GC33-0066-5

Customer Information Control System/Virtual Storage (CICS/VS) Version 1 Release 5

Program Product

General Information

Program Numbers 5740-XX1 (CICS/OS/VS)
5746-XX3 (CICS/DOS/VS)

IBM

GC33-0066-5

Customer Information Control System/Virtual Storage (CICS/VS) Version 1 Release 5

Program Product

General Information

Program Numbers 5740-XX1 (CICS/OS/VS)
5746-XX3 (CICS/DOS/VS)

IBM

| Sixth Edition (May 1980)

| This edition applies to Version 1 Release 5 (Version 1.5) of the IBM program product Customer Information Control System/Virtual Storage (CICS/VS), program numbers 5746-XX3 (for DOS/VS) and 5740-XX1 (for OS/VS). Until the OS/VS version is released, the information applicable to that version is for planning purposes only.

| This edition is a major revision of GC33-0066-4, which is now obsolete. Significant changes and additions to the text are indicated by vertical lines to the left of the changes.

Information in this publication is subject to change. Changes will be published in new editions or technical newsletters. Before using this | publication, consult the latest IBM System/370 and 4300 Processors | Bibliography, GC20-0001, to learn which editions and technical newsletters are current and applicable.

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

Publications are not stocked at the addresses given below; requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for reader's comments is provided at the back of this publication; if the form has been removed, comments may be addressed either to:

| International Business Machines Corporation,
| Department 812HP,
| 1133 Westchester Avenue
| White Plains, New York 10604.

or to:

IBM United Kingdom Laboratories Limited,
Programming Publications, Mail Point 095,
Hursley Park,
Winchester, Hampshire SO21 2JN, England.

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 manual introduces the CICS/VS program product primarily to data processing management and to system/application designers. Readers are assumed to have had some experience of computer applications -- either in management, programming, or operations.

Selected chapters of this manual also serve to introduce CICS/VS to other specific groups of personnel, namely: users of applications to be run under CICS/VS (Chapter 2); programmers responsible for coding CICS/VS applications (Chapter 3); programmers responsible for installing and preparing a CICS/VS system (Chapter 4); and the operations personnel responsible for running a CICS/VS system (Chapter 5). A brief introduction to how CICS/VS works is given in Chapter 6; a summary of the system requirements for running CICS/VS is given in Appendix A; and a list of terminals and subsystems supported by CICS/VS is given in Appendix B. A summary of the new facilities introduced for CICS/VS Version 1 Release 4 is given in Appendix C; for Version 1 Release 4 Modification 1 in Appendix D; and for Version 1 Release 5 in Appendix E.

In this publication, the term VTAM refers to ACF/VTAM, to ACF/VTAME (CICS/DOS/VS only), and to the Record Interface of ACF/TCAM (CICS/OS/VS only). The term TCAM refers both to TCAM and to the DCB Interface of ACF/TCAM. The term BTAM refers to BTAM (CICS/OS/VS only) and to BTAM-ES (CICS/DOS/VS only).

For further information concerning CICS/VS, see the following IBM publications:

Customer Information Control System/Virtual Storage (CICS/VS) Version 1 Release 5

System/Application Design Guide, SC33-0068

System Programmer's Reference Manual, SC33-0069

System Programmer's Guide (DOS/VS), SC33-0070

System Programmer's Guide (OS/VS), SC33-0071*

Application Programmer's Reference Manual (Command Level), SC33-0077

Application Programmer's Reference Manual (Macro Level), SC33-0079

Application Programmer's Reference Manual (RPG II), SC33-0085

IBM 3270 Guide, SC33-0096

IBM 3600/3630 Guide, SC33-0072

IBM 3650/3680 Guide, SC33-0073

IBM 3767/3770/6670 Guide, SC33-0074

IBM 3790/3730 Guide, SC33-0075

Operator's Guide, SC33-0080

Messages and Codes, SC33-0081

Entry Level System User's Guide (DOS/VS) SC33-0086

Problem Determination Guide, SC33-0089

Diagnosis Reference, LC33-0105

Data Areas (DOS/VS), LY33-6033

Data Areas (OS/VS), LY33-6035*

Application Programmer's Reference Summary (Command Level), GX33-6012

Master Terminal Operator's Reference Summary, SX33-6011

Program Debugging Reference Summary, SX33-6010

Master Index, SC33-0095*

---

* Available at the same time as CICS/OS/VS Version 1 Release 5

Contents

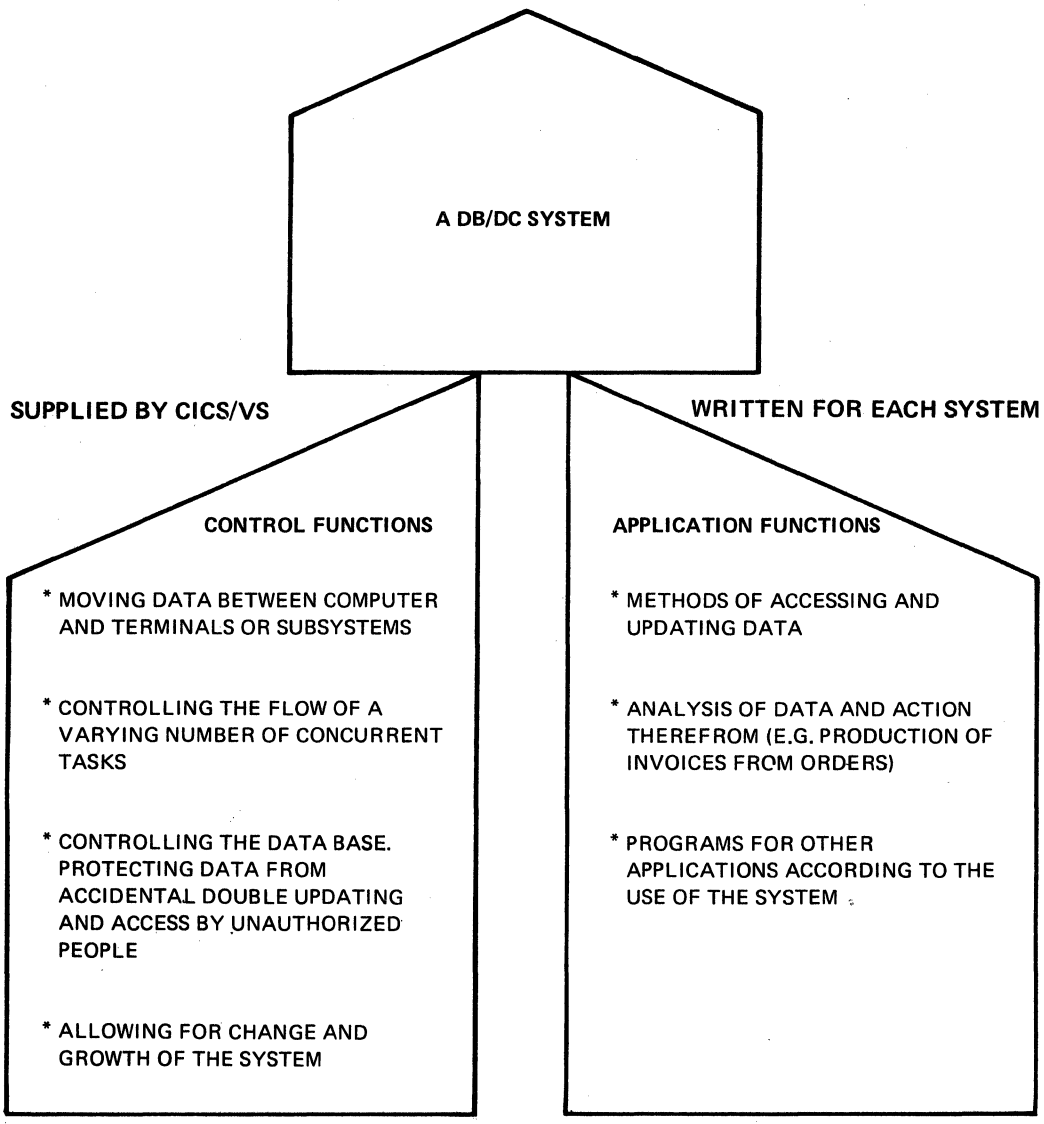


Figure 1. The Part CICS/VS Plays in a DB/DC System

Chapter 1. Introduction

In modern business computer applications, there is an increasing need for the computer to be more responsive to a user's information processing requirements. For example, a manager who needs information on which to base a decision, does not want to wait for an overnight batch run before he gets his answer — he wants the information in seconds. In other words, he wants his information online. To get online information he needs what is known as a data base/data communication (DB/DC) system - "data base" because the information (the data base) is held within the computer system; "data communication" because the information can be communicated from the computer to the terminals.

It is not easy for a user to develop all the components of a DB/DC system. With CICS/VS it is not necessary because CICS/VS supplies the basic components.

When developing a DB/DC system there are basically two types of problems that have to be solved. First there are the control problems, for example, moving the data between the computer and the terminals; preventing two people from trying to update one data item at the same time; matching the speeds of the lines and the terminals with those of the processing unit, and many more. Second there are the application problems. These are concerned with the purpose for which the computer is to be used (its application) rather than the mechanics of the system itself. Application problems are typically the manner in which the information is to be entered, displayed, and updated, and the production of meaningful statistics from the data.

By providing a standard set of services to solve the control problems, CICS/VS enables an online system to be set up quickly and efficiently, because those who set up the system can concentrate on the application problems - the problems directly concerned with their data and their business. Figure 1 shows the part CICS/VS plays in a DB/DC system.

What is CICS/VS?

The Customer Information Control System/Virtual Storage (CICS/DOS/VS or CICS/OS/VS) is a program product that controls online Data Base/Data Communication (DB/DC) applications. CICS/VS provides (1) most of the standard functions required by application programs for communication with remote and local terminals and subsystems; (2) control for concurrently running user application programs serving many online users; and (3) data base capabilities, in conjunction with the IBM data base management method, Data Language/1 (DL/I). CICS/VS Version 1 Release 5, and the application programs under its control, run under various virtual storage operating systems: VSE/Advanced Functions, OS/VS1, and OS/VS2 (MVS). (CICS/VS Version 1 Release 5 does not run under DOS/VS or SVS.) See Appendix A for details.

CICS/VS is a general purpose DB/DC control system that can be tailored to the needs of most combinations of concurrent online applications serving a network consisting of a wide variety of terminals and subsystems. Furthermore, the data files that are used by the online application users may also, in general, be used by batch programs in other partitions or regions.

The CICS/DOS/VS product includes also the CICS/DOS/VS Entry Level System (ELS), which gives the user the option of installing a simpler version of the full CICS/DOS/VS system. CICS/DOS/VS-ELS provides a first-time user of CICS/VS with a simplified system that compared with CICS/DOS/VS is easier to install and use. It is recommended for the user who does not require the concurrent processing of many applications. If at any time the user wishes to progress to a full CICS/DOS/VS system, application programs written for ELS will be compatible with CICS/DOS/VS.

What Does CICS/VS Do?

When a CICS/VS system has been installed, CICS/VS controls and services the system in the following manner. Whenever a request for some use of the system is sent from a terminal, CICS/VS identifies the application program required for the particular job. It loads the application program (if it is not already loaded), and starts a task to execute the application program. The application program can then send messages to the terminal and receive replies from the terminal until it has finished processing. When the application is complete, the program ends the task and returns the terminal to a standby state. During execution of the application program, CICS/VS facilities will have been requested by the program to handle such jobs as transmission of data between the program and the terminal, and accessing the data base. Thus CICS/VS is both controlling the overall flow of the online system and supplying an interface between the application programs of the online system and the operating system of the computer.

CICS/VS Intercommunication Facilities

CICS/VS can communicate with other online CICS/VS systems through a set of facilities generically termed the CICS/VS intercommunication facilities. There are two modes of intercommunication: intersystem communication and multiregion operation.

Intersystem communication is the facility that enables two CICS/VS systems to communicate with one another through SNA networking facilities. An application running in one system can access files and DL/I data bases, and initiate processing, on a connected CICS/VS system. The facility allows installations to distribute system resources such as application programs, files, and data bases in a network, without the need for locally required resources to be attached to the system requesting access to them.

The intersystem communication facilities of CICS/VS Version 1 Release 5 are also available for communication between CICS/VS and IMS/VS systems.

Multiregion operation allows CICS/VS regions within the same system to communicate with one another. It differs from intersystem communication in that SNA networking facilities are not required. Multiregion operation has several significant potential advantages for the user. For example, it allows new CICS/VS application programs to be tested concurrently with the production operation of CICS/VS, using a separate test region and so reducing possible impact on production without preventing access to existing resources and without requiring terminals to be dedicated to the test region. Multiregion operation was introduced in CICS/VS Version 1 Release 5, and the facilities it offers and potential uses are described in more detail in Appendix E.

Types of DB/DC Applications Supported By CICS/VS

Nearly all DB/DC applications correspond to one (or more) of the following generic types:

- Online inquiry
- Online inquiry-with-update
- Online data entry
- Remote batch processing
- Online message routing

All of these types of application can be handled concurrently by CICS/VS.

ONLINE INQUIRY APPLICATION

Provides the end-user with ready access to information in centralized files (the data base). In response to an operator's inquiry message, the application sends back a reply message containing data retrieved from the central files. This type of application requires the computer's response time to be short and is the simplest type of application, mainly because data files are not altered.

ONLINE INQUIRY-WITH-UPDATE APPLICATION (ORDER ENTRY)

Extends the inquiry type of application to the point where the end user can alter, delete, and store data in centralized files. This application also requires the computer's response time to be short. This application is more complex than simple inquiry because the integrity of the central files must be protected; safeguards are needed against system failure during the update process and against double updating.

The shared data base facility of CICS/OS/VS allows this type of application to operate on the same DL/I data bases as batch DL/I programs without loss of data base integrity.

ONLINE DATA ENTRY APPLICATION

Allows lines of data to be keyed at speed, with minimum interruption from the computer, and is similar to a card punching operation. It is used to create files at the central location for subsequent processing or access by other system users. The application checks (typically) that each entry (or line) of data is valid, and responds to the operator only to indicate an entry that is not valid. This application requires a very short response time so that the operator can continue entering data without delay, or can change an invalid entry with minimum disruption.

REMOTE BATCH PROCESSING APPLICATION

Allows data to be sent to the computer for batch processing — for example to perform a periodic update of the central files. The requirement here is not necessarily for a short response time; instead the remote operator needs the ability (1) to initiate the batch application that processes the data and (2) to direct the output of any batch application to his own terminal and/or other terminals in the network.

This type of application is often associated with remote terminals and subsystems that can store data — on disk, diskette, or on cards, for example. Data originally entered at the remote terminal can thus be stored and transmitted at high speed during off-peak hours. This technique is referred to as "batch store and forward".

ONLINE MESSAGE ROUTING APPLICATION

Handles the routing of messages from one terminal to other terminals in the network. No data processing or access to central files is required. If necessary, the application stores a message until each destination terminal is able to receive it.

How CICS/VS Helps You Set Up an Online System

CICS/VS provides the control and service functions needed to create a DB/DC system with a number of online applications. Thus the total amount of programming effort is reduced, and programmer productivity improved.

After the system has been designed, the programming effort is normally divided between two groups of programmers; the system programmers who install the system and get it running, and the application programmers who write the application programs that the system will use. (In some installations the same people may do both jobs.) CICS/VS has features to help both system and application programmers.

Some of the CICS/VS features that help the system programmer are:

- Pregenerated systems — reduce the amount of machine time (and programmer time) required to generate a system.
- Precompiled sample application programs — can be used to check that the system has been installed correctly.
- CICS/VS tables — make a system adaptable, for example to allow definition of the network configuration, line types, and file organizations.
- System modularity — allows a system programmer to choose or add only those functions needed to support a particular set of applications.
- System recovery functions — facilitate the recovery of the system from a variety of possible error. If desired, the CICS/VS recovery facilities can be extended by the system programmer to suit a particular installation's needs.

- System services — CICS/VS provides a variety of services that are required in most online applications, such as queuing and temporary data storage; message routing; terminal paging; data editing; time management.

- Data security – CICS/VS provides facilities to help prevent unauthorized access of information and has the flexibility to allow these facilities to be extended by the user, if required.

Some of the CICS/VS features that can help the application programmer are:

- Device independence and data independence — CICS/VS handles all communications and data accesses, so that, for most applications, the application programmer need be concerned only with the coding to perform the required application function; he does not have to handle the specific requirements of individual terminals or data storage devices.

- Choice of programming language — application programs can be written in American National Standard (ANS) COBOL, PL/I, RPG II, or Assembler language.

- Command-level (high-level) interface with CICS/VS — the application programmer needs to know little about how CICS/VS works: he requests CICS/VS functions for data access or for communications by issuing commands similar in appearance to those of the programming language being used.

- A command interpreter program that enables a programmer to enter and check individual commands interactively before they are executed.

- An execution (command-level) diagnostic facility (EDF) that enables a programmer to test a command-level application program in an interactive manner.

The same CICS/VS features that aid the setting-up of an online system also make the system very adaptable for future growth and change (of, for example, the processor, line disciplines, terminal types, operating systems, and intersystem communication) — with little or no impact on application programs.

How CICS/VS Helps an Online System Run Efficiently

Two vital areas in any online system are response time and recovery after error.

RESPONSE TIME

An objective of an online system serving a network of application users is that the computer should respond to each operator's input messages with a minimum of delay. Two of the CICS/VS features which make for shorter response times are: multitasking and priority processing.

- Multitasking. When an operator keys in a message (for example as part of an inquiry) CICS/VS immediately creates a task that is unique to the operator, and starts processing that task almost immediately. Several such tasks are processed concurrently by multitasking. CICS/VS controls its own multitasking and does not depend on the operating system's subtasking capability; therefore, tasks can be processed more rapidly and so the number of concurrent operator transactions can be increased.

