



IBM

World Trade Systems Centers

Boeblingen - Poughkeepsie - Raleigh - San Jose

DOS/VSE PLANNING GUIDE VOLUME 2

ZZ10-9826-00

IBM INTERNAL USE ONLY

ZZ10-9826

DOS/VSE Planning Guide
Volume II

IBM
World Trade Systems Center
Boeblingen

DOS/VSE PLANNING GUIDE
Volume II (ZZ10-9826)

(A Companion to Volume I, ZZ10-9825)

DOS/VSE RELATED
SOFTWARE PRODUCTS

March 1979

WTSC BOE

IBM Internal Use

NOTE TO THE READER

This WTSC publication is prepared for the use and guidance of IBM System Engineers and is shown to others for information only.

IBM gives no warranty nor makes any representation as to the accuracy of material in the publication, nor does it accept liability of any kind whatsoever arising out of the use of this document.

This document contains information believed accurate and current at the time of publication, with regard to DOS/VSE Release 1 and VSE/Advanced Functions Release 1 and related IBM licensed programs. The reader should be aware that subsequent changes either to the hardware features or to the above mentioned software, will not be reflected in this Guide. WTSC Boeblingen plans to provide update information on changes or enhancements through the normal WTSC distributions. However, this DAPS version of the the DOS/VSE Planning Guide with information on DOS/VSE Release 1 and VSE/Advanced Functions Release 1 will not be updated. Instead, the next edition of this Guide will contain specific planning information on details of VSE/Advanced Functions Release 2.

This DOS/VSE Planning Guide consisting of 2 volumes is a result of contributions from the following WTSC Boeblingen staff members:

Christa Grosse	-	VSE/ICCF and Data Security Chapters
Arnoldo Almeida	-	CICS/VS Chapter
Chris Engemann	-	Data Base Chapter
Erick Llaguno	-	Programming Facilities Chapter
Hans Merz	-	DOS/VSE Structure and DOS/VSE Basic Control Functions Chapters
Hansa Nishtala	-	CICS/VS and ACF/VTAME Chapters
Etienne Temmerman	-	Data Base Chapter
Garry Willinge	-	DOS/VSE Structure and DOS/VSE Basic Control Functions Chapters

Further, this Guide contains contributions from a number of other System Engineers. Consequently, WTSC Boeblingen wishes to thank the following people, whose input into various sections of the Guide, is greatly appreciated:

Kees Hage, (FSC Zoetermeer, Holland)	-	ACF/VTAME Chapter
Henk Middendorp, (FSC Zoetermeer, Holland)	-	FBM and CKD to FBM Migration Sections
Bill Schultz, (IBM Canada)	-	DOS/VSE Basic Control Functions Chapters
Ulrich Steinberg, (WTSC Poughkeepsie)	-	4300-Architecture Section

ZZ10-9826

DOS/VSE Planning Guide
Volume II

WTSC Boeblingen would like to thank, in anticipation, the readers of this Guide for their comments, corrections, recommendations and constructive criticism.

Any such comments or queries, with respect to this document, should be addressed to:

The Manager
WTSC Boeblingen
Dept. 3222, Bldg. 7030-44
Schoenaicher Strasse 220
7030 Boeblingen
West Germany

(ITPS code - EBOE)

This Technical Bulletin is an "IBM Internal Use Only" document and, therefore, its use is confined to IBM employees. The intent is to provide insight into the capability of IBM products for IBM Marketing Representatives and Systems Engineers.

This document has not been subjected to any formal review and has not been checked for technical accuracy. Results must be individually evaluated for applicability to a particular installation. You may discuss pertinent information from this bulletin with a customer, and you can abstract pertinent information for presentation to your customers. However, any code included is for internal information purposes only and may not be given to customers. If included code is identified as "incidental programming," its use must conform to the guidelines in the relative section of the Sales Manual.

The products referenced in this document may not be available in all countries.

Any performance data contained in this document was determined in a controlled environment; and therefore, the results which may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data in their specific environment. For this reason performance data contained in this document may not be abstracted and/or given to customers.

ZZ10-9826

DOS/VSE PLANNING GUIDE
Volume II

Chapter 1

CHAPTER 1

INTRODUCTION

CONTENTS

CHAPTER 1 INTRODUCTION. 1.3
1.1 Audience and Objectives. 1.3
1.2 Structure of the DOS/VSE Planning Guide. 1.5
1.3 Assumptions and General Comments 1.9

1.1 AUDIENCE AND OBJECTIVES

This guide is the result of a WTSC Boeblingen project, and is intended as a reference document to help Systems Engineers plan a customer's installation in terms of the new DOS/VSE related software.

The information contained in this document should eliminate duplicate fact finding efforts normally carried out for each customer installation.

The objectives of the Guide are as follows :

- (i) Provide planning information required for established customers migrating to new versions of IBM software.

Planning information is provided, where possible, for the use of new functions and the design of applications/systems exploiting these new functions.

Information is also provided to enable the Systems Engineer to prepare for the installation of, and migration to, the new version of the individual software component, in an efficient manner.

Productivity benefits, compatibility information, performance implications and examples of use of new functions are given where appropriate or necessary.

- (ii) Provide functional information for the Systems Engineer which may be valuable at proposal time for a potential first time user of a software product.

This is really a by-product of the Guide, but should be of value to Sales Representatives marketing these benefits of new version/releases of IBM software.

- (iii) Provide self education for the Systems Engineer on the use and exploitation of new functions in DOS/VSE related software.

- (iv) Provide a summary list of all available product reference material for the individual software components.
- (v) Provide an overview of the various DOS/VSE system offerings and their potential value to the DOS/VSE customer.
- (vi) Provide valuable material for the Systems Engineer which can be used in customer presentations on new functions of the various DOS/VSE software components.

1.2 STRUCTURE OF THE DOS/VSE PLANNING GUIDE

This is the second edition of this Guide and has been specifically produced to support planning and installation of the new 4300 range of machines as well as DOS/VSE and related software. Note also though, that DOS/VSE related information is also given for System/370 and 3031 hardware where relevant. Finally, Chapter 3 will also give a brief overview of the line items provided by VSE/Advanced Functions Rel 2 in 12/79. However, a new edition of this Guide will be released later, containing specific information about the new facilities of VSE/Advanced Functions Rel 2.

The DOS/VSE Planning Guide is logically divided into three parts contained in two separate volumes.

Volume I (ZZ10-9825) contains part i (DOS/VSE Basic Control Functions) and Volume II (ZZ10-9826) contains parts ii and iii (DOS/VSE related licensed programs and DOS/VSE system offerings).

The three parts are :

VOLUME I

Part i : DOS/VSE Basic Control Functions

The purpose of this part is to describe the enhancements and new features of the various DOS/VSE basic control functions. These include:

- DOS/VSE SCP (5745-020) (Chapter 4)
- VSE/Advanced Functions (5746-XE8) (Chapter 4)
- VSE/VSAM (5746-AM2) (Chapter 5)
- VSE/POWER (incl. RJE) (5746-XE3) (Chapter 6)
- ACF/VTAME (5746-RC7) (Chapter 7)

With regard to descriptions of new functions, the topics discussed generally address the following areas of interest :

- (a) What are the new functions?
- (b) How and when should they be used?
- (c) How are they implemented?
- (d) Examples of use of the new functions.
- (e) Compatibility information with regard to conversion from previous releases of the software component in question and with regard to other interfacing programs.
- (f) Performance implications of installation of new functions.

Besides descriptions of new functions, the following aspects have also been considered :

- (a) Where it is known that a particular topic is either not well documented in other manuals, or is of special interest to the field, additional information may be provided. For example, in Chapter 4, a special section has been included on Multi-CEC (or Multi-CPU) considerations to describe exactly what is supported under DOS/VSE at the present time. Of course, the next edition of this Guide, being oriented towards VSE/Advanced Functions Rel 2 details, will contain alot more information on this aspect because of the greatly enhanced DOS/VSE support of DASD sharing contained in the VSE/Advanced Functions Rel 2 licensed program.
- (b) Within each topic, tables summarizing what has been discussed, are included where relevant. For example in the topic that discusses DASD support under DOS/VSE, a table summarizing this support has been included.
- (c) With some topics discussed in the Guide, no new features or functions have been provided by DOS/VSE as compared to DOS/VS Release 34. These topics have been included for the sake of completeness or to set up a topic for update in future editions.

VOLUME II

Part ii : DOS/VSE Related Licensed Programs.

The purpose of this part is to describe supporting information for the Systems Engineer on a number of SPL (Selected/Standard Product Line) software components. It is intended to update this part in future editions of the DOS/VSE Planning Guide to include valuable planning information on the major strategic products in the IBM product line, particularly as new versions/releases are announced.

The software components discussed in this edition of the Guide are:

- (a) Data Base Function Group (Chapter 8)
 - DL/I DOS/VS (5746-XX1)
- (b) Data Communications Function Group (Chapter 9)
 - CICS/VS (5746-XX3)
- (c) DOS/VSE Programming Facilities (Chapter 10)
 - RPG II (5746-RG1)
 - DMS/CICS/VS (5746-XC4)
 - SORT/MERGE Version 2 (5746-SM2)
- (d) DOS/VSE Interactive Function Group (Chapter 11)
 - VSE/ICCF (5746-TS1)
- (e) DOS/VSE Data Security Facilities (Chapter 12)
 - VSE/ACCESS CONTROL - LOGREP (5746-XE7)

Not all of the above products have been discussed in detail, but where possible, the following topics have been discussed (especially for DL/I, CICS/VS and VSE/ICCF):

- (a) Description of product function.
- (b) Description of enhancements and use of the latest release of the software product.
- (c) Installation and migration considerations of the latest release of the software product.

- (d) Summary of all the product reference material, available to the Systems Engineer.

Part iii : DOS/VSE Software System Offerings.

The purpose of this part (Chapter 13), is to describe the benefits of the various offerings and include relevant information for the Systems Engineer on the following:

- (a) Objectives of each offering.
- (b) Components of each offering.

This Guide will place emphasis on the native/complex DOS/VSE System Offerings. In other words, those offerings which include VM software, will not be addressed here. It is intended to document useful installation experience on the non-VM offerings as soon as it becomes available.

1.3 ASSUMPTIONS AND GENERAL COMMENTS

- (i) The DOS/VSE Planning Guide, is aimed at the intermediate systems DOS/VSE user (optionally co-existing under VM/370). This includes entry level users.

It is assumed that the customer has been identified as such, before this Guide is consulted.

- (ii) The software products discussed in this DOS/VSE Planning Guide, are products specifically supported by WTSC Boeblingen. VM software does not fall into this category so no emphasis will be placed on VM planning.

- (iii) Finally it should be pointed out that this Guide, being the second in the series, attempts to repeat as little as possible, what has already been expressed in the first edition. Consequently, frequent references are made to the first edition, its DAPS number being ZZ10-9822-00. Therefore, it is suggested that the reader creates a library of DOS/VS and DOS/VSE Planning Guides, so that all referenced topics may be readily accessible when required.

Chapter 1.3
Introduction

DOS/VSE PLANNING GUIDE
Volume II

ZZ10-9826

ZZ10-9826

DOS/VSE PLANNING GUIDE Chapter 2
Volume II New System Architecture

CHAPTER 2

NEW SYSTEM ARCHITECTURE

(Refer to Volume I for this Chapter)

ZZ10-9826

DOS/VSE PLANNING GUIDE
Volume II

Chapter 3

CHAPTER 3

DOS/VSE STRUCTURE

CONTENTS

3.1 Overall Contents 3.1

3.2 General Structure. 3.2

3.2.1 Comments DOS/VSE Availability 6/79 3.3

3.2.2 Comments DOS/VSE FCS 12/79 3.3

3.3 DOS/VSE Related Product Highlights 3.4

3.3.1 DOS/VSE SCP. 3.4

 BASIC SCP. 3.4

 SCP - ASSEMBLER. 3.5

 SCP - Standalone Utilities 3.5

 SCP - System Utilities 3.6

 SCP - Removal of Programming Support 3.6

3.3.2 VSE/Advanced Functions 3.6

 VSE/Advanced Functions Rel 1 3.7

 VSE/Advanced Functions Rel 2 3.7

3.3.3 VSE/ICCF Rel.1 3.8

3.3.4 VSE/POWER. 3.9

 VSE/POWER Rel.1.0. 3.10

 VSE/POWER REMOTE JOB ENTRY FEATURE 3.10

 VSE/POWER Release 2. 3.11

 VSE/POWER Shared Spooling Feature. 3.11

3.3.5 VSE/VSAM 3.11

 VSE/VSAM Release 1.0 3.11

 VSE/VSAM Release 2.0 3.12

 VSE/VSAM SPACE MANAGEMENT. 3.12

3.3.6 BTAM-ES Rel.1. 3.12

3.3.7 Job Entry Program (JEP) and File Transfer Program
(FTP) 3.13

3.3.8 VSE/ACCESS CONTROL - LOGGING AND REPORTING 3.14

3.3.9 ACF/VTAME. 3.15

3.3.10 Languages, Sorts and Utilities. 3.16

 DOS/VS RPG II Release 3. 3.16

 DMS/CICS/VS. 3.17

 DOS/VS Sort/Merge Version 2 Release 2. 3.18

 VSE/DITTO. 3.18

 VSE/IBM SYSTEM/3-3340 DATA IMPORT. 3.19

3.3.11 Emulators 3.19

3.3.12 DB/DC Program Products. 3.20

 CICS DOS/VS Release 1.4. 3.20

 DL/I DOS/VS Release 1.5. 3.22

3.3.13 Other Licensed and Application Programs 3.22

3.4 DOS/VSE Supported Licensed Programs Summary. 3.24

3.5 Glossary of Names/Machine Numbers in This Guide. . . 3.29

ZZ10-9826

DOS/VSE PLANNING GUIDE
Volume II

Chapter 3.1
Contents

3.6 4300 Processor Bibliography.3.31
CHAPTER 3 DOS/VSE STRUCTURE i

3.1 OVERALL CONTENTS

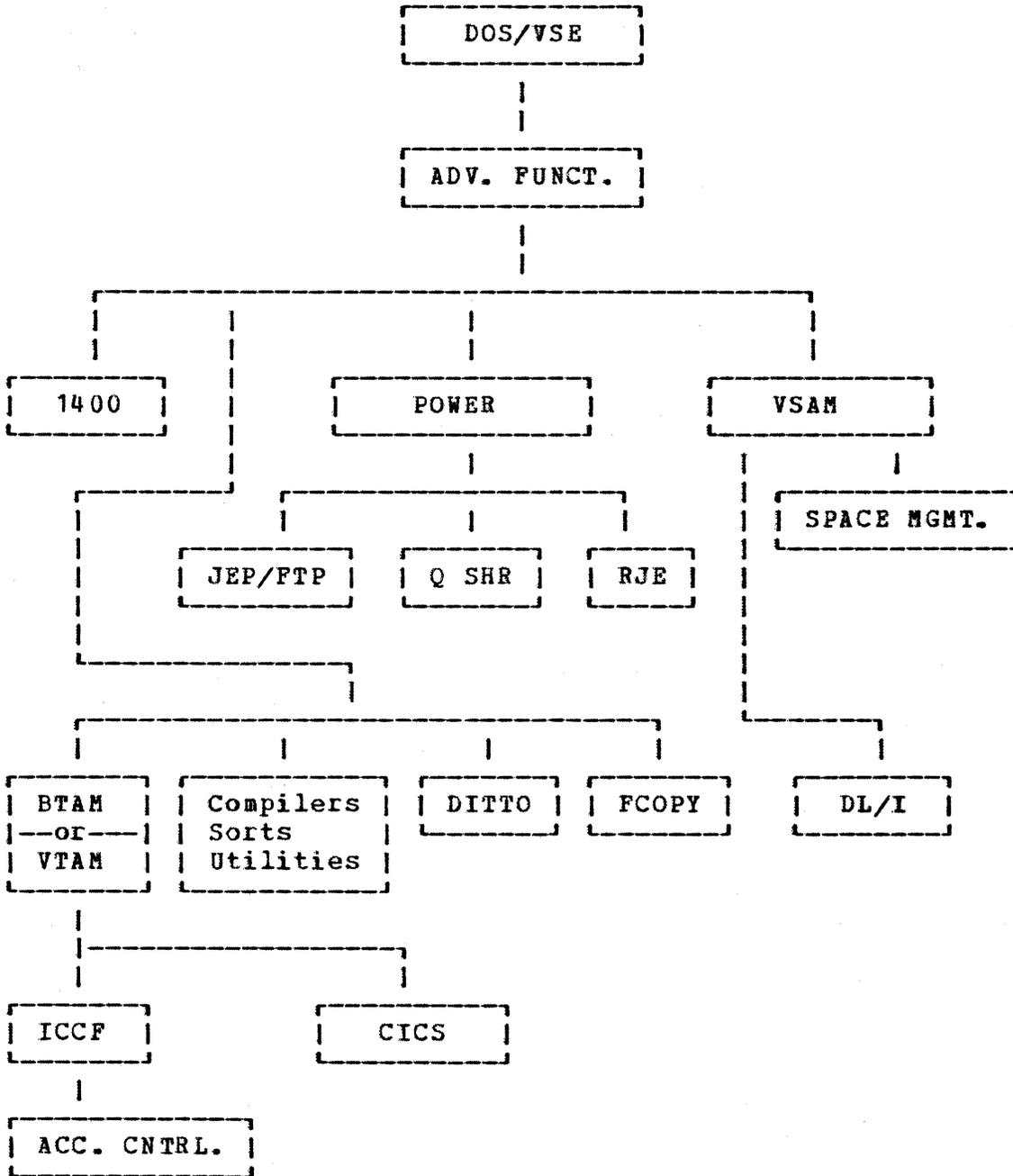
According to the 4300 processors and DOS/VSE announcement of 1Q79, DOS/VSE will be composed of the following components:

- DOS/VSE SCP
- VSE/Advanced Functions (Releases 1 and 2)
- VSE/VSAM (Releases 1 and 2)
- VSE/POWER (Releases 1 and 2)
- VSE/ICCF Release 1
- BTAM-ES or VTAM (ACF/VTAME or ACF/VTAM)
- Job Entry Program (JEP) and File Transfer Program (FTP)
- Languages, Sorts, and Utilities
 - includes DOS/VS RPG II (Release 3)
- Emulators
 - includes IBM Systems 1401/1440/1460 Emulator Program Release 1
- DB/DC program products
- Other Licensed and Application Programs

Notes:

- Only the DOS/VSE SCP will be free of charge. All other components are Licensed Programs, and therefore will be charged for.
- The System Utilities and Standalone Utilities are part of DOS/VSE.
- The ASSEMBLER is part of DOS/VSE.
- Do not forget that a an IR for 3800 printer support still exists. It is called the VSE/IBM 3800 Printing Subsystem IR, program number 5747-CC1.

3.2 GENERAL STRUCTURE



3.2.1 COMMENTS DOS/VSE AVAILABILITY 6/79

- All components require VSE/Advanced Functions (ADV. FUNCT. in above diagram) as a prerequisite product.
- The VSE/ACCESS CONTROL - LOGGING AND REPORTING (ACC. CNTRL. in above diagram) requires VSE/ICCF Rel. 1 as a prerequisite product.
- The ASSEMBLER is part of DOS/VSE.
- The System and Stand-alone Utilities are part of DOS/VSE.

3.2.2 COMMENTS DOS/VSE FCS 12/79

- The following components will have new Releases:
 - VSE/Advanced Functions Rel 2
 - VSE/POWER Rel.2
 - VSE/POWER SHARED SPOOLING FEATURE
 - VSE/VSAM Rel.2
 - VSE/VSAM SPACE MANAGEMENT FOR SAM FEATURE
 - VSE/FAST COPY DATA SET PROGRAM Rel.1 (this is the first availability of this licensed program)
 - Job Entry Program (JEP) and File Transfer Program (FTP)
- All other components will be at the DOS/VSE 6/79 level.

3.3 DOS/VSE RELATED PRODUCT HIGHLIGHTS

3.3.1 DOS/VSE SCP

The DOS/VSE SCP is available along with a priced extension - VSE/Advanced Functions, which is required for running all of the other products.

The following line items make up the enhancements incorporated in the DOS/VSE SCP:

BASIC SCP

- Hardware Support
 - 4300 Processor Support
 - /370 Support
 - Full 3031 Support
 - Support of High Resolution Timer
 - FBM Support
 - Extended AP-1 Program Support For 3310,3370 DASD
 - 3880-33xx / 3370 support
 - 8809 Support
 - 3800 ICR Integration
 - 3289-4 Printer Support
 - 3203-5 Printer Support
 - 3278-2A Support (in 3277 Compatibility Mode)
 - Improved Console Printer Support
 - 5424 Support
 - Common EREP (OS Compatible)
- Removal of Restrictions
 - More LUBS/Partition
 - Less Copy Blocks for CCW-Translation
 - Extent Information removed from JIBS
- Operational Improvements
 - Improved SVA Bring-up
 - DASD Volume Recognition
 - VOLUME Command
 - Improved 'LISTIO'
 - Remove X'...' in IPL and JCL
 - Continuation of Commands/JCL Statements
 - Cancel Mount Requests

- Automatic 1403U UCB Load
- COPYSERV (COPYSORT) Integration
- Performance Enhancements
 - IORB Concept
 - Alternate Path I/O for DASD
 - Improved Load Leveller
 - Fast CORGZ
 - EOJ
 - Console Buffering for CRT Consoles
- Improved Problem Determination / Installation
 - DUMPS in SVA
 - Other DUMP Improvements (PDUMP)
 - OLTEP Partition Independence
 - SDAID Improvements / PDAID Removal
 - MSHP
- Job Accounting Improvements
 - Use of CPU TIMER
 - Suppress SIO Counting for GENERIC ASSIGN
 - Repeatable 'Times'
 - Change in JOBACCT Table
- Miscellaneous Enhancements
 - Pageable Supervisor (optional)
 - Reduced Supervisor Options
 - Subsystem Identification Support
 - Label Area Support
 - New Parameters in OPTION Statement
 - Operator Communication for VTAM
 - GETVIS for Real Partitions
 - IOS RAS Support
 - New Parameter in DLBL Statement
 - Reentrant Supervisor Communication MACROS

SCP - ASSEMBLER

- Support of ECPS:VSE instructions
- Short Cross Reference List
- Support of Long SETC
- FBM Support of Workfiles

SCP - STANDALONE UTILITIES

- General - Tape Only
- New Utilities
 - Surface Analysis (FBM)
 - Format Emulated Extent
- Changed Utilities
 - Initialize Disk

- Restore System
- Fast Copy Disk

SCP - SYSTEM UTILITIES

- New Utilities
 - Assign Alternate Block
 - Printer Train Cleaning
- Changed Utilities
 - Backup System
 - Restore System
 - Copy and Restore Diskette
 - OBJMAINT
 - Fast Copy Disk (CKD and FBM)
 - Initialize Tape
 - Initialize Disk
 - Print HARDCOPY File (PRINTLOG)
 - VTOC Display

SCP - REMOVAL OF PROGRAMMING SUPPORT

- QTAM Support
- System /360 Mod. 20 Emulator
- 2321 Data Cell Support
- 2495 Tape Cartridge Reader Support
- Unblock Utility
- Copy/Restore Disk or Data Cell
- PTFHIST and HISTLIST Utilities
- Supervisor Seek Separation
- PDAID

3.3.2 VSE/ADVANCED FUNCTIONS

VSE/Advanced Functions Releases 1 and 2 are the successors of the ADVANCED FUNCTIONS-DOS/VS Licensed Program which was provided along with DOS/VS Release 34. The Release 34 functions are retained and new functions are provided.

The only pre-requisite for installing VSE/Advanced Functions is that the customer must have already installed DOS/VSE.

The customer must be aware that installation of VSE/Advanced Functions is mandatory before any IBM licensed programs may execute.

The following line items make up the enhancements incorporated in the VSE/Advanced Functions Release 1 product:

VSE/ADVANCED FUNCTIONS REL 1

- Operational Improvements
 - Implicit Invocation of Linkage Editor
 - Keep Messages with Outstanding Reply
 - Automatic System Initialization (ASI)
 - LBLTYP Statement Elimination
- Performance Enhancements
 - Fast CCW-Translate Switchable by JCL
 - High Level SDL Search
 - Fast B-Transient Fetch
 - Fast C-Transient Fetch
 - Fast Open of HARDCOPY File
 - Multiple Extent Page Data Set
 - Improvement of OPTION Statement
- Other Improvements
 - Area for Job-to-Job Communication
 - Label Area as Separate Data Set
 - Support of 7 Partitions for 3340 SYSRES
 - Dump Improvements
- VMLE Enhancements
 - Support of 3310 and 3370 in System/370 Compatibility Mode

The following line items make up the enhancements incorporated in the VSE/Advanced Functions Release 2 product:

VSE/ADVANCED FUNCTIONS REL 2

- Functional Description
 - SYSLOG on SYSLST (LISTLOG) Utility
 - Additional DLBL Information
 - Dynamic Device Assignment
 - SYSLNK in VSAM Managed Space
 - Extended Volume Command
 - Support of Volume Mount Requests
 - Shared Libraries
 - Concatenated Libraries
 - Multiple Procedure Libraries
 - DASD Sharing Across Systems
 - B-Transient Contention Removal

- General Services for IBM Components
- More Partitions / More Tasks
- System Tailoring at IPL
- New Label Area Support Functions
- On-line SYSGEN under Rel.1 Control

3.3.3 VSE/ICCF REL.1

The DOS/VSE Time Sharing System (VSE/ICCF) is an extension of the ETSS FDP's and is now offered as a LICENSED PROGRAM. It complements the interactive strategy on 4300 processors as well as on System /370. VSE/ICCF provides for interactive program development and execution, interactive system operation, and problem solving.

Pre-requisites for running of VSE/ICCF are that the following products are installed:

- DOS/VSE
- VSE/Advanced Functions

As well as this the user must generate a supervisor with 'ICCF=YES' in the SUPVR macro, after having installed VSE/Advanced Functions. Please note that specification of 'ICCF=YES' forces generation of 'FP=YES'; this means that the CPU must be equipped with the Floating Point feature. Note also, that if DOS/VSE data security support is also required (see a later section), then another parameter 'SEC=nn' should be specified in the FOPT macro of this supervisor.

The following line items make up the enhancements incorporated in the VSE/ICCF product :

- An interactive facility for DOS/VSE on System/370 and 4300 processors
- An online library system to store programs, jobs and data
- An online source program and data entry, update and maintenance facility, which enhances program development productivity
- An online library maintenance tool with copy, purge, insert, rename and resequence functions, data compression capability, library change control and auditing assist functions. This again should enhance productivity as well as add a level of a security in the DP installation
- A powerful editing function, including a context and full screen editor
- A direct job execution facility, providing compile, load and go compatibilities for all DOS/VSE language compilers, including interactive compilers

- A facility to run most existing single task batch programs interactively by submitting terminal I/O for card input/printer-punch output
- A submit-to-batch via VSE/POWER, allowing the user to initiate jobs from the terminal, for later batch processing. Printed results may be received at the terminal or the system printer or both.
- Interactive usability aids, in the form of prompter facilities to ease creation of DOS/VSE, VSE/POWER and VSE/ICCF jobs; ease installation and operation; and to lower skill requirements. These interactive usability aids are supported on display terminals of 1920 characters or more.
- System Data Security Aid (Access Control), allowing each installation to select the security provisions appropriate to its needs
- A procedure processor with logic capabilities, allowing both the installation and the users to build their own commands and procedures
- Dynamic disk space allocation, freeing the installation from having to pre-allocate or pre-define files
- Support of FBM devices for VSE/ICCF libraries
- User interface to VSE/ICCF through the IBM 3270 terminal system, utilizing such features as full screen support, program access keys, program FUNCTION KEYS (for command substitution) as well as forward/backward paging, right/left line shift and so on
- The ability to share terminal devices with CICS/VS, thus eliminating the need for separate terminal networks and duplicated terminal control functions
- Storage protection for INTERACTIVE PARTITIONS IS STANDARD

3.3.4 VSE/POWER

As in past DOS/VS releases, VSE/POWER provides the spooling and job scheduling facilities for DOS/VSE. The VSE/POWER SHARED SPOOLING FEATURE (prerequisites are VSE/AF Rel 2 and VSE/POWER Release 2) and the VSE/POWER REMOTE JOB ENTRY FEATURE are added to extend its Co-existence and Remote Job Entry capabilities.

Pre-requisites for VSE/POWER Release 1 are DOS/VSE and VSE/Advanced Functions Release 1, while those for VSE/POWER Release 2 are DOS/VSE and VSE/Advanced Functions Release 2.

The following line items make up the enhancements incorporated in the VSE/POWER Release 1 product :

VSE/POWER REL. 1.0

- Enhancements vs. Rel.34 POWER/VS
 - FBM Support
 - Expanded PALTER
 - PHOLD Command
 - Blocked Tape Reader
 - SLI Nesting
 - Page Count Improvements
 - Offloading of Queues
 - 3800 ICR Integration
 - 8809 Support
 - Quality Improvements

The basic VSE/POWER Release 1 system may be extended by the installation of a separately orderable optional VSE/POWER Remote Job Entry Feature that allows spooling via remote BSC or SNA work stations. The pre-requisites for this feature are: /

- DOS/VSE
- VSE/Advanced Functions
- DOS/VS ACF/VTAM Release 2 (5745-010) - for RJE,SNA support, if 370X control units used.
- ACF/VTAME (5746-RC7) - for RJE/SNA support in the 4331 ICA.
- DOS/VS NCP Version 5.0 (5747-AJ2) - for RJE,SNA support, if 370X control units used.
- DOS/VS EP (5746-AG1) - for RJE,BSC support, if 370X control units used.

The following line items make up the enhancements incorporated in VSE/POWER REMOTE JOB ENTRY FEATURE:

VSE/POWER REMOTE JOB ENTRY FEATURE

- Enhancements vs. Rel.34 POWER/VS
 - Full 3741 Support
 - SNA / Exchange Media Inbound (support for 3760 Data Entry Station)
 - Multileaving Support (BSC line connection)

The following line items make up the enhancements incorporated in VSE/POWER Release 2 as compared to VSE/POWER Release 1:

VSE/POWER RELEASE 2

- Support of 3262-1 Printer
- Support of Extended Number of Partitions
- Partial support for concatenated libraries (first in chain only)
- VSE/ICCF Interface allowing interrogation of POWER Queue etc.

The following line items show the facilities provided by the VSE/POWER Shared Spooling Facility:

VSE/POWER SHARED SPOOLING FEATURE

- Sharing of all or any POWER file(s) between CEC's (CPU's)
- Sharing of SLI (Source Library Include) Library

Note, that there are no new enhancements with regard to RJE support. The VSE/POWER RJE Feature as made available with VSE/POWER Release 1 is again available with VSE/POWER Release 2.

3.3.5 VSE/VSAM

As in past DOS/VS releases, VSAM provides Data Management functions and is the primary DASD Access Method for DOS/VSE. The VSE/VSAM SPACE MANAGEMENT FOR SAM FEATURE expands its space management capabilities to also support SAM files.

Pre-requisites for VSE/VSAM Release 1 and 2 are VSE/Advanced Functions Release 1 and 2 respectively. Similarly, the pre-requisites for VSE/VSAM SPACE MANAGEMENT FOR SAM FEATURE are VSE/AF Rel 2 and VSE/VSAM Release 2.

The following line items make up the enhancements incorporated in VSE/VSAM Releases 1 and 2 as well as in VSE/VSAM SPACE MANAGEMENT FOR SAM FEATURE :

VSE/VSAM RELEASE 1.0

- Enhancements vs. Rel.34 VSAM
 - No-Imbed for Catalog

- Control Interval Split Integrity
- FBM Device Support
- Fixed Head Support via Use of Space Classes
- Support of New Block Sizes
- EXPORT / EXPORTRA Performance
- IORB Support
- 8809 Support
- SHOWCAT Macro Improvement

VSE/VSAM RELEASE 2.0

- Enhancements vs. VSE/VSAM Release 1.0
 - DASD Sharing
 - Catalog Management Performance Improvements
 - Shareoptions 4 Performance Improvements
 - Control Area Split Integrity
 - AMS CANCEL Command
 - Serviceability Upgrade (User Exit Facility for De-bugging)
 - VSAM Space Management and Usability Enhancements

VSE/VSAM SPACE MANAGEMENT

- VSAM Space Management for SAM Files

3.3.6 BTAM-ES REL.1

As for BTAM SCP in DOS/VS Release 34, BTAM-ES provides teleprocessing control program support for communication between a host computer and a variety of start-stop, BSC and local channel attached terminals. The function of BTAM-ES has not been enhanced over DOS/VS Release 34 BTAM, but BTAM-ES will operate on 4300 processors and minor changes have been incorporated to allow proper interfacing with DOS/VSE.

Because BTAM-ES is substantially the same as it was in DOS/VS Release 34, no further details will be provided on this program in this Guide. However, the following migration considerations should be noted for the customer moving to BTAM-ES :

- The BTMOD macro must be re-assembled and re-linked. This is mandatory.

- For native BTAM users, it is suggested that application programs be re-compiled, so as to obtain the benefit of the latest maintenance level of the BTAM macros. However, it is possible that re-compilation of such application programs, may not be mandatory. Therefore, WTSC Boeblingen will investigate the alternatives and document clarification of this aspect in a later WTSC Boeblingen bulletin.
- Finally, for migration considerations with respect to CICS/VS users, the appropriate CICS/VS chapter later in this Guide, should be referenced.

3.3.7 JOB ENTRY PROGRAM (JEP) AND FILE TRANSFER PROGRAM (FTP)

The Job Entry Program, referred to as JEP, provides the basic interconnection mechanism between a DOS/VSE SUBHOST to a job entry subsystem HOST. It offers job submission from a SUBHOST to a job entry subsystem at the HOST and routing of job output back to the originating SUBHOST or to an alternate destination. JEP communicates with the following job entry subsystems at the HOST:

- OS/VS2 JES2
- OS/VS2 JES3
- OS/VS2 NJE/JES2
- OS/VS1 RES/JES
- DOS/VSE with VSE/Advanced Functions, VSE/POWER, and VSE/POWER REMOTE JOB ENTRY FEATURE.

