serial-by-bit information transfer over a communication channel. Transmission exchanges may be duplex or half duplex over switched or nonswitched data links. The communication channel configuration may be point-to-point, multipoint, or loop.

synchronous operation. In ACF/VTAM, a communication, or other operation in which ACF/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 ACF/VTAM, a request for a synchronous operation. Contrast with *asynchronous request.*

system services control point (SSCP). In SNA, a network addressable unit that provides configuration and session services via a set of command processors (network services) supporting physical units and logical units. The SSCP must be in session with each logical unit and each physical unit for which it provides services. It also provides services for the network operators or administrators who control the configuration. The SSCP is commonly located at a host node.

systems network architecture (SNA). The total description of the logical structure, formats, protocols, and operational sequences for transmitting information units through the communication system. Communication system functions are separated into three discrete areas: the application layer, the function management layer, and the transmission subsystem layer. The structure of SNA allows the ultimate origins and destinations of information–that is, the end users–to be independent of, and unaffected by, the specific communication-system services and facilities used for information exchange.

T

TC. Transmission control.

telecommunication subsystem. In ACF/VTAM, a secondary or subordinate network and set of programs that are part of a larger telecommunication system; for example, the combination consisting of an SDLC cluster controller, its stored programs, and its attached terminals. Synonomous with *terminal system.*

telecommunication system. The devices and functions concerned with the transmission of data between the central processing unit and the remotely located users.

terminal. (1) A device, usually equipped with a keyboard and some kind of display, capable of sending and receiving information over a communication channel. (2) In ACF/VTAM, the secondary end of a session; that is, a logical unit, a BSC device, a local non-SNA 3270 device, or an application program.

terminal component. A separately addressable part of a terminal that performs an input or output function, such as the display component of a keyboard-display device or a printer component of a keyboard-printer device.

terminal system. In a data communication network, a terminal control unit or a cluster control unit and its attached devices. Synonymous with *telecommunication subsystem.*

TH. Transmission header.

transit node. An intermediate node that is capable of routing path information units to another domain.

transmission. In data communication, one or more blocks or messages. For BSC devices, a transmission is terminated by an EOT character. See also *message.*

transmission control (TC). In SNA, one of three components of the transmission subsystem. It has three subcomponents: the connection point manager, session control, and network control. It establishes, controls, and terminates sessions, and also controls the flow of information into and out of the common network for a session between network addressable units. It provides access to the transmission subsystem; this direct access is used by function management components. A transmission control element exists for each active session. See also *data link control, path control.*

transmission control character. *(ISO) Any control character used to control or facilitate transmission of data between data terminal equipment. Synonomous with *communication control character.*

transmission header (TH). In SNA, a control field attached to a basic information unit (BIU) or to a BIU segment, and used by path control. It is created by the sending path control component and interpreted by the receiving path control component. See also *path information unit.*

transmission subsystem. In SNA, the innermost layer of the communication system. It provides the control in each session to route and move data units between NAUs, and to manage the NAUs and their interconnecting paths. Its three constituent parts are data link control, path control, and transmission control. See also *application layer and function management.*

TSO/VTAM. Time Sharing Option for the Virtual Telecommunications Access Method.

U

unformatted system services (USS). A portion of ACF/VTAM that translates a character-coded command, such as a logon or logoff command, into a field-formatted command for processing by formatted system services (FSS). Contrast with *formatted system services (FSS)* and *character-coded.*

unsolicited message. A network operator message, from ACF/VTAM to a program operator, that is unrelated to any command entered by the program operator. Contrast with *solicited message.*

USS. Unformatted system services.

INDEX

LY38-3021-0

ACF/VTAM Release 2 Logic: Overview (File No. S370-30) Printed in U.S.A. LY38-3021-0

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601

Advanced Communications Function
for VTAM Release 2
Logic: Overview

READER'S COMMENT FORM

Order No. LY38-3021-0

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. This form may be used to communicate your views about this publication. They will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

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.*

How did you use this publication?

[ ] As an introduction
[ ] As a reference manual
[ ] For another purpose (explain) ______________________________

[ ] As a text (student)
[ ] As a text (instructor)

Is there anything you especially like or dislike about the organization, presentation, or writing in this manual? Helpful comments include general usefulness of the book; possible additions, deletions, and clarifications; specific errors and ommissions.

Page Number: Comment:

What is your occupation? ______________________________

Newsletter number of latest Technical Newsletter (if any) concerning this publication: ______________

If you wish a reply, give your name and address:

IBM branch office serving you ______________________________

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or 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.)

Note: Staples can cause problems with automated mail sorting equipment.
Please use pressure sensitive or other gummed tape to seal this form.

LY38-3021-0

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, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Department 63T
Neighborhood Road
Kingston, New York 12401

Fold and tape — Please Do Not Staple — Fold and tape

IBM®

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601

Cut or Fold Along Line