- Priority processing. The fastest response can be given to those applications that need it — possibly at the expense of slightly longer response times for less critical applications. To do this, priority ratings can be allocated to each operator, to each terminal, and to each transaction. CICS/VS combines these priority ratings to determine the relative priorities of all tasks under its control. With CICS/VS Version 1 Release 5, further selective control over response time can be achieved by the use of multiregion operation. For information about this facility, see Appendix E.

System Control

Various tuning aids provide close control over the number and types of transactions being processed concurrently. A master terminal operator can monitor and control the network while the system is running.

ERROR HANDLING AND RECOVERY

If the processing work for an individual operator develops an error, the work done can be "backed out", without affecting other users: the affected files are restored to the state they were in before the processing began. Furthermore, an option can be specified at system initialization that under certain conditions and user control, allows for the immediate restart of the transaction associated with the processing in error.

After a system failure, an emergency restart facility backs out data base changes made by all tasks that were in progress at the time of failure.

System Design

There are many factors that may influence the design of a CICS/VS system. The subject is described in detail in the CICS/VS System/Application Design Guide, which contains chapters on the following topics:

- Application design
- Data communication design
- Data base design
- Recovery and restart design
- Performance design

- Intercommunication design

Two factors which must be considered in the design of all systems are:

1. Response time — some applications demand immediate responses; some applications can tolerate delayed responses. Also, the information contained in responses must be understandable and relevant to the user's needs.

2. Resource utilization — the system needs to be "balanced" to ensure that some online applications do not use so much processor time and storage that the response time for critical applications (and the ability to carry on processing in other partitions) is adversely affected.

Users of CICS/VS

The next four Chapters in this manual describe CICS/VS from the point of view of four groups of people:

- The application user departments who enter, retrieve, and make use of the data.
- The application programmer who writes the programs needed to make the data available, meaningful, and easy to handle.
- The system programmer who installs CICS/VS and sets up the complete DB/DC system.
- The operations staff who are concerned with the day-to-day running of the system.

Those responsible for the design of the system need an appreciation of all these chapters because they will be concerned with looking at the system from the point of view of each of these groups.

Chapter 2. CICS/VS for the Application User Department

CICS/VS brings the data processing capability of the computer into user departments — where input data is available and where output data is needed.

Under CICS/VS, many operators in different departments and in different locations can work with various types of terminals on independent applications at the same time. Normally, any one operator is not aware of CICS/VS — nor of the existence of other applications; most of the time, the operator interacts with online application programs that have been designed for the particular job in hand.

The operator need not be trained in data processing. The operator's procedure at the terminal (typically a display screen and keyboard) will have been determined by the systems designer or the programmer who wrote the application programs.

The following sections outline some of the characteristics of an online system under CICS/VS as seen by a user operator.

Because CICS/VS is a general purpose system that can be tailored to meet particular needs, the way a system appears to the user depends more upon the design of the system itself and the application programs it contains than on any features of CICS/VS. However there are a number of things that the user department and its management need to know so that they can cooperate in designing a system that meets their needs. The rest of the chapter covers these points and contains:

- An indication of the types of terminals available.
- A description of signing on to a CICS/VS system.
- An example of a job done at a terminal as seen by the operator.
- A discussion of response time and priority.
- A discussion of the security and integrity of data.

Operator Terminals

The type of terminal used by the operator can be chosen according to the needs of the application. For example, online applications such as inquiry and inquiry-with-update will typically require a display screen and keyboard so that reply messages can be displayed to the operator with minimum delay. On the other hand, in applications such as data entry or remote batch processing, a keyboard-printer type of terminal may be more appropriate.

CICS/VS supports a wide range of subsystems and terminals, which are listed in Appendix B.

Signing On

A terminal operator may be required to sign on to CICS/VS by keying in his name and a prearranged password; assuming his name is known to the system, and that the password is correct, CICS/VS allows the operator to proceed. (At sign-on time, CICS/VS also sets up priority and security keys for later use.) Alternatively, the operator can be required to identify himself by inserting an identity badge into a badge reader and entering a password.

For detailed information about security enhancements provided in CICS/VS Version 1 Release 5, see Appendix E.

Transactions

A session of application processing for an operator is called a transaction.

STARTING A TRANSACTION

To start a transaction, the terminal operator specifies the type of processing required by (usually) keying in a transaction identification code. The transaction identification code (which consists of one, two, three, or four characters) invokes the appropriate application program to perform the required processing.

In the following example of a conversational inquiry transaction, the transaction identification code consists of the four letters "SINQ" standing for student inquiry.

```
Operator enters:      SINQ
Terminal displays:  — WHAT IS STUDENT'S NO.?

(Pause)

Operator enters:      03534
Terminal displays:  —WHAT DATA IS WANTED?

(Pause)

Operator enters:      SUBJECT
Terminal displays:  —HISTORY —LATIN —LAW
```

It is worth noting that, to the operator, this appears as one transaction made up of three input messages and three responses. Unknown to the operator, however, the application programmer could have designed this as three separate CICS/VS transactions — a technique that makes better use of system resources and so helps CICS/VS to serve a greater number of concurrent users.

There are at least two other ways of initiating a transaction: (1) a terminal can be dedicated to one particular transaction so that the operator does not have to key in the transaction identification code; or (2) a transaction can be automatically started by the occurrence of a defined event — for example a print queue becoming full or a certain time of day being reached.

Note: In this chapter, the word "terminal" is used in its most general sense, that is, an input/output "device". It should be realized however that when an operator uses a terminal or device attached to a programmable system (such as the IBM 3600, IBM 3770, or IBM 3790), the operator is interacting with a subsystem program. A subsystem program, in turn, interacts with CICS/VS (and the transactions running under CICS/VS). In these circumstances, some of the actions that would generally be required from an operator might be performed by a subsystem program. Similarly, in the case of communication between CICS/VS systems, a transaction might be initiated to run on one CICS/VS system by an application already running on a connected CICS/VS system rather than through direct initiation by an operator at a terminal.

TRANSACTIONS AND TASKS

CICS/VS controls the processing of a transaction for an operator as a task. In this chapter, the following distinction is made between the terms transaction and task:

- Transaction means an application program (or programs) that can be used by a terminal operator. A given transaction can be used concurrently by one or more operators.
- Task means an execution of a transaction for a particular operator. A given task can relate to only one operator.

Response Times (Priority Processing)

Response times are potentially shortest for those tasks which have the highest priority. The priority that CICS/VS gives to a task depends on the following individual priorities:

- Terminal priority value (0 - 255)
- Operator priority value (0 - 255)
- Transaction priority value (0 - 255)

These priority values are assigned at the design stage according to the relative needs of users and their applications. For a particular operator, at a particular terminal, initiating a particular transaction, CICS/VS sums the individual priority values to make a task priority value (limited by CICS/VS to a maximum of 255); the larger the priority value, the higher the priority.

The user department can thus influence response times in the first instance by recommending suitable priority values for the department's terminals, operators, and transactions; and in the second instance, by initiating transactions from different terminals and by using different operator numbers.

Data Retrieval and Updating

CICS/VS transactions can access two kinds of data bases:

- Standard operating system data sets holding a data base (data set data bases).
- Data Language/I (DL/I) data bases.

DATA SET DATA BASES

The standard operating system data sets are processed by CICS/VS file control. File control relieves the programmer of buffer management, blocking and deblocking, access-method dependencies, provides browsing facilities and ensures the prevention of concurrent updates.

Using suitably designed applications under CICS/VS, an operator can retrieve, store, or update information in the system data set data bases.

Inquiries can be made directly or sequentially. This means that data can be obtained directly by name or by browsing forward or backward until the required data is found (sequential access).

DL/I DATA BASES

Using DL/I data bases allows a much greater degree of data independence than is given by file control. Used in conjunction with the IMS/VS, DL/I DOS/VS or DL/I Entry Program products, CICS/VS offers facilities for accessing DL/I data bases.

SHARED DATA BASES

CICS/OS/VS provides a facility that allows the sharing of DL/I data bases between concurrently active batch regions and CICS/OS/VS transactions, with preservation of the integrity of those data bases. Using the CICS/VS intercommunication facilities, it is possible for transactions running on different CICS/VS systems to access the same data bases.

Security

OPERATOR SECURITY

The system is able to check that an operator has the authority to use any transaction he tries to start. Every transaction can be given a security key (a number from 1 to 24), and, all operators can be given one or more security keys. Only if the security key of the transaction being initiated matches one of the security keys of the operator, does CICS/VS allow transaction processing to take place.

Further safeguards can be added, if warranted, in the application programs themselves. For example, the application program could require an operator to key in special passwords before allowing access to or change of data in the system files; also the application program could check that the transaction is being conducted from a terminal located in a department that is authorized to use that transaction.

For detailed information about security enhancements provided in CICS/VS Version 1 Release 5, see Appendix E.

PROTECTION OF DATA (DATA INTEGRITY)

Operator security measures (as already described) provide the primary means of preventing the intentional misuse of and accidental access to a user department's data files.

MVS users should note that the control over CICS/VS batch mode access to the CICS/VS data base can be provided through installation protection of the Authorized Program Facility (APF) library containing CICS/VS batch region controller programs.

Installations concerned with system wide security should consider the use of resource access control facilities for protection of the above library, data sets that make up the CICS/VS data base, and other sensitive system resources.

CICS/VS also protects the user's data from unintentional deletion or change arising from double updating, from transaction failure, or from system failure.

With CICS/VS Version 1 Release 5, further protection may be afforded by the use of multiregion operation. For detailed information about this facility, see Appendix E.

Double Updating

CICS/VS can prevent the loss of information that might otherwise occur when two or more operators attempt to update the same section of data at the same time.

By a technique called "exclusive control", CICS/VS ensures that one operator's updating of a system data set data base is finished before another's can start.

DL/I data bases are similarly protected, but different techniques are employed. Two mutually exclusive scheduling options are available; Intent Scheduling, and Program Isolation.

Intent scheduling ensures that a particular segment type of a DL/I data base is accessible for potential update by only one task at a time. Program Isolation, however, allows more than one task to have concurrent access, for potential updating, to a particular segment type of a DL/I data base, but ensures that only one task at a time can update a particular physical segment of the segment type being accessed. Furthermore, once a task has updated a particular physical segment, no other task can access that segment until the updating task has completed its processing and been descheduled. Other tasks scheduled to update that physical segment will be enqueued on the segment, until the descheduling previously described occurs.

Because of the enqueuing involved with program isolation scheduling, it is possible for a deadlock to occur. For example, task A updates physical segment 1 of segment type T, and then needs to update segment 2 of the same segment type. Concurrently, task B updates physical segment 2 of segment type T and then needs to update segment 1. The consequence is that each task is waiting for the other to deschedule and yet neither will be descheduled until its updating has been completed - hence deadlock. In such a case, one of the two tasks concerned is abnormally terminated.

The advantage of program isolation over intent scheduling is that it affords a much faster throughput in an online updating environment.

Transaction (Task) Failure

In the event of a transaction (task) failure, an optional CICS/VS function called dynamic transaction backout can be used to cancel (back out) all updates that have been performed by that task. The backout occurs within the same task and thus safeguards other tasks from the possibility of using corrupted data.

A CICS/VS function called transaction restart provides an optional facility that allows for the possible immediate restart of a task after transaction (task) failure and subsequent dynamic transaction backout. The facility is particularly useful after a deadlock has arisen from program isolation scheduling of access to DL/I data bases.

System Failure

After a system failure, CICS/VS can use an optional function called emergency restart to restore the data files of all interrupted transactions to the conditions they were in when those transactions started.

Another feature, called message protection, can postpone the transmission of an output message from a task until that task has ended successfully. This feature stops an erroneous "task complete" message (for example) from being sent to an operator in the event of a system failure occurring before the task has actually ended.

Other Transaction Functions

The earlier part of this chapter describes some of the functions that user departments may wish to specify for their applications. Other commonly used CICS/VS functions that are also of potential value to user departments include the following:

VERIFYING AND EDITING

Data entered by an operator can be checked for validity and can be edited so that, for example, only the numeric characters are used for inquiry or for updating files.

BATCHED DATA TRANSMISSION

The batched data transmission function is useful in applications that require the operator to key in large quantities of data at speed and without being interrupted by responses from the computer.

SENDING MESSAGES

Using a CICS/VS-supplied transaction, an operator can send messages to other operators, or terminals, for delivery at the earliest opportunity or at a specific time.

OBTAINING STATISTICS

Statistics that may be of day-to-day value to user departments include the numbers of transactions, input messages, and output messages per terminal. Any operator authorized to do so may obtain such statistics at a terminal.

CICS/VS also provides many other system statistics that can help management to make the most effective use of the system and to plan for future growth.

For detailed information about improved monitoring facilities provided in CICS/VS Version 1 Release 5, see Appendix E.

System and Configuration Changes

CICS/VS enables changes in the terminal configuration, data sets, data bases, and application transactions to be made easily and quickly by changing data in CICS/VS system control tables and without altering application programs. Furthermore, progressing to a CICS/VS system that has the capabilities of communicating with other CICS/VS systems can be effected by making changes to CICS/VS control tables, with little or no alteration to application programs.

Chapter 3. CICS/VS for the Application Programmer

Online application programs that are to run under CICS/VS can be written in ANS COBOL, PL/I, RPG II, or Assembler language.

In contrast to batch programs, most online application programs have the following characteristics: (1) they exist to provide fast computer services and data access to many operators concurrently; (2) requests for application program services occur randomly; and (3) some application programs may be required simultaneously by several operators.

To implement a DB/DC system with these characteristics requires many control functions that — without CICS/VS — an application programmer would find difficult to prepare for himself. Some of the control functions needed in all online DB/DC systems are listed below. These functions (and many others) are performed by CICS/VS and thereby free the programmer to concentrate on his application coding.

- Terminal management. The CICS/VS terminal control program:
  - Polls terminals to see if they have data to transmit; and addresses terminals to see if they can accept data.
  - Handles data link protocols.
  - Translates character codes to and from EBCDIC.
  - Detects and logs communication errors.
- Task management. The CICS/VS task control program:
  - Keeps track of each terminal's use of application programs (tasks).
  - Controls the execution of tasks in order of priority values associated with the transaction, the terminal, and the operator.
- Storage management. The CICS/VS storage control program:
  - Gets storage areas on demand from other management functions.
  - Releases storage areas when they are no longer needed.
- Program management.
  - Keeps track of which application programs are in use and where they are located.
  - Intercepts program checks so that they do not cause the entire CICS/VS partition or region to be shut down.

These four management functions are present in all CICS/VS systems and save the application programmer from many of the difficulties of system control coding. Other CICS/VS management functions used in most systems include: file management, time management, and journal management.

Application Programmer's Interfaces with CICS/VS

The application programmer can request CICS/VS functions through a high-level (command-level) or a low-level (macro-level) programming interface. (The CICS/VS functions available to the application programmer are described later in this chapter.)

Using the command-level programming interface, the application programmer makes requests for CICS/VS functions with commands. The interface is available to programmers who use ANS COBOL, PL/I, RPG II, or Assembler language.

Using the macro-level programming interface, the application programmer makes requests for CICS/VS functions with macro instructions in Assembler-language format. Existing programs that use CICS/VS macros will continue to run under Version 1 Release 5 of CICS/VS — normally with no need for recompilation. However, in creating new programs, programmers will almost certainly find it easier to use the command-level programming interface.

Only the command-level programming interface is available when using the CICS/DOS/VS Entry Level System or the CICS/VS intercommunication facilities.

For ANS COBOL, PL/I, and Assembler language programmers, details of the command-level programming interface are given in CICS/VS Application Programmer's Reference Manual (Command Level). For RPG II programmers, similar details are given in CICS/VS Application Programmer's Reference Manual (RPG II). Details of the macro-level programming interface are given in CICS/VS Application Programmer's Reference Manual (Macro Level).

COMMANDS

For ANS COBOL, PL/I, and Assembler language, a CICS/VS command contains the verb EXECUTE (or EXEC in its abbreviated form) and the system identifier CICS. This is followed by the function required and a sequence of options (depending on the particular function). For RPG II, because of constraints of the language, the command does not contain the system identifier CICS, and the function required precedes the verb EXEC.

Example of Command-Level Coding

The following example shows the use of an EXEC CICS command (in an ANS COBOL program) to write a record to a file.

```
IDENTIFICATION DIVISION.
PROGRAM-ID. SAMPLE.
DATA DIVISION.
WORKING-STORAGE SECTION.
01 FILEAS COPY FILEAS.
PROCEDURE DIVISION.
    -
    -
    -     (Set up fields in FILEAS)
    -
    -
    EXEC CICS WRITE DATASET ('FILEA') FROM (FILEAS)
         RIDFLD(NUMBI) END-EXEC

    (and so on.)
```

Command Interpreter

A command interpreter designed to help in writing and checking CICS/VS commands is available with CICS/VS Version 1 Release 5. The interpreter enables the operator of a display unit (with 1920 or more characters and supporting the 3270 data stream) to enter a command for immediate checking and optional execution. An application programmer can therefore construct a command interactively, thus using the command interpreter to complement the execution diagnostic facility (EDF), described later in this chapter, which enables an existing command-level program to be interactively tested.

In operation, the command interpreter takes the complete or partial command and, having checked its syntax, indicates default options, together with a diagnosis of missing or incorrect parameters. Once the command has been satisfactorily developed, the user may request its execution. In this way, both the syntax of the command and the result of executing it — including the screen display resulting from the operation — can be checked. The facility can also be used to create or inspect test data (for example, transient data) for use in application program testing.

Program Preparation

When they have been written, programs containing CICS/VS commands are prepared for execution by using a cataloged procedure provided by the system programmer. The procedure includes a translation step that yields a source program with the commands replaced by statements in the programming language being used together with any diagnostic messages. The translated source program is then compiled/assembled and link-edited in the usual way.