Note, that pre-requisites for JEP are:

- DOS/VSE with VSE/Advanced Functions
- VSE/POWER
- ACF/VTAM with MSNF (or ACF/VTAME if JEP is resident in the 4331 processor)

Be aware though, that JEP does not preclude the co-existence of VSE/POWER REMOTE JOB ENTRY FEATURE. In fact, with the VSE/POWER REMOTE JOB ENTRY FEATURE installed, JEP offers the capability for SNA workstations and BSC terminals connected to VSE/POWER in the DOS/VSE SUBHOST, to submit jobs through the SUBHOST to the HOST.

The File Transfer Program, referred to as FTP, expands the capabilities of JEP. Together with JEP, the FTP installed at the HOST, offers capabilities required to centralize program maintenance activities to the HOST, thus saving the user the

cost of keeping programmer teams at each subhost location. With JEP and FTP, jobs can be submitted from the HOST to the SUBHOST, and the job output can be transmitted back to the originating HOST or to an alternate HOST system. As well as this facility, with FTP installed at both the HOST and SUBHOST, JEP and FTP offer transmission of sequential files between the DOS/VSE SUBHOST and the HOST in both directions. The file transfers may be initiated in one or the other direction, either at the SUBHOST or the HOST.

The following line items show the facilities provided by the JEP and FTP programs:

- Collection of Data and Periodic Transmission of Jobs or Files to a HOST Computer for Central Processing
- Transmission of output data (job output or files) to remote locations for processing
- Transmission of Data Files Between the HOST and SUBHOST for Processing at the HOST or SUBHOST
- Submission of Jobs and Routing of job output from the SUBHOST to the HOST and from the HOST to the SUBHOST in order to offload the Submitting System (e.g. DOS/VSE to DOS/VSE environment) or to submit OS/VS Jobs at a DOS/VSE Installation or submit DOS/VSE Jobs at an OS/VS Installation etc.
- Sharing of the Communications Link among various Applications.
- Co-existence between BSC and SNA Components, together with VSE/POWER.
- Security and Integrity by means of JEP Generation Parameters and Authorization Passwords

3.3.8 VSE/ACCESS CONTROL - LOGGING AND REPORTING

This is a new program product which provides part of the data security functions offered for batch users of DOS/VSE. These security functions are designed for auditing purposes of a DOS/VSE data processing environment. In particular, the following events can be controlled, checked and logged for security reasons - data set access, library access, program access, user identification and user verification. A security log file is used for storing the logged information and a special report program is provided to allow printing of selected data from this file for auditing purposes. VSE/Advanced Functions and VSE/ICCF Release 1.0 are prerequisites for data security in DOS/VSE.

The following line items are the new facilities provided by this licensed program:

- Logging of information about access related events to protected resources (such as data sets, libraries and programs) to a sequential disk data set
- Logging (which is optional) is performed by a DOS/VSE subtask attached to VSE/ICCF
- Reporting provides a printout of selected data from the above mentioned log data set, for auditing purposes
- Selection criteria is provided to the reporting module during program execution by means of control statements
- Several reports corresponding to various criteria may be obtained in one program run
- Access event data may be archived on a tape and restored from tape for deferred reporting
- The log data set consists of two parts used alternatively; one part for logging and one part for reporting
- Bibliography : GH12-5045 - Design Objectives
GH12-5130 - General Information
SH12-5536 - Program Reference and Operations
GC33-6077 - Data Security Guide DOS/VSE
LY12-5032 - Logic

3.3.9 ACF/VTAME

ACF/VTAME is a new Licensed Program (program number 5746-RC7) for users of DOS/VSE. ACF/VTAME Release 1 offers support for the Communication Adapters of the 4331 and also provides support for the various channel attached devices. Key ACF/VTAM and ACF/NCP/VS functions are combined in ACF/VTAME. The functions provided by ACF/VTAME are compatible with analogous functions provided by ACF/VTAM Release 2. ACF/VTAME provides support for multi-system networking.

Highlights

- Support for the 4331 Communication Adapter instead of the 3705 communications controller.
- Support for a variety of devices:
 - SNA-SDLC
 - BSC 3270
 - SNA and non-SNA channel attached
- Upwards compatible with ACF/VTAM Release 2.
- Allows concurrent use of multiple ACF/VTAME application programs.
- Controls the sharing of network resources among ACF/VTAME application programs.

- Provides for application program to application program and application program to terminal sessions.
- A multisystem networking capability is provided as an integral part of ACF/VTAME functions.
- Support of CICS/VS
- Support of the Network Operation Support Program (NOSP) LP.
- Support of the Network Communication Control Facility (NCCF) LP and its related Network Problem Determination Application (NPDA) LP.

3.3.10 LANGUAGES, SORTS AND UTILITIES

Concurrently with the announcement of DOS/VSE, the following new languages, sorts and utilities are announced and covered in this Guide:

- DOS/VS RPG II Release 3
- DMS/CICS/VS
- DOS/VS Sort/Merge Version 2 Release 2
- VSE/Ditto
- VSE/IBM SYSTEM/3-3340 DATA IMPORT

Pre-requisites for all licensed programs except DOS/VS Sort/Merge V2R2, are DOS/VSE and VSE/Advanced Functions. The reader should refer to a later section in the Guide for support information with regard to back level DOS/VS releases for the Sort/Merge product.

DOS/VS RPG II RELEASE 3

DOS/VS RPG II is a programming language that can be used to create programs to perform a wide variety of commercial data processing jobs. Release 3 of DOS/VS RPG II is an enhancement of the current DOS/VS RPG II Release 2 compiler, mainly for interactive usage.

The following line items make up the enhancements incorporated in the Release 3 product :

- Fixed Block Mode Support
- Adaptation of the Compiler for VSE/ICCF
- Adaptation of the compiler for CMS in VM/370
- RPG II Source Entry Facility (RSEF) for CMS and ICCF
- DMS/CMS Call Interface Support (CMS)
- IFS/CMS Call Interface Support (CMS)
- Native Language Text Capability

- Error Cross Reference List

DMS/CICS/VS

Development Management System/CICS/VS (for DOS/VS) replaces DMS/VS (for DOS/VS). This program product simplifies application development using 3270 IDS. It aids in the design and generation of online application programs by providing a simple programming interface to the user.

The simplified application development and increased programmer productivity can be achieved through the facilities already known and the following support and capabilities.

The following line items highlight the facilities provided by Release 1 of this product:

- All the features of DMS/VS are preserved including large screen support. Compatibility is preserved.
- DMS/CICS/VS is divided into two components charged separately, a base product and an application generation feature.
- Customer Engineering Support.

The following line items highlight the facilities provided by Release 2 of this product:

- Calculations and edit capabilities to perform simple arithmetics (add, subtract, multiply and divide), and validate fields.
- DL/I enhancements to display repeated under repeated segments, and variable length segments. This is available for DL/I-DOS/VS not for DL/I-Entry.
- Add and Update files and data bases across multiple panels.

The following line items highlight the facilities provided by Release 3 of this product:

- File and data base support through CICS/VS ISC (Inter System Communication facility).
- RPG-II command level support of FDF (Formatted Data Facility).
- DMS/DBD generation from input supplied by the Data Dictionary program product.
- Interactive Application Generation facility to allow for Data Definition, Panel Definition and Application Definition in an interactive, online environment.

DOS/VS SORT/MERGE VERSION 2 RELEASE 2

This program provides a disk sort which runs under control of DOS/VS Releases 33 and 34, as well as DOS/VSE.

The following line items make up the enhancements provided by this release as opposed to Release 1 :

- Execution in both /370 mode and E-mode
- FBM device support for input, output, work and checkpoint data sets
- Variable length spanned records for SAM input and output files
- Ability to specify installation unique default values for the program options EQUALS, ERASE, DIAG, DUMP, VERIFY, PRINT=, ROUTE=, STORAGE=, SORTIN=, SORTOUT= and SORTWK=
- SORT or MERGE control fields may overlap each other
- OUTREC fields can be located beyond byte 4092 of a record
- When the program is invoked, the control statement images are optionally printed if desired
- When the program is invoked, messages may be routed to a printer other than SYSLST
- A new statement, ANALYZE, makes it possible to receive information on how the program will optimize, and what capacity a sort job will have, without actually sorting or merging
- More informative messages and improved diagnostic handling
- Automatic sequence checking in sort applications
- Use of command chaining when reading SAM input or writing SAM output to CKD or tape devices, where sufficient storage is available. This may enhance performance, especially where small input/output block sizes are being used

VSE/DITTO

The licensed program VSE/DITTO is a combination and extension of the FDP's 'DOS/DITTO' (5798-ARN) and 'Extensions to DOS/DITTO' (5798-CAF). It is a general purpose utility program for card, magnetic tape and DASD, providing a variety of facilities such as listing, copying, altering and creating files or portions of files.

The following line items make up the facilities provided by

this utility program :

- File to file functions between combinations of SAM, ISAM and VSAM, including the ability to create SAM, ISAM or VSAM files from tapes, cards or other SAM, ISAM or VSAM files. Blocking factors and file locations may also be changed
- Device to console functions, allowing displaying and verification of small amounts of tape or DASD data on the console
- Card functions to list, copy and interpret punched output and pre-punched cards
- Tape functions to scan, alter, list and copy tape files or portions of tape files
- Disk functions to scan, alter, list and display DASD data on a physical address basis, for both CKD and FBM devices. On FBM devices, data may be viewed at the block level or the control interval level
- A documentation function provides a program generated document of the VSE/DITTO functions and associated parameter requirements

VSE/IBM SYSTEM/3-3340 DATA IMPORT

The VSE/IBM SYSTEM/3-3340 DATA IMPORT program is a utility program that assists the user in converting System/3 3348 data module files into DOS/VSE SAM or VSE/VSAM files. This utility runs as a problem program requiring DOS/VSE and VSE/Advanced Functions as pre-requisites, and requires the System/3 - 3340 Compatibility feature on the 4331 processor to execute.

The following line items make up the facilities provided by this utility:

- Convert System/3 files on 3348 to SAM and VSE/VSAM Files
- Multi-Volume and Multi-Version files may be converted
- Multiple File Conversion during single execution of VSE/IBM SYSTEM/3-3340 DATA IMPORT

3.3.11 EMULATORS

The only new emulator to be announced along with DOS/VSE, is the IBM Systems 1401/1440/1460 Emulator Program Release 1. This program, hereafter referred to as the 1400 Emulator, allows the execution of 1400 programs on 4331 processors

equipped with the 1400 Compatibility Feature, and on 4341 processors with the 1400 Simulator generated as part of the 1400 emulator. The simulation of disk I/O on FBM devices is also provided. Note that the 1400 Emulator runs as a problem program under control of DOS Release 26, DOS/VS Release 34 or DOS/VSE with VSE/Advanced Functions.

The following line items make up the facilities provided by the 1400 Emulator licensed program:

- Emulation of 14XX with Core Storage up to 16KB
- Emulation of all Basic Features plus some Optional Features
- Availability of two Tape Formatting Programs with the Emulator

3.3.12 DB/DC PROGRAM PRODUCTS

The major DB/DC Products in the DOS/VSE range are CICS DOS/VS Release 1.4 and DL/I DCS/VS Release 1.5. Both of these licensed programs are covered in detail in later chapters in this Guide.

CICS DOS/VS RELEASE 1.4

This new release of CICS/DOS/VS Version 1 includes major new enhancements in the areas of intersystem communication, usability, and device and data base support. Additionally new facilities provide users with the potential to improve the application program development productivity.

Highlights

CICS/DOS/VS 1.4 provides a wide range of potential benefits to users including:

- Intersystems communication

Provides the capability for communication between CICS/VS systems connected by ACF/VTAM. Transactions may access files, DL/I data bases or initiate transactions or queue messages on a connected CICS system.

- Entry Level System in CICS/DOS/VS

An easy to use system with following features: Improved performance and reduced working set storage requirements, simplified installation (no SYSGEN), support of HLPI for Assembler, PL/I and COBOL, support for DL/I and DL/I Entry and upward compatibility with the CICS/DOS/VS system.

- HLPI ASSEMBLER

This removes the need for Assembler application programs to access the CICS/VS control blocks.

- EDF- Execution Diagnostic Facility to assist in the testing of application programs written in HLPI.

- Transaction Restart Facility

Permits abended transactions to be optionally restarted, without terminal user intervention. This feature is particularly useful in conjunction with Program Isolation.

- Support for the 3270 Information Display System enhancements:

- optional large screen sizes
- additional program function keys
- program attention key on SCS printer
- Print authorisation matrix
- magnetic stripe reader

Enhancements are available via BMS or native terminal control.

- Enhancements of CICS/VS SNA support for the 3774/ 3775

Supports programmable communications with user written function programs in 3774/ 3775.

- Object and Source code compatibility with CICS/VS 1.3 application programs.

NOTE

Functions specific to CICS/OS/VS are not listed here.

DL/I DOS/VS RELEASE 1.5

DL/I DOS/VS 1.5 is a new release of the proven data base management system called Data Language/1. It will execute on IBM 4300 processors or SYSTEM/370's operating under DOS/VSE. DL/I 1.5 will also support the 3310 and the 3370 FBM DASD.

The following line items make up the highlights provided by this new release of DL/I:

- Support of IBM 4300 processors. DL/I DOS/VS 1.5 will execute on IBM 4300 processors operating in ECPS:VSE or /370 mode, operating under DOS/VSE
- Support of FBM Devices. Data bases and utility work files may reside on the IBM 3310 or IBM 3370 devices in FBM mode. This support was made available in the DL/I DOS/VS 1.4 ICR.
- All Partition MPS Support. This permits the operation of DL/I MPS tasks to operate one at a time from a VSE/ICCF pseudo-partition
- Field Level Sensitivity. The user programs may be sensitive only to specified portions or a segment, rather than the entire segment
- Extended Logical Relationships facility. This support permits the definition of multiple logical relationships - as many as to satisfy the user's needs. This removes the current restriction of only one logical relationship per hierarchical path
- Unique segment support. This permits the specification of only one occurrence of a segment type for any particular parent.
- Extended SHOWCAT macro support. This function is used to find the high allocated RBA of a VSAM data set. The function will be provided by VSE/VSAM as opposed to the hard coded logic in previous DL/I releases.

3.3.13 OTHER LICENSED AND APPLICATION PROGRAMS

The major new licensed program that we will mention here, is the VSE/IPCS (Interactive Problem Control System) product. This product is not covered in detail in this Guide. It executes either in a DOS/VSE batch partition or in a VSE/ICCF pseudo-partition, where its execution may give the impression of running 'interactively'. To operate VSE/IPCS, commands are entered from either the system console, the system input device or a VSE/ICCF controlled terminal. It provides assistance in problem determination through a uniform mechanism for reporting and diagnosing software

failures. DOS/VSE contains facilities that collect problem data resulting from errors detected within the supervisor, subsystems or user programs. This problem data is recorded on a DASD file.

The following line items make up the facilities provided by VSE/IPCS :

- Problem Dump Data Collection and Management
- Creation of a problem description report by extracting specific keyword problem symptom information from the dump. This is especially intended to support the customer in evaluating whether he has a user problem or IBM problem, in the light of the new service terms and conditions announced with DOS/VSE and related software. It should also lead to much faster resolution of software problems, by producing clear, sufficient and concise problem related information.
- Problem Dump Data Display
- Special Formatting of DOS/VSE and VSE/POWER control blocks
- Assistance with APAR reporting
- Bibliography : SC34-2015 User's Guide and Reference
GC34-2016 Program Summary
GC34-2017 General Information
LY25-0004 Logic

3.4 DOS/VSE SUPPORTED LICENSED PROGRAMS SUMMARY

The following list of programs are some of those licensed programs that are officially supported by DOS/VSE. There are other licensed programs that will also run, but these will have to be reviewed individually. Finally, this list also includes some FDP's, IUP's etc., but again this list is far from exhaustive. Again, please refer to your own FSC locations or System Marketing Centres for support statements on products NOT mentioned here.

PROGRAM #	PROGRAM NAME	FBM	AVAIL.
MISCELLANEOUS PROGRAMS			
5746-AM2	VSE/VSAM	yes	6/79
5746-XE3	VSE/POWER with RJE Feature	yes	6/79
5746-SU1	IBM Systems 1401/1440/1460 Emulator	yes	6/79
5747-CC3	1401/1440/1460 & 1410/7010		6/79
	DOS/VS IR Emulators for S/370		
5746-AM3	VSE/IBM SYSTEM/3-3340 DATA IMPORT	yes	6/79
5735-CV1	RPG II S/3 Conv. Preprocessor		6/79
5746-TS1	VSE/ICCF	yes	6/79
5747-CC1	IBM 3800 Printing Subsystem IR	yes	6/79
5746-SA1	VSE/Interactive Problem	yes	6/79
	Control System (VSE/IPCS)		
5746-UT3	VSE/Data Interfile Transfer, Testing and Operations Utility (VSE/DITTO)	yes	6/79
5746-RC9	DOS/VSE Remote Entry Workstation	yes	6/79
5746-AM5	VSE/3270 Bysync Pass Through	yes	3/80
5748-MS1	Interactive Productivity Facility	yes	6/79
5746-XE6	IBM System/370 and IBM 4300 Processor	yes	5/80
	Job Entry Program		
5748-XE6	File Transfer Program	yes	5/80
LANGUAGES/SORTS/UTILITIES			
5746-RG1	DOS/VS RPG II	yes	6/79
5746-CB1	DOS/VS Cobol and Library	yes	6/79
5746-LM4	DOS/VS COBOL Library	YES	6/79
5736-PL3	DOS PL/I Optimizer and Library	yes	6/79
5736-PL1	DOS PL/I Optimizing Compiler	yes	6/79
5736-LM4	DOS PL/I Resident Library	yes	6/79
5736-LM5	DOS PL/I Transient Library	yes	6/79
5746-LM3	DOS Fortran IV Library Option 1	12/79	6/79
5746-SM2	DOS/VS Sort/Merge V2	yes	6/79
5748-XX1	VS Basic	yes	6/79
5748-AP1	VS APL	yes	6/79
DB/DC PROGRAMS			
5746-XX1	DL/I DOS/VS	yes	6/79
5746-XXC	DB/DC Data Dictionary	yes	6/79
5746-XX7	DL/I Entry	yes	6/79
5746-XX3	CICS DOS/VS	yes	6/79

PROGRAM #	PROGRAM NAME	FBM	AVAIL.
TELE-COMMUNICATION PROGRAMS			
5735-XX6	Network Communication Control Facility	yes	7/80
5735-XX3	SSP for ACF/NCP/VS	yes	6/79
5735-XX2	NOSP	yes	6/79
5735-XX7	Network Terminal Option	yes	9/79
5735-XX8	Network Problem Determination Application	yes	7/80
5736-RC3	VIDEO/370		6/79
5746-RC3	ACF/VTAM Release 2	yes	6/79
5746-RC7	ACF/VTAME	yes	8/79
5735-XX1	ACF/NCP/VS	yes	6/79
5747-AJ2	NCP 5		6/79
5747-AG1	EP 3		6/79
5746-RC5	BTAM-ES	yes	6/79
5747-CC6	Subsystem Support Services	yes	6/79
5747-BJ2	Subsystem Program Preparation Support (SPPS I)	yes	6/79
5747-BJ3	3650 RSS Host Support	yes	6/79
5747-BK1	3660 Supermarket Host Support, BSC	yes	6/79
5747-BK2	3660 Supermarket Host Support, SDLC	yes	6/79
5735-D16	3650 Subsystem Program Preparation Support (SPPS II)	yes	6/79
5747-CJ2	Programmable Host Support	yes	6/79
5746-BR1	3600 Host Support		6/79
5746-CJ1	3630 Host Services		6/79
5747-BQ1	3790 Host Support	yes	6/79
5747-BW1	Batch Transfer Program		6/79

PROGRAM #	PROGRAM NAME	FBM	AVAIL.
APPLICATION PROGRAMS			
5735-E91	EPIC Socrates	yes	6/79
5735-E93	EPIC Budget/Finance	yes	6/79
5736-N13	CFO II		6/79
5736-N14	ALPHA Search II	yes	6/79
5746-AM1	Interactive Personnel System	yes	6/79
5746-F11	PC/3600-DOS/VS		6/79
5746-F12	DOSCHECK		6/79
5746-N11	Life Inquiry/Data Entry	yes	6/79
5746-XC2	DMS DOS/VS V1	yes	6/79
5746-XM2	MPSX/370 DOS/VS	yes	6/79
5746-XP1	PROJACS		6/79
5746-XX2	STEPS		6/79
5746-XR4	STAIRS DOS/VS	yes	11/79
5746-XXA	PLANCODE/S DOS/VS	yes	6/79
5746-XXG	ATMS II DOS/VS		6/79
5746-XXM	CORMES	yes	6/79
5746-XXP	Online Routing	yes	6/79
5746-XXQ	Data Base Design Aids V2 DOS/VS		6/79
5746-XX9	PLANCODE/I DOS/VS	yes	6/79
5746-XXS	Customer Information Facility/VS (CICS DOS/VS)		6/79
5748-H11	HCS-Electrocardiogram Analysis II		6/79
5748-XC2	DMS/3790	yes	6/79
5748-XC3	DMS/3770	yes	6/79
5748-XX3	Chained File To DL/I Bridge	yes	6/79
5748-XX6	Interactive Instructional System V2	yes	6/79
5748-XXE	Document Library Facility	yes	6/79
5748-XX9	Document Composition Facility	yes	6/79
5795-AAG	FERST/VS - Message Switching		6/79
5795-AAJ	FERST/VS - Equipment Control		6/79
5795-AAL	FERST/VS - Freight Billing		6/79
5795-AAN	FERST/VS - Rating/Rate Audit		6/79
5746-M41	CAPOSS-E	3/80	6/79
5746-F52	IFS 1		10/79
5746-F53	IFS 2		10/79

NOTES

1. 6/79 is defined as the date of general availability of DOS/VSE
2. Be aware that the programming support for 5747-CC3 has been discontinued from this time.
3. The following programs are pre-requisites for 5746-LM3
 - FORTRAN IV Compiler (360N-F0-479)
 - FORTRAN Library (360-LM-480)

3.5 GLOSSARY OF NAMES/MACHINE NUMBERS IN THIS GUIDE

With the advent of a major hardware announcement and accompanying DOS/VSE announcement, many new names and machine numbers have been introduced. What follows is a glossary of all those new names/numbers that are used in this Guide, along with a brief description to what they refer.

MACHINE #	DESCRIPTION OF HARDWARE	AVAILABLE
3262-1	Printer, attaches natively to 4331/4341	7/80
3289-4	Printer, attaches natively to 4331/4341	6/79
3310	FBM DASD, attaches to 4331	6/79
3370	FBM DASD, attaches to 4331/4341	Note
3880	DASD control unit, attaches to 4341 block MPX	8/79
4331	Processor, .5M or 1M storage options	6/79
4341	Processor, 2M or 4M storage options	2/80
8809	Tape unit, attaches to 4331	6/79

NOTE

For 3370 attachment to DASD Adapter on 4331, FCS is 2Q/80

ABBREVIATION	FULL MEANING of ABBREVIATION
/370 mode	/370 Compatibility Mode
AMS	VSAM Access Method Services Program
BAM	Basic Access Method Component of DOS/VSE
CA	VSAM Control Area
CEC	Central Electronic Complex or CPU
CI	VSAM Control Interval
CKD	Count Key Data; referring to conventional DASD
DA	Direct Access
DAM	Direct Access Method
E-mode	ECPS:VSE Mode
FBM	Fixed Block Mode; specifically 3310,3370 devices
ICA	Communications Adapter on the 4331
ICR	Independent Component Release
IR	Independent Release
IS	Indexed Sequential
ISAM	Indexed Sequential Access Method
SAM	Sequential Access Method
VSAM	Virtual Storage Access Method
VSE	Virtual Storage Extended
VSE/AF R1	VSE/Advanced Functions Release 1
VSE/AF R2	VSE/Advanced Functions Release 2

3.6 4300 PROCESSOR BIBLIOGRAPHY

The purpose of this section is to summarise the available hardware publications that have been produced to support the announcement, installation and operation of the 4300 processors. Note that the software publications for the individual licensed programs addressed in this Guide, are contained in the relevant chapter/section that discusses that program.

FORM #	DESCRIPTION OF PUBLICATION	AVAILABLE
IBM 4331 PROCESSOR RELATED MANUALS		
GC20-xxxx	4300 Processors Bibliography	FCS
GA22-7000	System/370 Principles of Operation	CURRENT
GA33-1523	IBM 4300 Processors Summary and I/O and Data Communications Configurator	ANN
GA33-1524	IBM 4331 Communications Adapter Configurator	ANN+2mths
GA32-0039	IBM Input/Output Device Summary	ANN
GA27-3185	IBM Data Communications Device Summary	ANN
GA24-3667	IBM 4300 Processors Installation Manual-Physical Planning	ANN
GA27-3006	Remote Multiplexors and Communication Terminals Installation Manual -Physical Planning	ANN
GA22-7070	IBM 4300 Principles of Operation For ECPS:VSE mode	FCS
GA33-1525	4331 Operating Procedures and Problem Determination Guide	FCS
GA33-1526	IBM 4331 Functional Characterist. and Processor Complex Configurator	ANN+2mths
GA33-1527	IBM 4331 Channel Characteristics	FCS
GA33-1528	IBM 4331 Compatibility features	ANN
GC22-7064	IBM I/O Equipment Installation Manual-Physical Planning For System/360, System/370, and 4300 Processors	ANN
GX24-3670	4300 Processors Channel Worksheet	FCS
GX24-3725	4331 Processor Complex Installation Manual -	ANN
GX24-3728	Physical Planning Template	
G-57579	4331 Processor Template (adhesive)	ANN
G-57578	4331 Processor Template (non-adhesive)	ANN

FORM #	DESCRIPTION OF PUBLICATION	AVAILABLE
IBM 4341 PROCESSOR RELATED MANUALS		
GA24-3672	4341 Functional Characteristics and Processor Complex Configurator	ANN
GA24-3671	4341 Channel Characteristics	4/79
GX24-3666	4341 Processor Complex Installation Manual-Physical Planning Template (1:48 scale)	ANN
GX24-3729	4341 Processor Complex Installation Manual-Physical Planning Template (1:50 scale)	ANN
GA24-3669	4341 Operating Procedures and Problem Determination Guide	FCS
GA33-1523	4300 Processors Summary and Input/Output and Data Communications Equipment Configurator	ANN
GC22-7064	Input/Output Equipment Installation Manual - Physical Planning for System/360, System/370 and 4300 Processors	ANN
GN22-2159	TNL to above, for 3278-2A	ANN
GN24-3667	4300 Processor Complex Installation Manual-Physical Planning	ANN
GA22-7070	4300 Processors Principles of Operation of ECPS:VSE Mode	FCS 4331
GA32-0039	Input/Output Device Summary	ANN
GA27-3185	Data Communications Device Summary	ANN
GX24-3670	4300 Processors Channel Load Summary Worksheet	FCS

FORM #	DESCRIPTION OF PUBLICATION	AVAILABLE
4300 MARKETING SUPPORT MATERIAL INC. PROPOSAL INSERTS		
ZV11-6070	4300 Processors Product Excellence Foil Set	3/79
GK10-6240	4341 Processor Brochure 'How do I gain with the 4341 Processor'	ANN
GK10-6253	4300 Processor Overview Brochure	ANN
GK10-6243	4341 Processor	ANN
GK10-6246	3370 DASD	ANN
GK10-6247	3880 Storage Control Facility	ANN

ZZ10-9826

DOS/VSE PLANNING GUIDE
Volume II

Chapter 4

CHAPTERS 4 THRU 7

(Refer to Volume I for Chapters 4 to 7 inclusive)

CHAPTER 8

DL/I DOS/VS VERSION 1.5

WTSC BOE

IBM Internal Use

CONTENTS

CHAPTER 8 DL/I DOS/VS Version 1.5.	8.1
8.1 DEFINITION OF THE PRODUCT.	8.2
8.1.1 Product Information Table.	8.2
8.1.2 Product Function	8.3
8.1.3 Hardware Supported	8.8
8.1.4 Software Supported	8.9
8.1.5 System Reference Library	8.10
8.1.6 Announcement Details	8.11
8.2 ENHANCEMENTS PROVIDED BY DL/I VERSION 1.5.	8.12
8.2.1 Field Level Sensitivity.	8.12
Basic Support.	8.14
Automatic Definition Sequencing.	8.14
Automatic Segment Size Calculation	8.14
No Sequence Field Restriction.	8.15
SENFELD Macro	8.15
Subfields.	8.18
Extended Support	8.21
User Field Exit Routine.	8.22
Virtual Fields	8.25
Automatic Data Format Conversion	8.28
Dynamic Segment Expansion.	8.30
8.2.2 Extended Logical Relations	8.31
Logical Databases.	8.31
Ordering	8.33
8.2.3 4300 Systems Support	8.35
8.2.4 FBM Support.	8.36
DBDGEN	8.36
Space Management	8.38
Utilities.	8.38
Invocation	8.38
Error Messages	8.39
8.2.5 Unique Segment Support	8.40
Invocation	8.40
Error Message.	8.40
8.2.6 All Partition MPS Support.	8.41
8.2.7 Extended Showcat	8.42
8.2.8 CMS Support.	8.43
8.2.9 IPCS Dump Hook Support	8.44
8.3 INSTALLATION AND MIGRATION CONSIDERATIONS.	8.45
8.3.1 Field Level Sensitivity.	8.45
8.3.2 Extended Logical Relations	8.46
8.3.3 FBM Support.	8.47
8.3.4 Unique Segment Support	8.48
8.3.5 All Partition MPS Support.	8.49
8.3.6 Extended Showcat	8.50

8.3.7 CMS Support.	8.51
8.3.8 Multiple Partition Support - MPS (DL/I Rel 1.2)	8.52
8.3.9 Program Isolation - PI (DL/I Rel. 1.3)	8.54
8.3.10 Executing CICS/VS with DL/I MPS or PI	8.56
8.3.11 Executing Batch MPS Programs.	8.58
8.3.12 Dynamically Scheduling MPS or Non-MPS Execution	8.59
8.3.13 Job Accounting with Batch MPS Programs. . . .	8.61
8.3.14 Program Isolation Considerations.	8.62
8.3.15 Choosing the DL/I Log Medium.	8.63
DL/I Disk Logging Considerations	8.63
Logging and Performance.	8.65
8.3.16 DL/I Checkpoint (DL/I Rel. 1.3)	8.68
DL/I Checkpoint in Batch MPS Programs.	8.69
DL/I Checkpoint in Online Transactions	8.70
8.3.17 Abnormal Termination, Recovery and Restart. .	8.72
DL/I TERM Call and CICS Sync Point Processing. . .	8.72
Batch DL/I Program ABEND	8.73
Batch MPS program ABEND.	8.73
Online DL/I transaction ABEND.	8.74
DL/I System ABEND (internal error)	8.75
CICS/VS System ABEND	8.76
Power Failure.	8.77
Restarting MPS Batch Programs.	8.77
8.3.18 DOS/VSE Supervisor Generation	8.81
8.3.19 VSE/VSAM Considerations	8.83
VSE/VSAM Share Option Selection.	8.83
Conversion Considerations.	8.84
8.3.20 CICS/DOS/VS Generation.	8.85
CICS/DOS/VS System Generation.	8.85
CICS/VS-DL/I System Generation Example	8.87
CICS/DOS/VS Table Generation	8.88
DL/I DOS/VS Application Control Table.	8.94
8.4 PROGRAMMING AND OPERATING CONSIDERATIONS	8.100
8.4.1 Field Level Sensitivity.	8.100
8.4.2 Extended Logical Relations	8.101
8.4.3 Unique Segment Support	8.101
8.4.4 CMS Support.	8.101
8.4.5 New Status Code Summary.	8.102
8.5 Productivity Aids For Use With DL/I DOS/VS	8.103
8.6 Product Reference Material	8.104

P R E F A C E

This Installation Guide is intended for System Engineers who are proposing and installing DL/I DOS/VS. This guide covers functions specifically introduced with Version 1.5 of DL/I DOS/VS. However, some of the more important points of features made available in Releases of DL/I DOS/VS will be included (i.e. Program Isolation (PI), Multiple Partition Support (MPS), Checkpoint and Intersystems Communication (ISC) as appropriate in the discussion).

Note that if not otherwise explicitly stated within the text, DOS and corresponding DOS/VSE components are the same.

Section 1 gives a brief overview of the product, (i.e. Functions and Highlights, Hardware and Software supported, Summary of Version 1.5 Announcement Details etc.).

Section 2 discusses each new Version 1.5 enhancements in detail.

Section 3 gives guidance on installation and migration activities.

Section 4 is intended to draw attention to some aspects for consideration in the area of programming and operating of the new version.

Further, this Chapter contains contributions from a number of other System Engineers. Consequently, WTSC Boeblingen wishes to thank the following people, whose input into various sections of the Guide, was greatly appreciated:

DL/I Development Group, Endicott
Juhani Ellae, Finland
Keith Jones, UK
Etienne Temmerman, Belgium

8.1 DEFINITION OF THE PRODUCT

8.1.1 PRODUCT INFORMATION TABLE

VERSION	1.4	1.5
PRODUCT IDENTITY	DL/I DOS/VS	DL/I DOS/VS
PRODUCT NUMBER	5746-XX1	5746-XX1
ANNOUNCEMENT DATE	JULY 31, 1978	JAN 30, 1979
PID AVAILABILITY DATE	NOV 24, 1978	JULY 1979
SPL STATUS	DESIGNATED	DESIGNATED
PRE INSTALLATION TEST	60 DAYS	60 DAYS
SERVICE CLASSIFICATION	A	A
SERVICE CURRENCY	DEC 31, 1979	
ANNOUNCEMENT LETTER NR.	ZP78-0748	ZP79-0619

8.1.2 PRODUCT FUNCTION

DL/I DOS/VS is a data base management system developed to improve the user's ability to implement data base processing applications. It provides data organisation methods that help to create, interrelate, and maintain large common data bases similar to those on large systems. DL/I DOS/VS executes as an application program in a virtual storage environment under DOS/VS and DOS/VSE.