Use of the Command-Level Programming Interface

The command-level programming interface offers several advantages over the macro-level interface:

- An application execution (command level) diagnostic facility (EDF) for use on a 3270 device, to display the interaction between the application program and CICS/VS.
- Storage management is handled automatically; for most CICS/VS commands, the application programmer need not be concerned with the getting, freeing, or addressing of dynamic storage.
- Source programs tend to be shorter and more readable.
- The application programmer does not need to know the internal details of CICS/VS.
- The program-preparation process is able to give diagnostic information both in the translation step and the compilation/assembly step.

MACROS

The CICS/VS macro instructions are in assembler format. The name field must be blank if the macro is being used in a PL/I program, or it can contain sequence numbers in columns 1 through 6 in an ANS COBOL program. The operation field always contains the characters "DFH" followed by up to five more characters to complete the macro name. Operands specify the functions to be performed.

Program Preparation

Programs containing CICS/VS macros are generally prepared for execution by using a cataloged procedure provided by the system programmer. For ANS COBOL or PL/I programs, the procedure includes two preprocessing steps that yield a source program consisting entirely of high-level statements; the preprocessed source program is then compiled and link-edited in the usual way. For Assembler-language programs, the procedure consists of assembling and link-editing.

Functions Available to the CICS/VS Application Programmer

The categories of functions available to the application programmer are:

- Data base operations.
- Data communication operations.
- Control operations.
- Error-handling, debugging, and recovery.
- Built-in functions.

DATA BASE OPERATIONS

For data base operations, CICS/VS uses:

- The IBM data base management method — Data Language/1 (DL/I).
- The IBM data management access methods — indexed sequential access method (ISAM), direct access method (DAM), and virtual storage access method (VSAM).

DL/I Data Base Access

CICS/VS enables online application programs to access DL/I data bases via the DL/I data management method using a DL/I call interface. For CICS/DOS/VS users, there is also a command-level DL/I interface (COBOL and PL/I application programs only). (Online DL/I is implemented on DOS/VS systems by the DL/I DOS/VS program product or DL/I Entry program product running with CICS/DOS/VS; and on OS/VS systems by the IMS/VS data base system program product running with CICS/OS/VS.)

Note: A DL/I data base may be on the same or a connected CICS/VS system. However, users of CICS/DOS/VS should refer to the program directory for the current release of CICS/DOS/VS for a statement regarding availability of DL/I product support.

DL/I is a high-level data-management method that allows the CICS/VS application programmer to request data by name (rather than by location). DL/I also gives application programs greater data independence; this means that the application program does not have to deal with storage space problems caused by adding data to, or deleting data from, the data base.

Data Management Access Methods

CICS/VS enables online application programs to process records on data bases held in ISAM, DAM, and VSAM data sets with:

- Blocked and unblocked records.
- Fixed-length and variable-length records.
- Direct (random) access and sequential access.

Note: Such data set data bases may be on the same or a connected CICS/VS system.

Records in these data sets have the standard ISAM, DAM, or VSAM format and can, therefore, be created or processed by online application programs or by offline batch programs. Data can be retrieved sequentially (for "browsing") or by direct access, without the application programmer being involved with details of data set organization, actual locations, retrieval methods, block or record sizes, or logical relationships between data sets.

Note: CICS/VS does not support the ISAM interface program.

CICS/VS facilities (at macro level) permit data sets to be created also for indirect access. This enables an application programmer to

specify the field in the records of one data set (an index data set) that is to be used as the search key to access another data set. Data sets that are logically related in this way become, in effect, a data base for the application programmer.

DATA COMMUNICATION OPERATIONS

For data communication operations, CICS/VS Version 1 Release 5 uses the basic telecommunications access method (BTAM), the virtual telecommunications access method (VTAM), and the telecommunications access method (TCAM).

Terminal Control Functions

These functions include:

- Read from or write to a terminal
- Break connection with a terminal
- Wait for completion of a preceding operation

Many of these functions require application programs to include device-dependent code. Application programs can however be written without device-dependent code by the use of Basic Mapping Support (BMS) functions instead of terminal control functions; see the next section.

Basic Mapping Support (BMS) Functions

As distinct from terminal control functions, BMS functions (in their simplest form) give the application programmer a measure of device independence. Using BMS functions, the application programmer can handle data streams to and from a terminal without concern for the terminal's hardware characteristics.

For example, in communications with a 2740 typewriter terminal, the application program does not need to insert new-line characters into an output stream; nor does it need to remove them on input.

BMS Maps. Many BMS input and output functions are based on the use of BMS maps. A map is (usually) assembled separately from the application program and, in it, a programmer can include: (1) the format of input and output data (that is, how the data is presented to the operator); (2) fixed data, including page heading text, page trailer text, and text that identifies variable data (for example, "Name = ", and "Address = "); (3) the symbolic names of data fields within the format; and (4) the device type to which the map relates.

The application program then refers to the name of the appropriate map in its input and output operations, and handles data only in terms of the symbolic field names.

The use of a BMS map gives an application program a degree of format independence. This means that the format of input and output data (and other attributes such as the fixed data, image brightness, and cursor

position) can be changed — by changing the map — while leaving the application program unchanged.

A separate program product, Screen Definition Facility/Customer Information Control System (SDF/CICS) is available. SDF/CICS is an online application development program for the CICS/VS application programmer who wants to define or edit BMS maps interactively. The SDF program product runs as a CICS/VS application program.

Device Independence. Device independence means that an application program can be written without concern either for the format of input/output data or for the particular device with which it is communicating. Device independence can be achieved with a group of maps -- each map in the group relating to a particular device. The application program then needs to refer only to the name of the map group in its input and output operations.

This means, for example, that when new or different device types are installed, new maps can be added to the map group, but -- provided the same symbolic field names are used — the application program itself remains totally unchanged.

Terminal Paging. An application programmer can create and transmit output messages (with or without the use of maps) that are larger than can be physically displayed at a receiving terminal. The terminal paging facility consists of a set of commands available to the terminal operator for retrieving "pages" of messages in any order; that is, in the order they were prepared or by skipping forward or backward through the output pages.

Message Routing. This facility, which also can use maps, enables an application program to send an output message to one or more specific terminals or operators that are not connected to the application program's task. The application programmer is able to specify that messages be delivered immediately or at a particular time. (A "message switching" facility, available to operators, utilizes this BMS message routing function.)

Data Interchange Functions

Batch data interchange functions enable an application programmer to read batch data from, and write batch data to, a named data set in an outboard controller. (The term "outboard controller" is a generalized reference to a programmable subsystem such as the IBM 3790 or the IBM 3770 programmable terminals, which use SNA protocols.)

The value of batch data interchange functions to the application programmer is that he does not have to be aware of SNA protocols, or the format of data streams in order to communicate with the batch control function of an outboard controller.

CONTROL OPERATIONS

The following groups of functions enable an application program to control the way in which transactions are processed.

Interval Control Functions

Interval control functions enable the application programmer to initiate and to control time-dependent actions. For example:

- Wait (delay processing) for a specified interval, or until a specified time of day is reached.
- Signal when a specified period has elapsed or when a specified time of day is reached.
- Tasks may be automatically initiated at specified times, or at specified intervals of time.

Task Control Functions

In addition to scheduling CICS/VS transactions according to their priorities, CICS/VS task management enables the application programmer to influence the scheduling of the associated task either directly or indirectly. For example, he can postpone the execution of his application program until a specified event has occurred, or until a particular resource is available.

Program Control Functions

Program control functions enable application programs to pass control to each other — regardless of whether the programs were written in ANS COBOL, PL/I, RPG II, or Assembler language. The ways in which this can be done include:

- Linking to another program, expecting control to be subsequently returned. (The program linked-to is regarded as being at a lower logical level.)
- Transferring control to another program without expecting control to be returned. (The program to which control is transferred is regarded as being at the same logical level.)
- Returning control to a program at the next higher logical level.

Program control functions also allow designated programs to be explicitly or implicitly loaded and deleted.

Storage Control Functions

CICS/VS automatically controls the allocation of storage for itself and for application programs. This function is performed by the storage control program. However, an application program can also use storage control functions to provide work areas not provided automatically. These functions include:

- Acquisition of main storage and initialization to any required bit pattern such as blanks or zeros.
- Release of storage.

Transient Data and Temporary Storage Operations

CICS/VS provides mechanisms for queuing and also for temporarily storing data: transient data control and temporary storage control — both of which are independent of the more usual file control functions. The mechanisms can also be invoked when the system is communicating with another CICS/VS system.

Transient Data Control Functions. Transient data functions enable an application program to transfer data to and from predefined destinations. A destination is addressed symbolically and can be either an intrapartition destination or an extrapartition destination. (The destination name is independent of destination type and is predefined in a destination control table.)

- An intrapartition destination is a queue of data that is used subsequently as input data to another task within the CICS/VS partition or region.

  Optionally, when data is sent to an intrapartition destination and the number of items in the queue reaches a predefined trigger level, a task can be automatically initiated to process the queued data. (The trigger level and the transaction would normally be specified by the system programmer.)

- An extrapartition destination is a sequential data set. Data queued in an extrapartition destination can be used either (1) as input data within the CICS/VS partition or region; or (2) for offline processing (to be printed, for example).

Temporary Storage Control Functions. Temporary storage control functions enable an application program to temporarily store and retrieve data by dynamically-created name (which means that data names do not have to be predefined). Data can be retrieved by other transactions (or by the same transaction that stored it) either in sequence, or directly by name. (Temporary storage can be regarded as a "scratchpad" facility and can exist in main storage or on auxiliary storage.)

Some of the ways in which temporary storage control differs from transient data control are:

1. Data in temporary storage can be retrieved in any sequence (as well as in the storing sequence).

2. Data in temporary storage is not deleted in the process of reading, and can therefore be used by several tasks.

3. Data in temporary storage can be altered.

4. Temporary storage control cannot initiate a task.

Recovery of Transient Data and Temporary Storage. Both transient data (intrapartition) and temporary storage (on auxiliary storage) may be recovered (optionally) during CICS/VS emergency restart following an abnormal termination of the system, or a transaction abend.

ERROR HANDLING, DEBUGGING, AND RECOVERY

This section outlines those functions available to assist the application programmer in the general areas of error handling, debugging, and recovery.

Error Handling

An application programmer can handle exceptional conditions (such as file-end, or identification error) by specifying what conditions are to be checked, together with the names of routines that are to deal with those conditions. He can handle abnormal task terminations in a similar way.

Debugging Aids

Debugging aids available to the application programmer include: an interactive execution diagnostic facility (EDF) for programmers writing application programs to the command-level interface; trace control; dump control; and a formatted dump facility.

Execution (Command Level) Diagnostic Facility (EDF). This facility enables a programmer to test command-level application programs using a 3270 terminal.

EDF intercepts every CICS/VS command immediately prior to its execution and allows the programmer to modify it, if required. The intercepted command is displayed to the user on a 3270 display screen in source form, complete with keywords, options specified, and values of keyword arguments. The user then examines the displayed information and amends it, if necessary, before execution. The command is then executed.

After execution, but before control is passed back to the application program, the command is once more intercepted and displayed for the user to show the response and any argument values set by CICS/VS. At this point the user has the opportunity to modify returned values so displayed and to override the response returned from CICS/VS.

The EDF is extended in CICS/DOS/VS Version 1 Release 5 to support the command-level DL/I interface in addition to the existing CICS/VS commands. Because the DL/I commands can have many more operands than CICS/VS commands, and could therefore overflow the space available on one screen, a scrolling function is provided for the list of options in the command display panel.

Trace Control. The CICS/VS trace control function can create a trace table that records the execution of CICS/VS requests. The application program can generate requests that:

- Start and stop the recording of trace information about the program in the trace table.
- Select which CICS/VS functions are recorded; for example, the application program can specify that only the CICS/VS terminal control and BMS functions in the program are to be traced or that all CICS/VS functions are to be traced.

Dump Control. An application programmer (for example, in an exceptional condition routine) can request that the contents of selected areas of storage be stored on a sequential data set for printing as a dump. The selected areas can include: main storage areas related to the requesting task; main storage areas associated with a transaction and its associated task; CICS/VS control tables; and, particularly, the trace table which can assist in debugging. For further details of dump control see the CICS/VS Application Programmer's Reference Manual (Macro-Level).

Formatted Dump Program. The complete output from the formatted dump program contains a dump of the CICS/VS partition or region. The program is intended to make it easier to diagnose errors particularly by performing automatically the mechanical process of finding the control blocks in storage. Each control block is printed separately in the dump, and is preceded by a heading; for some of the blocks the important fields are printed by name. In addition the dump may contain error messages where certain easily detectable errors have been found.

Alternatively, or in addition, the formatted dump program may issue a SNAP or PDUMP macro to take a "snapshot" of the CICS/VS partition or region. This is done before the formatted dump itself is printed in case the formatted dump program fails to complete successfully. The output produced depends on the options specified.

The formatted dump program is usually invoked each time CICS/VS terminates abnormally, or because the operator requests a dump at system shutdown.

For further details of the formatted dump content, invocation and "snapshot" options see the CICS/VS Problem Determination Guide.

Recovery

Most recovery functions are under the control of the system programmer. However, the application programmer can exercise a degree of control towards recovery by way of journaling and synchronization point functions.

Journaling. In addition to automatic journaling performed by CICS/VS (controlled by data in control tables supplied by the system programmer), the application programmer can send any data he wishes to a journal.

A journal is a special-purpose sequential data set which can be used to provide (for example) an audit trail of operator and system actions, or a means of recovering superseded data.

Synchronization Points. During normal execution, CICS/VS makes a record of all protected data that is being changed so that, in the event of an uncontrolled shutdown those changes can be backed out during an emergency restart, or in the event of task failure, or in response to a ROLLBACK command. Such records are released at the end of each task, because there is no need to back out changes that have been successfully completed.

However, for long tasks, the application programmer can specify intermediate points in his program at which updates or modifications are logically complete. These points are called synchronization points (syncpoints), and denote to the CICS/VS recovery/restart facilities that the changes up to this point do not need to be backed out in the event of an emergency restart after a system failure.

CICS/VS SAMPLE PROGRAMS

The CICS/VS Source Statement Library includes the source code for several sample online application programs. The sample programs give coding examples for commonly required applications such as: simple online inquiry; browsing; adding a record to a file; and updating.

The sample programs are coded in ANS COBOL, PL/I, RPG II, and Assembler language (with CICS/VS commands): supporting maps and tables are also provided.

CICS/DOS/VS ENTRY LEVEL SYSTEM (ELS)

A CICS/DOS/VS application programmer who is new to online applications, or whose needs are uncomplicated, does not have to learn or to use the full range of CICS/VS functions. Instead, he can use the CICS/VS functions as described in the CICS/VS Entry Level System User's Guide (DOS/VS).

PROGRAMMING TECHNIQUES

Details of programming techniques are given in the following manuals:

- CICS/VS System/Application Design Guide
- CICS/VS Application Programmer's Reference Manual (Command Level) (for ANS COBOL, PL/I, and Assembler language programmers)
- CICS/VS Application Programmer's Reference Manual (RPG II)
- CICS/VS Application Programmer's Reference Manual (Macro Level) (for ANS COBOL, PL/I, and Assembler-language programmers)
- CICS/VS Entry Level System User's Guide (DOS/VS)

Reenterability

One of the basic requirements of an online application program is that it shall be able to serve many end-users concurrently. If multiple copies of a program were to reside in main storage at the same time (one copy for each user task), a considerable amount of storage space would be needed and response times would be long. To use storage more efficiently, and to achieve shorter response times, application programs need to be "reenterable." A completely reenterable program is one that does not modify itself and does not store data within itself.

Ideally, application programs should be completely reenterable in order to obtain maximum performance benefits. However, CICS/VS does not require an application program to be completely reenterable — it need only be "quasi-reenterable." This means that it must not permanently change itself between requests for CICS/VS functions. (Any storage

space required by the program to carry data across requests for CICS/VS functions must be outside the program, in task-related areas provided by CICS/VS.)

Quasi-reenterability allows a single copy of an application program to be executed by many tasks concurrently. A portion of a quasi-reenterable program can be executed by one task until it requests a CICS/VS function (such as a file access). While that task waits for the CICS/VS function to be performed, another task that is ready can execute a different (or the same) portion of the same program until it, too, requests a CICS/VS function. The whole process is known as "multitasking" and is controlled by the CICS/VS Task Control Program.

LANGUAGE CONSIDERATIONS

The use of CICS/VS requests in an ANS COBOL, PL/I, or RPG II application program precludes the use of certain high-level language features. Language features such as those related to input and output, opening and closing of files, and multitasking should not be used because these functions are performed by CICS/VS.

Chapter 4. CICS/VS for the System Programmer

CICS/VS is a general purpose system that can be tailored to the particular needs of the user. To obtain the best performance, the system programmer should select only those functions required by the applications to be run under CICS/VS. (Information about which functions and facilities are best suited to particular installations and types of application are given in CICS/VS System/Application Design Guide. The information also contains guidance for system programmers designing a system, the resources of which will participate in communication with other CICS/VS systems.)

To install CICS/VS quickly, the system programmer can use a pregenerated system (called a starter system), which provides a working base. Subsequently, the system can be changed as new application needs become evident.

Details of what operating system features are required, how to install a CICS/VS system (with and without a starter system), and how to prepare and run application programs are contained in CICS/VS System Programmer's Guide (DOS/VS) and CICS/VS System Programmer's Guide (OS/VS).

The following section outlines some of the main aspects of installing CICS/VS.

Installation

The major activities in the process of installation outlined here are:

1. Preparation of machine-readable information
2. CICS/VS system generation
3. CICS/VS table preparation

PREPARATION OF MACHINE-READABLE INFORMATION

The CICS/VS program product is distributed on tape and includes the following items of machine-readable information:

- A CICS/DOS/VS or CICS/OS/VS starter system, which includes preassembled (pregenerated) versions of all CICS/VS programs. (Also available for the DOS/VS user only is a system called the CICS/DOS/VS Entry-Level system (ELS), which includes preassembled versions of programs and also a full set of tables to run a small configuration.)

  The pregenerated versions of CICS/VS systems also include some preassembled system control tables and preassembled sample application programs.

- Source code for all CICS/VS programs

- Macro libraries for (1) system generation; (2) table preparation; (3) assembling BMS maps; and (4) assembling CICS/VS macros contained in ANS COBOL, PL/I, and Assembler-language application programs.

- Translator programs (object code) for translating CICS/VS commands (EXEC CICS requests) contained in ANS COBOL, PL/I, RPG II, and Assembler language command-level application programs.

The machine-readable information must be transferred to libraries on disk before system generation and table preparation can be performed.

CICS/VS SYSTEM GENERATION

The purpose of CICS/VS system generation is to establish assembled and link-edited system programs in the CICS/VS program library. Programs to be established in the library include:

- Online system control programs, such as the task control program, terminal control program, and program control program.

- Online system service programs, such as the sign-on/sign-off program, time-of-day program, and BMS programs.

- Offline system utility programs, such as the preprocessor and translator programs and a dump formatting program.

The online system control programs can be generated with a variety of options — and any one program can be generated several times with different combinations of options. Subsequently, when CICS/VS is initialized, the required version of each system control program can be selected.

There are two approaches to the process of CICS/VS system generation:

- Use of the CICS/DOS/VS or CICS/OS/VS starter system.

- Assembly of all required CICS/VS programs from the IBM-supplied source code.

Either, or both, of these approaches can be used to create a CICS/VS system.

Use of a Starter System

Use of a starter system saves a considerable amount of machine time that would otherwise be needed to generate a CICS/VS system by assembly. A starter system gives the user a ready-to-use system that greatly reduces the need for assembly.

The CICS/DOS/VS or CICS/OS/VS starter system provides at least one version of all system programs; some of them are provided in three or four different versions. The program versions provided have been chosen to meet the needs of the majority of installations — they include popular and frequently used options.

Use of CICS/DOS/VS Entry Level System (ELS)

The CICS/DOS/VS ELS provides a limited set of pregenerated system programs (and system tables) — sufficient to establish a working system controlling IBM display terminals working through BTAM and/or VTAM, and thus does not require a system generation process to be performed. This reduces the level of knowledge and skill required to understand, install, operate, and maintain ELS.

System Generation By Assembly

The system programmer who wishes to create versions of CICS/VS system programs with particular combinations of options not provided by the starter system can create those versions by the process of system generation by assembly, and then add them to the library of programs that have already been generated. The system programmer may wish to do this in order to tailor the system to meet the precise needs of the proposed applications — without taking up space with unwanted functions.

System generation is a two-stage process:

Stage 1
: In stage 1, the system programmer writes and assembles system generation macros in which he specifies the required system programs. (The system generation macros are described in the CICS/VS System Programmer's Reference Manual.) The output of stage 1 is a DOS/VS or OS/VS job stream.

Stage 2
: The output of stage 1 is run as a series of job steps in which the system control programs are assembled, link-edited, and stored in the CICS/VS program library.

When writing system generation macros to specify which CICS/VS system control programs are to be generated, the system programmer should realize that certain control programs are mandatory. A comprehensive list of the mandatory control programs is given in the CICS/VS System Programmer's Reference Manual. Among the mandatory system control programs are:

- Task control program
- Terminal control program
- Program control program
- Storage control program

Versions of these, and all other CICS/VS programs, are specified by means of appropriate system generation macros, which can include options relevant to each type of program. For example: the terminal control program can optionally support various telecommunication access methods and a wide range of device types, including those on other CICS/VS systems with which the system may communicate; the program control program can optionally provide the trace function for use by application programmers; and the storage control program can optionally be generated to include the ability to attempt storage recovery in the event of system failure.

System generation macros for system control programs also include an option that allows several versions of a control program to be generated, each with a different combination of options, and each uniquely identified by a two-character suffix. This facility enables specific versions of system control programs to be invoked at system initialization time — a feature that makes for great flexibility.

USER EXIT ROUTINES

Many CICS/VS management programs contain exit points from which user-written routines can be given control. In earlier versions of CICS/VS, it was customary to assemble user exit routines with the CICS/VS programs; CICS/VS Version 1 Release 5 provides a new mechanism that makes inline assembly unnecessary and permits dynamic control of exit routines. This means that the user can prepare more than one exit routine for a single CICS/VS exit point; the flow of control can then be determined dynamically according to circumstance. This gives the user greater flexibility in customizing CICS/VS to the unique requirements of the installation. Most existing exit routines can be easily adapted for the new mechanism.

CICS/VS TABLE PREPARATION

The system programmer describes all elements of the data base/data communications environment in tables. CICS/VS enables the user to maintain a number of versions of tables (as with system control programs) through the use of suffix characters. This allows the system programmer to maintain special versions (for example) for peak loads, for testing, or for special shift work, in addition to the normal operational tables.

Tables are prepared by assembling appropriate macro instructions as described in the CICS/VS System Programmer's Reference Manual. New tables can be assembled at any time before a CICS/VS system initialization.

Some CICS/VS tables are considered essential, and others optional. Some of them are described as follows. For a full list of CICS/VS tables, essential and optional, see the CICS/VS System Programmer's Reference Manual.

System Initialization Table (SIT)

A system initialization table (SIT) contains user-specified data that will control a system initialization process; in particular, an SIT can identify (by suffix characters) the user-specified versions of CICS/VS system control programs and CICS/VS tables that are to be loaded.

The system programmer can assemble several SITs — each SIT itself being identified by suffix characters. Then, whenever the system is initialized (see Chapter 5), the required SIT can be specified by its suffix. Parameters in the chosen SIT can also be overridden at system initialization time, thus giving the user even greater flexibility.

Terminal Control Table (TCT)

A terminal control table (TCT) is assembled to describe a configuration of terminals, logical units, or other CICS/VS systems in a CICS/VS network with which the CICS/VS system may communicate.

In general, there is an entry in each TCT to describe:

- Each communication line group control unit (BTAM and TCAM networks): device address; device-dependent options; and error recovery options.
- Each communication line (BTAM networks): access method; terminal types on the line; and device-dependent options.
- Each terminal or logical unit: terminal type; terminal identifier; terminal priority; and terminal address.
- Each remote CICS/VS system: local name; remote name.

Several TCTs can be assembled and identified by suffix characters.

Program Control Table (PCT)

A program control table (PCT) is assembled to define all transactions that may be processed by the system. Each transaction, including those on remote systems, is described in a table entry that includes as appropriate:

- Transaction identifier
- Transaction priority value.
- Security key.
- Name of program that (initially) processes the transaction.
- For a remote transaction, the name of the remote system.

Several PCTs can be assembled and identified by suffix characters.

Processing Program Table (PPT)

A processing program table (PPT) is assembled to define all application programs that are valid for processing under CICS/VS. Each valid application program is described in a table entry that includes:

- Program name
- Source coding language (ANS COBOL, PL/I, RPG II, or Assembler language).

The table is also used, online, to keep a note of whether an application program is in main storage or not.

Several PPTs can be assembled and identified by suffix characters.

System Recovery Table (SRT)

A system recovery table (SRT) is assembled to list the ABEND or abnormal condition codes that will be intercepted by CICS/VS. Each entry in the table includes:

- The ABEND code to be intercepted.
- The name of the program or routine to which control will be passed when the ABEND code is encountered.

Several SRTs can be assembled and identified by suffix characters.

File Control Table (FCT)

The table contains the characteristics of the files accessed by File Management, including where appropriate those of files that are on connected CICS/VS systems.

Destination Control Table (DCT)

The DCT is used by Transient Data Management. It describes each of the Transient Data destinations used in the system or in other CICS/VS systems to which the system is connected.

Temporary Storage Table (TST)

The TST is used to describe temporary storage data for which recovery is to be provided or which reside on a connected CICS/VS system. The TST lists the identification of data sets for which CICS/VS is to provide recovery in the event of abnormal termination of CICS/VS (and subsequent emergency restart) or abnormal termination of a transaction (and subsequent transaction backout).

The TST also lists the identification of data which resides in connected CICS/VS systems.

Performance and Design

The efficiency of a DB/DC system can be judged by two main factors:

1. Utilization of system resources — mainly processor time and storage. (Resource utilization is of prime interest to the system programmer.)
2. Response times — and how they vary with the number of online users being served concurrently. (Response times are of prime interest to the operators.)

A cost-effective system is one in which these two factors are balanced to give the best total performance.

Note: The two factors and indeed the distribution and placing of resources must also be considered when CICS/VS systems are to be interconnected for intersystem communication.

This section deals with a few of the parameters that affect CICS/VS system efficiency. A full discussion of the approach to creating a cost-effective CICS/VS system is given in the CICS/VS System Programmer's Guide (DOS/VS) and the CICS/VS System Programmer's Guide (OS/VS).

Multiregion operation and the improved CICS/VS monitoring facilities available with CICS/VS Version 1 Release 5 can also be used to control and monitor performance. For information about these facilities, see Appendix E.

GENERATION OF OPERATING SYSTEM

For performance reasons, supervisor programs in the operating system should be generated to use as little real storage as possible. This makes more space available for CICS/VS tasks and helps to reduce the path length (processor time) per transaction for the execution of supervisor code.

The system programmer should carefully assess the value of optional functions before including them in the operating system. For example, in DOS/VS, the inclusion of the job accounting option may increase processor usage for a small transaction by up to 10%.

Also, for optimum performance, sufficient virtual storage should be specified (for the partition or region being used by CICS/VS) to allow all application programs, CICS/VS online programs, and CICS/VS tables to be resident in virtual storage. (CICS/VS storage requirements are given in the CICS/VS System Programmer's Reference Manual.) On the other hand, to avoid unnecessary use of system resources the amount of virtual storage and real storage should not be excessive.

GENERATION OF CICS/VS

Ideally, the CICS/VS system should be generated so that for normal error-free processing:

1. Virtual storage paging is minimized.
2. The total path length (amount of processor execution time) per transaction is minimized.

Virtual Storage Paging

Virtual storage paging can be avoided during normal error-free processing by ensuring that sufficient real storage is available to accommodate the "working set" — that is, the set of virtual storage pages containing the programs, tables, and so on, required to run the system normally.

The size of the working set can be minimized by using CICS/VS system generation options to control the sequence in which programs are loaded

into virtual storage, to control their positions in virtual storage, and to specify which programs should be aligned on page boundaries. To take full advantage of these options, it is obviously beneficial to split application programs into modules of heavily used and lightly used code.

Furthermore, some CICS/VS management modules can be shared by more than one CICS/VS system within the same processor system (whether or not the separate CICS/VS systems communicate with one another). Such sharing reduces real storage requirements and virtual storage paging. Further details are given in the CICS/VS System Programmer's Guides.

Reduction of Path Length Per Task

The path length (amount of processor time) per task can be shortened in several ways — mainly by not using CICS/VS options that are not necessary to the application functions and which might carry significant overheads of processing time.

To help the system programmer in making such judgements, a discussion of those CICS/VS options that incur the more significant overheads (including those that also incur operating system overheads) is given in the CICS/VS System Programmer's Guide (DOS/VS) and the CICS/VS System Programmer's Guide (OS/VS).

High Performance Options (HPO) (CICS/OS/VS only)

A set of options intended to give significant performance improvement, especially to applications involving high volumes of fairly simple transactions, is available to CICS/VS running under MVS. The options, which if required are specified at system generation, are specifically intended to enhance the following facilities of CICS/OS/VS:

- Storage management
- Requesting of VTAM services
- VSAM support for accessing files
- Statistics collection
- Logging and journaling

Further details of the options are given in the CICS/VS System/Application Design Guide.

DESIGN OF DATA COMMUNICATIONS SYSTEM

The data communications path between the user's application program and the operator should be designed to provide sufficient resources in the various links along the path so that congestion does not occur. It is expensive in storage and processing power if, for example, CICS/VS is unable to accept incoming data from VTAM; in these circumstances, VTAM has to use processor time and more storage in order to hold that data temporarily in pageable buffers. This subject is discussed in the CICS/VS System/Application Design Guide.

In generating the telecommunication access method or indeed any aspect of a system, the system programmer should specify options that invoke only those resources that can be cost-justified.

When using intersystem communication (ISC) between CICS/VS systems, the usage of resources in the processors and on the communication-link can be significant. However, for certain ISC applications (such as online inquiry), this resource usage can be reduced by using the intersystem communication message performance option. This option is intended for those applications in which a transaction is initiated in a remote system and in which the system is not expected to perform automatic recovery in the event of failure. The ISC message performance option is discussed further in Appendix D and in the CICS/VS System/Application Design Guide.

With CICS/VS Version 1 Release 5, resource usage and performance in data communication systems can also be improved through the use of distributed transaction processing and SNA parallel session facilities. These facilities are discussed in more detail in Appendix E.

PERFORMANCE STATISTICS

CICS/VS can provide a wide variety of statistics covering such topics as transactions, tasks, terminals, application programs, and data sets. A brief outline of the types of statistical information available is given in the next chapter under the heading "Operating Statistics."

Such statistics are valuable to the system programmer because they enable him to alter system control parameters to suit the way in which the system is used in practice (as distinct from what was predicted). For example, the statistics show which programs are most frequently used; this information enables the system programmer to alter, if necessary, the sequence in which CICS/VS and application programs are loaded into virtual storage, and thereby reduce the amount of virtual storage paging.

Improved monitoring facilities are available with CICS/VS Version 1 Release 5. For information about these facilities, see Appendix E.

Chapter 5. CICS/VS for the Operations Staff

CICS/VS is a very adaptable system. Operations staff are given facilities for selecting and changing system characteristics — both at system initialization time and while the system is running.

This chapter describes these facilities in relation to the main duties of the operations staff, which are:

1. To initialize and start the CICS/VS system.
2. To monitor the CICS/VS system while it is running and, if necessary, to tune the system by changing certain parameters (dynamically).
3. To shut down the CICS/VS system in a controlled manner.

The initialization and starting of the CICS/VS system are processor console operator functions. The other duties (monitoring, changing, and shutting down the CICS/VS system) are CICS/VS master terminal functions.

Console operator functions and master terminal functions can be performed either by the same person, or by different individuals, depending on the size of the installation. In either case, the system programmer or designer should provide procedural guidelines for the operator(s).

System Initialization

In the simplest case, CICS/VS is initialized and started like any other job: after the required volumes and devices have been made ready, and the operating system has been initialized, a set of job control statements (called the "CICS/VS Startup Jobstream") is read into the job scheduler. A message is subsequently displayed to tell the console operator when CICS/VS is running. In this way, the CICS/VS system is initialized (and started) with system control parameters that are already specified in a CICS/VS system initialization table (SIT).

The way in which CICS/VS is initialized and started can be altered, however, by specifying the suffix characters of a particular SIT and/or by overriding system control parameters contained in the chosen SIT. This information can be provided either in the startup deck or via the console keyboard.

OVERRIDING SYSTEM CONTROL PARAMETERS

The system control parameters that can be overridden consist mostly of (1) the one- or two-character suffixes that specify particular versions of system control programs and tables; and (2) some numeric values that specify, for example, the maximum number of concurrent tasks allowed, or the size of certain storage areas.

Details of which SIT to specify and which system control parameters are to be overridden for any particular initialization of CICS/VS will usually be decided by the system programmer or designer.

The START Parameter

One of the key parameters that might well be specified at the console operator's discretion is the START parameter. Depending on the type of initialization required, and the way in which the previous shutdown occurred, START can be specified as COLD, WARM, or EMERGENCY.

- A COLD start disregards any previous system activity and initializes the system facilities exactly as specified in the chosen SIT (and in any overridden parameters).
- A WARM start reinitializes the system facilities (in particular the data sets) to the status that existed at the previous controlled shutdown.
- An EMERGENCY start reinitializes the system facilities to the conditions that existed at predefined points before an uncontrolled shutdown. For example, the initialization process backs out those data set changes that were in progress at the moment of failure.

System Monitoring and Tuning

To adjust the system to give the best performance in different conditions, many system control parameters can be changed dynamically (that is, while CICS/VS is running) through the use of master terminal functions.

MASTER TERMINAL FUNCTIONS

Master terminal functions provide the means of controlling (and shutting down) the entire CICS/VS system. These functions can be performed at practically any terminal, but their use is normally limited (by sign-on password and a security key) to a designated person known as the master terminal operator.

In CICS/VS Version 1 Release 5, an enhanced master terminal transaction, CEMT, provides a new interactive interface for controlling the CICS/VS system. The transaction is driven by commands similar in syntax to those of the command-level interface, allowing abbreviations and keyword defaults. CICS/VS interprets the command and diagnoses errors.

With CEMT, the master terminal operator can issue a single command to inquire about, or change the status of, groups of resources, such as all programs with names beginning with "ABC" or all terminals out of service.

The CEMT transaction is designed for use with display terminals that use the 3270 data stream and display 1,920 or more characters; however, the transaction can also be entered from the system console (and certain other terminal types), with limited support.

The pre-1.5 master terminal transaction, CSMT, will continue to be available with CICS/VS Version 1 Release 5.

Aspects of the system that can be controlled by master terminal functions include:

Transactions
: Transaction priority values can be changed.

  Transactions can be enabled or disabled.

Tasks
: Allowable number of tasks can be changed.

  All active tasks can be listed.

  An individual task can be terminated.

  Trigger levels (for automatic task initiation) can be changed.

Terminals
: Status of terminals can be inspected and changed.

  Terminal priority values can be inspected and changed.

  VTAM-supported terminals can be acquired and released.

Remote Control Units and Lines
: Status of control units and lines can be inspected.

  Control units and lines can be put in service or out of service.

Application Programs
: Status of application programs can be inspected

  Application programs can be enabled or disabled

Data Sets
: Status of data base data sets can be inspected and changed

  Data base data sets can be enabled or disabled, and opened or closed.