DL/I DOS/VS has been designed to permit the concurrent scheduling of multiple programs requesting DL/I DOS/VS services, thereby allowing access by more than one user to the same or different data bases at the same time. Application programs may utilise this design concept in conjunction with CICS/DOS/VS to access DL/I DOS/VS data bases in an online or batch environment.

Application programs call upon the data base management services through DL/I DOS/VS. In turn DL/I DOS/VS provides the application program with a level of interface to the data base management services which is independent of storage media programming considerations. Through installation management tools provided by DL/I DOS/VS, storage media characteristics may be changed without affecting application programs. Installation management can optimise productivity for the total community of applications and users without affecting the functional capabilities of any of the members of that community. Existing DL/I DOS/VS application programs can be insensitive to the reorganisation of stored data, the addition of new applications or data, changes in access methods, organisation, or access strategy, and the introduction of new storage devices.

Using DL/I DOS/VS utilities supplied with the system, the user describes the structure of the data base from two viewpoints: the stored data structure as seen by the system, and the logical data structure as seen by the application. Only one description of the stored data exists. However, multiple logical descriptions may exist. These data base descriptions are external to application programs. They exist as physical tables and are referred to by the system when it is processing an access request for application programs.

Data base descriptions define symbolic names for data items (fields), segments, and data bases. Within a single data base, the system description defines a hierarchical relationship among segments of the stored data structure. It also contains information about organisation, access strategy, the physical attributes of the data (length, format, etc), the physical structure of the stored data segments, and storage device characteristics.

To access or maintain stored data, the application program issues a CALL for data base management services. As part of the functional request, the application program supplies symbolic names which identify the data (segment type) to be processed. Through data base descriptions, the system relates the application-supplied logical data names to corresponding stored data names, and performs the requested function against the stored data.

Data may be stored in one of 6 different hierarchical organisations (SHSAM, SHISAM, HSAM, HISAM, HDAM, HIDAM). The use of HD (HDAM or HIDAM) is normally recommended where performance is critical. For HS (Hierarchical Sequential) organised data, the access method and basic processing strategy may be sequential or indexed sequential. For HD (Hierarchical Direct) organised data, it may be direct or indexed direct.

Reorganisation, recovery and backup of stored data bases is supported by a set of DL/I DOS/VS utilities supplied with the system.

The data management services are provided by VSAM for data bases of the HD organisation and indexed sequential data bases of the HS organisation.

Highlights

Application programs may process data bases independent of its physical organisation or storage device used.

Most DL/I DOS/VS main line modules are re-entrant and may reside in the DOS/VS Shared Virtual Area (SVA).

Data base buffer management capabilities retain frequently requested data records in main storage, resulting in fewer I/O accesses for these records.

Users may write application programs in COBOL/VS, PL/I, RPG-II, or ASSEMBLER language.

Data base recovery is facilitated through the use of data base activity logging.

Application programs using CICS/DOS/VS may issue DL/I DOS/VS calls to process DL/I DOS/VS data bases.

Data base integrity and security is provided through the concept of sensitivity: a description of a program's intended access to a data base.

Scheduling of multiple CICS-DL/I application programs for

concurrent execution is based on their sensitivity.

More than one CICS-DL/I application program can be served by DL/I DOS/VS at the same time.

HD-organised data bases may be defined such that logical relationships are established between segments of different hierarchical structures stored within the same or different data bases.

Secondary indexing provides indexed access to root or dependent segments within an HD-organised data base. Any field or field combination within most segments of a data base may be indexed.

Segments of HD-organised data bases may be defined as variable in length to accomodate variable length information.

Segment and field exits are provided for HD-organised data bases which allow e.g. user written encode/decode routines to improve utilisation of direct access storage space or to strengthen security of stored confidential data.

Path calls, such as path retrieval calls, which may simplify data base calls are available for all data base organisations.

Generation and support of low-level codes with continuity checking are functions provided in application support programs for the manufacturing industry oriented user. These are techniques that permit the user to specify hierarchical levels within a data structure and to ensure that a structure is not dependent upon itself. Although primarily provided for manufacturing industry applications, they can also be utilised in other environments.

In an on-line environment, the Shared Log option eliminates the need for separate tape drives to be assigned to CICS/DOS/VS System Log and DL/I DOS/VS Log, by optionally directing all DL/I DOS/VS log information to the CICS/DOS/VS tape/disk journal.

Through optional parameters, the tuning facility allows DL/I DOS/VS to be adapted to specific environments and/or application characteristics. The tuning facility provides flexible control mechanisms for buffer and main storage allocation.

Multiple Partition Support (MPS) enables multiple application programs executing in different partitions to access the same data base at the same time. For example, one or more online applications can retrieve information from a

data base while a batch program updates the same data base. MPS, following DL/I conventions, uses DL/I resources and the multitasking facilities of DL/I and CICS/VS. (CICS/DOS/VS, therefore, is a prerequisite for running MPS).

Reload Restart enables the user to restart the reloading of an HD-organised data base that failed before completion. The data base can be reloaded without rerunning the entire job by resubmitting the job with an additional JCL statement identifying the job as a restart of the Reload Utility. The system operator needs only to supply a checkpoint identification number obtained from a console message as the restart point. A checkpoint record is located approximately every 5000 segments.

Users may reserve Free Space in the HD organised data bases at load or reload time so as to reduce instances of extensive processing during space management search, and also to reduce access time for retrieving related segments.

Module identifiers and a system log print utility is provided to enhance serviceability.

Support is provided for multiple logs on a single tape, as well as a multiple volume log file.

Unload sequence checking is provided.

Disk logging is an option to allow the disk only user to take advantage of data integrity features of DL/I DOS/VS. These features are primarily Backout, Recovery and Change Accumulation.

The trace routine includes a short form.

DL/I application programs can access a data base that is resident in another CPU, using the CICS/DOS/VS Intersystem Communication (ISC) Support.

A PL/I user may use the PL/I diagnostic information to 'debug' an error condition, before the error is returned to DL/I DOS/VS.

A processing option (PROCOPT=0) supported on the PCB statement provides a performance improvement by inhibiting any locking (enqueueing) by the Program Isolation function during retrieve of the same segment type in a data base.

An enhancement notifies an MPS batch task if another task is waiting for a resource that is owned by the MPS batch task. This information can be used by the application programmer to determine when to take checkpoints to free resources.

Under CMS, batch DL/I application programs can be coded, compiled and tested to load, retrieve and update data bases. Additionally, DL/I control blocks may be generated.

Field Level Sensitivity gives the user the ability to have different views of a physical segment on an application basis.

8.1.4 SOFTWARE SUPPORTED

DL/I DOS/VS Version 1.5 will run on DOS/VSE and current DOS/VS release. VSE/Advanced Functions Rel 1 is required.

CICS/DOS/VS Rel. 1.4 is required for execution of DL/I DOS/VS 1.5

RPG-II Rel. 2.0 or subsequent release is required if DL/I DOS/VS application programs are to be written in RPG-II.

RPG-II Rel. 3.0 is required if an FBM device is used.

NOTE: Do not unload your data base under DL/I DOS/VS Version 1.3 and reload under Version 1.4 or 1.5. Changes made to DL/I DOS/VS Version 1.4 now incorporated in Version 1.5 will not allow Prefix Resolution to function properly if any data base contains logical relationships or secondary indices. If you are currently still on Version 1.3 then install Version 1.5, unload your data bases under this version and destroy saves made under version 1.3 or previous after reload test.

Dependency Information Chart

DEPENDENCY ITEM	REMARKS
CICS/DOS/VS	Required for online programming, Multiple Partition Support and for Intersystems Communication.
SORT/MERGE	Required for DL/I Change Accumulation Utility and for Prefix Resolution Utility
ASSEMBLER	Required by DL/I DOS/VS
COBOL/VS	Customer dependent
PL/I	Customer dependent
RPG-II	Customer dependent (Rel. 2.0 or subsequent)
DMS/VS DOS/VS	Customer dependent
DMS/CICS/VS DOS/VS	DOS/VS and DOS/VSE Customer dependent
GIS DOS/VS	Cannot be used (unmodified) with MPS.
VSE/VSAM	Mandatory for FBA and DL/I DOS/VS
VSAM	Mandatory for DL/I DOS/VS (without FBA)

8.1.5 SYSTEM REFERENCE LIBRARY

G510-0740 Sales Flyer

ZZ19-6250 DB/DC DOS Marketing Guide

GE19-5214 DL/I DOS/VS Foil Presentation

GE19-5284 Adding DL/I to CICS Presentation

GH20-1246 General Information Manual

SH24-5001 Guide For New Users

SH12-5413 System and Application Design Guide

SH12-5411 Application Programming Ref. Manual

SX24-5103 Application Programming Ref. Card

SH12-5412 Utilities and Guide for System Progr.

SX24-5104 System Programming Reference Card

SH12-5414 Messages and Codes

SH24-5002 Diagnostic Guide

LY12-5016 Program Logic Manual

LYB0-0839 Program Logic Manual (microfiche)

8.1.6 ANNOUNCEMENT DETAILS

Field Level Sensitivity

The user may see only portions of a segment, rather than the entire segment.

Extended Logical Relations

Today's restriction of only one logical relationship per hierarchical path is removed. The user can now define as many logical relationships as he needs to satisfy his requirements.

4300 Systems Support

DL/I DOS/VS 1.5 will run on 4300 Systems, either in /370 mode or in ECPS:VSE-mode.

FBM Support

FBM support will allow data bases and utility work files to reside on Fixed Block Mode devices (like 3310, 3370). This item is an inclusion of the DL/I DOS/VS 1.4 ICR FBM support.

Unique Segment Support

The ability to specify that only one occurrence of a segment type is allowed for that parent.

All Partition MPS Support

All Partition MPS Support allows one at a time operation of DL/I MPS tasks from an ICCF pseudo-partition.

Extended Showcat

The extended SHOWCAT function is used to find the high allocated RBA of a VSAM dataset. This function was hard coded by DL/I and will now be included and shipped in VSE/VSAM. However, DL/I DOS/VS Version 1.5 will function on DOS/VS Rel. 34 too. (Refer to detailed discussion of this modification).

CMS Support

Batch DL/I programs, written in ASSEMBLER, COBOL or PL/I can be created, executed and debugged using CMS and the interactive facilities of VM/370.

IPCS Dump Hook Support

IPCS Dump Hook Support provides a mechanism for obtaining standardised IPCS-format dumps instead of current DL/I dumps on a DL/I initiated abnormal termination of an application program.

8.2 ENHANCEMENTS PROVIDED BY DL/I VERSION 1.5

8.2.1 FIELD LEVEL SENSITIVITY

This feature gives the user the ability to have different views of a (physical or logical) segment on an application basis. Field sensitivity enhances the DL/I DOS/VS application program independence from DL/I physical data structures and allows a finer degree of data security for data stored in a DL/I DOS/VS data base.

In DL/I DOS/VS versions earlier than 1.5, the user had to execute the following steps in order to add fields to an existing segment type:

- Unload the affected data base. This was required because the physical structure of the data base is being changed - a segment size is being increased in size. If the existing data had to be preserved when the data base with the new segment size is created, then the existing data base must first have been unloaded.
- Modify the DBD. This was required to indicate the new segment size. With the exception of variable length segments, segment sizes were maintained in the DBD/DMB, not in the data base itself.
- Rebuild the existing PSBs and DMBs affected by the change. This was required because information from the DBD must be physically incorporated into the DMB used by DL/I at execution time so DL/I can properly access the data base.
- Modify all application programs sensitive to the segment containing the new field. This is required because the segment is increasing in size and the I/O area used to return the segment on a DL/I call must be large enough to accommodate the segment. This step can often be avoided if the application programs are written using an I/O area larger than the segment. If the new field is not added at the end of the segment however, the application programs' field mapping of the segment must be modified to reflect the presence of the new field. This step is required even if the program has no requirements to access the new field.

- Reload the data base. This was required because the data base must be created with segments the same length as reflected in the DBD/DMB.
- If logical relationships or secondary indexing is present in the data base, then the logical relationships must be resolved and new secondary indexes created.

These steps are time consuming and create unnecessary application program maintenance. Typically the physical unload and reload of the data base will take several hours.

With the availability of DL/I DOS/VS Version 1.5, the user is able to declare the fields in a segment to which a particular application program needs access. Therefore, if new fields are added to a segment, existing application programs will only have to be modified if they require access to the new fields. Typically, few existing application programs will need access to the new fields and thus program maintenance for non-application reasons will be reduced over that required before Version 1.5.

In the DL/I DOS/VS versions prior to 1.5, an application program is sensitive to a segment type. It can access all fields within that segment type. Security, privacy and legal requirements are going to force users to control access to field level type information. With the current product, the user can only accomplish this by storing those fields in separate segment types. At best this technique is awkward and often is infeasible because:

- DL/I is limited to 255 segment types and often data base records contain more than 255 fields.
- Application programs normally require access to related fields simultaneously. Storing each field in a separate segment would require the application program to issue a separate call for each field. This would significantly increase programming effort and degrade the performance of the application program.
- Logical relationships and secondary index usage may force the user to store several fields in the same segment type despite security or privacy needs.

Now with Version 1.5 of DL/I DOS/VS the user can declare the access privileges of a program to fields in a segment and the security, privacy and legal requirements can more easily be met.

BASIC SUPPORT

The basic support of Field Level Sensitivity contains the following features:

1. Automatic Definition Sequencing
2. Automatic Segment Size Calculation
3. No Sequence Field Order Restriction
4. SENFLD Macro
5. Subfields

AUTOMATIC DEFINITION SEQUENCING

Automatic Definition Sequencing is invoked by omitting the START parameter of the FIELD macro. With this feature, the user can avoid having to specify starting positions for fields that have to start at the end of the previously defined field. This feature is added to the physical field definition facility (FIELD macro) and will be present in the PCB view field definition facility (SENFLD and VIRFLD macro). For an example refer to the next page.

AUTOMATIC SEGMENT SIZE CALCULATION

Automatic Segment Size Calculation is invoked by omitting the BYTES Parameter of the SEGM macro. This feature allows the user to have DL/I DOS/VS calculate the size of a segment based on the locations and lengths of the fields identified as belonging to it. This feature is added to the physical segment definition facility (SEGM macro). However, be aware that DL/I DOS/VS will not be able to verify if the field(s)

match the intended segment size.

EXAMPLE:

```

SEGMENT NAME=EXAMPLE,.....
FIELD NAME=ABC,BYTES=6,TYPE=C
FIELD NAME=BCD,BYTES=7,TYPE=Z

```

generates:

```

| ABC | BCD |
|-----|

```

|
Start of Segment 'EXAMPLE'. Segment length is
calculated to be 13 Bytes.

NO SEQUENCE FIELD RESTRICTION

Sequence fields need no longer be the first field statement, allowing the user to define his fields in the order in which they appear in his segment.

SENFLD MACRO

The SENFLD Macro allows a user to specify those fields from the physical definition to be included in his view of the segment, including their location in the new view. DL/I DOS/VS will automatically map the fields from the physical segment into the applications view.

The user identifies the fields to which he wishes to be sensitive at PSB generation time via a new SENFLD statement following the SENSEG statement for the segment. (If no SENFLD statements are specified, sensitivity to the entire segment is assumed). Up to 255 fields within a segment and 4095 fields within a PSB may be specified.

The SENFLD Macro Format is an external macro used to specify sensitive fields within a sensitive segment. SENFLD statements are optional.

```

SENFLD NAME=fldname[ ,BYTES=n|P ][ ,START=Position ][ ,TYPE=t ]
                [ ,{OPT  }=REPL  ]
                {OPTION} NOREPL ]

```

NAME = fldname 'fldname' corresponds to the name of the related field in the physical definition.

BYTES = n|P 'n' specifies the length of this field in terms of bytes. This must be a numeric value in the range 1 through 256.

'P' must be specified if the length of the field in the application program's view of the field is to be the same as in the physical view. (Length derived from the DBD).

Note: Do not specify the TYPE parameter when you specify BYTES=P.

The 'BYTES' parameter must not be specified for field types, E, D, or L, and must be specified for types X, P, C, or Z. (If omitted for types H or F, values of 2 and 4 respectively will be assumed).

START = position 'position' specifies the starting position of this field in terms of bytes relative to the beginning of the user's view of the segment within which this field is defined. It must be numeric and in the range 1 through 32767. If omitted, the field will be appended to the previous field, or, at the beginning of the segment if there is no previous field.

Subfields are permitted. If a subfield starts in the same position as a previously defined field, the user may specify the name of the previously defined field, instead of a numeric value, to indicate the starting position (START=fieldname).

TYPE = t

't' specifies the type of data that is to be contained in the application program's view of this field. If you specify a data type that is different from that defined in the physical DBD for this field, DL/I DOS/VS will (in most cases) convert the data type to that needed by the program. (Refer to 'Automatic Data Conversion' later in this Chapter for additional information). If you do not specify this parameter, DL/I DOS/VS assumes the type to be the same as specified for this field in the physical DBD. If you specified 'BYTES=P', for this field, do not specify the 'TYPE=' parameter. The valid data types are:

- 'X' - hexadecimal data (binary)
- 'H' - half word binary
- 'F' - full word binary
- 'P' - packed decimal
- 'C' - character data
- 'Z' - numeric character data
- 'E' - floating point (short)
- 'D' - floating point (long)
- 'L' - floating point (extended)

If omitted, the type will be assumed to be the same as in the physical view. If 'BYTES=P' was specified, the TYPE parameter must be omitted.

```
[{OPT    }=REPL  ]
 {OPTION} NOREPL]
```

indicates whether the program using this PSB may replace this field. 'REPL' is the default value!

EXAMPLES

```
SENFLD NAME=ABC,BYTES=P
```

This statement defines sensitivity to a field named ABC. The applications view of the field is the same as the physical format. The field is located immediately following the previous field in the applications view. The program is allowed to replace this field.

```
SENFLD NAME=BCD,BYTES=8,TYPE=P,OPT=NOREPL
```

This statement defines sensitivity to a field named BCD. The application view of the field is a 8 byte packed decimal field located immediately following the previous field in the applications view. The program is not allowed to

replace this field.

SUBFIELDS

It might become necessary to divide a field into subfields (i.e. as redefining data structures in application programs).

```
      | MM | DD | YY |  
      |-----|  
DA    <----->    Field for Date  
MM    <---->       Subfield for Month  
DD    <---->       Subfield for Day  
YY    <---->       Subfield for Year
```

Defining Subfields in DL/I DOS/VS is meaningful only if these subfields are going to be used by SSAs. The 'subfields' the program requires are defined in the HLL data description.

You can reorder subfields in your application view if they are not subfields of a sequence field.

Rules on Subfields

1. If you define one subfield in a field, then the field has to be defined entirely by subfields (no holes).
2. If you have two fields that overlap, the overlapping area has to be entirely defined by subfield(s).
3. If a field is a subfield of another field in the physical view and both fields appear in the application view, then it must be a subfield in the application view too.
4. If a field is a subfield of another field in the application view and both fields appear in the physical view it must be a subfield in the physical view.

E X A M P L E :

NOTES

```

-----
DBD      NAME=.....
DATASET DD1=.....
SEGM     NAME=.....

SEGM     NAME=DATE,BYTES=6,PARENT=.....      (1)
FIELD   NAME=MM,BYTES=2,START=1,TYPE=C      (2)
FIELD   NAME=DD,BYTES=2,START=3,TYPE=C      (2)
FIELD   NAME=YY,BYTES=2,START=5,TYPE=C      (2)
FIELD   NAME=DA,BYTES=6,START=1             (3)

SEGM     NAME=ETCH,BYTES=15,PARENT=...       (1)
FIELD   NAME=DEPT,BYTES=7,START=1,TYPE=C    (2)
FIELD   NAME=EMP,BYTES=5,START=8,TYPE=C     (2)
FIELD   NAME=SAL,BYTES=3,START=13,TYPE=C    (2)
FIELD   NAME=FLD1,BYTES=12,START=1
FIELD   NAME=FLD2,BYTES=8,START=EMP

DBDGEN
FINISH
END

PCB      TYPE=DB,.....
SENSESEG NAME=.....

SENSESEG NAME=DATE,.....
SENFLD   NAME=DA,BYTES=P,START=1             (4) (2)
SENFLD   NAME=MM,BYTES=2,START=DA
SENFLD   NAME=DD,BYTES=2,START=3            (2)
SENFLD   NAME=YY,BYTES=2,START=5           (2)

SENSESEG NAME=.....
PSBGEN   .....
END

PCB      TYPE=DB,.....
SENSESEG NAME=.....

SENSESEG NAME=ETCH,.....
SENFLD   NAME=FLD1,BYTES=12,START=1         (2)
SENFLD   NAME=DEPT,BYTES=7,START=1
SENFLD   NAME=FLD2,BYTES=8,START=8         (2)
SENFLD   NAME=EMP,BYTES=5,START=FLD2
SENFLD   NAME=SAL,BYTES=3

SENSESEG NAME=.....
PSBGEN   .....
END

```

- (1) If you omit the BYTES-Parameter, the segment size would be automatically calculated from the information supplied by the Field Statements.
- (2) You may omit the START parameter as DL/I would calculate this value by default.
- (3) If you omit the START parameter in this field statement, DL/I DOS/VS will issue an error message when you have coded the BYTES parameter in the SEGM statement. In case you did omit the BYTES parameter in the SEGM statement, DL/I DOS/VS would not issue an error message but the segment would be 12 bytes long. (i.e. MM,DD,YY,DA).
- (4) BYTES=P means the field length and data type will be the same in the application view as defined in the DBD.

EXTENDED SUPPORT

In addition to the basic support, the following items are also supported under Field Level Sensitivity:

1. User Field Exit Routine: This gives the ability to specify a routine to get control whenever a field is retrieved or stored.
2. Virtual Fields: This gives the ability to identify a field in the application's view of a segment that doesn't exist in the physical view.
3. Automatic Data Format Conversion: This gives the ability to change the format of the physical data without re-compiling the application programs.
4. Dynamic Segment Expansion: This facility gives the ability to add fields to a segment without reloading the data base or re-compiling the application programs.

USER FIELD EXIT ROUTINE

During PSB generation the user may specify a routine to be given control whenever a field is referenced. The routine name must be the name by which the routine is cataloged in the DOS Core Image Library.

Within this exit routine, the user can have his own encryption facility; or he might use this facility to do his own data conversions.

I N T E R F A C E

Upon entry, register 15 will contain a pointer to the entry point, register 14 will contain the return address, register 13 contains a pointer to a standard format register save area, and register 1 points to a parameter list as follows:

NAME	DISP	LENGTH	CONTENTS	DESCRIPTION
FERPEC	0	1		Entry Code
FERPGET			G	Get
FERPPUT			P	Put
FERPFNCT	1	1		Function Code
FERPRET			G	Retrieve
FERPINS			I	Insert
FERPREP			R	Replace
FERPCSC	2	1		Conversion Status Code
FERPCSOK			(Blank)	OK
FERPCSNT			A	Numeric Truncation Error
FERPCSCT			B	Char. Truncation Error
FERPCSFE			C	Format Error
FERPCSTC			D	Type Conflict
	3	1		Reserved
FERPDSA	4	4		Physical Segment Address (If variable length, points to a two Byte length field)
	8	2		Reserved
FERPPFL	10	2		Physical Field Length
FERPPFA	12	4		Physical Field Address
FERPUSA	16	4		User Segment Address
	20	2		Reserved
FERPUFL	22	2		User Field Length
FERPUFA	24	4		User Field Address
FERPFSBA	28	4		FSB Address
FERPUWA	32	32		User Work Area

E N T R Y S E Q U E N C E

The user exit routines will be entered for GET calls after the field has been moved (and converted if necessary) from the physical segment to the application program.

On a PUT, the routine will be entered after the field has been moved (and converted) to the physical view.

S E G M E N T A D D R E S S A B I L I T Y

The addresses of both the physical segment and the user's view of the segment are provided to the user's routine. As the field processing order is arbitrary, the routine should not rely on other fields in the user's view during GET processing or fields in the physical view during PUT processing.

C O N V E R S I O N S T A T U S C O D E

If any problems arise during the system supplied field conversion, the conversion status code will indicate the problem. If the user routine corrects the problem, the code should be reset to a blank. Setting the code non-blank will likewise result in termination of the request with a status code of 'Cx' where 'x' is the code set by the user routine.

F I E L D S U S E D I N S S A ' s

A user exit routine for fields that are used in SSAs must not be dependent upon an other field in either view. This is because the routine will be called during processing of the SSA too and at that time there is no physical nor application view available.

M A C R O F O R M A T

The SENFLD Macro format is as follows, if the User Field Exit Routine is be used:

```

|-----|
| SENFLD NAME=fldname[ ,BYTES=n|P][ ,START=position][ ,type=t] |
|                               [ RTNAME=prog][ ,{OPT }=REPL ] |
|                               {OPTION} NOREPL ] |
|-----|

```

RTNAME=prog 'prog' is the name of a phase in the DOS Core Image Library that is to be given control whenever this field is retrieved or stored.

E X A M P L E

SENFLD NAME=X,START=9,TYPE=H,RTNAME=AUDIT

This SENFLD statement defines sensitivity to a field named X. The user's view of the field is a 2 byte binary field located in the 9th and 10th bytes of the users view. Routine 'AUDIT' will be called wherever the field is retrieved or stored. The Field 'X' can be replaced.

VIRTUAL FIELDS

Virtual Field statements are used to define fields in an application view of a segment that do not exist in the physical view. i.e. Fields may be identified in a application's view of a segment that do not exist in the physical segment via the VIRFLD statement during PSB generation.

When an application needs access to accumulated information for totals it might be better to use the exit routine to accumulate the information and pass it to the application in a VIRFLD. The integrity of detailed information is then maintained.

You can use the VIRFLD macro to regroup the content of a segment defined by SENFLD macros. You can not split up a field defined by a SENFLD macro by using VIRFLD macros alone.

VIRFLD statements are optional. If specified, they must, along with any SENFLD statements, immediately follow the SENSEG statement for this segment. There must be one VIRFLD statement for each virtual field you want included in your application view of the segment.

An initial value and/or the name of a user written routine to be called to create the field may be specified.

If both an initial value and an exit routine are specified, the value will be put in the user's view of the field before the routine is called during a GET procedure for the field.

If a routine is specified, it will be entered both on GETs and on PUTs involving the field. For GETs, the user exit routine will be entered after the field has been initialised. If no initial value for the field is specified it will be initialised with the 'null' value. (i.e. For numeric fields: Zeroes; for character fields: Blanks. The format of the zeroes correspond to the 'type' specification; packed, zoned, binary). On a PUT, the entry to the exit routine occurs immediately as no conversion is attempted.

V I R F L D M A C R O F O R M A T

```
VIRFLD NAME=fldname[ ,BYTES=n|P][ ,START=position][ ,TYPE=t ]  
          { ,VALUE=value[ ,RTNAME=prog ]}  
          { RTNAME=prog                                            }
```

NAME = fldname indicate the name of the field.

BYTES = n Indicates the length of this virtual field in bytes. It must be numeric and in the range of 1 through 256. This parameter must be specified for field types X, C, Z and P. It must not be specified for E, D or L field types. It is optional for field types H and F.

START = position 'position' specifies the starting position of this field in terms of bytes relative to the beginning of the application program's view of the segment for which this field is defined. Start position for the first byte of the segment is 1; the maximum specification is 32767.

Subfields are permitted. If a subfield starts in the same position as a previously defined field, you may specify the name of the previously defined field, instead of a numeric value, to indicate the starting position (START=fieldname).

TYPE = t 't' indicates the type of field as follows

- X hexadecimal data (binary)
- H half word binary
- F full word binary
- P packed decimal
- C character data
- Z numeric character data
- E floating point (short)
- D floating point (long)
- L floating point (extended)

TYPE must be specified if the VALUE parameter is used.

VALUE = value specifies an initial value for the field. If the RTNAME parameter is used also, the field is initialised before the routine is called.

RTNAME = prog 'prog' is the name of a phase in the DOS core image library that is to be given control whenever this field is returned or stored.

E X A M P L E S

```
VIRFLD NAME=FLD1,TYPE=X,BYTES=8,VALUE=40
```

This VIRFLD statement defines a field named 'FLD1' of 8 bytes length, initialised as XL8'40', and starting immediately after the preceding field.

```
VIRFLD NAME=FLD2,START=4,TYPE=H,VALUE=100
```

Above statement defines a field named 'FLD2' starting at byte 4 of the users view. The field is a half word field initialised to H'100'.

```
VIRFLD NAME=FLD3,TYPE=P,VALUE=0,RTNAME=ROUT,BYTES=4
```

This VIRFLD statement defines a four byte packed decimal field initialised to PL4'0'. Program ROUT will be given control each time the field is returned or stored.

AUTOMATIC DATA FORMAT CONVERSION

New Data Types

The following new data types are allowed on the field definition statements at physical (DBD-FIELD) and logical (PSB-SENFLD, VIRFLD) definition times:

- Z - zoned decimal
- H - halfword binary
- F - fullword binary
- E - floating point single
- D - floating point double
- L - floating point extended

Supported Conversions

The following conversions are supported:

<u>From</u>	<u>To</u>
X	H,F,P or Z
H	X,F,P or Z
F	X,H,P or Z
P	X,H,F or Z
Z	X,H,F or P
C	C

Conversion is not supported for field types E, D or L, as there are no instructions available. Only length conversion is supported for C types.

Conversion from one type to another will only be done if the field does not have subfields in either view.

Value Restrictions

- Binary (X,H,F) Restrictions
The maximum number containable in a four byte binary word is 2147483647 and the minimum is -2147483648. Packed decimal or zoned decimal fields to be converted to binary are therefore restricted to be within this range.
- Packed Decimal and Zoned Decimal Restrictions
The hardware instructions for conversion limits the size of the zoned decimal field to 16 characters. If conversion is needed, values must therefore be in the range ± 9999999999999999 .

Length Conversions

Numeric field types (X,H,F,P,Z) will be padded on the left with zeroes. Truncation will also occur from the left. Truncation of any significant digits will result in the field being set to the maximum (or minimum if negative) value and a status code of 'CA'.

For binary fields the maximum is X'7FFF---' and the minimum X'8000----'. For Zoned or Packed Decimal, \pm 999---

Character fields will be padded on the right with blanks. Truncation will result in dropping the rightmost characters. Truncation of non-blank characters will result in the 'CB' status code being passed back. Character field length conversion is done only if both 'to' and 'from' fields are type 'C'.

Format Checking

Zoned and Packed Decimal fields will be pre-scanned to insure valid format prior to conversion. An invalid format will result in the null value being provided for the converted field and a status code of 'CC'.

Non-supported Conversions

Non-supported field conversions (i.e. Physical type 'Z', user's view 'E') will not be allowed during ACB generation unless a field exit routine has been specified. A non-supported field conversion will result in a 'CD' status code when encountered during field access.

'C' Type Fields

'C' type fields are 'as is' representation and as such, no conversion will take place if the field is being moved into a different type of field. For instance, if a field is defined in the physical segment as 'C' type and in a user's view as 'P', the field will be assumed to be packed decimal in the physical view and any length adjustment would be done accordingly.

An uncorrected status code (not reset by a user exit routine) will result in immediate termination of the request. No more fields or segments will be processed.

DYNAMIC SEGMENT EXPANSION

The ability to dynamically add fields to a segment without re-compiling other applications that access the segment and without unloading and reloading the data base is supported in the following manner:

1. When the application's view of a segment is defined during PSB generation, the fields the application program is sensitive to within the segment must all be defined via SENFLD or VIRFLD statements.
2. The physical segment is defined as variable length with the maximum and minimum lengths both set to the data length plus 2 for the length field.

Programs that have specified fields to which they have sensitivity will always view the segment as fixed length. The two byte length field will be maintained by DL/I DOS/VS. The user will only see it if he has specifically identified it as a field he is sensitive to.

3. If a field has to be added to the segment, it is defined in the physical segment definition after the last currently existing field. The maximum and minimum length parameters are increased to include the field. ACB Generation has to be executed.
4. When a variable length segment is used with sensitive fields, it will be expanded or truncated to fit the user's view. (Expansion will be by adding null values for defined fields or binary zeroes for undefined areas).

Note that this technique cannot be used for segments intended to be source segments for secondary indexing. (i.e. Index Source segments cannot be variable length segments).

8.2.2 EXTENDED LOGICAL RELATIONS

The functional capabilities of DL/I DOS/VS Version 1.5 are improved by removing some restrictions in the area of

- Logical Relations
- Definition of PCBs.

The user gains more flexibility in constructing logical data bases compared with earlier versions of DL/I DOS/VS.

However, the user of these new capabilities has to consider the impact on performance and recovery/restart when designing his data base.

LOGICAL DATABASES

The restriction that only one logical relationship can be utilised in one hierarchical path of a logical database (i.e. each hierarchical path of a logical database may contain only one concatenated Segment) has been dropped. (For an example refer to LDB2 Segment B/A and E/M).

This allows repetition of a logical relationship down a hierarchical path (e.g. in describing product structures) or to relate three or more segments from different physical databases in one hierarchical path (e.g. Customer -> Insurance -> Accident).

Application programs can now more often deal with one single PCB and will less frequently require the need to maintain a synchronised position in different PCBs.

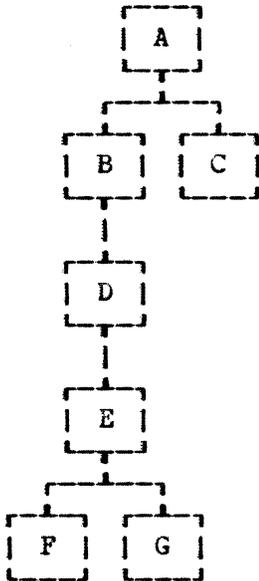
The restriction that the relative sequence of segments from left to right on any hierarchical level in a logical data base must correspond to the relative sequence of the corresponding segment in their physical databases has been dropped except for dependents of a logical relationship (For an example refer to LDB1, Segment L, M, N and S).

The rule for logical children is changed to: In its physical data base, a logical child segment cannot have another logical child as its immediate dependant. (For an example refer to PDB1 Segment B and E).