  A shared data base session can be started or terminated.

  The status of a shared data base can be displayed.

  Program isolation trace can be requested.

VTAM ACB
: The VTAM ACB can be opened or closed.

Intersystem Communication
: ISC sessions can be established and terminated.

Multiregion Operation (MRO)
: The use of MRO for a given CICS/VS system can be opened or closed. Connections to CICS/VS systems in other regions can be established or severed.

Trace and Dump
: Trace options can be set, and formatted dumps can be taken of the CICS/VS partition.

The master terminal functions are described in detail in the CICS/VS Operator's Guide.

SUPERVISORY TERMINAL FUNCTIONS

Supervisory terminal functions enable a supervisor to inspect or to change the status of terminals within a defined group. These functions can be performed at practically any terminal, but their use is normally limited (by sign-on password and security key) to designated people known as supervisory terminal operators.

By means of supervisory terminal functions, a supervisory terminal operator can:

- Place a terminal in service or out of service.
- Change a terminal's status so that, for example, it can or cannot initiate transactions, receive messages, or send messages.

For CICS/VS Version 1 Release 5, the supervisory terminal transaction has been enhanced similarly to the master terminal transaction described above.

OPERATING STATISTICS

A wide variety of operating statistics can be obtained by the master terminal operator (or any other authorized operator). The statistics can be produced either automatically at regular intervals, or on request at any time.

The statistics produced can apply to the whole of a CICS/VS system (in the case of interconnected systems, it will be the system from which the request was issued), or can be requested to cover only selected topics, such as transactions, tasks, terminals, application programs, and data sets. The following list gives a brief outline of the types of information produced.

- Transactions. For each transaction: frequency of usage; number of times that paging has been necessary; number of times transaction restarted after abend.
- Tasks. Peak number of concurrent tasks; number of times the maximum number of tasks has been reached.
- Terminals. For each terminal: number of input and output messages; number of transactions and transaction errors.
- Application programs. For each program: number of times used.
- CICS/VS files and DL/I data bases. For each CICS/VS file: number of requests for reading, adding, and updating. For each DL/I data base: number of requests for retrieval, insert, and delete operations.

Note: Statistics concerning accessing of DL/I data bases by IMS/VS are collected by CICS/OS/VS in the FCT. Such statistics are not collected by CICS/DOS/VS. However, DL/I DOS/VS collects these statistics and provides utilities that can be used to produce reports from the statistics.

Operating statistics are valuable, not only to the system programmer, but also to a master terminal operator in order to monitor the performance of CICS/VS. For example, if the statistics show that the peak number of concurrent tasks has approached or has equalled the

maximum number allowed, some improvement in performance might be possible by increasing the number of tasks allowed (a master terminal function).

A description of how an operator obtains the statistics is given in CICS/VS Operator's Guide.

Improved monitoring facilities are available with CICS/VS Version 1 Release 5. For information about these facilities, see Appendix E.

System Shutdown

The CICS/VS system can be shut down (through a master terminal function) in the following ways:

- Shut down CICS/VS immediately without regard for any active tasks that may exist.
- Shut down CICS/VS, but allow active tasks to finish. (This process automatically invokes an orderly dynamic close of the VTAM ACB.) Optionally, the operator can request that certain specified programs be executed during the termination process.

With a shutdown request, the master terminal operator can also request a CICS/VS storage dump.

Chapter 6. Introduction to How CICS/VS Works

The purpose of this chapter is to show how control flows between elements of a CICS/VS system during a normal sequence of execution. (For a more detailed description of how CICS/VS works, see CICS/VS Introduction to Program Logic.)

Elements of a CICS/VS System

The CICS/VS program product consists of six components*. Most online control programs are contained in the system management component and are referred to here as management functions. Associated with some of the management functions are CICS/VS control tables, which are prepared by the user to describe a particular DB/DC environment. A CICS/VS system also includes a number of control areas.

Some of the principal management functions were outlined in chapter 3, and the essential control tables were outlined in chapter 4. The main control areas, which hold dynamic data during the execution of CICS/VS control programs and application programs are briefly described here and include the following:

Common System Area (CSA)
: The CSA is a major CICS/VS control block. It includes, for example, pointers which can be used to locate all CICS/VS management modules and CICS/VS control tables.

Task Control Area (TCA)
: A TCA is created, by task management, for each task (that is, for each individual execution of a transaction). The TCA includes pointers to all storage areas and control fields that are related to the particular task. When the task ends, task management releases the TCA.

Transaction Work Area (TWA)
: A TWA (if called for in the program control table) is created by task management for each task as an extension to the TCA. The TWA can be used by the application programmer for the accumulation of data and intermediate results during an execution of the transaction. When the task ends, task management releases the TWA together with the TCA.

Terminal Input/Output Area (TIOA)
: A TIOA is created automatically by terminal management on receipt of an unsolicited input message.

---

* The six components are: (1) the system management component; (2) the system reliability component; (3) the system service component; (4) the system monitoring component; (5) the system support component; and (6) the application service component.

EXEC Interface Block (EIB)

In a command-level environment, each task has associated with it a control block called the EXEC interface block (EIB). Each application program has automatic access by name to the fields within the task EIB. The EIB contains information that is useful during the execution of an application program, such as the transaction identifier, the time and date, and the cursor position on a display device. The EIB also contains data flow control indicators that can be useful during execution of a program. Furthermore, the EIB contains information that is helpful when a dump is being used to debug a program. For further details see the CICS/VS Application Programmer's Reference Manual (Command Level), or the CICS/VS Application Programmer's Reference Manual (RPG II).

Example Sequence of Execution

The following example sequence of execution gives a much simplified description of how the elements of a CICS/VS system interact with each other and with the application program during normal error-free execution. The example is based on a simple inquiry application in which an operator enters an input message consisting of (1) a transaction identification code "INQY" and (2) an account number. The application is designed to return an output message to the operator giving, for example, the current balance in the account.

1. Terminal management periodically checks all terminals for input messages. Then, on receipt of the input message containing the transaction identification code ("INQY"), terminal management:

   a. Creates a terminal input/output area (by calling upon storage management).

   b. Places the input message in the terminal input/output area.

   c. Passes control to task management.

2. Task management creates a task for the transaction. To do this, task management:

   a. Validates the transaction identification code ("INQY") in the terminal input/output area against the program control table. (If the transaction identification code is misspelled or does not exist in the program control table, task management causes an error message to be sent to the terminal.)

   b. Requests storage from storage management, and creates a task control area.

   c. Assigns a priority value to the task (from the transaction priority in the program control table, from the terminal priority in the terminal control table, and from the operator priority in the sign-on table).

   d. Adds the task to the chain of tasks waiting to be executed.

   At this stage, task management has established the task, but the task is not yet executing.

3. At some subsequent time, task management dispatches the task. To do this, task management:

   a. Inspects the chain of tasks waiting to be executed and dispatches (causes execution of) tasks in order of their task priority values.

   b. When the INQY task is first dispatched, control is passed to program management.

4. Program management locates the required application program and starts its execution. To do this, program management:

   a. Inspects the program control table to find the name of the application program that services the transaction called "INQY."

   b. Inspects the processing program table to obtain the location of the application program. (If the program is not already in main storage, program management loads it.)

   c. Passes control to the inquiry application program.

5. The inquiry application program:

   a. Obtains the account number from the terminal input/output area.

   b. Issues a file control request using the account number as a record key — thereby passing control to file management.

6. File management initiates the retrieval of the required data. To do this, file management:

   a. Requests a file input area from storage management.

   b. Requests the operating system's data access method to read the required data from the appropriate file into the file input area.

   c. Specifies that this task will wait until the data has been completely read into the file input area. This causes control to be passed to task management.

7. Task management:

   a. Causes the inquiry task to wait while the data is being obtained.

   b. Dispatches other tasks.

   c. Redispatches the inquiry task when the requested data is in the file input area; this causes control to be returned to file management.

8. File management:

   a. Ensures that data has been read into the file input/output area correctly.

   b. Returns control to the inquiry application program.

9. The inquiry application program:

   a. Edits or processes the data to be sent to the operator as an output message in the terminal input/output area.

   b. Issues an output request to terminal management which subsequently causes the output message in the terminal input/output area to be transmitted. (See step 12(a) later in this sequence.)

   c. Finishes execution and returns control to program management.

10. Program management:

   a. Notes that execution of the program for this task is complete.

   b. Returns control to task management, requesting the termination of the inquiry task.

11. Task management releases storage areas associated with this task — including the task control area and the file input area, but excluding the terminal input/output area.

12. Terminal management:

   a. Sends the output message from the terminal input/output area to the terminal. It does this asynchronously (with respect to execution of the task) when the line to the terminal is free.

   b. Releases the terminal input/output area.

Appendix A. System Requirements

CICS/VS operates on any IBM System/370 or IBM 4300 with sufficient real storage to satisfy the combined requirements of CICS/VS, the host operating system, appropriate access methods, batch requirements and other customer required applications. Storage requirements are discussed briefly under "Storage Considerations" later in this appendix.

The configuration must include sufficient I/O devices to support the requirements for system output, system residence and system data sets. Sufficient direct access storage must be available to satisfy the user information storage requirements and may consist of any direct access facility supported by the system configuration and the programming systems. The host processor must be equipped with the Compare and Swap instruction. See also the documentation for the host operating system for details of any other features and engineering change levels that are required.

Distribution and support of CICS/VS requires the availability of one 9-track tape drive.

Programming Systems

CICS/VS operates as one or more tasks under VSE/Advanced Functions Release 2 or OS/VS (in a dedicated or multiprogramming environment). The selection of the environment is the user's responsibility, as is the selection of system options beyond those required for the operation of CICS/VS.

The versions, releases, and features of the programs listed below required for use of CICS/VS depend on the CICS/VS facilities that are to be used. Please ask your IBM representative for details.

CICS/DOS/VS UNDER VSE/ADVANCED FUNCTIONS

CICS/DOS/VS Version 1 Release 5 is designed to operate under VSE/Advanced Functions Release 2.

One or more of the following licensed programs may also be needed, depending on the user's requirements:

VSE/VSAM
: — required if the user has VSAM transient data, auxiliary temporary storage, or VSAM files.

ACF/VTAM
: — required if VTAM support is needed.

ACF/VTAME
: — required if VTAM support for the IBM 4331 Communications Adapter is needed.

BTAM-ES
: — required if BTAM support is needed.

The Extended Telecommunications Modules (EXTM) program product is not supported by VSE/AF, and therefore cannot be used with CICS/DOS/VS Version 1 Release 5.

For more detailed information relating to CICS/VS Version 1 Release 5, see Appendix E.

CICS/DOS/VS Version 1.4.1 will also run under VSE/AF Release 2.

CICS/DOS/VS Version 1.4 will run under DOS/VS Release 34 or VSE/AF Release 1, but not VSE/AF Release 2.

Data Language/I (DL/I)

Before using the DL/I facilities supported by CICS/DOS/VS, refer to the statement regarding availability of DL/I product support in the program directory shipped with CICS/DOS/VS.

CICS/OS/VS

CICS/OS/VS operates under the IBM System/370 Operating System/Virtual Storage (OS/VS1 and OS/VS2). (CICS/VS Version 1 Release 5 does not run under SVS.) Users of VTAM must have installed the OS/VS1 or OS/VS2 release that supports VTAM and the terminals to be used.

The following components of either OS/VS1 or OS/VS2 are required:

- Supervisor
- Sequential Access Method
- Basic Direct Access Method
- BTAM, TCAM, ACF/TCAM, ACF/VTAM, or a combination of these telecommunication access methods
- Linkage Editor
- Loader
- System Assembler
- System Utilities
- System Modification Program (SMP)

In addition to the above OS/VS1 or OS/VS2 components, the user may require any of the following:

- Virtual Storage Access Method (OS/VS1 or OS/VS2)
- Indexed Sequential Access Method (OS/VS1 or OS/VS2)
- Full ANS COBOL V2, 360N-CB-482
- Full ANS COBOL V3 Compiler and Library, 5734-CB1

- Full ANS COBOL V4 Compiler and Library, 5734-CB2, and Library only, 5734-LM2
- OS/VS COBOL and Libraries 5740-CB1, and Library only 5740-LM1
- PL/I Optimizing Compiler and Libraries, 5734-PL3
- PL/I Optimizing Compiler, 5734-PL1
- PL/I Resident Library, 5734-LM4
- PL/I Transient Library, 5734-LM5
- IMS/VS Data Base System, 5740-XX2
- 3735 Form Description Macro and Utility, 5744-AZ1

For more detailed information relating to CICS/VS Version 1 Release 5, see Appendix E.

Data Language/I (DL/I)

To use the optional CICS/OS/VS interface to the Data Language/I, (DL/I), facility of IMS/VS, the user must have installed the IMS/VS Data Base System (5740-XX2).

CICS/VS and Virtual Machines

CICS/OS/VS and CICS/DOS/VS can be run in a VM/370 virtual machine under the appropriate releases of VSE/AF or OS/VS.

The hardware and software requirements of CICS/VS in a virtual machine are the same as for a real machine, and should be considered as additional to the requirements of VM/370 itself and of any other virtual machines within the VM/370 environment.

Processor utilization and possibly terminal response times will be greater when CICS/VS is running under VM/370 than when it is running in a real machine.

If the impact on performance is significant, its effect can be partially or totally offset by the installation of additional main storage, or by upgrading the processor model, or both.

The operation of CICS/VS running in a virtual machine may differ in certain respects from that obtainable when running in a real machine. Briefly, the significant differences are:

1. VM/370 does not pass Channel Check and Interface Check conditions to virtual machines for action by the subordinate operating system. If these conditions arise, the virtual machine is terminated. This restriction is removed by VM/370 Release 5.0.
2. BTAM support of the Autopoll feature will not work correctly under SVS and MVS if Wraplist polling is specified.
3. VM/370 does not support the IBM 7770 Audio Response Unit.

Storage Considerations

CICS/VS is distributed in the form of assembler-language source statements (80-column card images), and requires that the system used for system generation has sufficient online storage devices and capacity to assemble a 30,000-statement program. However the Starter Systems are distributed as pregenerated CICS/VS modules and only require loading onto the online direct access storage device. The CICS/DOS/VS Entry Level System is provided only in the form of pregenerated modules with a minimum number of options to simplify installation. Thus a system generation is not required although a limited form of table generation is still needed.

The distributed and generated libraries require up to 75 million bytes of direct access storage, depending upon the system options generated and the blocking factors chosen by the user (where applicable).

The system configuration required for the use of CICS/VS in a data base/data communication system is largely determined by the scope of the environment to be supported and the nature of the user's application programs. CICS/VS is designed in a modular fashion with many system generation options so that the user can select the options that are meaningful to his operation with maximum economy of main storage.

A minimal CICS/DOS/VS system (using the Entry Level System), which could serve as a starting point for initial development, testing and implementation of a pilot application where response time is not a consideration, would require about 40K of real storage and 100K of virtual storage to support three or four 3277 locally-attached terminals. These sizes include BTAM, ISAM and the user application area. Where good response is required, 60K bytes of real storage should be provided. A production system giving reasonable response time using the CICS/DOS/VS Entry Level System to support about ten remote 3277s, using BTAM, BMS, ISAM and COBOL, would require about 90K of real storage and 300K of virtual storage. A full CICS/DOS/VS system using facilities not in the Entry Level System, for example, journal control, would require about 130K of real storage and 400K of virtual storage.

For more details on the estimation of storage requirements, see CICS/VS System Programmer's Reference Manual.

Appendix B. IBM Terminals and Subsystems Supported by CICS/VS

The IBM terminals and subsystems supported by CICS/VS Version 1 Release 5 are shown at two levels. Table 1 below gives an overview, showing the terminal/system type and an indication of the access method supported. Table 2 gives further details, including specific units and physical attachment where appropriate.

Two access method interfaces are shown in Table 1. Support of SNA networks was first introduced through VTAM. It is available through ACF/VTAM and, for devices also supported by these access methods, ACF/VTAME (CICS/DOS/VS) and the Record Interface introduced in ACF/TCAM Version 2 Release 2 (CICS/OS/VS). Table 1 shows this support under the column headed VTAM. The second interface is shown in the BTAM column. This support is available through BTAM (CICS/OS/VS) or BTAM-ES (CICS/DOS/VS).

The user should be aware that many terminals and control unit special features are transparent to programming and are therefore readily usable even though not specifically identified. In all cases, the appropriate line adapters or telecommunications control units must be used, as specified by the access method in use.

Another interface is available in CICS/OS/VS, supporting the use of TCAM with a Message Control Program. This, the TCAM GET/PUT interface, supports data streams rather than specific terminal types. CICS/OS/VS accepts any data stream from a TCAM or ACF/TCAM supported terminal that can be edited in the message handler portion of the TCAM MCP to appear as an EBCDIC, basic SCS, or IBM 3270 data stream. IBM 7770 support and IBM 2260 compatibility are not available through this interface. CICS/OS/VS BMS data stream support for IBM 3600, 3767, 3770 and 3790 logical units is also available through the TCAM direct interface to NCP/VS introduced in TCAM 10.

The CICS/DOS/VS Entry Level System supports only the IBM 3270 Information Display System, including enhancements such as larger screen sizes, through the VTAM or BTAM interface.

Table 1

Terminal/System Type	Access Method Type* VTAM	BTAM
1050 Data Communication System		X
2260/5 Display Station		X
2740 Communication Terminal		X
2741 Communication Terminal		X
2770 Data Communication System		X
2780 Data Transmission Terminal		X
2980 General Banking Terminal System		X
3101 Display Terminal	X	X
3270 Information Display System	X	X
3600 Finance Communication System	X	X
3630 Plant Communication System	X	X
3650 Programmable Store System	X	
3650 Retail Store System	X	
3660 Supermarket System		X
3680 Programmable Store System	X	

Table 1 (continued)

Terminal/System Type	Access Method Type* VTAM	BTAM
3730 Distributed Office Communication System	X	
3735 Programmable Buffered Terminal		X
3740 Data Entry System		X
3767 Communication Terminal	X	X
3770 Data Communication System	X	X
3780 Data Communications Terminal		X
3790 Communication System	X	
4300 Processors	X	X
5100 Portable Computer		X
5110 Portable Computer		X
5230 Data Collection System		X
5260 Retail System		X
5520 Administrative System	X	X
5937 Rugged Terminal	X	X
6670 Information Distributor	X	X
7770 Audio Response Unit		X
8100 Information System	X	X
Communicating Magnetic Card Selectric Typewriter		X
Office System/6		X
Series/1		X
System/3		X
System/7		X
System/32	X	X
System/34	X	X
System/38	X	X
System/370 (including 303x processors)	X	X
Teletypewriter Exchange Service (TWX 33/35)	X	X
World Trade Typewriter Terminal (WTTY)	X	X

* For explanation of these entries, see text above

Table 2 gives detail of device support, including some units supported. In general, devices and features supported by systems and programmable controllers are transparent to CICS/VS, but where CICS/VS provides specific device support the units are listed in Table 2. The column headed 'attachment' provides further information on physical attachment. When using the VTAM interface, the attachment is a function of the access method and the Network Control Program (where appropriate). Therefore the mode of connection of a terminal - local, SDLC, BSC, switched or non-switched - is transparent to CICS/VS, and CICS/VS support is dependent on the capabilities of the terminal, the Network Control Program and the access method. CICS/VS does not participate in the connection process and so switched/non-switched is not shown in Table 2 for VTAM level support. When using the BTAM interface, the attachment is shown in detail. CICS/VS supports host initiated switch connections where appropriate autocall hardware is installed, but does not support manually established connections for call-out. CICS/OS/VS support of locally attached 2260s uses GAM, and its support of 7770s is 'BTAM-like'.

Some IBM devices are listed as being 'supported as' a named IBM device, meaning that the level of CICS/VS support is that offered to the named device. Other IBM devices are listed as being 'attached as' a named IBM device, meaning that they are defined to CICS/VS as being the named device but they may operate with restrictions where they differ from the named device.

Detailed qualifications to the information in Table 2 are given in the notes that follow the table. The following abbreviations are used in Table 2.

local	–	local channel or adapter attached
SDLC	–	synchronous data link control
BSC	–	binary synchronous communication
s/s	–	start/stop transmission
sw	–	switched
non-sw	–	non-switched

Table 2

Terminal/ System Type	Units	Attachment	Notes
1050	1051,1052,1053,1056	s/s sw or non-sw	
2260/5	2260,2848,1053	local,s/s non-sw	1
	2265,2845,1053	s/s non-sw	
2740		s/s sw or non-sw	
2741		s/s sw or non-sw	2
2770	2772,0545,1053,2213,2265,2502	BSC sw or non-sw	3
2780		BSC sw or non-sw	4
2980	2972,2980	BSC non-sw	5
3101		Supported as TWX 33/35	15
3270	3271,3272,3274,3276	local,SDLC,BSC non-sw	6
	3275	BSC sw or non-sw	
	3277,3278,3279		
	3284,3286,3287,3288,3289		
3600	3601,3602,3690	SDLC,BSC non-sw	7,8
	3604		
	3610,3612,3618		
	3614,3624		
3630	3631,3632	attached as 3600	7,18
	3643,3604		
3650	3651	SDLC	7
	3653,3275,3284		
3660	3651,3661	BSC sw	7
3680	3684	Supported as 3790/3650	7
3730	3791	Supported as 3790	7
3735		BSC sw	
3740	3741	BSC sw or non-sw	16
3767		SDLC s/s supported as 2740/2741	
3770	3771,3773,3774,3775,3776,3777	SDLC BSC supported as 2770	7,9
3780		BSC sw or non-sw	17
3790	3791	SDLC or local	7,10
4300	4331,4341	BSC or SDLC	7,11
5100		s/s supported as 2741	7
5110		BSC attached as 2770	7
5230	5231	BSC supported as 3741	
5260	5265	BSC attached as 3741	
5520		SDLC supported as 3790 full function LU BSC supported as 3740	7
5937		SDLC/BSC attached as 3270	6
6670		SDLC BSC supported as 2770	
7770		local	12

Table 2 (continued)

Terminal/ System Type	Units	Attachment	Notes
8100	8130 or 8140 processors with		
	DPCX	Supported as 3790	7
	DPPX/Base using Host Presentation Services or Host Transaction Facility	Attached as 3790	7
	DPPX/DSC (inc 8775 attach)	Supported as 3270	7
CMCT		Supported as 2741	
Office System/6	6640,6670	Attached as 2770	
Series/1		Attached as System/3	7
System/3	5406,5408,5410,5415	BSC sw or non-sw	7
System/7	5010	s/s or BSC,	7
		sw or non-sw	7,13
System/32	5320	SDLC supported as 3770	7,14
		BSC supported as 2770	7
System/34	5320	SDLC supported as 3770	7,14
		BSC attached as System/3	7
System/38		SDLC attached as 3770	7,14
		BSC attached as System/3	7
System/370 including 3031,3032,3033		BSC or SDLC	7,11
TWX 33/35		s/s	15
WTTY		s/s	15

Notes

1. CICS/OS/VS provides support of the 2848 through GAM; CICS/DOS/VS provides support through BTAM.

2. For CICS/DOS/VS, using BTAM-ES, the Transmission Control Unit or ICA must be equipped as follows:

   2703-2741 Break feature (8055) required
   3704/3705 - UNITXC=NO required in Emulator Generation or equivalent
   ICA on Model 135 - Unit Exception Suppression Feature (9729-9736) required
   ICA on Model 115 or 125 - Line Mode must be specified with Unit Exception Suppression.
   The Receive Interrupt (4708) is supported under CICS/OS/VS only.

3. CICS/VS 2770 support includes optional 2772 features #3650 (EBCDIC Transparency), #9936 (WACK response), #1490 (Buffer Expansion), #1910 (Conversational Mode), #1340 (Automatic Answering), #4610 (Identification), #6310 (Security Identification), and #5010 (Multipoint).

4. 6-bit transcode is not supported. Support includes optional features #8030 (EBCDIC transparency) and #1340 (Automatic Answering) or #5020 (Multipoint).

5. 2980 support is for 2972 Model 8 (RPQ 858160) or Model 11 (RPQ 8582311) with 2980 Model 1 (RPQ 835504), Model 2 (RPQ 835505) or Model 4 (RPQ 858147) including options RPQ 858188 (Auditor key for Model 2) and RPQ 858165 (Buffer Expansion).

6. SDLC 3270s are supported only at the VTAM level, and the switched BSC 3275 (feature #3440) is supported only at the BTAM level. Printers attached to local or SDLC 3274s and SDLC 3276s are supported through the VTAM interface either as LU Type 3 using the 3270 printer data stream or as LU Type 1 using the SCS data stream (which is a subset of that used for SDLC 3767, 3770, and 3790

printers). The 3288 is supported as a 3286 Model 2. CICS/VS supports the 3270 copy feature (#1550).

7. Devices and features supported by a system or programmable controller are generally transparent to CICS/VS. In some cases CICS/VS provides specific device support, in which case the units are listed.

8. SDLC is supported at the VTAM level. BSC attachment (RPQ 8K0598) is supported at the BTAM level. The 3614 is supported both for loop attachment to the 3601 and SDLC attachment to the host via a 3704/3705 Communications Controller.

   The 3614 is supported by CICS/VS via BSC only when loop attached to the 3601/3602 Controllers. 3624 is supported as a 3614.

   The 3690 is supported as a 3602.

9. CICS/VS provides support for the Data Transfer Function of the SDLC Programmable Models of the 3770 Data Communication System. The user is responsible for allocating data sets and managing the program library.

10. CICS/VS does not support the 3790/Data Entry Configuration using 3760s. The #9165 or #9169 configuration is required to support the CICS/VS enhancement first made available in Version 1 Release 3.0.

    Printers on 3790 systems are supported with a user-provided function program, or 3270 data stream compatibility with a 3270 printer data stream (LU3), or an SCS data stream supporting a subset of that for SDLC 3767 (LU1). The 3288 Model 2 is supported as a 3286 Model 2 when operating in 3270 mode.

11. System/370 and 4300 attachment by BSC requires a suitable telecommunications program (for example, the VSE/3270 Bisync Pass Through Program) in the system connected to CICS/VS. Attachment by SDLC is supported by CICS/VS intersystem communication.

12. CICS/OS/VS provides support through GAM; CICS/DOS/VS provides support through BTAM.

13. Non-switched as a multipoint device: System/7 Remote IPL is supported.

    Switched and as a Point-to-Point Device: Remote IPL is not supported.

14. The System/32 with its SNA/SDLC work station system utility program, the System/34, and the System/38 are supported as compatible versions of an appropriately featured 3770 Communication System operating as a batch logical unit. The System/34 or System/38 user-written program is responsible for supporting the correct SNA sequences of the attached subsystem.

15. TWX and WTTY are supported at the VTAM level via the Network Terminal Option program product (5735-XX7), with attachments as defined by NTO.

    At the BTAM level, TWX (Line Control Type) is attached on eight-level code at 110 bps on common carrier switched 150 bps networks.

    WTTY is attached at 50 bps on common carrier switched network where the terminals supported are those interfacing through IBM WT Corporation Telegraph Terminal Control with Telegraph Line Adapter.

Transmission code used is International Telegraph Alphabet No. 2 (CCITT No. 2). Automatic host disconnect via WRITE break is not supported for OS. Autocall is not supported for DOS or OS.

16. 3740 support includes optional features #7850 (Terminal Identification), #1685 (Multipoint), #5450 (Operator ID Card Reader), and #1680 (Expanded Communications).

17. 3780 support includes features #3601 (EBCDIC Transparency), #9936 (WACK), #5010 (Multipoint), #7651 (Switched), and #1601 (Component Selection).

18. The 3643 is supported as a 3604.

Appendix C. New Facilities of CICS/VS Version 1 Release 4

This appendix is intended to present a brief outline of the facilities and enhancements available in CICS/VS Version 1 Release 4. (Facilities introduced by CICS/VS Version 1 Release 4.1 are outlined in Appendix D. Facilities introduced by CICS/VS Version 1 Release 5 are outlined in Appendix E.)

Facilities discussed in the ensuing text are as follows:

CICS/VS Intersystem Communication
Shared Data Base (CICS/OS/VS)
Transaction Restart
3270 Information Display System enhancements
Command-level programming (EXEC) interface enhancements
Entry-level system (ELS) (CICS/DOS/VS)
Execution (Command level) Diagnostic facility (EDF)
IBM 3774/5 Programmable Communications Terminal support

Also discussed briefly are the following additional enhancements available for CICS/VS 1.4. For further details of these enhancements, the reader is directed to the various CICS/VS publications listed in the preface of this manual.

Good Morning Message for VTAM Logical Units
Transient Data queue improvements
Dynamic OPEN of VTAM ACB
Initialization improvements
3790 Keyed message support
Problem Determination improvements
Improved Data Dump placement and packaging
Usability and tuning enhancements
Simplification of table building
Extension to syncpoint facility (ROLLBACK)
Intersystem communication message routing sample application
Tunable Short Wait
Dynamic Allocation Sample Program (OS/VS (MVS) only)
Support for Field Level Sensitivity (IMS/VS)

INTERSYSTEM COMMUNICATION (ISC)

The intersystem communication facility provides for the connecting of CICS/VS systems through ACF/VTAM, such that a transaction running in one CICS/VS system may access resources on a connected CICS/VS system, or may initiate a transaction to be run on that connected system.

Note: If the connected systems are not running in the same processor then the Multi-Systems Networking Facility (MSNF) of ACF/VTAM is also needed.

The resources that may be accessed on a connected CICS/VS system are files, transient data, temporary storage, and DL/I data bases.

The facility of intersystem communication enables installations to distribute system resources such as application programs, files, and

data bases in a network, without the need for such resources to be attached to the system requiring them.

SHARED DATA BASE (CICS/OS/VS)

The shared data base facility allows IMS/VS batch application programs to access and update an IMS/VS data base through CICS/VS. In this way, the data base can be shared concurrently by CICS/VS transactions and batch application programs, and its integrity preserved.

The facility obviates the current requirement that a user must close an IMS/VS data base in his CICS/VS system when running a batch program to access that same data base.

TRANSACTION RESTART

Transaction restart is an optional facility that allows an abended transaction satisfying certain conditions, to be restarted without terminal user intervention. The facility allows user exit code to perform any additional recovery required by the user, and to override the decision whether or not to restart.

The facility is particularly useful where program isolation scheduling is being used, because any deadlocks can be made transparent to terminal operators.

CICS/VS SUPPORT FOR 3270 INFORMATION DISPLAY SYSTEM ENHANCEMENTS

The enhancements supported by CICS/VS include:

Display Devices with Large Screens. Each device can be used with its full screen size or with a default screen size. The default screen sizes provide compatibility for existing 3270 programs.

CICS/VS allows screen sizes to be switched to either full size or default size for different transactions. The facility is available via both the Basic Mapping Support (BMS) and Terminal Control (TC) interfaces.

Additional Program Function Keys. The keys can be used as transaction identifiers and also as BMS paging commands and can be referred to by CICS/VS application programs.

Program Attention Key on the SCS Printer. When the 3270 printer is operated as an SCS printer logical unit (via VTAM), a program attention key may be used to communicate with the host. CICS/VS allows CICS/VS application programs to receive the attention. The key may also be used to initiate a transaction.

Character Sets. Provision of alternative character sets for various countries. Although CICS/VS does not support multiple character sets directly, it does allow hexadecimal character strings, representing character sets other than EBCDIC, to be used as transaction identifiers and also as initial data for BMS maps.

Print Authorization Matrix. A matrix is provided to determine which printer can be allocated for host or operator initiated local copy operations. The matrix is defined by the installation through application programming. The CICS/VS user must send a particular 3270 data stream (defined by the 3270 Display System) to set up the matrix. If the matrix is not set up, no printers will be available for local operations.

Magnetic Stripe Reader. CICS/VS now permits a magnetic stripe reader attention identification (AID) or an operator identification card reader AID to be used to initiate a transaction.

New Selector Pen Designator Character. Selection of the ampersand character (by light pen or cursor) will cause not only the address but also data for modified fields to be transferred to the host. The facility is supported by BMS and by Terminal Control (TC).

Note: With the exception of the program attention key on the SCS printer, all the facilities described in the foregoing text are supported on the CICS/DOS/VS Entry Level System (ELS).

Data Security - Large Screen Display Devices. The large screen devices are updated without the screen being cleared; this prevents the blink which occurs with a standard size 3270 device. Because the large screen devices are structured to update the screen in this way, non-display data may be temporarily displayed if and when the screen is updated without first erasing the contents of the entire screen.

In order to avoid this potential security exposure it is important therefore to either:

1. Reformat the entire screen with an ERASE/WRITE command, or
2. Ensure the screen format is unchanged, or
3. Ensure that the entire message to be entered is contained in a single transmission.

COMMAND-LEVEL PROGRAMMING (EXEC) INTERFACE

The scope of the EXEC interface is extended to permit Assembler language and RPG II to be used for CICS/VS command-level application programming.

Segmented records are supported at the command-level interface.

In addition, the existing IMS/VS - DB CALL interface has been improved to remove the need for any CICS/OS/VS application programmer to have to access CICS/VS control blocks directly or be aware of CICS/VS storage management mechanisms. Similar enhancements will also be available to CICS/DOS/VS users through support from DL/I DOS/VS. However, users of CICS/DOS/VS should consult the Release 1.4 program directory for a statement regarding availability of DL/I product support.

The DEEDIT built-in function has been added to the command-level interface to provide the same function as provided by the equivalent macro interface. This is particularly useful for RPG II programs that may require the function as an equivalent to the PL/I and COBOL picture function.

CICS/DOS/VS - ENTRY LEVEL SYSTEM (ELS)

The CICS/DOS/VS product also includes the CICS/DOS/VS Entry Level System (ELS), to give the user the option of installing a simpler version of the full CICS/DOS/VS system.

CICS/DOS/VS - ELS provides a first-time user of CICS/VS with a simplified system that compared with CICS/DOS/VS is easier to install and use, and is generally smaller in real and virtual storage requirements.

CICS/DOS/VS - ELS offers:

- Simplified control and selecting mechanisms.
- Terminal support limited to display devices (via BTAM or VTAM).
- A complete set of pregenerated programs - allowing simplified installation, without the need for system generation.

The system is recommended for the user who does not require the concurrent processing of many transactions, or for the first time user wishing to gain familiarity with CICS/VS, before expanding to the use of the facilities of the full CICS/DOS/VS system.

If at a later time the user wishes to progress to a full CICS/DOS/VS system, application programs written for ELS will be upwards compatible with CICS/DOS/VS.

EXECUTION (COMMAND LEVEL) DIAGNOSTIC FACILITY (EDF)

The facility enables a programmer to test an application program written to the command-level interface, in an online interactive manner at the source level, using a 3270 terminal.

IBM 3774/5 PROGRAMMABLE COMMUNICATIONS TERMINAL SUPPORT

CICS/VS SNA support for the 3774/5 Programmable Communications Terminals has been extended to assist a user to write 3770 programs that communicate with a host system. Such communication will be similar to that already permitted for 3790 programs for a full function logical unit.

GOOD MORNING MESSAGE FOR VTAM LOGICAL UNITS

For any VTAM logical unit (LU), a 'Good morning' message may be sent to the LU immediately after the LU has logged on to CICS/VS. The message sent to the LU may be specified by the user if required. For a 3270 display terminal a CICS/VS logo is generated.