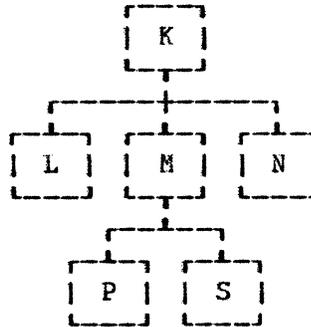
Rules for defining logical data bases are expanded in the following way: The physical parents of the destination parent and their dependents up to the root of the physical data base in inverted hierarchy can be included as dependents of the concatenated segment. (For an example refer to LDB2 Segment L and N).

E X A M P L E :

PDB1:
 =====

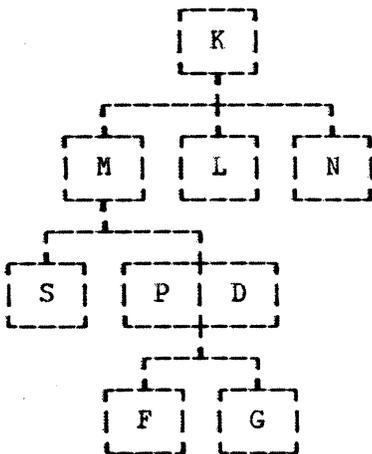


PDB2:
 =====

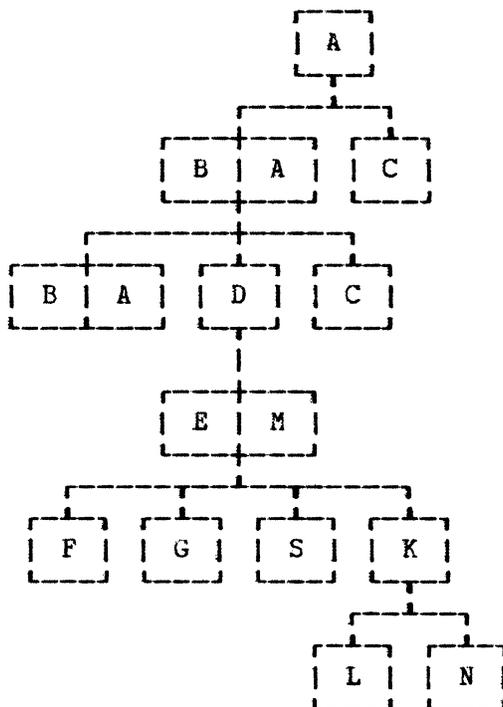


B and E are logical child segments
 A is the logical parent of B
 M is the logical parent of E
 P is a virtual logical child, paired to E

LDB1:
 =====



LDB2:
=====



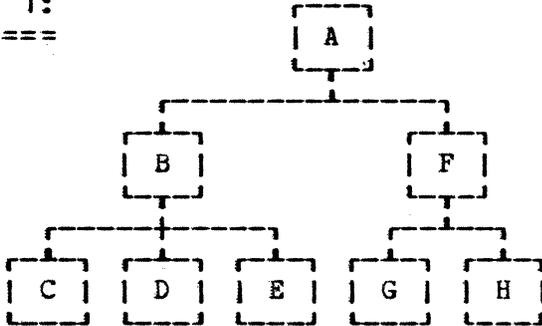
ORDERING

The rule that the SENSEG statement in the PCB must appear in the same hierarchical sequence as in the DBD has been changed. For HDAM and HIDAM data bases, segments on the same level need no longer be defined in the same sequence as in the DBD except if the PCB specifies an alternate sequence and except if the segments are dependents of a concatenated segment. However it is recommended that the required hierarchical structure is defined in a logical DBD rather than a PCB wherever possible.

E X A M P L E :

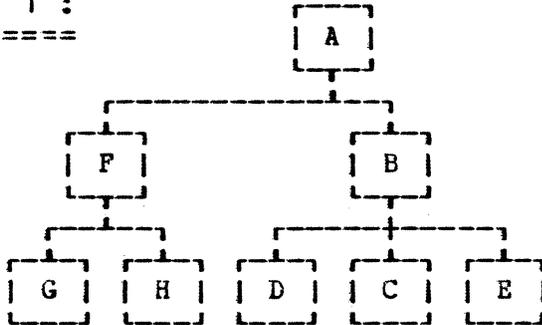
DBD 1:

=====



PCB 1 :

=====



8.2.3 4300 SYSTEMS SUPPORT

There are no changes necessary to DL/I DOS/VS to support the 4300 Systems. All support is through DOS/VSE

8.2.4 FBM SUPPORT

The FBM support is actually made available through an ICR for DL/I DOS/VS Version 1.4.

FBM support in DL/I DOS/VS is designed to allow the DL/I DOS/VS data bases and/or log file to reside on an FBM device. The major difference between FBM and count-key-data (CKD) devices is the addressing scheme. FBM is laid out as a series of blocks, which are presently 512 bytes long, starting at zero and numbered sequentially to the capacity of the device. It is addressed by block number rather than the familiar CKD disk address. The concept of tracks and cylinders no longer exists in addressing an FBM device, as it is looked at as a contiguous series of physical records. VSAM and SAM are the two access methods that support FBM devices. Functions impacted by FBM are:

- DBDGEN
- Space Management
- Utilities

DBDGEN

The DBDGEN macros will be changed to accept FBA as a device type. It also makes no sense to specify the number of cylinders to scan for space any longer. This will be changed to scan a certain number of blocks if the device specified is FBA. Because device characteristics are not known at DBDGEN time and are intended to be transparent to the user, device sensitive parameters such as track capacity, number of control intervals per track and number of control intervals per cylinder are not set at this time.

External changes to DBDGEN are confined to the DATASET macro

parameters which will be changed as follows:

```

DATASET DD1=fname1
        { 3330 }
        { 3340 }
        { 3350 }
        ,DEVICE={ 2314 }
        { TAPE }
        { FBA }

        { cyls }
        [ ,SCAN={ blks } ]
        { 3 }

        .....

```

The device parameter specification of 'FBA' will indicate that the database is to be placed on an FBM device. It will also cause other DBDGEN macros to avoid device sensitive code.

If FBA is specified in the DATASET macro on the DEVICE statement, the number specified in the SCAN parameter will no longer be the delta cylinders to be scanned when looking for space; but will represent the number of physical FBM blocks to be scanned. A default for FBM will be generated the first time that space management is entered for each data base, that is equivalent to three cylinders on a count-key-data device. If specified, this number may be any integer from 0 to 32767 blocks. The default for a CKD device is still three cylinders. The following macros will be impacted by this support:

- DATASET
- DBDGEN
- DLZDBGLB
- DLZDBVSI

SPACE MANAGEMENT

Space Management contains routines that are device sensitive to improve the performance. It is desirable to keep this philosophy, which scans for space by tracks and cylinders, for FBM-support. To accomplish this, space management will issue the DOS/VSE GETVCE macro the first time it is entered for each data base to obtain the device characteristics. The device characteristics will then be used to calculate the number of control intervals per track and cylinder to be inserted into the DMB for space management's use. Secondly, the number of physical FBM blocks to be scanned will be rounded up to a cylinder boundary. This is done to insure maximum efficiency in the scan routine and to eliminate changes to this routine.

UTILITIES

The utility support of FBM will be done by changing the DLZDVCE and the DLZDTPRM macros. These macros will be changed to accept FBA as a device type and also recognise the FBM device code of X'90' in the PUB. When an FBM device is encountered DLZDVCE will do the same as it does for 3350 and 3330 mod-11, and insert only the file name and logical unit into the DTF. All other changes to the DTF will be made at OPEN time by DOS/VSE.

INVOCATION

The ASSGN, DLBL, and EXTENT statements must be supplied as always. A difference does arise when defining space for VSAM, and when generating DBD's for DL/I DOS/VS. For DBDGEN the DATASET statement DEVICE= parameter must specify FBA, and the SCAN= parameter indicates the physical number of FBM-Blocks to be scanned instead of cylinders.

ERROR MESSAGES

The same errors already known for DBD Generations can occur with FBM as well. If the user specifies DEVICE=FBA but in fact the data base resides on a count-key-data device, Space Management will issue message DLZ831I and abend the job. The user must then insure that the data base is on a FBM device, or rerun the DBD-generation with the correct DEVICE= parameter.

8.2.5 UNIQUE SEGMENT SUPPORT

INVOCATION

A new parameter will be supported for the POINTER operand on the SEGM statement in the DBD - NOTWIN - which will specify that only a single occurrence of this segment type will be allowed for a parent.

```
-----  
|      SEGM  NAME=seg-name1      |  
|          [,PARENT=.....]      |  
|          .....                 |  
|                   {.....}      |  
|          [,POINTER={NOTWIN}]   |  
|                   {.....}      |  
|          .....                 |  
|-----|
```

NOTWIN saves 4 bytes in the prefix of the segment being defined and some processing activity at load time.

The condition of the specification is passed from DBDGEN to the action modules as a bit in the PSDB (Physical Segment Description Block). Retrieve, while processing position for insert, will insure that no other occurrence of the segment occurs under the same parent.

ERROR MESSAGE

The existence of another segment under the same parent at the same time will result in an 'II' status code returned for insert or 'LB' for load returned to the program in response to an insert or 'load' call.

8.2.6 ALL PARTITION MPS SUPPORT

All-Partition MPS Support removes a restriction from current DL/I DOS/VS Multi Partition Support (MPS) to allow one-at-a-time operation of DL/I MPS tasks from VSE/ICCF pseudo-partitions. During MPS initialisation, a set of XECBs will be defined for every partition in the system (including the online CICS/DOS/VS partition itself).

No changes to the method of invoking MPS support nor changes of error messages are made.

Note however, that the TWASIZE specified in the DFHPCT for the DLZMPC00 Program has to be increased to 268 Bytes.

8.2.7 EXTENDED SHOWCAT

The extended SHOWCAT function is used to find the high allocated RBA of a VSAM dataset. This function is used in DL/I DOS/VS presently by hard coding the macro expansion of the VSAM CATALOG macro. This had to be done since DL/I DOS/VS ships source code to the user, but VSAM does not ship the CATALOG macro. To avoid this code dependency in the future, the SHOWCAT macro, which is shipped with VSAM, was extended to include this function.

The code for this item is in the Space Management module (DLZGGSP0) and in the OPEN/CLOSE module (DLZDLOC0).

To find the high allocated RBA of the DL/I DOS/VS data base the SHOWCAT Macro is coded as follows:

SHOWCAT CI=addr,AREA=addr,EXTOPT=HARBADS

CI= specifies the address of the work area for the number of the first CI of the catalog entry.

AREA= specifies the address of the work area used to return the high RBA.

EXTOPT=HARBADS specifies that the high allocated RBA of the data set is desired.

If an error occurs while trying to find the high allocated RBA, DLZGGSP0 will return to the caller with R15=4. This indicates that the requested space was not aquired because something was wrong. OPEN/CLOSE will also return an error to the user using existing messages.

To avoid DOS/VSE Rel.1 becoming a prerequisite for DL/I DOS/VS version 1.5, this extended showcat macro is used for FBM devices only. This will be changed in later versions of DL/I DOS/VS.

8.2.8 CMS SUPPORT

CMS libraries can be used to store DL/I macros and application programs. Coding and generation of DL/I control blocks as well as creation, modification and compilation of application programs is facilitated by the CMS text manipulation and EXEC facilities. Languages supported are Assembler, Cobol, PL/I and RPG-II.

The DL/I DOS/VS macros must reside on a CMS maclib to permit the use of the CMS assembler.

Batch DL/I programs can be link-edited and executed either interactively or in CMS Batch. On-line DL/I programs using CICS functions must be submitted to a DOS machine for link-edit and execution.

Execution under control of CMS can fetch phases from a DOS library. The data used for the test can be in CMS LIBS or it can be a DOS Data Set. Once the programs, tested under CMS, are ready for production, they must be submitted to the DOS Machine for cataloging into the core image library.

8.2.9 IPCS DUMP HOOK SUPPORT

IPCS Dump Hook Support provides a mechanism for obtaining standardised IPCS format dumps instead of current DL/I dumps on a DL/I initiated abnormal termination of an application program. These dumps are intended to be used by the IPCS service routines to assist the system user with problem determination. These dumps are captured on disk storage devices using a logical unit called SYSDMP.

There are no direct interactions between the user and DL/I for this function. If the user specified the SYSDMP option for the DL/I partition, the dump hooks are activated. Otherwise, DL/I dumps are taken in the current manner.

8.3 INSTALLATION AND MIGRATION CONSIDERATIONS

8.3.1 FIELD LEVEL SENSITIVITY

The following features of Field Level Sensitivity are not supported by IMS:

1. Field type conversion.
2. Field types Z, E, D or L.
3. Field exit routines
4. Virtual fields.
5. Automatic segment length calculation for physical segments.
6. Automatic field start location calculation for physical fields.
7. Automatic field start location calculations for sensitive fields.
8. 'P' parameter on SENFLD BYTES parameter for 1:1 match to physical fields.
9. Arbitrary order for FIELD statements for sequence fields.
10. Coverage sensitivity sufficient for sequence fields on insert. IMS requires sensitivity to sequence fields by name.

8.3.2 EXTENDED LOGICAL RELATIONS

IMS does not support arbitrary order of sensitive segments in HDAM or HIDAM data bases. They must be in the same order as in the physical data base definition.

By dropping restrictions from earlier DL/I DOS/VS releases, the design of a data base must be carried out very carefully. Logical data bases can be constructed by combining many physical data bases. Therefore, when designing logical databases (especially in the online environment) the user has to consider very carefully performance and recovery/restart procedures.

When there are many physical data bases involved in one logical data base, the record level enqueueing (Program Isolation) per data base call and resulting segment enqueue after the update can become significantly high. (All access paths to the inserted/deleted segments will remain enqueued; replacement calls enqueue only the affected segment).

Procedures for recovery/restart must be setup very carefully. If any of the physical data bases involved in a logical data base become damaged due to hardware or software errors, the user will normally have to recover and restart all physical databases; dependent on the data structure complexity, database availability in case of an error can be degraded significantly.

8.3.3 FBM SUPPORT

DL/I DOS/VS programs that have run on previous releases of DL/I DOS/VS will still execute properly on version 1.5. No reassembly will be necessary. If FBM support is desired, the meaning of the 'SCAN=' parameter on the DATASET statement will now indicate physical FBM blocks. This change is triggered internally in the DBDGEN macros when DEVICE=FBA is specified.

If you have to move your data bases to an FBM device, the following steps must be executed:

- Unload, Image Copy or Export your data base(s) from Count-Key-Data device.
- Regenerate the affected DBDs and run the block builder to catalog them to the system.
- Define data base space and cluster as a new FBM file with the AMS DEFINE command (i.e. modify AMS Job streams and JCL to reflect blocks instead of tracks and cylinders).
- Use Reload or Import to have your data base on the FBM device.

Note : If you migrate from an earlier release of DL/I DOS/VS than 1.4 do not use unload reload, as changes made to the method the Prefix Resolution Utility works, will not allow for successful execution. (The Control Data Set normally created by Pre-Reorg contains the sort size required by Prefix Resolution).

Upward migration to IMS is incompatible since IMS does not currently support FBM devices. (Note that Import will need changes from CKD; Device=FBA).

8.3.4 UNIQUE SEGMENT SUPPORT

The NOTWIN parameter is compatible with the IMS implementation of this feature.

8.3.5 ALL PARTITION MPS SUPPORT

When generating a DOS/VSE supervisor, the increased number of XECBs required for the MPS-Support must be taken into account. As soon as MPS is started, 2 XECBs for each partition in the system are defined plus 2 additional ones for the whole system. Each active batch MPS task requires 2 other XECBs during its operation. So the number of XECBs required for MPS support is:

$2 + (2 * (N + M))$ N: Number of partitions generated
in the Supervisor.
M: Maximum number of concurrent
running MPS batch programs.

N and M can vary from 2 to 7.

In the CICS/DOS/VS Program Control Table (DFHPCT) you must increase the TWASIZE to 268 Bytes due to that support.

8.3.6 EXTENDED SHOWCAT

The SHOWCAT Macro replaces the CATALOG Macro previously used. The function performed is the same; therefore, there is no migration consideration.

8.3.7 CMS SUPPORT

VM/370 release 5 is required to run DL/I under CMS. VM/BSE 1.1 may be used with VM/370 release 5. DL/I 1.4 or subsequent versions must be used for this support. DL/I must first be installed to a DOS system. Once this task completed, the DL/I macros must be copied to a CMS MACLIB for use by the CMS assembler.

The installation and the maintenance of the DL/I system itself and the execution of the DL/I utilities are not supported under CMS. These functions need to be executed in the DOS machine.

8.3.8 MULTIPLE PARTITION SUPPORT - MPS (DL/I REL 1.2)

Actually MPS was provided in DL/I Release 1.2 and is not 'new' to the 1.5 release.

FUNCTION

DL/I DOS/VS Multiple Partition Support (or MPS for short) is a facility that allows several application programs running in different partitions to access the same data bases concurrently with full data base integrity.

MPS is implemented as an extension to the CICS/VS - DL/I DOS/VS interface and thus requires CICS/VS to operate. Batch DL/I programs communicate their DL/I requests to the CICS/VS partition via the cross partition event control facilities of DOS/VS. Special transactions in the CICS/VS partition receive these requests, issue the appropriate DL/I calls and pass back the results to the batch partitions. All data base I/O is performed by the DL/I facility within the CICS/VS partition. MPS operation is transparent to the application program. Existing application programs, both batch and online, need not be changed when used in this environment.

MPS is especially applicable to users who must keep data bases online for the major part of a day and need to be able to run batch reports or perform minor file updating during this period. Without MPS, the data bases would have to be closed in the online system while the batch programs were being run. DL/I DOS/VS is the only DOS/VS data management facility that provides full multiple partition support with complete data base integrity.

SECURITY

Special CICS/VS transactions are provided by DL/I which allow the user to dynamically enable and disable the multiple partition support facility.

Additional security is provided via the DL/I Application Control Table (ACT). The ACT acts as a master list of all authorized programs and PSB combinations that may be used to access a data base. This facility, in conjunction with VSAM's share option 1, will prevent batch programs from accessing online data bases except under control of MPS.

The DL/I PSB (Program Specification Block) provides an additional level of security on a program-by-program basis. The PSB allows the user to control the access rights of individual programs to any segment within a data base record.

INTEGRITY

Because all data base requests from both batch and online programs are channeled to a single facility, DL/I is able to prevent programs from attempting to update the same information simultaneously. In a similar way, DL/I is able to detect deadlocks between programs and resolve them.

In addition, DL/I provides full before and after data base segment image logging and a comprehensive set of utilities to allow for recovery of the data base in the event of a software or hardware failure. With MPS a single log is written for all DL/I partitions (in the CICS/VS partition). If desired, CICS/VS emergency restart will invoke the DL/I backout utility during emergency restart processing to restore the DL/I data bases to a known point where processing may be restarted, backing out any updates made by programs executing at the time of failure. CICS/VS emergency restart provides this function for both batch MPS programs and online DL/I transactions. CICS/VS Dynamic Transaction Backout may be used to automatically back out the changes made to DL/I data bases by failing batch MPS application programs.

RESTRICTIONS

Certain DL/I programs are restricted from running in the MPS environment, i.e. the data bases they access may not be shared across several partitions while these programs are executing. The programs in this category are:

- All DL/I Utilities
- All programs that access SHSAM or HSAM data bases
- All programs that load data bases

In addition any batch DL/I programs that modify the contents of the DL/I control blocks cannot run under MPS since the DL/I control blocks no longer exist in the batch partition.

All data bases accessed by a program running under MPS control must be defined in the CICS/VS partition and accessed through MPS. A program running under MPS control cannot access any data bases not known to MPS, i.e. not defined in the CICS/VS partition.

DL/I MPS can be active in only one CICS/VS partition at a time, i.e. two concurrently executing CICS/VS partitions (test and production) cannot both have DL/I MPS active.

8.3.9 PROGRAM ISOLATION - PI (DL/I REL. 1.3)

Actually Program Isolation was provided in DL/I Release 1.3 and is not new to the 1.5 release.

Program Isolation is an optional addition to the segment intent scheduling technique used by prior releases of DL/I to insure data base integrity when multiple tasks are concurrently accessing and modifying data bases.

All PSBs in DL/I, whether originally intended for batch or online usage, contain indicators as to the access intent (no access, read-only access, update access or exclusive access) of each segment in each data base described in the PSB. During scheduling of the PSB, DL/I checks the segment access intent of the scheduling PSB against all other scheduled and waiting PSBs. Scheduling of PSBs for online programs occurs when the application program requests it via a scheduling (PCB) call. Batch MPS programs are automatically scheduled by DL/I during their initialization phase.

With intent scheduling, if a potential simultaneous update or deadlock problem could occur, DL/I will suspend the program attempting to schedule until the conflicting programs have released their PSBs.

With Program Isolation, with the exception of tasks that have requested exclusive access to a segment type, all tasks are allowed to process concurrently. To provide integrity for the concurrent update tasks, all data base calls made by tasks are monitored by DL/I so it can prevent simultaneous updating of the same segment occurrence by multiple tasks. By monitoring the DL/I calls is also able to detect and resolve deadlocks between tasks due to conflicting data base call patterns.

Program Isolation gets its name from the fact that all data base activity of an application program is isolated from all other application programs active in the system. This concept makes it possible to dynamically back out the data base activities of an application program that terminates abnormally without affecting the integrity of the data bases controlled by DL/I. This backing out does not affect the activity of other application programs processing concurrently in the system.

The use of Program Isolation requires that the CICS/VS system include support for the dynamic transaction backout facility and that all DL/I transactions be specified in the PCT with DTB=YES.

Note that Program Isolation is not supported for SHISAM. For HISAM data base processing, Program Isolation cannot be

used if PROCOPT=I is specified on root segments.

8.3.10 EXECUTING CICS/VS WITH DL/I MPS OR PI

There are several operational considerations when executing DL/I MPS or PI under CICS/VS additional to those for executing CICS/VS with a non-MPS or PI version of DL/I.

With MPS you may have defined additional data bases in the ACT that were formerly used in batch partitions only. If this is so, then any ASSGN, DLBL, and EXTENT statements necessary for these data bases will have to be placed in the JCL stream used to execute CICS/VS. In addition the presence of additional data bases in the CICS/VS partition will increase the GETVIS requirements of the partition. This is due to the additional VSAM control blocks and buffers needed to support the new data bases. Therefore it may be necessary to change the SIZE parameter on the EXEC statement or increase the virtual partition size of the CICS/VS partition.

With the addition of MPS to your CICS/VS system you should expect the volume of log records written by CICS/VS-DL/I to increase. Therefore if you are logging to disk you should consider allocating larger extents to the log data sets.

If MPS operation is started or stopped from a terminal (CSDA and CSDD transactions), no message is sent back to the terminal acknowledging the request. A message is sent to the system console indicating the success or failure of the start or stop request.

DL/I DOS/VS PI uses the partition GETVIS area to maintain information about segments and records that are enqueued as a result of a reference to them by concurrently executing DL/I tasks. This space is acquired in 2K increments dynamically by DL/I as required. Once acquired it is never released. Therefore it may be necessary to change the SIZE parameter on the CICS/VS EXEC statement or to increase the virtual partition size of the CICS/VS partition to provide additional GETVIS space in the partition.

For PI operation the UPSI statement used with the CICS/VS job stream must specify DL/I log function active (Bit 6 = 0) and DL/I log assigned to the CICS/VS system journal (Bit 7 = 0). PI also requires the CICS/VS Dynamic Transaction Backout facility be active. If either the UPSI bit settings are incorrect or dynamic transaction backout is not active in your CICS/VS system, DL/I support will terminate with a DLZ031I message.

With Dynamic Transaction Backout and PI active be sure to leave enough temporary storage space to contain the dynamic log records. To estimate the size of the dynamic log requirements for DL/I, all x'50' DL/I log records are

maintained within the dynamic log for the duration of a CICS/VS logical unit of work (LUW). The CICS/VS LUW for a batch MPS program is from initialization to termination of the program unless CHKP calls are issued by the batch MPS program. CHKP calls indicate the completion of a LUW to CICS/VS and the dynamic log records will be released at that point.

CAUTION: An operator should never do an Immediate Shut-down of CICS/VS with DL/I. CICS/VS Emergency Restart may be required following an Immediate Shut-down of CICS/VS to recover the data bases.

8.3.11 EXECUTING BATCH MPS PROGRAMS

In order to execute a batch MPS application program CICS/VS must be running in another partition and MPS operation must have been started (CSDA transaction).

The batch UPSI byte settings for MPS are the same as for non-MPS batch execution with the exception that bits 6-7 are not used. Data base logging, normally controlled by UPSI bit 6, is controlled in the CICS/VS partition under MPS operation. STXIT linkage to DL/I for abnormal task termination, normally controlled by UPSI 7, is always active under MPS operation. It may not be turned off as is the case in non-MPS batch execution.

No ASSGN, DLBL, EXTENT, or TLBL statements are required to describe the data bases or DL/I log in the batch MPS job stream. This information is contained in the JCL for the CICS/VS partition.

The 'DLI' parameter information for batch MPS operation need only specify the program name and the PSB name. Any other parameter information such as buffer pool options, etc. are ignored as this is controlled in the CICS/VS partition.

The phase name on the EXEC statement must be 'DLZMPI00'. A SIZE parameter is not required unless your application program invokes some DOS/VS function that requires a partition GETVIS area; however the use of the SIZE parameter is recommended as an operational standard. DL/I MPS does not require a partition GETVIS area in the batch MPS partition.

Batch MPS programs that were written to issue PCB and TERM calls for DL/I Release 1.2 will require changes. With PI active, batch MPS programs may not issue PCB and TERM calls for read-only or update PSBs. If this is attempted the program will be abnormally terminated with a DLZ102I message.

8.3.12 DYNAMICALLY SCHEDULING MPS OR NON-MPS EXECUTION

The requirement that a different phase name must be used on the EXEC statement to distinguish between MPS and non MPS operation can cause operational difficulties. The operations staff must be aware of when CICS/VS is active and MPS in operation and modify the JCL of batch DL/I jobs to specify the appropriate phase name on the EXEC statement.

An alternate approach would be to use a program to dynamically test to see if MPS was in operation or not and fetch the corresponding phase. The program can use the XECBTAB TYPE=CHECK macro to test for the presence of the XECB 'DLZXCB00'. If this XECB is currently defined (R15=0) then MPS operation is active and the program can fetch the phase 'DLZMPI00'. If MPS operation is not active (R15≠0) then the program should fetch the phase 'DLZRRRC00'. Note that this technique will not work if logging is required as there is no way to dynamically assign the DL/I log device if non MPS operation is required.

The JCL to execute a batch DL/I Program in this environment would use an EXEC statement that specifies the XECB testing program's name, not DLZMPI00 or DLZRRRC00. No other changes in JCL would be required.

An example program to do this for read-only (no logging) DL/I programs is given below.

```

DLZCTRL  CSECT
          BALR      R12,0           ESTABLISH BASE REG (R12)
          USING    *,R12          AND TELL ASSEMBLER
          OPEN     PRINTER,CONSOLE
          XECBTAB  TYPE=CHECK,XECB=DLZXCB00
          LTR      R15,R15         IS MPS ACTIVE?
          BZN      NOMPS           NO
          LA       R2,=C'DLZMPI00' YES, USE DLZMPI00
          MVC      IOAREA(L'MPSMSG),MPSMSG
          LA       R7,L'MPSMSG     SET UP MPS ACTIVE MSG
          B        OUTPUT         GO WRITE MSG & FETCH DL/I
NOMPS    EQU      *
          LA       R2,=C'DLZRRC00' NOT ACTIVE, USE DLZRRC00
          MVC      IOAREA(L'NOMPSMSG),NOMPSMSG
          LA       R7,L'NOMPSMSG'  SET UP MPS NOT ACTIVE MSG
OUTPUT   EQU      *
          PUT      PRINTER         INDICATE CHOICE TAKEN
          PUT      CONSOLE         ON SYSLST AND SYSLOG
          CLOSE   PRINTER,CONSOLE
          FETCH   (R2)            FETCH APPROPRIATE PHASE
MPSMSG   DC       C'MPS ACTIVE - WILL EXECUTE DLZMPI00'
NOMPSMSG DC       C'MPS NOT ACTIVE - WILL EXECUTE DLZRRC00'
PRINTER  DTFDI    DEVADDR=SYSLST,IOAREA1=IOAREAC,RECSIZE=81
IOAREAC  DS       0CL81
          DC       X'F1'
IOAREA   DC       CL80' '
CONSOLE  DTFCN    DEVADDR=SYSLOG,IOAREA1=IOAREA,BLKSIZE=80,  C
          RECSIZE=(7),RECFORM=UNDEF,MODNAME=DLZCONSL
          END

```

8.3.13 JOB ACCOUNTING WITH BATCH MPS PROGRAMS

Note that since the data base I/O operations are being carried out in the CICS/VS partition for the batch MPS job, the DOS/VS job accounting information for the batch partition will not reflect any data base I/Os or associated CPU time for the data base call processing.

If the Performance Analyzer II FDP (5798-CFP) or similar program is being used to collect job accounting information for the CICS/VS partition, the data base I/Os and associated CPU time for the data base call processing will be charged to the DL/I batch partition controller task, CSDC.

The very same consideration applies to tasks using the ISC support. There all remote data base call activities will be charged to the CICS/VS mirror transaction if the task is local or to the remote task if the task have been invoked locally.

8.3.14 PROGRAM ISOLATION CONSIDERATIONS

When a deadlock occurs between two or more DL/I tasks because of conflicting call patterns, the DL/I Program Isolation facility will resolve the deadlock by abnormally terminating one of the tasks. The task with the fewest DL/I records and segments enqueued will be the one ABENDED. The only exception to this is that if an online task is deadlocked with a batch MPS program, the online task will always be ABENDED regardless of which has the fewest enqueues.

Because a deadlock only occurs between tasks, and is dependent on the time sequence of data base calls made by the various tasks, it is impossible to predict at what point in processing any particular program will be at when a deadlock ABEND occurs. The task that is ABENDED may not be the one that issued the data base call causing the deadlock.

If a batch MPS program is ABENDED, only the changes it made to the DL/I data bases will be backed out by the CICS/VS Dynamic Transaction Backout facility. Any changes the batch MPS program made to other batch files will not be backed out since they are not under control of DL/I or CICS/VS. It is your responsibility to provide a mechanism to back out any other changes made to other batch files by the batch MPS program after an ABEND. (Refer as well to chapter 'Abnormal Termination, Recovery and Restart').

During the dynamic transaction backout processing, CICS/VS will issue messages only to CSMT for batch MPS programs. No message is written to the CPU console directly. Therefore in the event of a batch MPS program ABEND (for any reason) you will have to verify that backout was successful by examining CSMT.

If you have programs that should not be exposed to the potential of a deadlock ABEND, then you can use PROCOPT=E in the programs' PSBs to eliminate deadlock potential. Specifying PROCOPT=E on all sensitive segments gives the program exclusive control over the segments. Since no other task sensitive to the same segment types can be concurrently scheduled with this PSB, the program cannot become deadlocked on DL/I resources.

8.3.15 CHOOSING THE DL/I LOG MEDIUM

DL/I has the capability to write its log records on several different media. Only one medium can be chosen at any one time. Your choices are:

- VSAM ESDS on disk.
- SAM data set on standard labeled tapes.
- Assigned to the CICS/VS system journal.

In a batch non-MPS environment you may use either of the first two choices. If the disk log option is chosen, you must define the VSAM cluster with the 'REUSE' parameter.

In a batch MPS environment the logging is performed from the CICS/VS partition, and no log device is assigned in the batch MPS partition.

In a CICS/VS environment only the last two choices are available. However if any of the CICS/VS recovery facilities (Emergency Restart or Dynamic Transaction Backout) are to be used with DL/I tasks, the DL/I log must be assigned to the CICS/VS system journal. This choice usually performs better in a CICS/VS environment. The CICS/VS system journal is a SAM workfile and can reside on either disk or tape. Note that CICS/VS does not use standard labels on its tape journals unlike batch DL/I. The DL/I utilities, with the exception of the backout utility, will accept all forms of log input however. The DL/I backout utility, when executing in a batch partition, will not accept CICS/VS SAM disk journals as input. Any CICS/VS journal records on the log input are ignored by the DL/I utilities.

If the DL/I log is assigned to the CICS/VS system journal, CICS/VS assumes all responsibility for opening and closing the logging device and performing I/O as required. DL/I still maintains its write ahead logging logic correctly when its log is assigned to the CICS/VS journal.

DL/I DISK LOGGING CONSIDERATIONS

DL/I disk logging is an option only in batch non MPS program executions. The choice of whether to log to disk or tape is controlled through a parameter (LOG=...) on the DLI control statement. The default is LOG=TAPE so that no changes are

required in operating procedures from previous releases (previous to DL/I 1.3) for the same log option.

The DL/I disk log is a VSAM ESDS unlike the CICS/VS disk journal which is a SAM data set. Thus before using disk logging you must define the disk log ESDS in the VSAM catalog. The data set must be defined with the reusable and recovery attributes. When DL/I opens the disk log for output logging it specifies RST which resets the VSAM catalog high RBA to zero, effectively destroying any log records currently in the data set. Therefore before you reuse a disk log data set for another program execution you should first take steps to preserve the existing log records. This could be done by running the DL/I change accumulation utility against the log data set or by copying the log records to another data set.

Because the volume of log records is difficult to estimate accurately, there are a number of options which may be specified to avoid job termination when the disk log data set fills up.

1. You can define the VSAM cluster with secondary allocation. Thus when the primary allocation fills up during program execution, VSAM will automatically acquire additional space for the data set and allow the program to continue. This support is inherent in VSAM and is transparent to DL/I or the application program. All DL/I disk log data sets should normally be defined with secondary allocations.
2. You can specify two log data sets (LOG=DISK2) and DL/I will switch from one to the other when the log data set currently being used becomes full. When the switch occurs DL/I will close the current log data set and write a DLZ076A message on the CPU console. When the operator responds 'GO' to this message, DL/I will then open the other data set and write a DLZ077I message. At this point the operator should execute a program in another partition to preserve the old log data set contents in case another log switch becomes necessary.
3. If you only specify one log data set (LOG=DISK1) then DL/I will close the data set when it becomes full and issue a DLZ076A message on the CPU console. At this point the operator must execute a job in another partition to preserve the log data set contents. After the log data set contents have been copied or otherwise preserved, the operator can reply 'GO' to the DLZ076A message and DL/I will reuse the log data set. If you choose to operate with this option, then the asynchronous operator communication feature of the Advanced Functions (DOS/VS Program Product 5746-XE2) should be

used to facilitate console operations.

Through the use of the NOPAUSE specification in the LOG parameter it is possible to bypass the DLZ076A console message and consequent operator interaction. Only the DLZ077I message will be issued with this mode of operation. Because of the danger of losing a set of log records, especially when operating in the environment described above in point 3, you should not use the NOPAUSE specification.

LOGGING AND PERFORMANCE

The performance of your application programs can be very dependent on the efficiency of the DL/I logging mechanism. You can affect the efficiency of the DL/I logging mechanism in the design of your data base application. Minimizing the number of physical writes to the log will improve the logging mechanism efficiency. Physical writes to the log are required whenever:

1. The log buffer fills up.
2. A modified data base record is about to be written back to disk and the log records corresponding to the change have not yet been written.

The second condition described is known as a 'force write' of the log. Physical writes to the log because of the first condition are performed asynchronously by DL/I. Force writes are by their very nature synchronous. Thus the execution of your application program will be delayed until the log record is written. If your application program causes many force writes to the log, you will find that the performance of your application program is dependent on the speed of the log device. Note that the effect of using the asynchronous logging option is to skip the force writing of the log under the condition described in (2) above.

There are several things you can do in designing your data base applications to minimize writing to the log:

- Avoid the use of HISAM. The HISAM access method uses VSAM logical record processing. As a consequence, DL/I has no control over when a physical write to the data base will occur. Therefore DL/I must assume every PUT request to VSAM will result in a physical write to the data base. This causes DL/I to force write the log more

often than would normally occur if an HD access method were chosen. With the HD access methods, DL/I has complete knowledge of when physical writes to the data base are actually occurring. Physical writes to the data base in the HD access methods tend to be deferred longer, thus allowing a greater opportunity for an asynchronous log write.

- INDEX data bases, either HIDAM primary index data bases or secondary index data bases are also processed using VSAM logical record processing. Therefore the same conditions exist when DL/I updates these data bases as exist for HISAM data bases. Be particularly careful with secondary indices. Every time the source field value changes for a secondary index, DL/I will write at least three log records. If the source field is very volatile, considerable logging activity will occur.
- Consider the relative data base activity when assigning several HD data bases to the same HD buffer subpool. If two highly active data bases are assigned to the same subpool, activity in one data base will probably cause DL/I to write back updates to the other data base sooner than would otherwise be necessary. DL/I will do this to free up buffer space in the subpool. This, in turn, could cause force writes to the log. Balancing the activity across subpools, where possible, will reduce the occurrence of this premature log I/O.

You should attempt to use the device that will give you the best performance for your log. The following table compares the time required to write the default (1K) DL/I log record on a variety of IBM devices. This table assumes no seek is required on disk and no channel contention is encountered during the write.

IBM Device type	Data transfer time for 1K bytes	Start-Stop rotational delay time	Total log write time
3410/11-1	51.2 ms	15.0 ms	66.2 ms
3410/11-2	25.6 ms	12.0 ms	37.6 ms
3410/11-3	12.8 ms	6.0 ms	18.8 ms
3420-3	8.5 ms	4.0 ms	12.5 ms
3420-5	5.1 ms	2.9 ms	8.0 ms
3420-7	3.2 ms	2.0 ms	5.2 ms
3330-1	1.3 ms	8.4 ms	9.7 ms
3340/44	1.2 ms	10.1 ms	11.3 ms
3350	.9 ms	8.4 ms	9.3 ms
3420-4	2.2 ms	2.3 ms	4.5 ms
3420-6	1.3 ms	1.6 ms	2.9 ms
3420-8	.8 ms	1.1 ms	1.9 ms

As can be seen on above table, device start-stop time (rotational delay for disk devices), is generally the major factor in the log write time. This is more pronounced if the log records are shorter than 1K, which is often the case in an online environment.

8.3.16 DL/I CHECKPOINT (DL/I REL. 1.3)

Since Release 1.3, DL/I provides a way for an application to checkpoint DL/I data base processing. The CHKP-call, when used in conjunction with other facilities provided by DOS/VS will allow DL/I application programs to establish checkpoints periodically during processing. If a failure should occur, after recovery, processing can be restarted at the last checkpoint rather than at the beginning of the program.

With the use of the CHKP-call, DL/I application programs can be written that will permit faster recovery and require less reprocessing after a failure has occurred.

Upon restart after a failure and subsequent recovery, it is the application program's responsibility to resume processing logic at the point of the last DL/I checkpoint call. This may require the application program to reposition itself in other related files, restore intermediate totals or other control figures being maintained within the program, etc. DL/I provides no mechanism for this function. Refer to the chapter 'Abnormal Termination Recovery and Restart'.

By issuing a CHKP (Checkpoint) call the application program causes DL/I to:

1. Write any log records currently held in the log buffer on to the log.
2. Write all modified data base records currently held in main storage buffers onto the data bases. At this point any position in the data bases is lost.
3. Write a special checkpoint record on the DL/I log.
4. Write a DLZ681I message on the CPU console indicating successful completion of the CHKP-call.
5. Returns control to the application program.

Since any position in the data bases is lost at step 2, the application program must reposition itself in its data bases after executing a CHKP-call.

Once a CHKP call is successfully executed, any data base changes made by the application program up to that point are reflected in the data base on secondary storage. If a failure should occur subsequent to the CHKP call, only the data base changes reflected on the data base since the last CHKP call need be backed out. When the DL/I backout utility is executed it will stop when encounters a checkpoint record on the log.

Related File Processing Considerations

Note that changes the application program made to other files (ISAM, VSAM, etc.) which were still in main storage buffers at the time a DL/I CHKP call is issued are not written back to the files on secondary storage. It is the application program's responsibility to devise a means of insuring this information is written to those files at the time a DL/I CHKP call is issued, for example by closing the related files.

After a failure occurs and DL/I backout has been run, the data bases reflect all the changes made to them up to the last DL/I checkpoint. If other files (ISAM, VSAM, etc.) must logically synchronize to the DL/I data bases, then you must provide some way of synchronizing them to the data bases. No such facility is provided by DL/I.

DL/I CHECKPOINT IN BATCH MPS PROGRAMS

DL/I checkpoint should only be used in a batch MPS program operating in the following environment:

- The DL/I log is assigned to the CICS/VS system journal.
- Dynamic transaction backout support has been generated into the CICS/VS system.
- The DL/I Batch Partition Controller transaction, CSDC, has 'DTB=YES' specified in its CICS/VS PPT entry.

When a DL/I checkpoint call is issued in a DL/I batch-MPS program under these conditions, DL/I performs the following functions:

1. Writes any altered data base records owned by the task that are currently in main storage buffers back to the data bases. This action will also force write the log records for these changed data base records, if necessary. At this point any position in the data bases is lost.
2. Writes a CICS/VS synch-point record on the CICS/VS system journal.

3. Dequeues any records or segments enqueued for this program if PI (Program Isolation) is being used.
4. Writes the 'DLZ105I' message on the system console indicating that a DL/I checkpoint call has been successfully executed.
5. Returns control to the application program.

If a failure subsequently occurs, CICS/VS will invoke its dynamic transaction backout facility and back out all data base changes made since the last checkpoint call was issued.

Note that since no DL/I checkpoint records are written to the CICS/VS system journal, the system journal cannot be used as input to the DL/I backout utility operating in batch. CICS/VS dynamic transaction backout is the only way the effects of failing batch MPS programs can be correctly backed out.

As is the case for non MPS batch programs you must provide any synchronization between the DL/I data bases and other related files (ISAM, VSAM, etc.). Restart of the batch MPS program has the same considerations as non MPS batch programs discussed later.

DL/I CHECKPOINT IN ONLINE TRANSACTIONS

It is technically possible to issue a CHKP call from an online DL/I transaction. If a CHKP-call is issued by an online transaction DL/I performs the following functions:

1. Writes any altered data base records owned by the task that are currently in main storage buffers back to the data bases. This action will also force write the log records for these changed data base records, if necessary. At this point any position in the data bases is lost.
2. Writes a CICS/VS synch-point record on the CICS/VS system journal. No DL/I checkpoint record is written on the CICS/VS journal. A DL/I checkpoint record is written on the DL/I log only if it is not assigned to the CICS/VS system journal.
3. Dequeues all records or segments enqueued for this task if program isolation is being used.
4. Returns control to the application program.

Because no DL/I checkpoint record is written to the CICS/VS system journal in step 2, it is impossible to use the CICS/VS system journal with the DL/I backout utility in a batch partition. The only way to correctly back out the effects of this task after a failure is through CICS/VS emergency restart.

If instead of issuing a CHKP call, the online transaction issues a TERM call, then in step 2 a DL/I termination record will be written on the CICS/VS system journal; steps 3 and 4 remain the same. Because of the presence of the DL/I termination record on the log, the CICS/VS system journal can then be used with either CICS/VS emergency restart processing or with the DL/I backout utility in a batch partition.

Because of the added flexibility in recovery with TERM calls, CHKP calls should not be used in online transactions.

8.3.17 ABNORMAL TERMINATION, RECOVERY AND RESTART

Abnormal termination of a DL/I program can occur in a variety of ways:

- Batch DL/I Program ABEND.
- Batch MPS program ABEND.
- Online DL/I transaction ABEND.
- DL/I system ABEND (internal error).
- CICS/VS system ABEND.
- Power failure.

The state of the data bases and DL/I log after each of these ABEND conditions is discussed below. You must have tested operational procedures to recover from each of these types of failures.

DL/I TERM CALL AND CICS SYNC POINT PROCESSING

DL/I DOS/VS TERM call processing for a read-only PSB does NOT implicitly invoke CICS/VS Sync Point Processing. This leads to some application recovery guidelines:

- When a transaction first schedules a read-only PSB and then later schedules an update PSB, the user must not assume that updated CICS resources (e.g. VSAM data sets etc) are synchronised by DL/I Termination of the read-only PSB. Termination of the update PSB will cause CICS Sync Point to be taken in the normal way.
- If the user needs to synchronise CICS resources before scheduling the update PSB, he should issue a CICS SYNCPOINT (DFHSP TYPE=USER) request to implicitly terminate the read-only PSB.
- If there is no need to synchronise CICS resources, a DL/I TERM call should always be used in preference to CICS SYNCPOINT request for performance reasons.

- Since a DL/I TERM call for a read-only PSB does not force the end of CICS Logical Unit-of-Work, the transaction remains eligible for Automatic Transaction Restart (CICS 1.4 only).

BATCH DL/I PROGRAM ABEND

In case of a failure in a batch environment of a DL/I program the backout utility may be run to backout DB changes occurring since the last checkpoint.

For concurrently updated other files, the user has to take care for recovery. A possible way to do so is described later under 'Restarting MPS Batch Programs'.

BATCH MPS PROGRAM ABEND

The DL/I abnormal termination routine in a DL/I batch MPS program cannot be bypassed. In a batch MPS environment the DL/I abnormal termination routine performs the following functions:

1. Writes the 'DLZ096I' message on the system console if the abnormal termination was caused by a program check or other error that causes DOS/VS to invoke a STXIT AB exit. Possible causes of a STXIT AB can be found in the DOS/VS Supervisor and I/O Macros manual in the section that discusses the STXIT macro. If DL/I's abnormal termination routine was entered directly from DL/I, rather than through a STXIT, then the 'DLZ096I' message is not produced. Instead DL/I produces messages indicating the explicit reason for the ABEND.
2. Notifies DL/I in the CICS/VS partition that the batch MPS job is ABENDING. If the condition that caused the ABEND originated in the batch MPS partition, this will cause the batch partition controller task supporting this batch partition to issue a CICS/VS ABEND request.
3. Deletes the XECB defined by this partition (DLZXCbn1).
4. Depending on the setting of the UPSI byte, optionally produces a storage dump if the abnormal termination routine was entered via a STXIT PC or STXIT AB. If

DL/I's abnormal termination routine was entered directly from DL/I rather than through a STXIT, a dump is always produced, regardless of the UPSI setting.

5. Cancels the job. Any remaining job steps in the job are not executed.

Because DL/I always uses STXIT AB and STXIT PC in a batch MPS environment, your application programs should not use any facilities that require STXIT AB or STXIT PC. To do so would disrupt DL/I's abnormal termination routine and would cause unpredictable errors in the CICS/VS partition, possibly causing it to abnormally terminate as well.

If DL/I MPS fails to complete its ABEND processing the DL/I XECB defined in the batch partition will be deleted by the supervisor at end of job if not deleted by the program.

When CICS/VS 1.3 or later Release is installed and Dynamic Transaction Backout is generated, all updates done against a Data Base known by CICS/VS will be backed out by DTB up to the last Synch-Point record.

How to restart a MPS Batch program is described later under the heading 'Restarting MPS Batch Programs'.

ONLINE DL/I TRANSACTION ABEND

For a DL/I application task ABEND, including the ABEND of a MPS batch partition controller task, DL/I's abnormal termination routine performs the following functions:

1. Writes a dump of the DL/I control blocks and related information that are unique to that task on the CICS/VS dump data set.
2. If the ABENDING task is a MPS Batch Partition Controller task, the XECBS defined by that BPC are deleted.
3. Issues a TERM call on behalf of the ABENDING task to cause any data base records altered by that task which are still in main storage to be written back to the data bases. This action will also cause any log records reflecting these changes to be written to the log. If PI was used for this task, any DL/I records or segments enqueued for this task are released.
4. Writes a DL/I termination record on the log and a CICS/VS synch-point record on the CICS/VS system jour-

nal.

5. Releases any DL/I resources that belong to this task, e.g. PST, PPST, PSB, etc.
6. Returns control to CICS/VS.

DL/I and CICS/VS do not terminate in the event of a DL/I application task ABEND. DL/I data bases are not closed on a task ABEND condition. If CICS/VS Dynamic Transaction Backout was specified for this task, CICS/VS would have performed that processing prior to DL/I's abnormal termination routine gaining control.

If for this online task Automatic Transaction Restart was specified, CICS/VS would perform this function.

DL/I SYSTEM ABEND (INTERNAL ERROR)

For the condition where DL/I has detected an internal error which does not permit it to continue processing, the DL/I abnormal termination routine performs the following functions:

1. ABENDS all DL/I tasks for which it is currently processing a call with a CICS/VS ABEND code of 'D062'.
2. Disables the CICS/VS-DL/I interface. An error code will be returned in the TCA to any DL/I tasks attempting to issue a DL/I call after this condition occurs.
3. Writes the 'DLZ062I' message on the system console.
4. Returns control to CICS/VS.

Note that no attempt is made to close DL/I data bases on this type of error. No DL/I task termination records are written on the log either. However, CICS/VS will write its own task termination record on the CICS/VS system journal. Because of the presence of the CICS/VS end of task record on the CICS/VS system journal, CICS/VS emergency restart processing will not consider these tasks to be 'in-flight' and thus will not attempt to back them out. However the DL/I Backout Utility, if executed in batch, will back out the data base updates made by these tasks since there is no DL/I termination records on the log.

DL/I does not terminate CICS/VS on this condition. CICS/VS remains operational so that non-DL/I tasks may continue to

be processed.

If the DL/I tasks that were ABENDED as a result of the DL/I error have dynamic transaction backout specified, CICS/VS will attempt to back out the effects of the DL/I updates made by the tasks. This attempt will fail however, as the DL/I interface will be disabled. Dynamic Transaction Backout provides a user exit to handle such error conditions. Depending on the critical nature of the data bases it may be appropriate to terminate CICS/VS at this point. DL/I itself does not provide any user ABEND exits. When a CICS/VS shut-down is later attempted, DL/I will issue a 'DLZ068I' message and no attempt will be made to close its data bases at that time either. The DL/I log, if not assigned to the CICS/VS system journal, will be closed at CICS/VS shut-down however.

CICS/VS SYSTEM ABEND

If CICS/VS is terminating abnormally, the DL/I abnormal termination routine performs the following functions:

1. If the DL/I log is not assigned to the CICS/VS system journal, the log buffer is flushed, the log is closed, and the DOS/VS subtask for the DL/I log is detached.
2. If MPS was active it deletes all XECBs that had been defined.
3. Writes the 'DLZ070' message on the system console.
4. Attempts to load and execute the DL/I online formatted dump program 'DLZFSDP0'. This program must be specified in the CICS/VS PPT for this step to execute.
5. Returns control to CICS/VS.

Note that no attempt is made to close the DL/I data bases on this error condition. However the VSAM automatic close facility will be invoked by DOS/VS end-of-job processing to close the VSAM data sets used by DL/I. DL/I task termination records are not written to the DL/I log on this error condition. CICS/VS will not write its task termination records on the CICS/VS system journal on this ABEND condition. Thus either CICS/VS emergency restart or the DL/I Backout Utility executed in batch can back out the effects of the inflight tasks.

If CICS/VS fails to complete its ABEND processing, the DL/I

XECBs will be deleted by the supervisor at end of job.

POWER FAILURE

In case of a Power Failure, execute emergency restart of CICS/VS which will backout all in flight tasks at failure time. MPS Batch jobs will be backed out as well up to the last synch point/checkpoint issued.

RESTARTING MPS BATCH PROGRAMS

If the user takes advantage of the MPS and PI facility, he is recommended to issue periodic checkpoint calls from his batch program to avoid excessive queue length (built up by PI) and keep control block size to a minimum.

If he is using checkpoints, he needs a restarting procedure which is his responsibility. DL/I provides no mechanism for this function.

If CHKP-calls are used by Batch and MPS-Batch programs, the user must take care on repositioning sequential input/output files, restoring interim results like subtotals etc. into the appropriate program location.

Subsequently the steps to be performed for a MPS Batch restart are listed. Please note that this is a proposal. This solution has not been submitted to any formal IBM test. Users should evaluate its usefulness in their own environment prior to justification or implementation.

Step 1: Create a 'User Checkpoint Data Base' containing root only segments, one root per program. The root segment consists of the following fields:

- Keyfield: Program Number/Name
- Field 1: DL/I Checkpoint number
- Field 2: Number of input records read
- Field 3: Number of output records written
- Field 4-n: Interim results

Step 2: Application programs must be modified to contain the following code:

- a) At start the user application program checks the

Checkpoint number of the corresponding segment in the 'User Checkpoint Data Base'. If the number is 0, branch to normal program execution (this is not a restart). If the Checkpoint number is not 0 (it is a restart), open an additional output file (if sequential output is used to copy the number of output records to that tape). Read the number of input records already processed into storage and branch as many times around the get input processing routine as necessary. If the number is reached, the input file is positioned correctly for continuation of normal processing.

If a sequential output file is used, copy the appropriate number of records from original output file to the new output file. At completion, the sequential output is positioned correctly.

Before continuing with normal processing, restore the interim data into the appropriate fields within the program.

- b) Within application program logic itself, the user must immediately before the DL/I Checkpoint Call include a 'User Checkpoint Call'. In this call the user must update the appropriate segment in the 'User Checkpoint Data Base' with:
 - the value of the DL/I Checkpoint number he is going to write,
 - the number of input records read,
 - the number of output records written
 - the values of the interim data maintained in the appl. program.
- c) Issue the DL/I Checkpoint Call. Note that position is lost within the data base after the DL/I Checkpoint. The user must reposition himself in his data base before resuming further processing.
- d) At successful termination of program execution, update the appropriate segment in the 'User Checkpoint Data Base' with the Checkpoint number of 0.

This is a way to restart a MPS batch job, if DL/I Checkpoints are used. The DOS/VS Checkpoint in connection with the DL/I Checkpoint cannot be used, as there is no 'reconnect' of the MPS-batch partition to the CICS/VS partition.

This solution has the following additional benefits:

- Apart from mounting an additional new tape if sequential output processing on tape is performed by the application program, no special operator intervention is required in case of restart. The operator has just to read in the job-cards again. (Thus reducing exposure of damages due to wrong operator handling).
- Even in case of an ABEND between updating the 'User Checkpoint Data Base' and writing the DL/I Checkpoint, the 'User Checkpoint Data Base' will reflect the correct values, as Dynamic Transaction Backout will backout this data base as well to the previous DL/I Checkpoint.
- Users might expand on the content of the 'User Checkpoint DB's' root segment to include a field, containing a value each time the program terminated successfully. This value can be checked/reset before a subsequent program starts its execution etc.

The only ways to restart if files (ISAM, DAM, VSAM etc.) are updated besides the DL/I data bases are the following ones:

- a) Migrate the files to DL/I root only data bases.
- b) If the file updates are completely independent from the DB processing, users might consider restoring the file content from a SAVE tape before starting the application program. Execute during restart processing all the necessary code to update the file again.

Following is an example of User/DL/I Checkpoint and restart.

JOB 1

- Read from 'USER CHKP-DB' root segment JOB-1
- Checkpoint number is 0.
- Read first input record.
- Issue 'REPL'-Call
-
-
- Read 100th input record
- Issue 'REPL'-Call
- Write 'USER-CHKP' Record | 1 | 100 | Interim Results |
 to 'USER CHKP-DB' |-----|
- Issue DL/I CHKP-Call
-
-
- Write 'USER-CHKP' Record | 3 | 275 | Interim Results |
 to 'USER CHKP-DB' |-----|
-
-
FAILURE

BACKOUT PROCESSING

RESTART JOB 1

- Read from 'USER CHKP-DB' root segment JOB-1
- Checkpoint number is not 0
- Skip 275 input records
- Restore Interim Results
- Read 276th input record
- (Continue normal processing)
-
-

8.3.18 DOS/VSE SUPERVISOR GENERATION

The following parameters must be specified in the DOS/VSE Supervisor in order to run DL/I DOS/VS Version 1.5 with MPS and PI. Note that DOS/VSE or a subsequent release is required.

SUPVR MODE=370|E This is a new parameter. Specifies whether the supervisor is generated for /370-Mode or ECPS:VSE-Mode. The default is ECPS:VSE-Mode.

NPARTS=3|n Note that this parameter has changed. Default is 3; the minimum value is 2.

PAGEIN=n Required because DL/I DOS/VS and CICS/DOS/VS use the RELPAG macro. This is also required if the anticipatory paging feature of CICS/DOS/VS is used (ANTICPG parameter in the PCT). Set PAGEIN at least equal to the AMXT value of your CICS/DOS/VS system when using the ANTICPG feature.

TP=BTAM|VTAM Required by CICS/DOS/VS. Note that the specification of VTAM includes BTAM-Support automatically.

The parameter AP=YES (in the SUPVR macro), necessary in previous releases of DOS/VS is obsolete and standard in DOS/VSE.

FOPT XECB=n Required for DL/I DOS/VS MPS. A specification of 'YES' is not enough anymore as with Version 1.5 of DL/I the all partition MPS Support is included. The following formula can be used to calculate the minimum number of XECBS to be generated:
 $2 + (2 * (N + M))$ where N is the number of partitions generated in the Supervisor; M is the maximum number of concurrent MPS tasks. M and N can vary from 2 to 7.
 As soon as MPS is started, 2 XECBS for each partition in the system are defined plus 2 additional ones for the whole system. (e.g. 5 partitions generated and MPS started would result in 12 XECBS which are used up without any MPS processing in progress). Each active

batch MPS task requires 2 other XECBS during their execution. Do not forget to consider that you might need more XECBS defined than used by MPS only, if there are other programs using XECBS. In particular, the use of the VSE/POWER cross partition communication macros GETSPOOL, CTLSPPOOL and PUTSPOOL will require two XECBS.

If you already are using XECBS, you should ensure that the names you are using do not conflict with those of DL/I DOS/VS. The XECB names DL/I uses are:

- DLZXCBn0
- DLZXCBn1
- DLZXCBn2
- DLZXCBn3

where 'n' is a number from 1 to 7,

- DLZXCB00
- DLZXCB01

Note that the XECB names DL/I DOS/VS MPS uses cannot be changed by the user. Thus you cannot run copies of DL/I MPS in different CICS/DOS/VS partitions concurrently. The second CICS/VS-DL/I MPS partition to start would detect that the unique MPS XECB names already exist and message DLZ082I would be issued to the terminal.

All the other parameters needed in previous DOS/VS Releases in the FOPT macro (i.e. AB=YES, IT=YES, GETVIS=YES, OC=YES, PC=YES, PCIL=YES, PPIX=YES, RELDR=YES, TOD=YES, VSAM=YES) are obsoleted parameters and became standard.

8.3.19 VSE/VSAM CONSIDERATIONS

DL/I DOS/VS is using VSE/VSAM as its access method. Therefore all considerations outlined in the 'DOS/VSE Planning Guide' Volume 1 Chapter 5 apply basically for DL/I DOS/VS data bases too.

When VSE/VSAM Access Method Services (AMS) is used, the new architecture is transparent to users, except that disk addresses must be specified in terms of block numbers rather than relative track numbers. Similarly the size of a data space or data set can be specified only in terms of 'number of records' or 'number of blocks', but not in terms of cylinders or tracks. The specification of disk addresses is generally necessary in VSAM and AMS for the allocation of data spaces. Any information concerning disk addresses or extent sizes will also be in units of blocks.

The fundamental unit of transfer between the system and the FBM devices are 512 byte blocks of data, which is externally invisible to the VSE/VSAM user.

Following are some chapters you should read in the DOS/VSE Planning Guide in order to avoid mistakes in using VSE/VSAM and FBM.

- VSE/VSAM/AMS Changes for FBM Support
- Definition of Data Space and Catalog
- CLASS/USECLASS Parameter

VSE/VSAM SHARE OPTION SELECTION

Share option 1 is the only VSE/VSAM share option that should be used with any VSAM and VSE/VSAM data set being accessed by DL/I DOS/VS. DL/I DOS/VS operating in an MPS environment does not require any different VSAM share options than DL/I DOS/VS operating in a non-MPS environment. Since any DL/I batch MPS programs residing in different partitions are really accessing the data bases through a common partition, the CICS/DOS/VS partition, VSAM share option 1 is sufficient.

VSAM share option 2 provides no functional benefits because DL/I always opens its VSAM data sets for update.

VSAM share option 3 should never be used in conjunction with DL/I. It provides no protection against inadvertent scheduling of non-MPS DL/I programs that update the same data base(s) in different partitions simultaneously. Damage to data bases used in this manner may go undetected for some time, after which recovery becomes very difficult.

VSAM share option 4 does not apply to ESDSS processed at the control interval level (the way DL/I processes its HD ESDS data bases). In addition the use of share option 4 may cause the loss of data base (SHISAM, HISAM and INDEX) updates because of the way in which DL/I accesses a VSAM KSDS for update.

DL/I does not support concurrent access to data bases from more than one partition, except in an MPS environment. No attempt to access data bases concurrently should be made except through MPS, as data base damage may occur.

CONVERSION CONSIDERATIONS

If you have your data bases to move to an FBM device, the following steps must be executed:

- Unload or Export(ra) your data base(s) from CKD device.
- Regenerate the affected DBD's and run the block builder to catalog them to the system.
- Define data base space and Cluster as a new FBA file on an FBM device with AMS Define command.
- Use Reload or Import(ra) to have your data base(s) on the FBM device.

Note that the meaning of the 'SCAN=' parameter on the DATASET statement of the DBD-Macro will now indicate physical FBM blocks to be scanned. This change is triggered internally in the DBDGEN macros when DEVICE=FBA is specified. Evaluate the usefulness of the CLASS/USERCLASS parameter in connection with Databases. For details refer to the DOS/VSE planning Guide Volume 1, Chapter 5.

If you do not move your data bases to an FBM device you can process your existing databases without any modification in the area of DBDGEN/ACBGEN or VSAM.

8.3.20 CICS/DOS/VS GENERATION

CICS/DOS/VS Rel. 1.4 or a subsequent release is required for DL/I DOS/VS Version 1.5.

CICS/DOS/VS SYSTEM GENERATION

CICS/DOS/VS Rel. 1.4 or a subsequent release is required for DL/I DOS/VS Rel. 1.5.

The parameter 'DLI=YES' must be specified in the DFHSG TYPE=INITIAL macro to generate support for DL/I DOS/VS in a CICS/DOS/VS system. This is independent of whether or not MPS or PI will be used. There are no parameters for CICS/DOS/VS system generation specifically for DL/I MPS. DL/I PI requires CICS/VS Dynamic Transaction Backout support.

For ease of maintenance the preassembled CICS/VS programs, distributed in the starter system core-image library, are highly recommended.

Certain DL/I macros, identified by their DLZ names, are included in the CICS/DOS/VS Release 1.4 in order to avoid assembly errors in CICS/VS intersystems communication modules. (i.e. DLZRIB, DLZRPDIR, DLZRPCB, DLIUIB0). When DL/I DOS/VS Version 1.5 is installed the dummy CICS supplied macros are replaced with real DL/I code. After installation of DL/I, the CICS/VS modules which include the replaced macros must, of course, be re-assembled. The affected modules are:

DFHISP DFHXFP

The following module should be reassembled for performance reasons:

DFHMIR

The following modules reference DL/I DSECT and reassembly is suggested for security reasons:

DFHRUP DFHTBP DFHDBP DFHJCP

There is also a set of DL/I modules which must always be re-assembled when a new CICS/VS release is installed:

DLZOLIO0	DLZSTP00	DLZRDBL1	DLZMPC00	DLZBPC00	DLZMPI00
DLZMSTRO	DLZMSTPO	DLZODP	DLZFTDPO	DLZQUEFO	

If you choose not to use the preassembled programs provided in the CICS/DOS/VS starter system library, you should ensure that your Stage I CICS macros include the appropriate programs and parameters for DL/I.

If you already have an operational CICS/VS system and you want to add support for DL/I without performing a complete new generation, the following CICS/VS management modules must be reassembled with the DL/I global, &DLIGEN, turned on (&DLIGEN SETB 1). Note that the DL/I global in the Journal Control Program is &JCCLI not &DLIGEN.

DFHSRP	System Recovery Program
DFHSTP	System Termination Program
DFHPCP	Program Control Program
DFHSIP	System Initialization Program
DFHFCP	File Control Program
DFHJCP	Journal Control Program (if logging to CICS/VS journal)
DFHTBP	Transaction Backout Program (if CICS/VS Emergency Restart is to be used)
DFHDBP	Dynamic Transaction Backout Program (if CICS/VS dynamic transaction support is to be used)
DFHEIP	Exec Interface Program (if RPG-II online processing or ISC is to be used)
DFHEXP	Command Language Translator Program (if RPG-II online processing is to be used)
DFHISP	Intersystem Program (if ISC is to be used for remote DB access)
DFHXFP	Transformer (if ISC is to be used for remote DB access)
DFHTCP	Terminal Control Program (if remote DB access is used)

Note that although a DL/I global exists in FCP, DL/I does not currently use any FCP support. (VSAM support in FCP is not required).

The batch Partition Controller program (DLZBPC00) was non-reentrant in DL/I releases 1.2 and 1.3 and so MPS could not be used in conjunction with the subset option of CICS/VS. This restriction no longer applies since DL/I release 1.4.

CICS/VS-DL/I SYSTEM GENERATION EXAMPLE

The following is an example of the parameters required in the various CICS/VS system generation macros to support DL/I. This example is not intended to be a complete CICS/VS generation. Rather it only shows those parameters affected by the presence of DL/I in a CICS/VS system. The numbers on each line refer to the comments below. To generate the ISC support, please check in Chapter 8 of this planning guide.

1. DFHSG TYPE=INITIAL,DLI=YES,VSAM=YES,....
 2. DFHSG PROGRAM=DBP,SUFFIX=01,.....
 3. DFHSG PROGRAM=EIP
 4. DFHSG PROGRAM=EXP,LANG=(.....
 5. DFHSG PROGRAM=FCP,DUMMY=YES,...
 6. DFHSG PROGRAM=ISC,....
 7. DFHSG PROGRAM=JCP,AUTOJRN=YES,DTB=AUX|MAIN,...
 8. DFHSG PROGRAM=TCP,VTAMDEV=LUTYPE6,....
 9. DFHSG PROGRAM=TBP,....
 10. DFHSG PROGRAM=TSP,AUX=....
- DFHSG TYPE=FINAL

1. DLI=YES must be specified whether or not MPS or PI support is used. If PI support is used, then VSAM support will normally be required for CICS/VS Temporary Storage (but is not essential since Temporary storage could be provided in MAIN storage).
2. The CICS/VS Dynamic Transaction Backout program is required if DL/I PI support is used. In this example it is generated with a suffix of '01'.
3. The CICS/VS Exec Interface Program is required if online RPG II programs are to be supported. Note that MPS-batch programs written in RPG do not require the presence of EIP in the CICS/VS system.
4. The CICS/VS Command Language Translator Program is required to support all online programs written in RPG II.
5. Note that VSAM support is not required in FCP to support

DL/I. A dummy FCP may be used in the (unlikely) situation in which a CICS/VS system supports only DL/I data bases.

6. The CICS/VS Intersystem Communication Program must be generated, if a data base on a remote system has to be accessed. Note that VTAM=YES is required in the DFHSG TYPE=INITIAL macro.
7. Required to support CICS/VS Emergency Restart of a system with DL/I transactions and dynamic transaction backout when selected. Also required in all systems for which the DL/I logging function is routed to the CICS/VS system log. Note DL/I Program Isolation requires DTB and that AUTOJRN=YES must always be specified, VSAM support must be specified in the DFHSG TYPE=INITIAL macro and the DFHSG PROGRAM=TSP is also required.
8. If a remote data base has to be accessed, the LUTYPE6 parameter is required to generate an appropriate Terminal Control Program. This parameter requires that VTAM support has been generated in the DFHSG TYPE=INITIAL request.
9. If CICS/DOS/VS Emergency Restart support of DL/I transactions is used, the transaction backout program must be generated.
10. TSP is required by DL/I only when DTB=AUX has been specified to support Program Isolation (see above). However it is very likely that this function will also be required to support online user programs.