TRANSIENT DATA QUEUE IMPROVEMENTS

Intrapartition transient data queues may optionally be supported using VSAM in place of the currently assumed DAM. For most users this could mean a reduction in the number of I/O operations and associated path lengths.

DYNAMIC OPEN OF VTAM ACB

This allows VTAM to be initialized after CICS/VS has been activated.

INITIALIZATION IMPROVEMENTS

Certain operands that previously could only be specified at system generation time can now be provided at system initialization time. This allows the user to alter preset parameters to suit his requirements without resorting to system generation.

All parameters that can be specified in the System Initialization Table (SIT) may also be entered as overrides at CICS/VS initialization. Overrides may be obtained from either the master terminal operator and/or the system input device.

3790 KEYED MESSAGE SUPPORT

3790 now has the capability to retrieve messages by key/identifier rather than sequentially as at present. The route list fields for 3790 have been extended to allow specification of 3790 keyed messages through the use of the LDC fields. This support is available in user route lists or on CMSG requests.

PROBLEM DETERMINATION IMPROVEMENTS

As a problem determination aid, certain types of storage, which are currently freed during transaction processing, can be optionally frozen and only freed at the end of the transaction. Thus any abend or other dump may contain all the storage relevant to the problem.

A timing trace entry has been added. This is automatically created approximately every 128 entries. This will enable the various traces (for example, VTAM and CICS/VS traces) and dumps to be tied together.

Additional trace entries are now decoded.

A subfield has been added to the trace table to identify different request types, for example, user trace, or FE.

Additional trace entries are generated by management modules which are using the LIFO storage scheme provided in this release, to indicate the start and completion of requests.

When an ABEND or program check occurs an entry is added to the abend trace table. This improves the debugging ability in the event of a CICS failure.

IMPROVED DUMP DATA PLACEMENT AND PACKAGING

The output of the Formatted Dump is now put on to the same data set as the output from the Transaction Dump.

The data set is now in variable blocked record format to improve disk utilization.

USABILITY AND TUNING ENHANCEMENTS

When a sign-off in the form CSSF LOGOFF is issued at a VTAM terminal, the status of the terminal now remains unchanged. It is therefore possible for the terminal to logon again without intervention from the master terminal operator.

Upper case translation is now available for all VTAM LUs instead of for only 3270 devices.

The entry-point name on the END statement of table generations is no longer mandatory.

It is now possible, as an option, for an application not to lose control when the requested resource is already in use (conditional ENQUEUE).

A facility is provided whereby the user can improve the performance of certain BSC devices which differentiate between the end of transmission block (ETB) and end of transmission text (ETX).

SIMPLIFICATION OF TABLE BUILDING

The generation of Program Control Tables (PCT) and Processing Program Tables (PPT) is simplified by the addition of the GROUP parameter of the TYPE operand. The appropriate macro then generates a complete set of transaction or program entries for a specific function with all their attributes suitably defaulted.

The INDEX option replaces the 'page index' for the Program Control Table (PCT), Processing Program Table (PPT), and Destination Control Table (DCT). It allows for more efficient access to large tables and removes the need for the user to sort tables into alphabetic order. Entries can now be placed entirely for efficiency and convenience.

EXTENSION TO SYNCPOINT FACILITY (ROLLBACK)

The syncpoint facility has been extended to include ROLLBACK, so that an application program may at any time cancel the changes it has made to all recoverable resources during the current logical unit of work. These are then restored to the state prevailing at the previous

syncpoint or task initiation. Transaction execution continues following the syncpoint request.

INTERSYSTEM COMMUNICATION MESSAGE ROUTING SAMPLE APPLICATION

A sample application is provided with CICS/VS to show the user how to redirect messages to a remote system based on the message content.

TUNABLE SHORT WAIT (CICS/DOS/VS ONLY)

It is now possible for the user to override the default wait time used by CICS/DOS/VS in order to regain control from DOS/VS and check for the completion of disk I/O events.

DYNAMIC FILE ALLOCATION SAMPLE PROGRAM (CICS/OS/VS (MVS) ONLY)

A sample program is provided to give CICS/OS/VS users the ability to dynamically allocate and deallocate any file which CICS/VS can open and close. The program incorporates a HELP feature to provide assistance at the syntax level.

SUPPORT FOR FIELD LEVEL SENSITIVITY (IMS/VS)

CICS/VS now supports the Field Level Sensitivity (FLS) facility provided by IMS/VS 1.1.5 (PTF UP90001). This facility allows DL/I application programmers independence from the sequence or position of the fields in a physical segment.

Appendix D. New Facilities of CICS/VS Version 1 Release 4.1

This appendix briefly outlines the new facilities made available with CICS/VS Version 1 Release 4.1:

- Intersystem Communication - Message Performance Option
- Dynamic Close of VTAM ACB
- Support for the SNA Type 4 Logical Unit Protocol (LUTYPE4)
- VSE/Advanced Functions support
- Support for Fixed Block Architecture (FBA) Direct Access Storage Devices (IBM 4300 series only)
- Support for Field Level Sensitivity (IMS/VS)

INTERSYSTEM COMMUNICATION - MESSAGE PERFORMANCE OPTION

The ISC - Message Performance Option is designed for transaction switching between CICS/VS systems and is invoked by specifying a new parameter on the START command. Performance is improved by eliminating certain CICS/VS-generated control sequences that are normally exchanged between two CICS/VS systems. The application designer is given the option of exchanging certain aspects of data integrity for improved performance.

Although recoverable resources are still protected within each system, the synchronization points are not coordinated between the systems. This option is therefore of value for those applications where transactions are switched between CICS/VS systems, but are updating recoverable resources in only one system. Careful consideration should be given to application integrity design before selecting this option.

The ISC - Message Performance Option will be of particular value in those cases where the terminal operators are interested in making an inquiry only (that is, no resources are updated) and in getting an answer as soon as possible. If no answer is received after a certain time, the inquiry is considered lost and can be retried. Thus the system need not provide any recovery capability in the event of failure, either of the session or of the system; the operator provides all the recovery necessary, merely by reissuing the original request.

DYNAMIC CLOSE OF VTAM ACB

Dynamic close of VTAM ACB is complementary to the dynamic open of VTAM ACB, which was introduced with Version 1 Release 4 of CICS/VS. Together, these two facilities allow CICS/VS and ACF/VTAM to be initialized, run, and terminated independently if required.

Dynamic Close can be invoked either by issuing a CICS/VS master terminal command or an ACF/VTAM halt command.

Dynamic Close results in the termination of all CICS/VS sessions with ACF/VTAM. This can be done in two ways:

- Through an orderly dynamic close.
- Through a quick dynamic close.

An orderly close will wait for transaction termination on all CICS/VS sessions and then issue a CLSDST against each active session before issuing an ACF/VTAM CLOSE ACB macro. A quick close will abnormally terminate all transactions currently in session with ACF/VTAM before issuing a CLOSE ACB macro.

SUPPORT FOR THE SNA TYPE 4 LOGICAL UNIT PROTOCOL

A new SNA protocol, Logical Unit Type 4 (LUTYPE4), has been added to CICS/VS. This support is an extension of CICS/VS batch logical unit support and provides for communication with word-processing media.

The Logical Unit Type 4 protocol has been implemented in the IBM 6670 Information Distributor announced by the IBM Office Products Division.

Except for minor operating restrictions, the CICS/VS LUTYPE4 support is such that a CICS/VS application program designed for an IBM 3770 batch logical unit is compatible with the IBM 6670. Thus the IBM 6670 can be used with existing applications, or as a basis for creating new applications in word processing.

For the application programmer, communication with the IBM 6670 is provided at the command level and macro level by batch data interchange, by terminal control, and by BMS. (The macro-level support is restricted to Assembler language only.)

For further information, see the CICS/VS IBM 3767, 3770, and 6670 Guide (SC33-0074).

When the IBM 6670 is used with CICS/VS BTAM support, it is attached by CICS/VS as an IBM 2770.

VSE/ADVANCED FUNCTIONS SUPPORT

CICS/DOS/VS Version 1 Release 4.1 can be run under Release 2 of VSE/Advanced Functions.

SUPPORT FOR FIXED BLOCK ARCHITECTURE (FBA) DIRECT ACCESS STORAGE DEVICES

CICS/DOS/VS data sets can reside on FBA devices. These data sets include:

- CICS/VS and user journals
- Dump data sets
- Auxiliary trace data sets
- Restart data sets
- Statistics data sets
- Extrapartition transient-data data sets
- Terminal control sequential data sets

In addition, the intrapartition data set can be a VSAM file and thus can be allocated to an FBA device. Users' VSAM files can also reside on FBA devices. The VSE system, however, does not support DAM or ISAM data sets on FBA devices.

SUPPORT FOR FIELD LEVEL SENSITIVITY (IMS/VS)

CICS/VS now supports the Field Level Sensitivity (FLS) facility provided by IMS/VS 1.1.5 (PTF UP90001). This facility allows DL/I application programmers independence from the sequence or position of the fields in a physical segment.

Appendix E. New Facilities of CICS/VS Version 1 Release 5

This appendix briefly outlines the new facilities and enhancements available with CICS/VS Version 1 Release 5. The new facilities are:

- New intercommunication facilities, offering:
  - Multiregion operation (MRO) — a new mechanism that allows communication between multiple connected CICS/VS regions within the same processing system without the use of SNA networking facilities.
  - Distributed transaction processing (DTP) — direct transaction-to-transaction communication across systems. (This facility is not available on MRO.)
  - Intersystem Communication between CICS/VS and IMS/VS.
  - Improved throughput by support of SNA parallel sessions.
- Enhanced master terminal facilities for interactive control of CICS/VS
- Command-level interface enhancements:
  - an interactive command interpreter.
  - a new command-level interface with DL/I DOS/VS.
  - Execution Diagnostic Facility enhancements to support DL/I commands.
- Security enhancements, including support for an external security manager (for example, the Resource Access Control Facility (RACF) program product).
- Improved monitoring facilities
- Further device support, including:
  - additional 3270 support.
  - use of the OS/VS console as a CICS/VS terminal.
  - networking of TWX and WTTY terminals through the Network Terminal Option (NTO) program product.
- Usability and serviceability aids, including a new user exit mechanism and facilities in CICS/DOS/VS similar to those provided by the FERS service aid.

Intercommunication Facilities

The CICS/VS intercommunication facilities allow two or more systems or regions to communicate and/or share terminals and other resources.

Prior to CICS/VS Version 1.5, the intersystem communication facilities allowed CICS/VS function requests to be shipped between CICS/VS systems that were connected through SNA networking facilities. CICS/VS Version 1.5 intercommunication facilities allow CICS/VS regions within the same processing system to be connected without the use of SNA. At the same time, the SNA support is enhanced to include a direct transaction interface, and SNA parallel session facilities are used to improve throughput.

MULTIREGION OPERATION (MRO)

Note: In the following discussion, the word "region" covers DOS/VS and OS/VS1 virtual partitions and OS/VS2 address spaces.

CICS/VS MRO allows a number of CICS/VS regions to be connected, without the use of VTAM, so that the systems can be run in parallel, while sharing files, transactions, and other resources. The connected systems normally look like a single system to the terminal user.

MRO has three main functions:

- Transaction routing, which allows operators of terminals in one CICS/VS region to run transactions in any connected CICS/VS region.
- CICS/VS function request shipping, which allows command-level application programs (or programs written for the DL/I CALL interface) to address resources in any connected CICS/VS region without knowledge of the placement of the resources; the CICS/VS functions supported are file control, transient data control, temporary storage control, interval control for transaction initiation, and DL/I access.
- Sharing of CICS/VS management modules, which allows many CICS/VS management modules to be placed in the system link pack area (CICS/OS/VS) or shared virtual area (CICS/DOS/VS). This allows the code to be shared between regions and thus reduces paging. It also offers increased integrity in both MRO and single-region environments.

  Support for management module sharing is independent of the other MRO facilities — CICS/VS management modules can be placed in the common area even for a single CICS/VS system, to increase data and system integrity.

These facilities have a number of potential uses. The user can choose to have one or more CICS/VS regions, and each region can run many applications or one. For example:

- New CICS/VS application programs can be tested concurrently with the production operation of CICS/VS, using a separate test region and so reducing possible impact on production without preventing access to existing resources and without requiring terminals to be dedicated to the test region.
- CICS/VS operation in a multiprocessor system can be improved by running in multiple OS/VS MVS address spaces.
- A CICS/VS system can be divided on a departmental basis, providing independent operation for one department while allowing controlled access to resources belonging to another department.

The system programmer specifies which regions are to be associated with which terminals, transactions, and resources; this enables the total system to be optimized across the CICS/VS regions. Terminals and other resources normally associated with a production application should be defined in the same region as the application's programs so that the overheads associated with interregion communication are minimized.

Most existing application programs will require no change in order to be invoked from terminals in other regions. Most existing command-level programs will also be able to use data resources defined in a different region without requiring changes. A case where changes might be required would be an application that communicates with a user exit.

INTERSYSTEM COMMUNICATION ENHANCEMENTS

The intersystem communication facility is enhanced to allow direct communication between transactions across systems (distributed transaction processing), communication between CICS/VS and IMS/VS systems, improved throughput, and the ability to tailor CICS/VS to other subsystems.

Distributed Transaction Processing (DTP)

The provision of a direct transaction-to-transaction communication interface supports distributed transaction processing, where an application runs partly on a local system and partly on a remote system. User-defined functions can therefore be distributed between systems in an SNA network (contrasting with the pre-1.5 support, where only CICS/VS functions are distributed). An application program using CICS/VS commands in PL/I, COBOL, or Assembler language can initiate, maintain, and terminate a conversation with an application program across a CICS/VS ISC link.

Direct communication between CICS/VS transactions can be a more efficient method of transferring data than CICS/VS function request shipping, since several data records can be placed in each message instead of needing a message for each CICS/VS command. On the other hand, CICS/VS function request shipping needs no special programming for many applications. CICS/VS provides sample programs to show how the CICS/VS facilities can be used for distributed transaction processing.

Distributed transaction processing uses SNA protocols and is dependent on functions introduced in ACF/VTAM Release 2. All CICS/VS ISC functions depend on the facilities of ACF/VTAM, with its multisystem networking facility for cross-domain linkage. Access methods offering compatible interfaces (ACF/TCAM Version 2, Release 2 and later, and ACF/VTAME) can also be used.

Note: Distributed transaction processing is not available for MRO.

Intersystem Communication between CICS/VS and IMS/VS

Intersystem communication functions available with CICS/VS Version 1 Release 5 for CICS/VS to CICS/VS communication are also available for communication between CICS/VS and IMS/VS Version 1.1.6.

Communication is with the DC component of IMS/VS, and can take place at two distinct levels. At the first, the CICS/VS application program communicates directly with an IMS queue; this level of communication is supported through extensions to the command level interface for Terminal Control. At the second level, the CICS/VS application program issues transaction initiation requests that are transferred to an IMS queue; this level of communication is supported by extensions to the command level interface for Interval Control.

The choice of level is influenced by the characteristics of the IMS transaction that processes the data, the volume of the data, the response time required, and the available programming effort.

As an example, a CICS/VS transaction in the system connected to the terminal could be an application program that performs all the terminal mapping functions in addition to any local file access or inquiry functions, while the connected IMS/VS system could contain the application program for updating the master data base. The CICS/VS application program need only initiate the IMS/VS application program when a particular data record is required. This would reduce the line traffic to occasional data transfer. It would also allow the device mapping function to be localized in a system nearer to the terminal.

Improved Throughput

CICS/VS Version 1 Release 5 introduces further use of SNA facilities for all ISC facilities. The parallel session capability introduced in ACF/VTAM Release 2 is used to allow a choice of the number of concurrent sessions between systems. This can offer throughput improvements for ISC traffic.

Improved Compatibility with other Subsystems

Use of the negotiable session initialization parameters introduced in ACF/VTAM Release 2 enables CICS/VS to adapt, where appropriate, to the capabilities of complementary subsystems in the ISC network.

Enhanced Master Terminal Transaction

The enhanced master terminal transaction, CEMT, provides a new interactive interface for controlling a CICS/VS system. The transaction is driven by commands similar in syntax to those of the command-level interface, allowing abbreviations and keyword defaults. CICS/VS interprets the command and diagnoses errors.

With CEMT, the master terminal operator can issue a single command to inquire about, or change the status of, groups of resources, such as all programs with names beginning with "ABC" or all terminals out of service.

The CEMT transaction is designed for use with display terminals that use the 3270 data stream and display 1920 or more characters; however,the transaction can also be entered from the system console (and certain other terminal types), with limited support.

The existing master terminal transaction, CSMT, will continue to be available with CICS/VS Version 1 Release 5.

A similarly enhanced supervisory terminal transaction (CEST) and operator terminal transaction (CEOT) are also included in CICS/VS Version 1 Release 5.

Command-Level Interface Developments

CICS/VS Version 1 Release 5 further develops the command-level interface introduced in CICS/VS Version 1 Release 3.

Command Interpreter

The command interpreter is a display-oriented program designed to help in writing and checking CICS/VS commands. It enables the operator of a display unit (with 1920 or more characters and supporting the 3270 data stream) to enter a command for immediate checking and optional execution. An application programmer can therefore construct a command interactively, thus using the command interpreter to complement the execution diagnostic facility (EDF), which enables an existing command-level program to be interactively tested.

In operation, the command interpreter takes the complete or partial command and, having checked its syntax, indicates default options, together with a diagnosis of missing or incorrect parameters. Once the command has been satisfactorily developed, the user may request its execution. In this way, both the syntax of the command and the result of executing it — including the screen display resulting from the operation — can be checked. The facility can also be used to create or inspect test data (for example, transient data) for use in application program testing.