CICS/DOS/VS TABLE GENERATION

For DL/I DOS/VS support within CICS/DOS/VS the following CICS/VS tables require specific entries.

FCT	File Control Table
JCT	Journal Control Table (if Logging to CICS Journal)
PCT	Program Control Table (if MPS or PI used)
PLT	Program List Table
PPT	Program Processing Table
SIT	System Initialization Table
TCT	Terminal Control Table (ISC Support)

File Control Table (FCT)

There must be an entry in the FCT for each DL/I data base referenced either by an online DL/I transaction or by a batch DL/I MPS program. This includes not only the data bases accessed logically by the application code, but also includes all data bases that are related physically by DL/I to the logically accessed data bases. Therefore the following types of data bases must have an entry in the FCT:

HDAM Data Bases
 HIDAM Data Bases
 HIDAM Primary INDEX Data Bases
 HDAM and HIDAM Secondary INDEX Data Bases
 (S)HISAM Data Bases

For example, where a logical data base is composed of a HDAM and HIDAM data base and each of these physical data bases has a secondary index, the FCT would have to contain the following five entries:

- 1 HDAM Data Base
- 2 HDAM Secondary INDEX Data Base
- 3 HIDAM Data Base
- 4 HIDAM Primary INDEX Data Base
- 5 HIDAM Secondary INDEX Data Base

There would not be an entry in the FCT for the logical data base. The only parameters that should be specified are dataset name (use the DBDname from the PCB statement), access method (DL/I), and open option (INITIAL or DEFERRED).

Since these entries are only referenced during system initialization processing they should be grouped with other low activity entries in your FCT.

Remote data bases do not have to be defined in the FCT by using the DFHFCT TYPE=REMOTE macro request as you have to do for other files which CICS has to access in the remote system. DL/I get the necessary information from the ACT.

Journal Control Table (JCT)

For DL/I logging to the CICS/VS system journal, the buffer size can be specified in the range of 1024 Bytes (minimum) to 32767 Bytes (maximum).

You might choose to increase the CICS/VS Log Blocksize, when there is a small percentage of Force-writes and a high percentage of Log-waits. Therefore review your existing JCT specifications.

```
DFHJCT TYPE=ENTRY,JFILEID=SYSTEM,...  
      JOUROPT=(CRUCIAL,RETRY,INP...  
      BUFSIZE=4096,...  
      SYSWAIT=STARTIO (1.4 CICS)
```

- Note 1: If a buffer size other than (the default) 1024 Bytes is specified, it will be necessary to change LI-statement input to the following DL/I utilities: Log Print, Data Set Recovery, Backout, Change Accumulation. The buffer size must be specified in columns 31 to 35 (left-justified) of the LI-statement.
- Note 2: If PI is used, the DL/I log must be assigned to the CICS/VS system journal.
- Note 3: If ISC is used, the DL/I log entries go into the CICS/VS journal for the system which supports the data base(s) locally.
- Note 4: Force-Write to the log causes short blocks but is required when DB-buffers are written to DASD. Such force-writes occur when Index, HISAM and SHISAM data bases are updated and the DL/I buffer subpools are too small. The task waits when the journal buffers are full. There are no statistics available to help in log tuning.
- Note 5: Be aware of increased working set size with larger DL/I log blocksize and note that the Utilities are affected too.

Program Control Table (PCT)

If DL/I MPS is used there are four DL/I MPS transactions which must be defined in the PCT. They should be coded as follows:

```
DFHPCT TYPE=ENTRY,PROGRAM=DLZMSTRO,TRANSID=CSDA  
DFHPCT TYPE=ENTRY,PROGRAM=DLZMPC00,TRANSID=CSDB,TWASIZE=268  
DFHPCT TYPE=ENTRY,PROGRAM=DLZBPC00,TRANSID=CSDC,TWASIZE=256,DTB=YES  
DFHPCT TYPE=ENTRY,PROGRAM=DLZMSTP0,TRANSID=CSDD
```

These four transactions are used as follows:

```
CSDA  Start MPS operation  
CSDB  Master Partition Controller  
CSDC  Batch Partition Controller  
CSDD  Stop MPS operation
```

Since these transactions are probably of low activity compared to most of the transactions in your system, their

entries should be placed with other low activity entries in your PCT. Use the CICS/VS statistics to determine their placement in the PCT. These transactions should be specified as CLASS=LONG.

Only the CSDA and CSDD transactions should ever be entered from a terminal. The CSDB and CSDC transactions are internally attached by DL/I MPS. If either CSDB or CSDC is entered from a terminal, the Master Partition Controller and Batch Partition Controller programs will ignore the request (without sending a message to the terminal).

Since the CSDA and CSDD are special transactions you will probably want to put a transaction security on these so that they can only be executed by the master terminal operator or other such users.

Note that since these transaction-ids start with the letter 'C' you cannot use the CICS/VS TCLASS/CMXT facilities to control their execution.

If PI is used, then all DL/I transactions must be marked for dynamic transaction backout (DTB=YES). Failure to do so can cause loss of data base integrity. For batch MPS operation with program isolation, the CSDC transaction must be marked for dynamic transaction backout. The other three DL/I MPS transactions do not issue DL/I calls and need not have DTB specified when PI is used.

There are two ways of specifying remote DB-access in the PCT:

- a) All processing is done in the remote system including the DB-access;
- b) DB-access only is done in the remote system.

If the processing program is in the remote system, the following PCT entries are required:

```
DFHPCT TYPE=ENTRY,SYSID=name,RMTNAME=name,TRANSID=name  
DFHPCT TYPE=GROUP, FN=ISC
```

For this type of remote processing, there is no need to code remote PSBs in the ACT. There is, as well, no need to have DL/I running in the local system.

If the processing program is in the local system, but the referenced data base(s) in the remote system, the following entry has to be done

DFHPCT TYPE=GROUP, FN=ISC

Note, that in this case you have to support DL/I in both CICS/VS systems. Note, as well, that the Mirror transaction (DFHMIR) must not be eligible for CICS/VS automatic transaction restart.

Program List Table (PLT)

DL/I will not process any MPS batch programs until the Master Partition Controller program is initiated in the CICS/VS partition. This program can be initiated in two ways:

1. By a terminal operator entering the transaction 'CSDA'.
2. By letting CICS/VS execute the MPS start program (DLZMSTRO) during its system initialization processing.

If the second option is taken, a start-up PLT must be coded with the DL/I MPS Start program as one of the entries. The suffix of the start-up PLT must be specified in the 'PLTPI' parameter of the SIT.

If you want to have DL/I MPS stop automatically when CICS/VS starts its shut-down processing you should make an entry in the shut-down PLT for the program DLZMSTP0. This entry must appear before the DFHDELIM entry.

DL/I requires an entry in a shut-down PLT for the program DLZSTP00 following the DFHDELIM entry. This is required whether or not MPS is used. The suffix of this shut-down PLT must be specified in the 'PLTSD' parameter of the SIT.

Do not forget to code the PLT names in the PPT!

Processing Program Table (PPT)

When DL/I MPS is used the four programs referenced by the four DL/I MPS transactions must be defined in the PPT. They should be coded as follows:

```
DFHPPT TYPE=ENTRY,PROGRAM=DLZMSTRO
DFHPPT TYPE=ENTRY,PROGRAM=DLZMPC00
DFHPPT TYPE=ENTRY,PROGRAM=DLZBPC00
DFHPPT TYPE=ENTRY,PROGRAM=DLZMSTPO
```

Failure to define all these entries in the PPT will yield unpredictable results if MPS operation is attempted.

Since these programs are probably of low activity relative to the regular online programs they should be placed with other low activity entries in your PPT. Since DLZMSTRO and DLZMSTPO are low usage (normally used only once per CICS/VS session) they should be specified as RES=PGOUT. DLZMPC00 and DLZBPC00 should be specified as RES=YES for best performance.

DL/I also requires an entry in the PPT for the program 'DLZFSDP0' if a DL/I formatted dump is to be produced on a CICS/VS ABEND. Since this may aid in problem determination when a CICS/VS ABEND occurs, it is a good idea to include this program in your PPT. Since both DLZSTP00 and DLZFSDP0 are only used once per CICS/VS session, they should be specified as RES=PGOUT for best CICS/VS system performance.

Note that the specification for program residence options can also be made via an ALT.

For ISC support, the following entry is required:

```
DFHPPT TYPE=GROUP, FN=ISC
```

This entry generates the Mirror transaction DFHMIR (Supplied with CICS/VS) referenced in the DLZACT definition.

System Initialization Table (SIT)

The SIT must have 'DLI=YES' or 'DLI=xx' specified. In addition, DL/I requires entries in a shutdown PLT and optionally in an initialization PLT. Specify the suffix of the appropriate initialization and shutdown PLTs in the PLTPI and PLTSD parameter respectively.

Each active batch MPS program will require a CICS/VS task to support it. If you have remote programs accessing your local DL/I data base, you must add for each such remote system one CICS/VS task to support it as well. Therefore

you may want to increase your current AMXT and MXT specifications. Note that the DL/I MPS Master Partition Controller task (CSDB) is not counted by CICS/VS when calculating the number of active tasks for AMXT purposes, just as the CICS/VS terminal control, task control and journal control tasks are not counted.

For support of Inter System Communication the ISC=YES parameter must be specified. (If you need suffixed modules specify ISP=xx and XFP=xx instead of ISC=YES).

Terminal Control Table (TCT)

The CICS control blocks required to define the VTAM link to another CICS system are generated by the DFHTCT TYPE=ISLINK macro. This entry is required to establish the ISC support. Refer to the CICS/DOS/VS SRL for detailed information.

DL/I DOS/VS APPLICATION CONTROL TABLE

The online DL/I DOS/VS Application Control Table (ACT) has two functions. One function is to provide DL/I with environmental information such as buffer pool size, DMB assignments, maximum number of concurrent DL/I tasks that may execute, PI support, ISC etc.

With DL/I DOS/VS Version 1.3, PI support is provided by default. If you do not wish to use PI with your DL/I system, then you must also specify the parameter 'PI=NO' in your ACT. For good documentation you should specify 'PI=YES' in those ACTs where PI support is desired, instead of letting it default. This way your intentions are clearly documented.

An entry in the ACT for the CICS/DOS/VS Dynamic Transaction Backout Program is needed if PI=YES is specified. Reason: If the DFHDBPxx program abends trying to back out a transaction, DL/I DOS/VS will receive control to service the abend and prevent a possible CICS/DOS/VS-DL/I DOS/VS abend. The PSBNAME you specify for this program does not matter since it does not issue a PCB call.

If you have an existing CICS/DOS/VS-DL/I DOS/VS system without MPS generated, you should review your current ACT environmental parameters (MAXTASKS, etc.) to see if they require changes for MPS or ISC. Each active batch DL/I MPS program will require a Batch Partition Controller (BPC)

transaction in the online system to support it. For ISC you need a Mirror transaction for each remote system. Therefore you may need to increase the values of MAXTASK and CMAXTASK. If you are introducing more data bases into the online system (those that were formerly used only in batch and now are to be accessed via MPS) you should examine the size of your buffer pool and DMB assignments.

The second function of the ACT is to define the valid application program and PSB combinations for the online system. When an online DL/I transaction attempts to schedule a PSB, DL/I checks to ensure that the program issuing the scheduling call is authorized to use that PSB. DL/I accomplishes this by scanning its ACT for the name of the requesting program. Once DL/I locates the program name in the ACT, it then checks to see if the requested PSB is authorized for that program.

Since all batch MPS programs actually communicate to DL/I through the BPC running in the online partition, you must make an entry in the ACT for this program. You should list with this program the names of all PSBs that will be used with MPS batch application programs. Note that the DL/I utility programs cannot run under MPS. Nor can you execute any batch DL/I programs under MPS that require a PCB PROCOPT of L or use SHSAM or HSAM data bases. The form of this ACT entry is as follows:

```

DLZACT      TYPE=PROGRAM,                Col 72
            PGMNAME=DLZBPC00,           C
            PSBNAME=(name,name,..)     C

```

Potentially, a large number of PSBs can be associated with a Batch Partition Controller task for use by different batch programs. The DLZACT macro will accept up to 4095 PSB names (and program names). That should be sufficient for any user. The macro language of the DOS/VS assembler, however, will accept no more than 255 characters (including parentheses) in a sublist used as the operand in a macro. A sublist is one or more entries separated by commas and enclosed in parentheses, such as the list of PSB names following the PSBNAME key word parameter in the DLZACT macro. Since this

probably won't be sufficient for the BPC entry in the ACT, DL/I has provided a continuation facility. To continue the PSBNAME sublist you code a 'CONT=YES' parameter on each line that is to be continued. An example will make this clearer.

```
DLZACT TYPE=PROGRAM,PGNAME=DLZBPC00,          cc 72
      PSBNAME=(PSBA,PSBB,PSBC),CONT=YES      C
DLZACT PSBNAME=(PSBD,PSBE,PSBF),CONT=YES
DLZACT PSBNAME=(PSBG,PSBH)
```

In this example PSBs A through H are associated with the BPC. The use of the continuation facility is not limited to the BPC entry in the ACT.

All programs which issue DL/I schedule (PCB) call and the subsequent TERM call must be defined in the ACT. These programs must be defined in the ACT even if they do not actually issue any retrieve, update or delete/add calls.

When assembling an ACT a size parameter of at least 272K is required (// EXEC ASSEMBLY,SIZE=272K), otherwise the assembler will terminate with an IPK100 error.

If you want to access a remote data base via the usage of CICS/VS ISC, you must specify the location of the remote PSBs by using a 'DLZACT TYPE=RPSB' statements. Each PSB of this type defined for use by PL/I programs must be identified as a PL/I PSB by the LANG=PL/I parameter on the DLZACT TYPE=RPSB statement. All DLZACT TYPE=RPSB statements must immediately follow the last DLZACT TYPE=PROGRAM statement. The format of this statement is:

```
DLZACT TYPE=RPSB,PSB=psbname,SYSID=systemid[,RNAME=remname]
```

- where:
- TYPE=RPSB identifies this statement as a definition of a PSB on a remote system.
 - psbname is the name of the PSB specified on any previous DLZACT TYPE=PROGRAM statement.
 - systemid is a 4-character identifier of CICS/VS system to which the PSB and associated data bases are assigned.
 - remname is the name given to the PSB in remote system. May be 1-8 chars and defaults to psbname.

Intersystems Communication support is invoked when the DL/I

Program Request Handler (PRH) detects a request to schedule a remote PSB.

When generating a DL/I nucleus to process requests from another system, a DLZACT TYPE=PROGRAM statement must be supplied for a mirror transaction, PGMNAME=DFHMIR. The mirror transaction issues DL/I calls on behalf of the other system. Therefore, the PSBNAME parameter must include all PSB names which can be referenced by another system. In order to simplify nucleus generation for processing requests from other systems, an optional parameter, REMOTE=YES, can be specified on the DLZACT TYPE=CONFIG statement. This will generate the equivalent of a DLZACT TYPE=PROGRAM,PGMNAME=DFHMIR statement with PSBNAME equal to all PSBs defined in the DL/I nucleus.

ACT SUMMARY:

The ACT Statements MUST be coded in the following order:

```
DLZACT TYPE=INITIAL,....
DLZACT TYPE=CONFIG,.....
DLZACT TYPE=PROGRAM,....
DLZACT TYPE=RPSB,.....
DLZACT TYPE=BUFFER,.....
DLZACT TYPE=FINAL
```

E X A M P L E S

- 1. Program A in CPU-1 accesses DB-W on CPU-1 using PSB-W
Program B in CPU-1 accesses DB-X on CPU-1 using PSB-X
Program C in CPU-2 accesses DB-Y on CPU-2 using PSB-Y
Program D in CPU-2 accesses DB-Z on CPU-2 using PSB-Z

The ACT in CPU-1 should be specified as follows:

```
DLZACT TYPE=INITIAL,SUFFIX=C1
DLZACT TYPE=CONFIG,...
DLZACT TYPE=PROGRAM,PGMNAME=A,PSBNAME=W
DLZACT TYPE=PROGRAM,PGMNAME=B,PSBNAME=X
DLZACT TYPE=FINAL
```

The ACT in CPU-2 should be specified as follows:

```
DLZACT TYPE=INITIAL,SUFFIX=C2
DLZACT TYPE=CONFIG,.....
DLZACT TYPE=PROGRAM,PGMNAME=C,PSBNAME=Y
DLZACT TYPE=PROGRAM,PGMNAME=D,PSBNAME=Z
DLZACT TYPE=FINAL
```

- 2. Program A in CPU-1 accesses DB-W on CPU-1 using PSB-W
- Program B in CPU-1 accesses DB-X on CPU-1 using PSB-X
- Program C in CPU-2 accesses DB-Y on CPU-1 using PSB-Y
- Program D in CPU-2 accesses DB-Z on CPU-2 using PSB-Z

The ACT in CPU-1 should be specified as follows:

```
DLZACT TYPE=INITIAL,SUFFIX=C1
DLZACT TYPE=CONFIG,.....
DLZACT TYPE=PROGRAM,PGMNAME=A,PSBNAME=W
DLZACT TYPE=PROGRAM,PGMNAME=B,PSBNAME=X
DLZACT TYPE=PROGRAM,PGMNAME=DFHMIR,PSBNAME=Y
DLZACT TYPE=FINAL
```

The ACT in CPU-2 should be specified as follows:

```
DLZACT TYPE=INITIAL,SUFFIX=C2
DLZACT TYPE=CONFIG,.....
DLZACT TYPE=PROGRAM,PGMNAME=C,PSBNAME=Y
DLZACT TYPE=PROGRAM,PGMNAME=D,PSBNAME=Z
DLZACT TYPE=RPSB,PSB=Y,SYSID=CPU1
DLZACT TYPE=FINAL
```

- 3. Program A in CPU-1 accesses DB-W on CPU-1 using PSB-W
- Program A in CPU-2 accesses DB-W on CPU-1 using PSB-W
- Program B in CPU-1 accesses DB-X on CPU-1 using PSB-X
- Program C in CPU-2 accesses DB-Y on CPU-2 using PSB-Y
- Program D in CPU-2 accesses DB-Z on CPU-2 using PSB-Z

The ACT in CPU-1 should be specified as follows:

```
DLZACT TYPE=INITIAL,SUFFIX=C1
DLZACT TYPE=CONFIG,REMOTE=YES,.....
DLZACT TYPE=PROGRAM,PGMNAME=A,PSBNAME=W
DLZACT TYPE=PROGRAM,PGMNAME=B,PSBNAME=X
DLZACT TYPE=FINAL
```

The ACT in CPU-2 should be specified as follows:

```
DLZACT TYPE=INITIAL,SUFFIX=C2
DLZACT TYPE=CONFIG,.....
DLZACT TYPE=PROGRAM,PGMNAME=A,PSBNAME=W
DLZACT TYPE=PROGRAM,PGMNAME=C,PSBNAME=Y
DLZACT TYPE=PROGRAM,PGMNAME=D,PSBNAME=Z
DLZACT TYPE=RPSB,PSB=W,SYSID=CPU1
DLZACT TYPE=FINAL
```

4. Program A in CPU-1 accesses DB-W on CPU-1 using PSB-W
 Program A in CPU-2 accesses DB-W on CPU-1 using PSB-W
 Program B in CPU-1 accesses DB-X on CPU-1 using PSB-X
 Program B in CPU-2 accesses DB-X on CPU-1 using PSB-X
 Program C in CPU-2 accesses DB-Y on CPU-1 using PSB-Y
 Program D in CPU-2 accesses DB-Z on CPU-2 using PSB-Z

The ACT IN CPU-1 should be specified as follows:

```
DLZACT TYPE=INITIAL,SUFFIX=C1
DLZACT TYPE=CONFIG,REMOTE=YES,.....
DLZACT TYPE=PROGRAM,PGMNAME=A,PSBNAME=W
DLZACT TYPE=PROGRAM,PGMNAME=B,PSBNAME=X
DLZACT TYPE=PROGRAM,PGMNAME=DFHMIR,PSBNAME=Y
DLZACT TYPE=FINAL
```

In this ACT, the mirror transaction with PSB 'Y' has to be specified in addition to 'REMOTE=YES' because the PSB 'Y' is not used by a local running program as PSB 'W' and 'X' are.

The ACT in CPU-2 should be specified as follows:

```
DLZACT TYPE=INITIAL,SUFFIX=C2
DLZACT TYPE=CONFIG,.....
DLZACT TYPE=PROGRAM,PGMNAME=A,PSBNAME=W
DLZACT TYPE=PROGRAM,PGMNAME=B,PSBNAME=X
DLZACT TYPE=PROGRAM,PGMNAME=C,PSBNAME=Y
DLZACT TYPE=PROGRAM,PGMNAME=D,PSBNAME=Z
DLZACT TYPE=RPSB,PSB=W,SYSID=CPU1
DLZACT TYPE=RPSB,PSB=X,SYSID=CPU1
DLZACT TYPE=RPSB,PSB=Y,SYSID=CPU1
DLZACT TYPE=FINAL
```

8.4 PROGRAMMING AND OPERATING CONSIDERATIONS

8.4.1 FIELD LEVEL SENSITIVITY

Several points in the area of programming using Field Level Sensitivity have to be discussed. In particular

- Usage of the SSA
- INSERT-Calls
- Format of the Key-Feedback-Area

SSA

The information supplied as an SSA should be in the format of the application's view of the field. The field in the SSA will be converted to the physical view before the compare is done. Any fields overlapping either end of the SSA field will not be converted.

An applications view of a secondary index can be defined in either of two ways:

- Following the SENFLD/VIRFLD statements of the target segment (i.e. the root of the PCB) include the SENFLD statement referencing the XFLD statement followed by SENLFD statements referencing all the fields specified in the SRCH-parameter of the XFLD statement.
- Make the application program field sensitive for the index source segment and include following its SENSEG statement SENFLD statements for all fields specified in the SRCH-parameter of the concerned XFLD statement.

If both specifications are made, those specified under the target segment will supercede the others for the interpretation of a SSA referring to the XFLD. The applications view of the index source segment will be as specified below its SENSEG statement.

If a PCB specifies an alternate processing sequence and contains some field level sensitivity for the index source segment then one of the two ways must be used to define an application view for the secondary index.

INSERT-Calls

If field sensitivity for a segment is specified in conjunction with insert sensitivity, sensitivity must be specified

for any sequence fields in the segment. The field need not be identified by name as long as its area is included in some field for which sensitivity has been specified.

Sensitivity for insert of bi-directional logical children requires sensitivity to both physical and logical twin sequence fields.

If insert sensitivity is specified for a destination parent, the application program must be sensitive to the entire concatenated key.

Key Feedback Area

The information supplied in the key feedback area will be unconverted.

8.4.2 EXTENDED LOGICAL RELATIONS

There are no implications on the programmer by this line item. For the operator, procedures for recovery must be adapted corresponding to the extension of the logical database structure.

8.4.3 UNIQUE SEGMENT SUPPORT

If NOTWIN is specified, any attempt to insert or load more than one occurrence of this segment per parent results in an incomplete DL/I call. The application program issuing the call is notified via a status code (II for insert and LB for load) that one occurrence of this segment already exists in the data base.

8.4.4 CMS SUPPORT

All functions available to the DL/I DOS/VS application

programmer are available under CMS. In addition the application programmer disposes now of VM/370 interactive facilities and CMS editors and EXECs to develop and test his programs.

By using this support of DL/I under CMS the user can insure a better separation of development and production environments. Note however that the DL/I utilities can not be executed under CMS.

8.4.5 NEW STATUS CODE SUMMARY

- CA Numeric Truncation Error during length conversion occurred.
- CB Character Truncation Error during length conversion occurred.
- CC Invalid Packed/Zoned Decimal Character found during conversion.
- CD Non Supported Field Conversion (Type conflict).
- CE A call was issued trying to replace a field whereas the scheduled PCB does not have replace as procopt.
- II extended to: only one occurrence of the segment is permitted.

8.5 PRODUCTIVITY AIDS FOR USE WITH DL/I DOS/VS

The following aid programs may be used in conjunction with DL/I DOS/VS :

Data Base Design Aid (5748-XX4)
DB/DC Data Dictionary (5746-XXC)
Chained File - DL/I Bridge (5748-XX3)
DL/I HDAM Data Base Tuning Aid (5796-AKF)
DOSMAP - DL/I (5796-PCW)
Data Base Dump Utility (5796-AJE)
DL/I Status and Debugging Aid (5798-CGL)
ISAM-to-DL/I COBOL Translator (5796-PFB)
Data Base Call Macros (5798-CHK)
VANDL/I - DL/I Data Base Utility (5798-CDH)
DL/I Batch Monitor (5796-AKY)
DL/I Performance Aid (5785-DAX)
CICS/DOS/VS Testing DL/I DOS/VS Calls Online (5796-ARN)
DL/I DOS/VS Space Management Utilities (5796-PKF)
(Includes Pointer Validation Utility)
ISAM to DL/I Bridge (5798-CYT)
High Speed Utility for DL/I DOS/VS (5798-CYJ)

The following application programs may be used in conjunction with DL/I DOS/VS :

GIS/DOS/VS (5799-ALX)
Bill of Material - DL/I (5798-CLF)
Material Requirements Planning - DL/I (5798-CNF)
STEPS DOS/VS (5746-XX2)
PLANCODE/I DOS/VS (5746-XX9)
CAPOSS-E DOS/VS (5746-M41)
DMS/VS DOS/VS (5746-XC2)
DMS/CICS/VS DOS/VS and DOS/VSE (5746-XC4)

8.6 PRODUCT REFERENCE MATERIAL

The following material is available to be used in conjunction with DL/I DOS/VS :

ZP72-0812 Program Product Announcement Letter
ZP73-0618 CICS/DOS/VS Interface Announcement Letter
ZP73-0741 Version 1.0 Availability Notice (w/o CICS IF)
ZP74-0620 Version 1.0.2 Availability Notice
ZP74-0631 Version 1.1 Availability Notice (Log. Relationship)
ZP75-0642 Command Code F ICR Announcement Letter
ZP76-0611 Tuning Options ICR Availability Notice
ZP76-0630 Tuning Options ICR Availability Notice
ZP77-0648 Version 1.2 Announcement Letter
ZP77-0648 Version 1.3 Announcement Letter
ZP78-0748 Version 1.4 Announcement Letter
ZP79-0620 ICR Availability Notice for FBM and VSE/ICCF Support
ZP79-0621 Version 1.4 Announcement for CMS Support
ZP79-0619 Version 1.5 Announcement Letter
GH20-4230 Program Product Design Objectives
GH20-4350 Program Product Specifications
G520-2687 Sales Flyer
G520-2943 Sales Flyer
GK10-6050 Sales Flyer
GH20-1246 General Information Manual
SH24-5001 Guide For New User
SH12-5411 Application Programming Reference Manual
SH12-5412 Utilities and Guide for Systems Programmers
SH12-5413 System/Application Design Guide
SH12-5414 Operators Reference Manual, Messages and Codes
SH24-5002 Debugging Guide
LY12-5016 Program Logic Manual I
LYB0-0839 Program Logic Manual II (Microfiche)
ZZ20-2833 MPS Planning Guide
G320-6011 1.3 Planning Guide
B77-03 DL/I DOS/VS 1.2 Performance Report produced by Endicott
(See Item 3)

CHAPTER 9

DOS/VS DATA COMMUNICATIONS FUNCTION GROUP

- CICS/DOS/VS (5746-XX3) -

Note :

CICS/VS 1.4 RELEASE GUIDE published in
WTSC Boeblingen Distribution BO-06-78, item 4, Aug. 15, 1978,
describes in detail the CICS/DOS/VS 1.4 new functions, use,
API, installation and SYSGEN.

SECTIONS IN CHAPTER 9

- 9.1 Introduction
- 9.2 Functional Enhancements since CICS/VS 1.1
- 9.3 Release Objectives
- 9.4 Release Highlight Summary
- 9.5 Currency and Programming Systems
- 9.6 Terminal Support Summary Table
- 9.7 Terminal - Communication Controller Support table
- 9.9 Dependency Information Chart
- 9.9 Programs Interfacing with CICS/VS
- 9.10 Performance Factors Summary
- 9.11 Miscellaneous enhancements in 1.4
- 9.12 DOS/VS Supervisor Generation Options
- 9.13 Installation, Verification and Migration Considerations
- 9.14 Fixed Block Architecture
- 9.15 CICS/VS SRL Bibliography

CONTENTS

CHAPTER 9 DOS/VS DATA COMMUNICATIONS FUNCTION GROUP. . . 9.1

9.1 Introduction 9.1
 CICS DOS/VS (5746-XX3) 9.1
 9.1.1 Product Function 9.1

9.2 FUNCTIONAL ENHANCEMENTS since CICS/DOS/VS 1.1. . . . 9.6

9.3 CICS/VS 1.4 Release Objectives 9.8

9.4 CICS/VS 1.4 Release Highlight Summary. 9.9

 INTERSYSTEM COMMUNICATION. 9.9
 The systems. 9.9
 The connecting link. 9.9
 Application facilities across the link 9.9
 The functions available to the application program
 : 9.9
 CICS/DOS/VS Entry Level System 9.10
 Replaces subset option : 9.10
 Functions supported :. 9.10
 functional limitation :. 9.10
 Ease of installation and migration : 9.10
 Usability Enhancements 9.11
 Programming Language Support 9.11
 Assembler and RPG HLPI (EXEC). 9.11
 Command Level Interface Restrictions:. 9.11
 Enhanced DL/I CALL Interface 9.11
 CICS/OS/VS and CICS/DOS/VS 9.11
 EXECUTION Diagnostic Facility. 9.12
 Interactive Program Testing and Debugging :. . . . 9.12
 PROGRAM ISOLATION. 9.12
 CICS/DOS/VS :. 9.12
 Transaction Restart. 9.12
 3270 Enhancements Support. 9.12
 Features Supported : 9.13
 Screen Sizes : 9.13
 Additional PF keys : 9.13
 3770PC Support 9.13
 Full Function Logical Unit 9.13
 DASD Device Support Enhancements 9.13
 VSAM for intra-partition Transient Data :. . . . 9.13
 SAM for Restart Dataset :. 9.14
 3350 DASD :. 9.14
 Compatibility. 9.14
 Source Compatibility : 9.14
 Object Compatibility : 9.14

- 9.5 CURRENCY AND PROGRAMMING SYSTEMS 9.15
 - 9.5.1 Currency 9.15
 - CICS/VS 1.2 :. 9.15
 - CICS/VS 1.3 :. 9.15
 - APAR Service for CICS/VS 1.3 : 9.15
 - 9.5.2 CICS/DOS/VS Programming Systems 9.15
- 9.6 Summary of CICS/DOS/VS 1.4 Terminal Support. 9.17
- 9.7 CICS DOS/VS 1.4 TERMINAL/COMMUNICATIONS CONTROLLER SUPPORT 9.20
- 9.8 CICS/DOS/VS 1.4 Dependency Information Chart 9.22
- 9.9 Programs Interfacing with CICS/DOS/VS. 9.24
- 9.10 Performance Factors Summary 9.25
 - 9.10.1 Process Time Factors. 9.25
 - 9.10.2 Wait Time Factors 9.27
- 9.11 Miscellaneous Enhancements. 9.28
 - Summary. 9.28
 - CICS/DOS/VS and CICS/OS/VS 9.29
 - 9.11.1 Good Morning Message. 9.29
 - 9.11.2 Transient Data Queue VSAM Option. 9.29
 - 9.11.3 Dynamic OPEN of VTAM ACB. 9.30
 - 9.11.4 Initialisation Improvements 9.30
 - 9.11.5 Character Field Edit. 9.30
 - 9.11.6 3790 Keyed Message support. 9.31
 - 9.11.7 Problem Determination Improvements. 9.31
 - Storage Freeze in Transactions 9.31
 - Suspension Timeout of tasks. 9.32
 - Timing Trace 9.32
 - 9.11.8 Dump data placement and packaging 9.32
 - 9.11.9 Useability and Tuning 9.33
 - Logoff by VTAM LU. 9.33
 - Upper case Translation. 9.33
 - END statement in TABLE generation. 9.33
 - Conditional ENQ. 9.33
 - ETB support for BSC Devices. 9.34
 - Journal STARTIO option 9.34
 - Simplified Table Generation. 9.34
 - PCT / PPT GROUP statement. 9.34
 - INDEX option 9.34
 - 9.11.10 SYNCPOINT ROLLBACK 9.35
 - 9.11.11 ISC Message Switching Sample Application 9.35
 - 9.11.12 CICS/DOS/VS only 9.35
 - Tunable Short Wait 9.35
 - 9.11.13 Compatibility. 9.36

9.12 DOS/VS Supervisor Generation Options.9.37

9.13 INSTALLATION VERIFICATION & MIGRATION
CONSIDERATIONS.9.39

 9.13.1 TRANSFERRING TO CICS/DOS/VS 1.4 General
 Considerations.9.39

 Reassembly of Application Programss.9.39

 9.13.2 INSTALLATION CONSIDERATIONS9.40

 C-TYPE RPG-II APPLICATIONS9.40

 THE EXECUTION DIAGNOSTIC FACILITY.9.40

 INTERSYSTEMS COMMUNICATION SAMPLE APPLICATIONS . .9.41

 SYSGEN CHANGES9.41

 SAMPLE APPLICATIONS.9.41

 SAMPLE TABLES.9.41

 DFHGEN9.42

 TRACE ENTRIES.9.42

 INITIALISATION PROGRAMS.9.42

 VSAM INTRAPARTITION TRANSIENT DATA9.42

 9.13.3 MIGRATION CONSIDERATIONS.9.45

 Conversion from Version 1 Release 3.0 to Release 4.0 .9.45

 CICS/DOS/VS and DOS/VSE Migration Planning9.48

9.14 FIXED BLOCK ARCHITECTURE.9.50

 9.14.1 Overview.9.50

 9.14.2 Description of Supported Functions.9.51

 File-Control9.51

 Temporary Storage.9.51

 Journalling.9.52

 Transient-Data9.52

 Terminal-Control9.53

 Dump-Control9.53

 Auxiliary Trace.9.53

 Statistics9.54

 Warm Keypointing and Emergency Restart9.54

 9.14.3 Impact on users9.55

 Impact on application programmer9.55

 Impact on system programmers9.55

 Impact on computer operator.9.55

 9.14.4 COMPATIBILITY/ MIGRATION ASPECTS.9.56

9.15 CICS/VS 1.4 BIBLIOGRAPHY.9.57

 Promotional Material9.57

 Unlicensed Documentation9.57

 Licensed Documentation9.57

 Licensed Optional Microfiche Listings.9.57

9.1 INTRODUCTION

This chapter presents a brief overview of the data communication product, CICS/DOS/VS. In particular CICS/DOS/VS Version 1 Release 4 is addressed.