Command-Level Interface with DL/I (CICS/DOS/VS only)

CICS/DOS/VS Version 1 Release 5 provides a new command-level interface with DL/I for application programs written in COBOL or PL/I. The new commands permit the following DL/I data base operations:

- Direct retrieval of records (GET UNIQUE)
- Limited direct retrieval of records (GET NEXT IN PARENT)
- Sequential retrieval of records (GET NEXT)
- Insertion of records
- Replacement of records
- Deletion of records
- Checkpointing

For CICS/VS online operation, scheduling and termination are also supported. In summary, the command-level DL/I interface is functionally

| equivalent to the existing DL/I call interface, but has the advantages
| of a less rigid syntax that is consistent with the CICS/VS commands.

| The existing DL/I call interface continues to be available as before,
| but command-level DL/I cannot be mixed with DL/I call statements in the
| same application program.

| Execution (Command Level) Diagnostic Facility (EDF)

| The Execution Diagnostic Facility is extended in CICS/DOS/VS Version 1
| Release 5 to support the command-level DL/I interface in addition to the
| existing CICS/VS commands. Because the DL/I commands can have many more
| operands than CICS/VS commands, and could therefore overflow the space
| available on one screen, a scrolling function is provided for the list
| of options in the command display panel.

Security Enhancements

A new interface is provided to enable the user to supplement CICS/VS security facilities by means of external security management programs. In particular, the new interface allows the MVS user of CICS/OS/VS to make use of the Release 3 facilities of the Resource Access Control Facility (RACF) program product (5740-XXH) to control user access to CICS/VS and its transactions. RACF can also be used to authorize access to one CICS/VS address space from another (for example, when using the
| multiregion operation facilities available with CICS/OS/VS Version 1
| Release 5.)

Increased authorization checking is available to application programs using the command-level interface. In addition to the existing check on user access to a transaction, there is a new option to check on a transaction's access to a resource. This option is particularly relevant to generalized transactions which may access many different resources (for example, files) according to user input. The new option is used by CICS/VS-provided transactions for the intercommunication facilities, the execution diagnostic facility, and the command interpreter. For transactions requiring resource authorization checks, the user specifies security keys for the resources to be protected in this way, for example, in file control table entries.

The CICS/VS sign-on procedure is made more flexible: passwords can be entered from a non-display field on an IBM 3270 or from an overstruck field on terminals such as the IBM 3767; use of an operator ID card can be stipulated in addition to keyed-input; and all sign-on and sign-off actions can be logged.

Improved Monitoring Facilities

A new data gathering mechanism improves the facilities for performance monitoring and for providing accounting data for individual users. The mechanism consolidates and extends data previously available in CICS/VS, offering a single interface for analysis routines. The user can choose the level of recording, depending on his needs for accounting, performance, or exception data. In this way, the user is allowed a high degree of selectivity, while analysis programs have a common mechanism for processing and extracting relevant information.

The improved monitoring facilities on CICS/OS/VS support System Management Facility (SMF) files. Monitoring data on SMF is intended for use on MVS in conjunction with MVS/SE Release 2.

The formats of many statistics fields have been changed to allow a greater magnitude and accuracy in the statistics.

New Device Support

CICS/VS terminal support is increased in several areas:

- Support for new 3270 devices and features, including extended color, extended highlighting, and the Programmed Symbols features.
- Support for the Network Terminal Option (NTO) program product.
- Support for the OS/VS console as a CICS/VS terminal.

NEW 3270 DEVICES AND FEATURES

Color, Extended Highlighting, and Programmed Symbols

New 3270 devices - the IBM 3279 Color Display Station and the IBM 3287 Printer Models 1C and 2C - provide color capability, extended highlighting, and have optional Programmed Symbols features that can provide graphic capability. Extended highlighting and Programmed Symbols are also available on some models of the IBM 3278 Display Station and on the IBM 3287 Printer Models 1 and 2. CICS/VS provides support for these new facilities through extensions to the programming interfaces, to permit the specification (and dynamic alteration) of the relevant attributes.

The Basic Mapping Support (BMS) interface allows extended color, extended highlighting, and programmed symbols attributes to be specified at field level in maps; the attribute values can also be changed dynamically (that is, during application program execution). The BMS SEND TEXT and TEXTBLD facilities allow the specification and alteration of the attribute values at individual character level.

Existing BMS maps will be compatible with those supporting the new function. The extension of application programs can thus be undertaken gradually, without the need for extensive alterations all at one time.

The Terminal Control (TC) interface allows users to create their own data streams, and thus specify the new 3270 data stream commands and

orders directly. Programs using this interface can specify extended color, extended highlighting, and programmed symbols attributes at both the field and the character level. If an application requires operator selection of the new attributes, the Write Structured Field command, which is required to set the inbound reply mode to include such attributes, must be sent using the TC interface.

Magnetic Stripe Reader

CICS/VS supports 3270 systems with the 10/63 magnetic stripe reader, so that data from a magnetic stripe can invoke a transaction or be used as input to a transaction.

Alternate Screen Sizes

Basic mapping support is extended to enable application programs to support 3270 alternate screen sizes, including new sizes such as that of the IBM 3278 Model 5, while remaining independent of screen size. Applications can also route messages to such displays independently of screen size.

Two-Byte Character Data

Some 3270 display stations can handle character sets that require each character to be represented in the data stream by two bytes of data. The facility is used, for example, to support the Kanji Japanese language character set.

CICS/VS allows the use of two-byte character data streams. BMS maps can be used to format the output data stream.

IBM 8775 DISPLAY TERMINAL

BMS supports functions of the IBM 8775 Display Terminal attached to IBM 8100 processors via the Data Stream Compatibility facility of 8100/DPCX. In output maps, application programs can specify (and dynamically modify), on a field basis, the field validation attributes 'mandatory enter' and 'mandatory fill'.

NETWORK TERMINAL OPTION

CICS/VS supports the Network Terminal Option (NTO) program product (5735-XX7) so that additional types of terminal can be used with CICS/VS through ACF/VTAM or ACF/TCAM Version 2 with ACF/NCP/VS. Support through NTO offers an alternative to the existing (BTAM) support for these terminal types, so that they can use SNA networking facilities. CICS/VS support is specifically for Western Union Teletypewriter Exchange Service (TWX Model 33/35) terminals and World Trade Teletypewriter (WTTY) terminals, and is designed so that existing TWX/WTTY application programs require minimal change for use with NTO. Physical attachment of terminals is as specified by NTO.

OS/VS CONSOLE AS A CICS/VS TERMINAL

CICS/OS/VS Version 1.5 introduces support for use of the OS/VS console as a CICS/VS terminal. This enables the console operator to invoke CICS/VS transactions from the console and allows other CICS/VS operators to communicate with the console operator. In particular, the console can be used for CICS/VS master terminal functions to control CICS/VS terminals or to control several CICS/VS regions in conjunction with multiregion operation. Normal operating system use of the console is not inhibited, and CICS/OS/VS supports multiple consoles where present.

Similar support is already available for CICS/DOS/VS.

Usability and Serviceability Aids

Improved usability and serviceability are provided by the addition of a new user exit mechanism and further problem determination aids.

NEW USER EXIT MECHANISM

A new user exit mechanism offers easier customization of CICS/VS. New exits can be incorporated without reassembly of CICS/VS management modules, so facilitating the use of CICS/VS pregenerated modules. The new mechanism encompasses most of the existing customization points and is designed so that logical groups of exits can be controlled dynamically and independently. New exits are written in assembler language, but have an improved interface with their host modules. Several exit programs can coexist independently at a single exit point, and, for CICS/VS Version 1 Release 5, existing exits can coexist with new exits; also, most existing exits can easily be converted to the new mechanism.

PROBLEM DETERMINATION AIDS

CICS/DOS/VS includes facilities designed to improve serviceability in a VSE environment. It incorporates functions similar to those of the service aid Facility Error Recognition System (FERS). This provides problem determination data for communications problems, particularly for 3270 displays supported through BTAM-ES. The user can display relevant statistics and details on BTAM-ES terminals and lines, individually or globally.

The VSE SYSDMP file can be used for CICS/DOS/VS problem data, such as partition dumps, so that this data can be accessed through the VSE/Interactive Problem Control System (VSE/IPCS).

CICS/DOS/VS supports the Maintain System History Program (MSHP) in VSE for use in installing CICS/DOS/VS and its service updates. This facilitates the maintenance of a history file containing service level information on CICS/DOS/VS.

System Requirements

This section gives the major system requirements for CICS/VS Version 1 Release 5. For full details of release dependencies, please consult your IBM representative.

CICS/DOS/VS VERSION 1 RELEASE 5

CICS/DOS/VS Version 1.5 is designed to run with Release 2 of the
| VSE/Advanced Functions program.

CICS/DOS/VS Version 1.5 supports BTAM-ES, ACF/VTAME, and ACF/VTAM Release 2. EXTM is not supported.

ACF/VTAME is supported as equivalent to ACF/VTAM Release 2 for those functions that are supported by ACF/VTAME.

Release 3 of ACF/VTAM is supported as equivalent to Release 2.

CICS/OS/VS VERSION 1 RELEASE 5

CICS/OS/VS Version 1.5 is designed to run under OS/VS1 or OS/VS2. SVS is not supported.

| CICS/OS/VS Version 1.5 supports BTAM, TCAM, ACF/VTAM Releases 1 and 2, and ACF/TCAM Version 1 and 2.

The Record Application Programming Interface of ACF/TCAM Version 2.2 is supported as equivalent to ACF/VTAM for those functions supported by TCAM.

ACF/VTAM Release 3 is supported as compatible with Release 2.

Glossary

This glossary includes definitions developed by the American National Standards Institute (ANSI). 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. ANSI definitions are preceded by an asterisk (*).

abend: abnormal end of task

ACB: Access method control block (VTAM and VSAM)

ACF: Advanced communications function

ANSI: American National Standards Institute

* ASCII (American National Standard Code for Information Interchange): The standard code, using a coded character set consisting of 7-bit coded characters (8 bits including parity check), used for information interchange among data processing systems, data communication systems, and associated equipment. The ASCII set consists of control characters and graphic characters.

BMS: Basic mapping support

BSC: Binary synchronous communication

BTAM: Basic telecommunications access method

COBOL: Common business-oriented language. A business data processing language.

| command: In CICS/VS, an instruction similar in format to a high-level
| programming language statement. (Contrast with macro.) CICS/VS
| commands invariably include the verb EXECUTE (or EXEC).

CSA: Common storage area

DAM: Direct access method

DB/DC: Data base/data communication

DCT: Destination control table

DL/I: Data Language/I

DOS: Disk operating system

DTP: Distributed Transaction Processing

* EBCDIC: Extended binary-coded decimal interchange code. A coded character set consisting of 8-bit coded characters.

EDF: Execution (command level) diagnostic facility

EIB: EXEC interface block

ELS: Entry Level System

EXTM: Extended telecommunication modules

EXEC: EXECUTE (as used in a CICS/VS command)

EREP: Error recording, editing, and printing

FBA: Fixed Block Architecture

FCT: File control table

FERS: Facility Error Recognition System

GAM: Graphics access method

ICR: Independent component release

Intercommunication Facilities: A generic term covering ISC and MRO.

ISAM: Indexed sequential access method.

ISC: Intersystem communication - communication between separate systems by means of SNA networking facilities.

| macro: In CICS/VS, an instruction similar in format to an assembler
| language instruction. (Contrast with command.)

map: In CICS/VS, a format established for a page or a portion of a page.

master terminal operator: Any CICS/VS operator whose security key(s) allow use of the master terminal functions transaction.

MRO: Multiregion operation — communication between CICS/VS systems in the same processor without the use of SNA networking facilities.

MVS: Multiple virtual storage. An alternative name for OS/VS2 Release 3.

NCP: Network control program

OPD: IBM Office Products Division

OS: Operating system

PCT: Program control table. The table defines the transactions known to the system.

* PL/I: A programming language designed for use in a wide range of commercial and scientific applications.

PPT: Processing program table. Defines all the application programs and maps in the system, and also various CICS/VS modules and tables.

Processor: Host processing unit

RACF: The Resource Access Control Facility program product.

* rollback: A programmed return to a prior checkpoint.

region: A section of the dynamic area that is allocated to a job step or system task. In this manual, the term is used to cover partitions and address spaces as well as regions.

RPG II: Report program generator, version 2. A commercially oriented programming language specifically designed for writing application programs that meet common business data processing requirements. (RPG is available only with CICS/DOS/VS.)

SAM: Sequential access method

SCS: SNA character stream

SDLC: Synchronous data link control

SIT: System initialization table

SNA: Systems network architecture

SRT: System recovery table

starter system: A set of pregenerated programs provided as part of the CICS/VS program product.

supervisory terminal operator: Any CICS/VS operator whose security key(s) allow use of the supervisory terminal functions.

SVS: Single virtual storage. An alternative name for OS/VS2 release 1.7.

task: (1) A unit of work for the processor; therefore the basic multiprogramming unit under the control program. (CICS/VS runs as a task under DOS/VS or OS/VS.) (2) Under CICS/VS, the execution of a transaction for a particular user. Contrast with transaction.

TCA: Task control area

TCAM: Telecommunications access method

TCT: Terminal control table

TIOA: Terminal input/output area

transaction: A transaction may be regarded as a unit of processing (consisting of one or more application programs) initiated by a single request. It may require the initiation of one or more tasks for its execution. Contrast with task.

Note: In many cases the request will originate at a terminal.

transaction identification code: Synonym for transaction identifier. For example: A group of up to four characters entered by an operator when selecting a transaction.

transaction identifier: Synonymous with transaction identification code.

TST: Temporary storage table

TWA: Transaction work area

* USASCII: Deprecated term for ASCII.

VS: Virtual storage

VSAM: Virtual storage access method

VTAM: Virtual telecommunications access method.

WT-A/FE: IBM World Trade, Americas/Far East

WT-E/ME/A: IBM World Trade, Europe/Middle East/Africa

Index

Each page number in this index refers to the start of the paragraph containing the indexed item.

Customer Information Control System/Virtual Storage (CICS/VS)
General Information

GC33-0066-5

READER'S COMMENT FORM

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.*

Number of your latest Technical Newsletter for this publication. . . . . .

Note: *Staples can cause problems with automated mail sorting equipment. Please use pressure-sensitive or other gummed tape to seal this form.*

Cut Along Dotted Line

If you want an acknowledgement, give your name and address below.

Name . . . . . .

Job Title . . . . . Company . . . . . .

Address . . . . . .

. . . . . Zip . . . . . .

Thank you for your cooperation. No postage stamp is necessary if mailed in the U.S.A. (Elsewhere, your IBM representative or IBM branch office will be happy to forward your comments.)

GC33-0066-5

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 40 ARMONK, NEW YORK

Postage will be paid by addressee:

International Business Machines Corporation
Department 812HP
1133 Westchester Avenue
White Plains, New York 10604

Fold and tape Please do not staple Fold and tape

CICS/VS General Information Printed in USA GC33-0066-5

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

Customer Information Control System/Virtual Storage (CICS/VS)
General Information

GC33-0066-5

READER'S COMMENT FORM

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.*

Number of your latest Technical Newsletter for this publication. . . . . .

If you want an acknowledgement, give your name and address below.

Name . . . . . .

Job Title . . . . . Company . . . . . .

Address . . . . . .

. . . . . Zip . . . . . .

Thank you for your cooperation. No postage stamp is necessary if mailed in the U.S.A. (Elsewhere, your IBM representative or IBM branch office will be happy to forward your comments.)

Note: *Staples can cause problems with automated mail sorting equipment. Please use pressure-sensitive or other gummed tape to seal this form.*

Cut Along Dotted Line

GC33-0066-5

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 40 ARMONK, NEW YORK

Postage will be paid by addressee:

International Business Machines Corporation
Department 812HP
1133 Westchester Avenue
White Plains, New York 10604

Fold and tape Please do not staple Fold and tape

CICS/VS General Information Printed in USA GC33-0066-5

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

Customer Information Control System/Virtual Storage (CICS/VS)
General Information

GC33-0066-5

READER'S COMMENT FORM

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.*

Number of your latest Technical Newsletter for this publication. . . . . .

Note: *Staples can cause problems with automated mail sorting equipment. Please use pressure-sensitive or other gummed tape to seal this form.*

Cut Along Dotted Line

If you want an acknowledgement, give your name and address below.

Name . . . . . .

Job Title . . . . . Company . . . . . .

Address . . . . . .

. . . . . Zip . . . . . .

Thank you for your cooperation. No postage stamp is necessary if mailed in the U.S.A. (Elsewhere, your IBM representative or IBM branch office will be happy to forward your comments.)

GC33-0066-5

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 40 ARMONK, NEW YORK

Postage will be paid by addressee:

International Business Machines Corporation
Department 812HP
1133 Westchester Avenue
White Plains, New York 10604

Fold and tape Please do not staple Fold and tape

CICS/VS General Information Printed in USA GC33-0066-5

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

Customer Information Control System/Virtual Storage (CICS/VS)
General Information

GC33-0066-5

READER'S COMMENT FORM

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.*

Number of your latest Technical Newsletter for this publication. . . . . .

Note: *Staples can cause problems with automated mail sorting equipment. Please use pressure-sensitive or other gummed tape to seal this form.*

— Cut Along Dotted Line —

If you want an acknowledgement, give your name and address below.

Name . . . . . .

Job Title . . . . . Company . . . . . .

Address . . . . . .

. . . . . Zip . . . . . .

Thank you for your cooperation. No postage stamp is necessary if mailed in the U.S.A. (Elsewhere, your IBM representative or IBM branch office will be happy to forward your comments.)

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 40 ARMONK, NEW YORK

Postage will be paid by addressee:

International Business Machines Corporation
Department **812HP**
1133 Westchester Avenue
White Plains, New York 10604

Fold and tape Please do not staple Fold and tape

CICS/VS General Information Printed in USA GC33-0066-5

IBM®

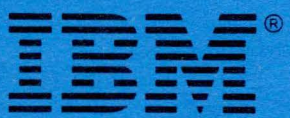
International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

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

GC33-0066-5

CICS/VS General Information Printed in USA GC33-0066-5



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