CICS DOS/VS (5746-XX3)

Status	: SPL program product
Latest Release	: 1.4
System IPO/E Available	: planned 3Q79
Installation Licence	: N/A
Pre-Installation Test	: 60 days

9.1.1 PRODUCT FUNCTION

CICS/VS is a general-purpose DB/DC interface between the operating system and the application programs written in ANS COBOL, PL/I or Assembler. The user can generate a CICS/VS system configuration applicable to the needs of the installation and to define the environment in which the system is to execute. User exits are provided within the CICS/VS management module for optional processing as required for specific system operation. An interface is provided to allow application programs to request CICS/VS facilities such as task and terminal management, formatting of data to terminals and interfaces to system service programs.

Although CICS itself provides a comprehensive file handling capability (sequential, direct, indexed and segmented), that is satisfactory for many applications, the full capabilities of a data base/data communication system can be obtained by the use of DL/I with CICS/VS. This allows the CICS/VS user to take advantage of the data independence capabilities of DL/I.

The operating system supervisor and data management services

are used by CICS wherever they are functionally applicable. In the area of security, CICS/VS provides facilities to prevent unauthorized access to information. These security facilities can be extended by user programming, if required.

By handling activities such as storage management, file management, terminal management, task management and so on, CICS/VS removes the need for the programmer to concern himself with such activities or with the physical characteristics of terminals, line control protocols and so on, and to allow him to concentrate on developing applications to process the transactions in a form suitable for his particular business.

CICS/VS provides many of the facilities necessary for the general data communication applications of inquiry, order entry, data entry, data collection, message switching, broadcasting, order distribution and data base browsing together with the more specific type applications including credit authorization, point of sale applications and those applications of the store and forward type.

The functions needed to support these applications are provided through the following CICS/VS logical components :

- System Management
- System Service
- System Monitoring
- System Reliability
- System Support and
- Application Services.

Each of these components contain one or more functions which provide services to the user written application programs. Most services are requested directly by the application programs, but some are performed automatically by CICS/VS to support certain functions basic to a DE/DC environment.

- The system management component contains:
 - Task management
 - Storage management
 - Time management
 - Terminal management
 - File management
 - Transient data management
 - Temporary storage management
 - Journal management
 - Sync point management

- The system service component contains :
 - Sign-on / sign-off
 - Master terminal function
 - Supervisory terminal function
 - Operator terminal function
 - System statistics
 - Asynchronous transaction processing
 - Dynamic open / close
 - Time of day control
 - Terminal test
 - Message switching

- The system monitoring component contains :

- Trace management
- Dump management

- The system reliability component contains
 - System recovery management
 - Dynamic transaction backout
 - Transaction Restart
 - Emergency restart
 - Abnormal condition program
 - Program error program
 - Terminal / Node abnormal condition program
 - Terminal / Node error program

- The system support component contains:
 - System Generation
 - Environment definition
 - System initialisation
 - System termination
 - Program preparation
 - Trace utility
 - Dump utility
 - Formatted dump program
 - System log / journal utilities

- The application service component contains :
 - Basic mapping support
 - Execution diagnostic facility (command level)
 - IBM 2260 compatibility
 - Built-in functions

Table search, Phonetic conversion, Field verify, Field edit, Bit manipulation, Input formatting and weighted retrieval.

9.2 FUNCTIONAL ENHANCEMENTS SINCE CICS/DOS/VS 1.1

ENHANCEMENT AREA	RELEASE LEVEL		
	1.2.0	1.3.0	1.4.0
DATA COMMUNICATIONS SUPPORT	3767 COMMUNICATIONS TERMINAL 3770 DATA COMMUNICATIONS SYSTEM (BATCH LOGICAL UNIT) 3270 SDLC AND 3270 BSC THROUGH VTAM 3650 HOST COMMAND PROCESSOR LOGICAL UNIT	3790 COMMUNICATION SYSTEM ENHANCEMENTS -NATIVE MODE -BATCH DATA INTERCHANGE -3270 COMPATIBILITY 3770 SDLC PROGRAMMABLE FUNCTIONS EXTENSIONS 3660 SUPERMARKET SYSTEM 3606/3608 FINANCIAL SERVICES PIPELINE SUPPORT 3600 FINANCE COMMUNICATION SYSTEM BSC	INTERSYSTEMS COMMUNICATION 3270 ENHANCEMENTS SUPPORT 3774/5P SUPPORT DYNAMIC OPEN OF VTAM ACB VTAM GOOD MORNING MESSAGE SIGNOFF ENHANCEMENTS FOR VTAM TERMINALS UPPER CASE TRANSLATION FOR ALL VTAM LUS
USABILITY	SINGLE KEYSTROKE PAGE RETRIEVAL COMMANDS USING 3270 PA/PF KEYS IMPROVED VTAM SUPPORT -CHAIN ASSEMBLY -LOGICAL RECORD PRESENTATION BMS ALIGNMENT OPTION ERASE ALL UNPROTECTED SUPPORT IN BMS	HIGH LEVEL PROGRAMMING INTERFACE VSAM ENHANCEMENTS -BUFFER POOL SHARING -REUSABLE DATA SET -GET PREVIOUS -RELATIVE RECORD DATA SETS -ALTERNATE INDEXES	ASSEMBLER and RPG II COMMAND LEVEL PROGRAMMING INTERFACES RESTRUCTURING OF VTAM TERMINAL CONTROL PROGRAM AND TABLE DL/I CALL INTERFACE ENHANCEMENTS

<p>PERFORMANCE</p>	<p>RESTRUCTURING TO REDUCE WORKING SET</p>	<p>ENHANCED JOURNALLING SHARED VSAM RESOURCE</p>	<p>VSAM INTRAPARTITION TRANSIENT DATA IMPROVED TCT SCAN DASD DEVICE SUPPORT ENHANCEMENTS TUNABLE SHORT WAIT</p>
<p>RELIABILITY, AVAILABILITY, SERVICEABILITY</p>	<p>FORMATTED DUMP DUMP ENHANCEMENTS SAMPLE NODE ERROR PROGRAM</p>	<p>DYNAMIC TRANSACTION BACKOUT STORAGE VIOLATION PARTITION DUMP IMPROVED DIAGNOSTIC INFORMATION EXIT AB OF STXIT AB</p>	<p>IMPROVED DATA DUMP PLACEMENT AND PACKAGING IMPROVED TRACE TABLE DECODING AND AUXILIARY TRACE PROGRAM CHECK ABEND TRACE TABLE IN SRP IMPROVED SYSTEM MESSAGES EXECUTION DIAGNOSTIC FACILITY TRANSACTION RESTART DFHGEN FACILITY</p>
<p>INSTALLABILITY</p>	<p>REPLACEMENT SOURCE AND GENERATED MODULES SUPPLIED WITH PTFS MACHINE_READABLE MATERIAL SHIPPED IN DOS/VS RESTORE FORM</p>	<p>IMPROVED SUBSET OPTION NEW SYSGEN OPTION MULTIPLE PRE-GENERATED CICS/VS MODULES SAMPLE APPLICATION PROGRAMS AND TABLES</p>	<p>ENTRY LEVEL SYSTEM FEWER SYSGEN OPTIONS EASIER CREATION AND MODIFICATION OF TABLES NEW SAMPLE APPLICATIONS</p>

9.3 CICS/VS 1.4 RELEASE OBJECTIVES

Meet key user requirements:
Shared Data Base
Program Isolation
Transaction Restart

Increase user productivity:
High level programming interface (HLPI) for Assembler and RPG II
DL/I Call Enhancements

Improve usability:
Entry Level System (ELS)
Execution Diagnostic Facility (EDF)
Object Code Compatibility

Extend Function to strategic direction:
Distributed data / processing through intersystem communication (ISC)
SNA device support - 3270 IDS enhancements support

DL/I DOS/VS 1.4
ann. 6/78, fcs 12/78
provides access to remote DL/I Data Base
Call Enhancements

DL/I Entry
Call Enhancements - yes.

RPG/VS R II
HLPI

All DL/I items related to CICS/OS/VS are part of
the CICS/VS 1.4 base release.

9.4 CICS/VS 1.4 RELEASE HIGHLIGHT SUMMARY

INTERSYSTEM COMMUNICATION

THE SYSTEMS

CICS/VS Inter-system communication support provides the capability for two or more CICS/VS systems to communicate, whether running in the same or remote location or in the same CPU. This is in addition to the ACF VTAM function allowing terminal logon and communication with any application in any host in a network, but provides communication on a transaction rather than session basis. Communication may be between CICS/DOS/VS, between CICS/OS/VS, or between CICS/DOS/VS and CICS/OS/VS systems, through VTAM.

Inter-system communication is not supported by CICS/DOS/VS-ELS.

THE CONNECTING LINK

The CICS/VS systems must be connected by an SDLC link which may be part of a SNA multi-system network supported by VTAM/NCP. However, the terminal network attached to the CPU's may consist of S/S, BSC or SNA terminals. The type of connection between CICS/VS systems is transparent to the CICS/VS application program.

APPLICATION FACILITIES ACROSS THE LINK

These extensions allow CICS/VS application programs to access files and databases on a connected system and allow CICS/VS transactions to route messages to another CICS/VS system. The data transmitted may initiate a transaction to process it on the connected system. The interface for the CICS/VS application programmer is at the command level (EXEC), allowing existing applications using the EXEC interface to run unchanged, or with very little change apart from error handling, to use the inter-system connection environment.

THE FUNCTIONS AVAILABLE TO THE APPLICATION PROGRAM :

The function provided extends the current File Control, Transient Data, Temporary Storage, Interval Control Transaction Initiation (START, CANCEL) commands and IMS DL/I requests, to allow the requested file, database, queue transaction requested to be on an attached system. The definition of which system is to be used to satisfy a

particular request is held in CICS/VS table extensions and is specified by the installation system programmer. This is therefore transparent to the application programmer, although applications may specifically select the particular system on which a particular command is to be executed. Existing EXEC transactions can run unchanged even though a file, database, or application program is moved to another system, or the network reconfigured, since exactly the same application program interface is used to access both local and remote resources.

Accesses to remote DL/I database managed by DL/I DOS/VS is supported. Access to remote DL/I database managed by DL/I DOS/VS is provided by DL/I DOS/VS 1.4.

CICS/DOS/VS ENTRY LEVEL SYSTEM

REPLACES SUBSET OPTION :

CICS/DOS/VS-ELS replaces the 'Subset Option' that has been part of previous CICS/DOS/VS. While providing approximately the same functional capabilities the CICS/DOS/VS-ELS will be able to make savings in both working set and path length.

FUNCTIONS SUPPORTED :

Terminal support is provided via BTAM or VTAM for 3270 devices only attached locally or via remote leased lines. BMS support for communication with online terminals provides device independence for application programs. Intrapartition transient data queues and in particular ATI are also supported. DL/I DOS/VS and DL/I Entry DOS/VS are supported together with ISAM and VSAM access methods.

FUNCTIONAL LIMITATION :

Since full recovery/restart facilities are not supported through ELS, it is recommended that the use of DL/I be limited to online inquiry, with batch update. Users wishing to update online are recommended to use full CICS/DOS/VS.

EASE OF INSTALLATION AND MIGRATION :

No system generation is required to install ELS, although some table generation will still be necessary. Only the command-level (EXEC) interface is available to the application programmer. Application programs may be written in PL/I, COBOL, RPG or Assembler. The migration to full CICS/DOS/VS is simplified since application programs, maps and tables will be object compatible with full CICS/DOS/VS.

USABILITY ENHANCEMENTS

Programming Language Support

ASSEMBLER AND RPG HLPI (EXEC)

In CICS/VS 1.3.0 the HLPI was introduced for PL/I and COBOL application programs. This removed the need for application programmer to understand and manipulate CICS/VS internal control blocks and to perform CICS/VS internal storage managements operations. The isolation of the application programmer from the CICS/VS control block logic reduces the possibility of inadvertent errors giving therefore a higher degree of protection. The HLPI helps to increase programmer productivity and reduces the probability of coding errors.

This facility will be extended in CICS/VS V1 R4.0 to provide a similar interface for Assembler and RPG application programs. The RPG interface supports the DOS/VS RPG II compiler (5746-RG1)rel.2.

COMMAND LEVEL INTERFACE RESTRICTIONS:

Following Functions are NOT supported by the command level API

- Built-in functions (except for de-edit)
- Indirect accessing of files
- Certain system programmer functions such as Task control Attach and Purge.

NOTE :

- Segmented files are now supported by command level.

ENHANCED DL/I CALL INTERFACE

CICS/OS/VS AND CICS/DOS/VS

CICS/VS application programs requiring access to a DL/I database today have a high level interface available in the form of DL/I CALLs. This still requires the application programmer to obtain the response by examining CICS/VS internal control blocks. CICS/VS V1 R4.0 enhances the interface by removing this requirement. This enhancement for DOS/VS requires DL/I DOS/VS 1.4 (FCS 12/78).

EXECUTION DIAGNOSTIC FACILITY

INTERACTIVE PROGRAM TESTING AND DEBUGGING :

This facility enables a CICS/VS application programmer to test an application written using the EXEC commands in an online interactive manner. It intercepts application EXEC commands, immediately before and after their execution, as well as transaction initialization and termination. This enables the user to step through the CICS/VS commands in the application checking their validity. The user is also allowed to make temporary modifications to argument values and response as required. No translated or other specially prepared form of the application program is required. Information will be displayed on a 3270 terminal (with a screen size of at least 1920 characters) which may be the terminal being used to run the application under test. Thus the application programmer needs only one terminal to run the new transaction and to interact with EXEC Debug Facility.

The necessity for printing a transaction dump to debug an application under development is thus considerably reduced.

PROGRAM ISOLATION

CICS/DOS/VS :

For CICS/DOS/VS DL/I DOS/VS user the program isolation scheduling is provided by DL/I DOS/VS V1 R3.0. and subsequent releases.

TRANSACTION RESTART

This facility enables the user to specify those transactions which should be immediately restarted if they have been abnormally terminated and backed out by the CICS/VS Dynamic Transaction Backout facility. In particular this facility is useful for transactions which have been abnormally terminated because of Program Isolation scheduling deadlock, although it is not restricted to this type of ABEND. Wherever possible, the terminal operator will be unaware of the ABEND, back out and restart of the transaction.

3270 ENHANCEMENTS SUPPORT

FEATURES SUPPORTED :

Current CICS/VS support for the 3270 Information Display System is extended in CICS/VS V1 R4.0 by the following enhancements: larger screen sizes than those available on previous 3270 displays; additional Program Function Keys; Program Attention key on the SCS printer; a print authorization matrix and a magnetic stripe reader.

SCREEN SIZES :

Models 1, 3 and 4 of the 3270 displays have a choice of two screen sizes. CICS/VS V1 R4.0 supports the use of either screen size on a transaction basis. For every model, one screen size coincides with current 3270 screen sizes, thus providing compatibility for existing 3270 applications.

ADDITIONAL PF KEYS :

The enhancements provide additional program function keys 13 through 24 as a feature. CICS/VS V1 R4.0 allows these to be used as transaction identifiers and BMS paging commands, or will pass them through to a CICS/VS application program when received. Thus an application program and its end users may assign a particular meaning to a program function key with CICS/VS acting only as the message medium.

3770PC SUPPORT**FULL FUNCTION LOGICAL UNIT**

CICS/VS SNA support for the 3770PC Online Programmable Communications Terminal is extended to assist users to code 3770PC programs to communicate directly using read and write macros in a similar manner to 3790 FP's for full function logical unit. This requires no change to the existing 3770PC support in CICS/VS. CICS/VS publications will be updated to outline the sequences used by CICS/VS and a sample program will be provided.

DASD DEVICE SUPPORT ENHANCEMENTS**VSAM FOR INTRA-PARTITION TRANSIENT DATA :**

CICS/VS V1 R4.0 enhancements will offer improved use of DASD devices. These enhancements allow the optional use of VSAM as the access method for Intrapartition Transient Data, giving enhanced performance by reducing the number of physical I/Os compared with the existing EDAM support.

SAM FOR RESTART DATASET :

This allows the Restart Dataset to be managed sequentially via SAM instead of DAM, allowing faster CICS/VS initialization.

3350 DASD :

CICS/DOS/VS users may also specify 3350 in certain device-related parameters for CICS/VS generation; this complements functional (transparent) support introduced with CICS/VS V1 R3.0 for 3350 and 3330 model 11.

COMPATIBILITY

SOURCE COMPATIBILITY :

CICS/VS V1 R4.0 will be source compatible with previous releases of Version 1 at application program interface (Macro or Command level). This means that application programs need, at most, be re-assembled or re-compiled to operate with Version 1 Release 4.0.

OBJECT COMPATIBILITY :

Object compatibility is guaranteed to applications written to the Command Level Interface as defined in the APRM-CL (SC33-0077). In addition there will be object compatibility between V1 R4.0 and V1 R3.0 at the level of macros and control blocks as described in the APRM-ML (SC33-0079). Object code compatibility cannot be assured for those programs that use facilities other than those described in the above manuals.

9.5 CURRENCY AND PROGRAMMING SYSTEMS

9.5.1 CURRENCY

CICS/VS 1.2 :

All releases of CICS/VS Version 1 up to the Release 2.0 are no longer current.

CICS/VS 1.3 :

Version 1 Release 3.0 was available 28 February 1977 for CICS/DOS/VS and 29 April 1977 for CICS/OS/VS. This release will be current until 13 months (DOS) and 12 months (OS) after the availability of the respective CICS/VS V1 R4.0. It will therefore be current until 31 August 1979 for CICS/DOS/VS, and 30 October 1979 for CICS/OS/VS.

APAR SERVICE FOR CICS/VS 1.3 :

This means that corrective (APAR) service will be provided for CICS/VS V1 R3.0 until the expiration of the currency periods as defined above. Preventive (PTF) service will be available for the CICS/VS V1 R3.0 up to the availability of CICS/VS V1 R4.0 (i.e. 30 June 1978 for CICS/DOS/VS and 30 September 1978 for CICS/OS/VS).

9.5.2 CICS/DOS/VS PROGRAMMING SYSTEMS

The following table shows, for users of VTAM, EXTM or BTAM, the DOS/VS Release under which CICS/DOS/VS will operate. Release numbers are provided where appropriate.

<u>CICS/DOS/VS</u>	<u>DOS/VS</u>	
	33	34
1.3.0 BTAM	X	X
1.3.0 VTAM	X	X
1.3.0 EXTM	3.1	3.1
1.3.0 ACF/VTAM		X
1.4.0 BTAM		X
1.4.0 VTAM		-
1.4.0 EXTM		3.1
1.4.0 ACF/VTAM		X

Notes:

- CICS/DOS/VS VTAM and ACF/VTAM support IBM 3270,3600,3650,3767,3770,3774/5 Programmable and 3790 (configuration support 9165,9169 and 9431)
- CICS/VS 1.4 with VTAM-supported logical units requires the ACF/VTAM PP (5746-RC3). CICS/VS 1.4 ISC facilities, other than for communications within the same CPU, require Multi-System Networking Facility feature of ACF/VTAM.
- CICS/DOS/VS ELS does not support EXTM.
- DOS PL/1 version 1 release 5.0 must be installed to use the EXECUTE interface.
- COBOL V3 or DOS/VS COBOL must be installed to use the EXECUTE interface.

To use the optional CICS/DOS/VS interface to Data Language/I (DL/I), the user must have installed either the DL/I DOS/VS (5746-xx1) or the DL/I-Entry DOS/VS (5746-xx7) Program Product. The following indicates the releases of DL/I DOS/VS and DL/I-ENTRY DOS/VS that run with CICS/VS.

CICS/DOS/VS	DL/1-E DOS/VS	DL/I DOS/VS
1.3.0	2.0	1.1, 1.2, 1.3
1.4.0	2.0	1.3, 1.4

Notes:

1. DL/I Entry DOS/VS does not support access to a remote database through CICS/DOS/VS inter-system communication. DL/I DOS/VS 1.4 supports access to remote database.
2. CICS/VS device support through VTAM, or (for SNA devices only) through EXTM, requires the COMPARE AND SWAP instruction on the host CPU.

9.6 SUMMARY OF CICS/DOS/VS 1.4 TERMINAL SUPPORT

TERMINAL	BTAM	EXPM	VTAM	REMARKS
START/STOP				
1050	B			
2260-1,2	NS			
2265-1,2	NS			
2740-1	B	NS		
2740-2	NS	NS		
2741	B	NS		
3767-1,2,3	B	NS		
5100	B			Supported as a 2741
COMM. MAG.				
CARD SELECT.	S			Supported as a 2741.
SYSTEM/7	B			
CPT-TWX 33/35	S			
WORLD TRADE				Program Controlled
TELEX	S			Disconnection Supported.

contd.

Terminal Support (continued)

TERMINAL	BTAM	EXTM	VTAM	REMARKS
BSC				
2770	B	NS		
2780	B	NS		6 Bit Transcode NOT supported.
2980	NS			
3271-1,2	NS	NS	NS	3288-2 supported as 3286-2
3274-1c	NS	NS	NS	
3275-1,2	B	NS	NS	
3276-1,2,3,4	NS	NS	NS	
3600	NS			RPQ 8K0598 required. 3614 supported only if attached to a 3601/3602.
3660	S			
3735	S			
3741-2,4	B			With Terminal Identification Feature (7850) only SWITCHED lines supported. With Expand. Comm/Multipoint Data Link Control (1685) only NON-SWITCHED lines supported.
3770	B	NS		
3780	B	NS		
SYSTEM/32	B			Supported as a 2770
SYSTEM/34	B			Supported as SYSTEM/3
SYSTEM/6	B			Supported as a 2770.
SYSTEM/3	B			
SYSTEM/7	B			
SYSTEM/360	B			
SYSTEM/370	B			
IBM 4300	B			

contd.--

Terminal Support (continued)

TERMINAL	BTAM	EXTM	VTAM	REMARKS
SDLC =====				
3600		X	X	EXTM does NOT support : 3601-2A,2B,3A,3B 3604-3,4,5,6 OR 3610-4
3630		X	X	Supported as 3600
3650		X	X	
3790		X	X	3790 Data Entry with 3760 NOT supported. 9165 Configuration IS required to support CICS/VS 1.3 enhancements. EXTM 3.1.0 WILL support 9165 Config.
3767-1,2,3		X	X	
3770		X	X	Data transfer func supported
3271-11,12		X	X	
3274-1c		X	X	
3275-11,12		X	X	
3276-11,12, 13,14		X	X	
SYSTEM/32			X	Supported as a 3770.
SYSTEM/34			X	Supported as a 3770.
SYSTEM/370			X	Intersystem communication only
IBM 4300			X	Intersystem communication only
LOCAL ATTACH =====				
2260-1,2	X			
3272-1,2	X		X	
3272-1b	X		X	
3274-1a		X	X	
3790		X	X	See remarks for SDLC 3790
7770	X			

NOTES.

- NS = NON-SWITCHED
- S = SWITCHED
- B = BOTH SWITCHED and NON-SWITCHED
- X = Supported

CICS/DOS/VS Entry Level System does not support EXTM ACF/VTAM (DOS/VS) PP 5746-RC3 is required to support CICS VTAM functions.

MSNF feature is also required for Intersystem communication with another processor.

9.7 CICS DOS/VS 1.4 TERMINAL/COMMUNICATIONS CONTROLLER SUPPORT

S/S TERMINAL	TCU / COMMUNICATIONS CONTROLLER				
	2701	2703	ICA/115, 125, 135, 138	3704/05	4300/CA
1050	X	X	X	X	X
2260-1,2	X		ICA/135 ONLY	X	
2265-1,2	X		ICA/135 ONLY	X	
2740-1	X	X	X	X	X
2740-2	X	X	X	X	X
2741	X	X	X	X	X
3767-1,2,3	X		X	X	X
COMM. MAG.					
CARD SELECT.	X	X	X	X	
SYSTEM/7	X	X	X	X	
5100	X	X	X	X	X
CPT-TWX 33/35	X	X	X	X	
WORLD TRADE					
TELEX	X	X	X	X	

Contd.-

Note :

1. IBM 4300 / CA : Only terminals using the IBM Terminal control type 1 are supported.

BSC TERMINAL	TCU / COMMUNICATIONS CONTROLLER				
	2701	2703	ICA/115,125,135,138	3704/05	4300/CA
2770	X	X	X	X	X
2780	X	X	X	X	X
2980	X	X	X	X	
3271-1,2	X	X	X	X	X
3274-1C	X	X	X	X	X
3275-1,2	X	X	X	X	X
3276-1,4	X	X	X	X	X
3651-A50,B50				X	X
3651-A60,B60			X	X	
3660			X	X	X
3735	X	X	X	X	X
3741-2,4	X	X	X	X	X
3747	X	X	X	X	X
3770	X		X	X	X
3780	X	X	X	X	X
SYSTEM/32	X	X	X	X	x
SYSTEM/34	X	X	X	X	X
SYSTEM/6	X	X	X	X	X
SYSTEM/3	X	X	X	X	X
SYSTEM/7	X	X	X	X	X
SYSTEM/370	X	X	X	X	X
IBM 4300	X	X	X	X	X
SDLC TERMINAL					
3271-11,12				X	X
3274-1C				X	X
3275-11,12				X	X
3276-11 to 14				X	X
3600				X	X
3650				X	X
3767-1,2,3				X	X
3770				X	X
3790				X	X
SYSTEM/32				X	X
3704/ 05				X	(1)
IBM 4300				X	(2)

Notes : IBM 4300 Communications Adapter is supported as -
 (1). a secondary station in ACF/VTAM environment.
 (2). a primary or a secondary station in an SNA Network.

9.8 CICS/DOS/VS 1.4 DEPENDENCY INFORMATION CHART

DEPENDENCY ITEM	REMARKS
Program Prerequisites	
- DOS/VS EXTM	SDLC Terminals require VTAM or the EXTM feature (5746-XXB) of CICS/DOS/VS
- VTAM	VTAM function require the installation of ACF/VTAM (DOS/VS) PP 5746-RC3. Additionally, the Intersystem communication facility, other than for communication within the same CPU requires the MSNF feature(VTAM).
Program Facilities Support	
- ASSEMBLER	Now supported via HLPI also.
- COBOL/VS	for HLPI COBOL V3 or DOS/VS COBOL is prerequisite.
- PL/I OPTIM.	For HLPI use DFHPL1I supplied by PL/I Opt. rel 5. Module supplied in CICS private rel lib is called DFHPL1I\$ and may be renamed and used if HLPI is not used.
- DMS/VS	
Data Management Support	
- VSAM	See VSAM support table earlier.
- SAM	Supported as CICS/VS Extrapartition Data Set.
- DAM	Blocked DAM record support.
- ISAM	
- DL/I	DL/I 1.4 required for extended CALL interface.
- SEGMENTED	Special CICS/VS unique organization.
- INDIRECT	CICS unique support for DAM,ISAM and VSAM.

-----Contd.-----

Unit Record Support	Any DOS/VS supported I/O device.
DASD Support	
- 2314	
- 2319	
- 3330-1,2	
- 3333	
- 3330-11	
- 3340	
- 3344	
- 3350	
Tape Support	Any supported DOS/VS tape device.
Current Release	
- 1.3	Current until August 31, 1979.
- 1.4	Current
Compatibility Information	
	(i) EXTM 3.1.0 users MUST use CICS DOS/VS 1.3, DOS/VS Rel 33 or above, and NCP/VS 5.0 (or ACF/NCP/VS when available).
	(ii) See MIGRATION section for further information.

9.9 PROGRAMS INTERFACING WITH CICS/DOS/VS

The following programs can be used in conjunction with CICS/VS to facilitate the installation and development of an online system.

- ETSS II (5798-CLR)
- EXTM (5746-XXB)
- DMS DOS/VS (5746-XC2)
- Chained File - DL/I Bridge (5748-XX3)
- CICS SPMOL-II (5798-CFT)
- CICS Performance Analyser-II (5798-CFP)
- CICS Dynamic Map (5798-AXR)
- CICSPLOT (5798-CCG)
- CICS Network Activity Simulator (5798-CCH)
- CICS Forward Recovery (5798-CJF)
- CICS 3270 Simulator (5798-AXC)
- CICS Online Test/Debug-II (5798-AHJ)
- CICS/MICS Screen Format Utility (5798-CBH)
- CICS COBOL Call Interface (5798-AHK)
- CICS Bill of Material (5798-CKP)

In addition the following application systems require CICS DOS/VS for data communications support.

- ATMS-II DOS/VS (5746-XXG)
- STAIRS DOS/VS (5746-XR4)
- RIRMS DOS/VS (5740-XR1)
- PLANCODE/I DOS/VS (5746-XX9)

9.10 PERFORMANCE FACTORS SUMMARY

The purpose of this section is to provide a checklist type overview of some of the important factors influencing performance of CICS/VS systems.

The user should refer to the CICS/VS System Programmers Guide and the CICS/VS System Programmers Reference Manual for detailed discussion of some of the individual parameters.

Good or poor performance may be judged in many different ways, dependent on the environment. However typical criteria for the level of performance are :

- (a) System Throughput
- (b) Response Time
- (c) CPU Utilization
- (d) Storage Utilization

Performance factors can be divided into two major groups :

- (a) Those that affect process time i.e. CPU process time and paging rate influence on process time.
- (b) Those that affect wait time i.e. Wait time due to un-overlapped I/O and time due to tasks being 'queued' waiting for execution.

The following list documents the factors in each group :

9.10.1 PROCESS TIME FACTORS

- (a) CPU process time.

- DOS Factors - Job Accounting
 - Fast CCW Translate for high activity VSAM files

- CICS/VS Services - Trace
 - Dump
 - Automatic Statistics

- CICS/VS Access Method - BTAM wraplist, general poll, negative

- poll delay
- VTAM Definite Response, Immediate I/O,
Message Integrity

CICS/VS Management - Number, size and sequence of tables and entries

CICS/VS Integrity - Use of logging and protection of
Temporary Storage, Transient Data, File Control

I/O Blocking - Locate/move mode

- Application Functions - Terminal control BMS
- File control browsing
 - Getmain/initialization image
 - Program calling strategy - linking or XCTL
 - Error recovery

Data Base Support - DL/I call processing

(b) Paging Rate Influence on Process Time

- CICS/VS Batch Contention
- TP Balancing
- Storage over-commitment
- Conversational transactions
- Automatic Journalling
- Anticipatory paging
- Level of CICS/VS multi-tasking (MXT, AMXT parameters)
- Transaction Rate
- Resident Programs
- Nucleous Load Table (NLT)
- Dynamic File Open
- Storage Cushion size
- Trace Table size
- VSAM - Number of strings, buffers
 - Local Shared Resources
 - Number of index levels
- DL/I buffer pool - number of subpools and buffers

9.10.2 WAIT TIME FACTORS

- (a) Un-overlapped I/O (between partitions and within CICS/VS partition across tasks)

Paging I/O
Synchronous tape and disk I/O
CICS/VS Protected Resource logging
Auxilliary Trace
ISAM non-resident index
ISAM overflow records

- (b) 'Queued' Wait Time (at the task level)

Application Exclusive Resource Control

- VSAM file control
- DL/I PSB
- Transient data
- Temporary storage
- VTAM message integrity

System/Network

- Channel queues
- DOS/VS transients
- VSAM strings
- DL/I Maximum Tasks parameter (MXT)
- CICS/VS Maximum Tasks parameter (MXT)
- Short On Storage (SOS) condition in CICS/VS
- VTAM Receive Any Pool

9.11 MISCELLANEOUS ENHANCEMENTS

SUMMARY

Following is a list of miscellaneous enhancements in CICS/VS
1.4

CICS/DOS/VS and CICS/OS/VS :

- Good Morning message
- Transient Data Queue improvements
- Dynamic Open of VTAM ACB
- Initialisation Improvements
- Character field edit in HLPI
- 3790 Keyed Messages Support
- Problem Determination Improvements
- Improved dump data placement and packaging
- Usability and tuning enhancements
- Simplified table generation
- SYNCPOINT ROLLBACK

CICS/DOS/VS only :

- High Level Programming Interface for RPG II
- Access to remote Database
- DL/I DOS/VS call enhancements
- Tunable short wait

CICS/DOS/VS AND CICS/OS/VS

9.11.1 GOOD MORNING MESSAGE

A "Good Morning" message may be sent to any VTAM LU after the LU logs on to CICS/VS. The message may be specified by the user. For the IBM 3270 display terminals a CICS/VS logo (like VM/370) is generated. The 'Good Morning' message is displayed on the bottom line below the logo. The text of the message is specified by the user in the DFHTCT TYPE=INITIAL macro.

```
DFHTCT TYPE=INITIAL,
        ACCMETH=(VTAM,..),
        APPLID=...,
        GMTEXT='Hello! CICS is now online.'
```

The good morning message is sent after VTAM logon by LU to CICS/VS. The message is sent by the transaction CSGM, which is invoked in the OPENDST exit for the LU.

An option in the terminal entry specifies if the message is to be sent to the LU.

```
DFHTCT TYPE=TERMINAL,
        ACCMETH=VTAM,
        GMSG=NO | YES,           NO is the default.
        ...
```

9.11.2 TRANSIENT DATA QUEUE VSAM OPTION

Optionally VSAM may be used as the access method for the Intra-partition Transient Data queue management. This is instead of the current DAM support which is the default. For most users this means a significant reduction in the number of I/O operations and the associated disk I/O and CPU instruction overhead.

The VSAM option is specified in the TDP program
DFHSG PROGRAM=TDP,INTRA=NO|DAM|VSAM,...

There is a device option for DAM in DFHDCT TYPE=INITIAL.

9.11.3 DYNAMIC OPEN OF VTAM ACB

VTAM may be started after CICS has been activated. However VTAM may not be terminated prior to shutdown of CICS/VS. Instead of at the time of CICS initialisation, master terminal operator can initiate a deferred open of VTAM ACB.

9.11.4 INITIALISATION IMPROVEMENTS

This permits high flexibility in system operation. All the parameters that can be specified in the Systems Initialisation Table (SIT) may also be entered as overrides at initialisation time. Overrides may be entered via the console or the SYSIN device.

EXEC=YES can be specified in SIT or overrides. There is no need for a PPT entry for DFHEIP program. It is loaded as a part of the CICS nucleus.

PRINT=PAN, specifies the 3277 print key support in SIT or override.

SKRxxxx='page-retrieval-command', xxxx is any PA/PF key.

9.11.5 CHARACTER FIELD EDIT

This CICS built-in function is now available in HLPI.

The DEEDIT built-in function of the macro interface is now also available via HLPI.

This is particularly convenient for RPG programs needing the function equivalent to the PL/I and COBOL picture function.

Also segmented file support in local CICS system is available in the Command Level.

9.11.6 3790 KEYED MESSAGE SUPPORT

3790 (EC52) extends the capability to retrieve messages by key / identifier rather than sequentially. 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 the user route lists or on CMSG requests.

9.11.7 PROBLEM DETERMINATION IMPROVEMENTSStorage Freeze in Transactions

Currently certain types of storages are freed during transaction processing. As a problem determination aid, these areas are optionally frozen and only freed at the end of the transaction. Then ABEND and other Dumps contain all the storages relevant to the transaction.

The storage freeze is activated/deactivated by the trans. CSFE.

CSFE DEBUG TRANID=... | TASKREQ=... STGFRZ=ON | OFF

TRANID or TASKREQ as specified in PCT (PA/PF keys) but not LPA or OPID.

Suspension Timeout of tasks

This is the deadlock or suspension time out that may be caused for example due to ENQ on a busy resource, Temporary Storage full, SOS etc. The abend code is AKCT, like that for terminal read timeout.

DFHPCT TRANSID=...,DTIMOUT=...,RTIMOUT=.....

Timing Trace

A timing trace entry is added. This entry is created automatically approximately every 128 trace entries. This enables the correlation of various subsystem traces and dumps (for example CICS and VTAM).

More trace entries are now decoded. Program Debugging Guide gives more information on this.

A new subfield in the trace entry identifies the request type (for eg. user, FE or LIFO module).

Additional trace entries generated by the management modules using the LIFO storage scheme, indicate the beginning and the completion of the request to the module.

On an ABEND or program check processing in the System Recovery Program a new trace entry is made.

9.11.8 DUMP DATA PLACEMENT AND PACKAGING

The CICS Transaction Dump and Formatted Dump facilities now direct their output to the same data set. Moreover variable length record format is used to reduce the dump data set space requirements.

The dump datasets DFHDMPA / B can be used in any order.

The dump utility program prints dumps on both datasets.

9.11.9 USEABILITY AND TUNING

Logoff by VTAM LU

On a 'logoff' from CICS/VS by a VTAM LU the LU status is left unchanged. This means that the LU can logon again to CICS/VS without intervention from the master terminal operator.

The LU is not set to RECEIVE status.

Format of sign off is :
CSSF LOGOFF

This applies to BTAM switched line connections. This type of logoff causes a DFHRC TYPE=DISCONNECT to be issued by CICS. The switched line connection is disconnected. Some users may have already written transactions to do such a disconnect.

Upper case Translation

This is now available to all VTAM LU's instead of only 3270's.

END statement in TABLE generation

It is no longer mandatory to specify the entry point name on the END statement.

Conditional ENQ

Transactions have now the option not to lose control when the requested resource is already in use.

```
DFHRC TYPE=ENQ,QARGADDR=...,COND=YES|NO default is NO
```

```
EXEC CICS HANDLE CONDITION ENQBUSY(label) ...
```

```
EXEC CICS ENQ ...
```

ETB support for BSC Devices

ETB (end of transmission block), is now supported on certain BSC devices (like 2770,2780). BSC Devices which can differentiate between ETB and ETX (end of transmission text) can obtain performance improvements by the use of ETB.

Journal STARTIO option

In the Journal Control Table a new option specifies whether Start I/O is to be forced or not. NOTE that the default is to force start I/O.

Simplified Table Generation

PCT / PPT GROUP STATEMENT

The GROUP statement generates a complete set of transaction or prog. entries in PCT and PPT, with all their attributes suitably defaulted.

```
DFHPCT / DFHPPT TYPE=GROUP,  
FN=STANDARD|ISC|EDF|VTAM|VTAMPRT|FE|CONSOLE|TIME  
ATP|AKP|AUTOSTAT|MSWITCH|BMS|HARDCOPY|MASTTERM  
JOURNAL|SOGNON|...
```

INDEX OPTION

The INDEX option replaces the 'page index' option for the PCT, PPT and DCT. It provides an efficient access to large tables and removes the user responsibility to sort tables into alphabetic order. Entries can now be sequenced for efficiency and or convenience.

9.11.10 SYNCPOINT ROLLBACK

Rollback is an extension to syncpoint facility. Application program can use this to request the cancellation of the changes it has made to all recoverable resources. These resources are restored by CICS to the state they were in at the previous syncpoint or task initiation.

The facility is available to the application programmer in the Macro level and Command level.

```
DFHSP TYPE=ROLLBACK,...
```

```
EXEC CICS SYNCPOINT ROLLBACK,...
```

9.11.11 ISC MESSAGE SWITCHING SAMPLE APPLICATION

This sample application is supplied with CICS/VS, and helps to illustrate / verify the ISC function in the installation. It can send a message from a terminal operator in one CICS system in the network to a specified destination in another CICS system in the network. The msg is sent on the ISC link and uses the ISC function of CICS/VS 1.4. The application is coded in PL/I and uses the command level interface. The receiving destination may be a terminal.

More information is given in the installation documentation.

9.11.12 CICS/DOS/VS ONLY

HLPI for RPG II

Access to remote DL/I Database using DL/I DOS/VS. The Database may be a DL/I DOS/VS or IMS/VS-DL/I Database.
DL/I DOS/VS Call Enhancements

Tunable Short Wait

User can now override the default wait time used by CICS/DOS/VS to regain control from DOS/VS to check for the completion of disk I/O events. This SIT operand ICVSWT can be overridden at start-up time.

9.11.13 COMPATIBILITY

CICS/VS 1.4 is source compatible with previous releases of V1 at the API Macro or Command level, with the following exceptions :

The TIOA prefix parameter is now valid for Assembler programs. Previously this parameter was ignored.

Object level compatibility is assured to command level programs.

Additionally object level compatibility exists for macro level programs between 1.3 and 1.4 at the level of macros and control blocks as described in the APRM Macro Level.

9.12 DOS/VS SUPERVISOR GENERATION OPTIONS

The following is a breakdown of the mandatory and suggested Supervisor parameters within the Supervisor generation macros :

MACRO	PARAMETER	REMARKS
SUPVR	AP=YES	Multitasking - for ATTACH macro use. Required for multitasking if journalling or asynchronous loader is used, or if TP=VTAM is specified. AP=YES forces WAITM=YES in FOPT macro
	NPARTS=2+ PAGEIN=n TP=a/method	At least 2 partitions must be specified Must be greater/equal to 1. BTAM or VTAM or (BTAM,VTAM) Specification of TP=VTAM forces following parameters for FOPT : AB=YES OC=YES PC=YES IT=YES TOD=YES RELLDR=YES PPIX=YES ECPREAL=YES GETVIS=YES WAITM=YES
STDJC	SPARM=YES	Required for &SYSPARM for BMS map assemblies
	EDECK=YES	For maintenance of macros.
	ALIGN=YES	CICS/VS requires data alignment

-----Contd.-----

FOPT	WAITM=YES PCIL=YES IDRA=YES FASTTR=YES RPS=YES VSAM=YES AB=YES OC=YES PC=YES IT=YES TOD=YES RELDR=YES PFIX=YES GETVIS=YES SYSFIL=YES	Forced by TP=VTAM Reqd. if the starter system or sample sequential terminal job stream is used. optional. If AP=NO in SUPVR, IDRA=YES reduces the wait time for CICS partition when a FETCH or LOAD macro is issued by CICS Only if VSAM is the main access method If it is to be used in CICS. For CICS ISAM files also specify RPS=SVA in DFHFCT TYPE=LOGICMOD, in FCT. For aux. temporary storage, VSAM and DL/I. If system recovery table or DL/I are used for CICS console support for system recovery and system init. pgms. for task control program for interval control and keypoint pgm. for loading and relocating all CICS modules reqd. by CICS journaling, BTAM or CICS progs. defined FIX=YES in NLT/ALT or RES=FIX in PPT. Forced by TP=VTAM, and VSAM=YES. Reqd. if parameters passed to trace utility pgm. Reqd. for working storage of formatted dump and dump utility programs, unless sufficient storage allocated in DFHFDP, DFHDUP. Initial allocations 4k and 8k resp. reqd. if DASD used as intermediate storage for HLPI translator output
VSTAB	SVA=(nK,mK)	Where n = 302K (for VSAM) + 70K for DL/I
IOTAB	NRES=10	Forced by VSAM - 10 is the minimum value

9.13 INSTALLATION VERIFICATION & MIGRATION CONSIDERATIONS9.13.1 TRANSFERRING TO CICS/DOS/V5 1.4 GENERAL CONSIDERATIONS

CICS/DOS/V5 Version 1 Release 4.0 is source code compatible with all previous releases of Version 1 at the application program interface (i.e. macro level), with minor exceptions noted below as conversion considerations. This means that application programs developed for previous releases need, at most, to be reassembled or recompiled to operate with Version 1 Release 4.0.

CICS/DOS/V5 Version 1 Release 4.0 is object code compatible with Releases 1.0, 1.1, 2.0 and 3.0 of Version 1 at the level of the macros and control blocks as described in the Application Programmer's Referen Manual (SC33-0056) for Releases prior to 4.0. Object code compatibility cannot be guaranteed for those programs that use facilities outside those described in the above manual.

Note: All tables from previous releases of CICS should be reassembled against CICS/DOS/V5 Version 1 Release 4.0

Reassembly of Application Programss

- All programs assembled before Version 1 Release 1.0, (strictly a those which address the TCTTE), must be reassembled for release 1.4 This is due to the change in offsets of the TCTTE control block, notably TCTTEDA. This will therefore include all applications using BMS.
- 3767. The Interactive Logical Unit can execute applications written for 2740/2741 terminals, provided that BMS is used, and that applications do not make use of the Attention and Interrupt features of the device. A new BMS map will be needed.

- 3770. Applications written for present batch devices can use the Batch Logical Unit if they make use of BMS and Logical Record Presentation, and do not make use of the Interrupt and Attention features of the device. A BMS map re-assembly will be necessary to include the logical device code.

9.13.2 INSTALLATION CONSIDERATIONS

This section contains additional information on CICS/DOS/VS Version Release 4 Modification Level 0.

C-TYPE RPG-II APPLICATIONS

C-type RPG -II applications, that is those which have CICS without DL/I, will give an unresolved external reference for DLZLI000 if Standard Data Transfer is used and DL/I is not installed. This unresolved reference should be ignored.

THE EXECUTION DIAGNOSTIC FACILITY

The Execution Diagnostic Facility (EDF) programs cannot be generate using the CICS/DOS/VS sysgen process. The EDF programs are supplied on the pregenerated full function core image library (A5746XX3.STARTER.R14 Therefore, if a user is not intending to use the starter system but wants to use EDF, he must install the starter system temporarily and copy the EDF programs to his CICS/DOS/VS private or system core image library.

The EDF programs are:

DFHEDFP DFHEDFX DFHEDFD DFHEDFM DFHEDFF DFHEDFR

In order to use EDF in two-terminal mode the EDF display terminal must have TRANSCEIVE status. Users of ELS must, therefore, use the Master Terminal facility to alter the status of the EDF display terminal from TRANSACTION (the default) to TRANSCEIVE before using EDF in two-terminal mode.

INTERSYSTEMS COMMUNICATION SAMPLE APPLICATIONS

Information on the ISC message routing sample application program (in PL/I) supplied in the starter system, is given in the Systems Programmer's Guide, Appendix C.

SYSGEN CHANGES

DFHSG will allow a suffix to be coded on the PROGRAM=TBP macro, as as the PROGRAM=DBP. It is also possible to override the value of DLI= coded on the TYPE=INITIAL macro by coding DLI= on the PROGRAM= and PROGRAM=TBP macros. Both of these facilities were used when th starter system was generated (the stage 1 is in A.DFHSPSG in the primary source library).

SAMPLE APPLICATIONS

The sample programs supplied with the starter system and the operating information to execute them is given in the Systems Programmer's Guide Appendix C.

SAMPLE TABLES

Although ELS users do not have to code a System Initialisation Tabl (SIT), there is a SIT in the ELS core image library which is used during an ELS start up. It is called DFHSIT on the ELS core image library. The source is in Z.DFHSIT1# on the primary source library and it is called DFHSIT1# in the microfiche.

Sign on tables are not suffixable. The sample SNTs on the ELS and full function libraries are different in content but have the same name (ie DFHSNT). The ELS version is called DFHSNT1# in the source library and microfiche. The full function version is called DFHSNT1 in the source library and microfiche.

All source books of the sytems initialisation tables supplied with the starter system are in Z.source book with the same name as the table in question.

DFHGEN

It is important to remember that DFHGEN does not produce JCL for private libraries, ie. it does not produce DLBL, EXTENT and ASSIGN cards for any private libraries being used. Any private libraries should therefore be permanently assigned before running jobs produced by DFHGEN.

TRACE ENTRIES

The internal format of some CICS trace entries has been rearranged. Users who have their own auxiliary trace decoding programs should check their code.

INITIALISATION PROGRAMS

Register usage has been changed in the initialisation programs DFHSIP and DFHSIA1 - DFHSIJ1. Users who write their own initialisation overlay programs should check their code. If DFHSICOM is used to pick up register equates etc, then reassembly of the overlay will probably be sufficient to allow for the change. As a result of this change initialisation overlays have 8K bytes of addressability and the statement USING •, SIPBAR2, SIPBAR3 may be included at entry point.

VSAM INTRAPARTITION TRANSIENT DATA

1. Sysgen of TDP - Transient Data Program

1) DFHSG PROGRAM=TDP

Specify INTRA=VSAM to indicate that the intrapartition queues are to be used with VSAM. This option cannot be used when DAM or BDAM is specified.

2) DCT - Destination Control Table

DEVICE=device specifies the type of device on which the DAM intrapartition data set resides. The default is DEVICE=2314. This operand replaces the DEVICE operand that was formerly in DFHSG PROGRAM=TDP.

NOTE: If DFHSG PROGRAM=TDP is generated with INTRA=VSAM, the device type specified in this operand will be ignored at execution time and need not be specified. The default of DEVICE=2314 will nevertheless be applied and the resulting MNOTE can be ignored.

2. DOS/VS Supervisor Generation

FOPT VSAM=YES is required if the intrapartition transient data set defined as a VSAM file.

3. System Initialisation

The following additional messages may appear on the system console depending on the message level setting the System Initialisation Table.

DFH1500 - OPENING INTRAPARTITION ACB
DFH1500 - INITIALIZING INTRAPARTITION STORAGE

4. Designing an Efficient CICS/VS System

DFHSG PROGRAM=TDP,
INTRA=VSAM

This operand implies the use of VSAM to provide the Intrapartition Transient Data feature. While this will normally give substantially better performance than the DAM version, some additional storage will be required to accommodate the VSAM control interval. If VSAM is not required for other features allowance must be made for the additional storage required.

5. Evaluation of the CICS/VS System's Performance

DFHSG PROGRAM=TDP,
INTRA=VSAM

the VSAM version of the Transient Data program provides significant improvements over the DAM version,

especially when automatic transa initiation is used. The improvements result from the reduced number of physical I/O operations taking place, and reduced processor activity. Another significant saving over the DAM version arises because VSA does not require the CICS/VS partition to wait while an output operation takes place. (For DAM, as used by Transient Data, a system wait is issued following a PUT, thus preventing the CICS/VS partition from doing work until the wait is satisfied.)

6. Transient Data Intrapartition Data Set

The Transient Data intrapartition data set may be either a VSAM ESD or direct access (DAM) data set, and is used for queuing messages and data within the system. The DTF for the intrapartition data set is automatically generated in the DCT, and is opened by system initialisation if intrapartition queues are being used.

If the Transient Data Program to be used was generated with INTRA=DAM (the default), the intrapartition data set must be direct access. Only one of the access methods supported can be used for intrapartition transient data during a particular execution of CICS/DOS/VS, since the choice between the two is made by selecting a particular TDP when CICS/VS is initialised.

VSAM Intrapartition Data Set

The Systems Programmer's Guide chapter 1.10 has more information on the subject.

9.13.3 MIGRATION CONSIDERATIONSCONVERSION FROM VERSION 1 RELEASE 3.0 TO RELEASE 4.0

The following considerations apply:-

- In CICS/VS Release 1.4 the Formatted Dump Program writes dumps to the dump data set using the dump control program.
 - In some cases the dump data set may need to be enlarged.
 - It will no longer be necessary to assign SYSLSST to the CICS/DOS/VS partition in order to use the Formatted Dump Program.
 - A SIZE option should be specified when executing DFHDUP if GETVIS support is available in the DOS/VS supervisor. SIZE=80K is sufficient.
 - If GETVIS support is not included then the initial allocation of working storage can be increased in DFHDUP by changing the field DUWEAREA.
- A 'DEVICE=' parameter card may be supplied in the jobstream for DFHDUP, DFHTUP and DFHSTUP. If this is omitted it must be specified on the console.
- The meaning of the SIT override parameter FDP=(,SNAP) has been changed. FDP=(,PARTN) provides an equivalent function.
- The 'short wait interval' can now be changed by the user. The ICVSWT SIT operand or start-up override controls the length of time CICS/DOS/VS will wait before testing to see if disk I/O is complete.
- Several Release 3.0 sysgen operands are now available as operands on the SIT or as start-up overrides. For compatibility reasons they can still be specified at sysgen time.

They are:-

```
NSD          on DFHSG PROGRAM=CSO
WRKAREA      "    "    PROGRAM=CSA
DATFORM      "    "    "    "
```

- The following sysgen operands have been removed and will be ignored if encountered.

```
DEVICE=      on DFHSG PROGRAM=CSU1
TRDEV=       "    "    "    " 1
TLT=         "    "    "    =MTP2
DEVICE=       "    "    "    =TDP3
PRTREQ=      "    "    "    =TCP4
```

Notes:

1. The device type must be specified as SYSIPT input to the appropriate utility or as an answer by the system operator to a question from the utility.
 2. TLT support is always included.
 3. DEVICE= must be coded on the DFHDCT TYPE=INITIAL macro if DAM INTRAPARTITION is used.
 4. PRTREQ as a sysgen operand has been replaced by PRINT as a SIT operand or as a start-up override.
- The CICS/DOS/VS starter system contents have been changed slightly. A complete list of the changes since Release 3.0 can be found in the comments in the stage 1 used to generate the starter system. The stage 1 is in A.DFHSPSG.
 - The SUBSET has been discontinued.
 - The Entry Level System (ELS) has been introduced to replace it. Major differences are:
 - Terminal support is for the 3270 family of terminals only.
 - Terminal support can be via BTAM or VTAM.
 - File control via DAM is not supported.
 - ELS supports Auxiliary Trace and Intrapartition Transient Data.

- ELS is completely pregenerated and so no sysgen step is necessary.
- ELS supports only the command level application interface.
- Simplified high level start overrides are provided. A DFHSIT table does not have to be prepared.

For more information about ELS please refer to:-

- CICS/VS General Information (GC33-0066-2)
- System/Application Design Guide (SC33-0068-1)
- Entry Level System User's Guide (SC33-0086-0)
- Applications and maps written for the 3270 group of terminals do NOT need to be changed in order to run on terminals which have the alternate screen size facility. However, anyone using these terminals should consult the information concerning them in the CICS/VS manuals. New operands on the DFHTCT and DFHPCT table macros allow user control of the alternate screen size facility.
- There are changes to the CICS/DOS/VS PCT,PPT,DCT macros. On their TYPE=IN macros the PAGENXD keyword has been replaced by the INDEX keyword. On the PCT TYPE=ENTRY macro the RESTART keyword has been added. TYPE=GROUP is added to PCT and PPT. Existing source for all these tables will still assemble validly, but occurrences of PAGENX will be re-interpreted as INDEX=YES.
- VSAM may be selected for intrapartition transient data support. In order to obtain VSAM intrapartition TD:
 - Select VSAM TDP at sysgen if you are not using the pregenerated VSAM TDP.
 - Before the first COLD CICS startup using VSAM TDP, run a job to define the VSAM cluster for DFHNTRA.
 - Use appropriate DLBL and EXTENT statements for a VSAM DFHNTRA in CICS startup jobstream.

- Note that DCTs assembled for Release 4.0 are independent of whether VSAM or DAM is used for intrapartition TD.

CICS/DOS/VS AND DOS/VSE MIGRATION PLANNING

The Valid Combinations of CICS/DOS/VS (1.4.0 Base and 1.4.0 ICR) and DOS/VSE R 35 are as follows :

CICS/DOS/VS	* DOS/VS * * Rel 34 *	* DOS/VSE Rel 35 * * CKD dev * * FBM dev *
1.4.0 Base (1)	* Yes *	* Yes (2) * * NO *
1.4.0 ICR (3)	* No (4) *	* Yes (5) *

Notes:

(1)-All pre-assembled supplied modules are assembled on DOS/VS Rel 34. Applies for all maintenance PTFs for the base release.

(2)

(a)-DFHTCP and DFHTCT must be re-assembled (BTAM-ES must be used) after installing DOS/VSE-SCP, VSE/AF, BTAM-ES, ACF/VTAM or VTAME, VSE/VSAM, etc.

(b)-Migrate from (DOS/VS Rel 34 + CICS/VS 1.4.0 base) to (DOS/VSE Rel 35 + CICS/VS 1.4.0 ICR) in one step (position corresponding to 5). If you cannot manage then use this as the intermediate step.

(3)-All pre-assembled supplied modules are assembled on DOS/VSE Rel 35. Applies for all maintenance PTFs for the ICR release.

(4)

(a)- NOT RECOMMENDED. THIS NOT A VALID COMBINATION

(b)- You dont need this combination, because makes no sense.

(c)- Correct intermediate step for migration is CICS140 base + DOS/VSE Rel 35 (step 2b).

(5)-Should be your target, if you plan to use DOS/VSE or CICS/VS 1.4.0 ICR in your installation.

9.14 FIXED BLOCK ARCHITECTURE

The description below applies to the IBM 3310 and IBM 3370 FBM DASD.

9.14.1 OVERVIEW

FBM is a new DASD architecture in which the main difference to Count-Key-Data resides not in how to address the device but in the addressing within the device.

FBM appears to the programming as comprising a contiguously numbered set of physical records, starting from zero and going to the device capacity.

The size of each physical record is constant and is a multiple of 512 bytes. In the current implementation is 512 only.

Channel programs may read or write single or multiple consecutive records at each request.

There is no hardware implementation of a key-search capability.

The Access Methods providing explicit support for FBM devices are VSAM and SAM.

The CICS/VS support of FBM devices coming in CICS/VS V1 R4.1 has the objective of providing most of the functions for datasets on FBM devices.

The following areas of CICS/VS are affected in order to provide support for FBM devices:

- File Control
- Temporary Storage
- Journalling
- Transient Data
- Terminal Control
- Dump Control
- Auxiliary Trace
- Auto-Statistics
- Initialization / Termination
- Keypointing

9.14.2 DESCRIPTION OF SUPPORTED FUNCTIONS

FILE-CONTROL

The File Control Program will have no changes caused by the FBM support. Datasets residing on CKD-devices will continue to be supported by VSAM, ISAM, or DAM as in the current CICS/VS release. Datasets residing on FBM-devices will be supported only by VSAM.

TEMPORARY STORAGE

FBM support will be transparent to CICS/VS since Temporary Storage uses VSAM when auxiliary storage is requested.

JOURNALLING

Journal-control will be modified to support disk journals on FBM devices through the regular SAM workfile interfaces.

The corresponding specification in JCT is JDEVICE=FBA.

TRANSIENT-DATA

CICS/VS 1.4.1 will support both Extra and Intrapartition destinations on FBM devices.

DEVICE=FBA will be accepted in the DFHDCT macro for extrapartition destinations.

The support for intrapartition destinations on FBM devices is given via VSAM only.

The following must be specified in the generation of Transient Data Program if intrapartition destinations are to be resident on FBM devices.

```
DFHSG PROGRAM=TDP,INTRA=(...,VSAM),...
```

Internally, when VSAM is chosen, intrapartition transient data PUT requests will be satisfied by the allocation of space from the current control-interval for the specified destination and the chaining together of such control-intervals. This is analogous with the current DAM approach, which allocates space from the current track for the specified destination, and chains together tracks to form queues. Transient-Data Recovery, Acquisition, Disposition, and Initialization functions will support this option.

TERMINAL-CONTROL

Simulated terminals using FBM devices will be supported as they are currently supported using CKD devices.

Generation of the Terminal Control Table will allow FBM in the DEVICE= operand

```
DFHTCT TYPE=SDSCI,....,DEVICE=FBA,...
```

DUMP-CONTROL

DFHDCP and DFHDUP will support dump datasets residing on FBM devices. FBM will be valid in the DEVICE= operands of the following macros:

```
DFHSG PROGRAM=CSU,....,DEVICE=FBA  
DFHSG PROGRAM=DCP,....,DEVICE=FBA
```

and also in the Operator dialogue when executing the Dump Utility Program.

A dynamic test for DTF device-type will be made before entering end-of-extent processing, and, for non-FBM, the current code will be executed; for FBM, new code will avoid writing outside of the extent.

AUXILIARY TRACE

DFHTRP and DFHTUP will support Trace datasets residing on FBM devices. The following are the correspondent specifications:

```
DFHSG PROGRAM=CSU,....,DEVICE=FBA  
DFHSG PROGRAM=TRP,....,DEVICE=FBA
```

FBM will also be accepted in the start-up parameters when

executing the Trace Utility Program.

DFHTRP uses contents of the DTFSD control-block to anticipate end-of-extent, and the code will be modified in the same way as in Dump Control Program.

STATISTICS

No modification of the online modules will be made in order to support FBM. The only difference will be in the offline printing of statistics where FBM (besides of TAPE, 2314, 3330 and 3340) will be accepted as a reply to message:

ENTER INPUT DEVICE: TAPE, 2314,

if the statistics summarization utility program is generated to be used for more than one input device type.

WARM KEYPOINTING AND EMERGENCY RESTART

The Restart Dataset is used during Warm Keypointing and during the Emergency Restart. In the current implementation of CICS/VS (CICS/VS 1.4.0) Restart Dataset is a DAM dataset. In order to support Restart Dataset residing on FBM device it will be modified in order to use SAM workfile.

Although CICS/OS/VS will not support FBM at 1.4.1 level, it will have also the Restart Dataset modified to SAM in the interest of consistency.

FBA will be valid in the DEVICE= operand when generating Keypoint Program.

9.14.3 IMPACT ON USERS

IMPACT ON APPLICATION PROGRAMMER

Since the only existing support for user databases residing on FBM is via VSAM interface, application programs which use macros or command-level requests containing parameters peculiar to DAM will have to be modified.

IMPACT ON SYSTEM PROGRAMMERS

The impact on system programmer is minimum, in order to get FBM support:

- a) Check for VSAM option in DFHSG PROGRAM=TDP
- b) Check for FBA option in the following macros:

```
DFHSG TYPE=INITIAL,...,DEVICE=  
DFHSG PROGRAM=CSU,...,DEVICE=  
DFHSG PROGRAM=DCP,...,DEVICE=  
DFHSG PROGRAM=TRP,...,DEVICE=  
DFHSG PROGRAM=KPP,...,DEVICE=  
DFHTCT TYPE=SDSCI,...,DEVICE=  
DFHDCT TYPE=SDSCI,...,DEVICE=  
DFHJCT TYPE=ENTRY,...,JDEVICE=  
DFHSG PROGRAM=CSU,...,TRDEV=
```

IMPACT ON COMPUTER OPERATOR

Job Control statements for all DASD datasets residents on FBM-devices are affected. Changes are the following:

- 1) Intrapartition transient dataset, DFHNTRA, may be optionally a VSAM dataset.

- 2) The Restart dataset, DFHRSD, must be defined as sequential dataset (SD instead of DA in the DLBL card of JCL).
- 3) When executing the Trace, Dump and the Statistics Utility Program use of FBM in device-specification in JCL and operator dialogues.

9.14.4 COMPATIBILITY/ MIGRATION ASPECTS

- The FBA specification will be accepted as previously stated. Some degree of device-independence will however be provided by DOS/VS SAM allowing the user in many cases, to maintain the current specification (2314, 3330, 3340) when moving to a FBM device. A 2314 specification, for example, with a blocksize of 7294 bytes (or fewer) will be accepted when the dataset is assigned to a FBM device (DOS/VS will change from 2314 to FBM at OPEN time). If, however, the blocksize has to be increased to more than 7294, then due to access method restrictions the 2314 specification will not be accepted any longer, and the 2314 specifications has to be changed to FBM.
- An Intrapartition Dataset created by DAM implementation of Transient Data Program will not be usable under the VSAM option of TDP or vice-versa.
- The Asynchronous Transaction Program (ATP) will execute only with the DAM option of Transient Data Program.

9.15 CICS/VS 1.4 BIBLIOGRAPHY

PROMOTIONAL MATERIAL

GV20-0643 DB/DC Program Product Extensions
G520-2725 CICS/VS Program Product Flyer
GH20-4496 CICS/VS Program Product Design Objectives
GC33-0084 CICS/VS Program Product Specifications

UNLICENSED DOCUMENTATION

GC33-0066 General Information Manual
SC33-0067 Introduction to Program Logic
SC33-0068 System/Application Design Guide
SC33-0069 System Programmer's Reference Manual
SC33-0070 System Programmer's Guide (DOS/VS)

SC33-0071 System Programmer's Guide (OS/VS)
SC33-0072 IBM 3600/3630 Guide
SC33-0073 IBM 3650 Guide
SC33-0074 IBM 3767/3770 Guide
SC33-0075 IBM 3790 Guide

SC33-0077 Application Programmer's Reference Manual
(Command Level)
SC33-0079 Application Programmer's Reference Manual
(Macro Level)
SC33-0080 Operator's Guide
SC33-0081 Messages and Codes Manual
SC33-0082 Entry Level System User's Guide
SC33-0089 Problem Determination Guide
SX33-6010 Reference Summary - Program Debugging
SX33-6011 Reference Summary - Master Terminal Operator
GX33-6012 Application Programmer's Reference Summary
(Command Level)

LICENSED DOCUMENTATION

LY33-6028 Program Logic Manual CICS/DOS/VS V1 R4
LY33-6029 Program Logic Manual CICS/OS/VS V1 R4

LICENSED OPTIONAL MICROFICHE LISTINGS

LYA4-3003-0 Microfiche Listings CICS/DOS/VS (use feature 7050)
LYA4-3004-0 Microfiche Listings CICS/OS/VS

CHAPTER 10

PROGRAMMING FACILITIES

DOS/VS COBOL Compiler and Library	(5746-CB1)
DOS PL/I Optimizing Compiler and Libraries	(5746-PL3)
DOS/VS RPG II	(5746-RG1)
DOS FORTRAN IV Library Option 1	(5746-LM3)
DMS/CICS/VS-DOS	(5746-XC4)
DOS/VS SORT/MERGE Version 2	(5746-SM2)

CONTENTS

10.1 Introduction. 10.1

10.2 Programming Facilities Described. 10.2

10.3 DOS/VS COBOL and Library (5746-CB1) 10.3

10.3.1 Product Function. 10.4

10.3.2 Product Use 10.5

 DOS/VS COBOL 2.5 Dependency Information Chart. 10.6

 Supervisor Generation Options. 10.7

 VSE/VSAM Support 10.8

10.3.3 Product Reference Material. 10.9

10.4 DOS PL/I Optimizing Compiler and Libraries
(5746-PL3). 10.10

10.4.1 Product Function. 10.11

10.4.2 Product Use 10.12

 PL/I Dependency Information Chart. 10.13

 Supervisor Generation Options. 10.14

 VSE/VSAM Support 10.15

10.4.3 Product Reference Material. 10.16

10.5 DOS/VS RPG II (5746-RG1). 10.17

10.5.1 Summary of Releases 10.18

10.5.2 Product Function. 10.20

 Objectives 10.20

 User Requirements. 10.21

 Online Operating Modes 10.26

 Screen Formats 10.28

 Native Language Texts. 10.29

10.5.3 Product Use. 10.30

 FBA Support. 10.30

 RPG-II Source Entry Facility (RSEF) Commands 10.30

 DOS/VS RPG-II Compiler 10.45

 DMS/CMS CALL Interface support 10.53

 VM/IFS CALL Interface Support. 10.61

10.5.4 Dependencies. 10.64

 Hardware and Software. 10.64

 Compatibility With Previous Releases 10.65

 DOS/VS RPG II Dependency Information Chart 10.66

 Supervisor Generation Options. 10.68

10.5.5 Examples of Use 10.69

 Command Menu Example 10.69

 EDIT Mode Display. 10.70

 INPUT/ALTER Display Examples 10.71

10.5.6 Product Reference Material. 10.74

10.6 FORTRAN IV Library Option 1 (5746-LM3)	10.76
10.6.1 Product Function.	10.77
10.6.2 Product Use	10.78
DOS FORTRAN IV Library Option 1.	10.79
Supervisor Generation Options.	10.80
10.6.3 Product Reference Material.	10.81
10.7 DMS/CICS/VS-DOS (5746-XC4)	10.82
10.7.1 Summary of Releases	10.83
DMS/VS-DOS Line Items.	10.84
DMS/CICS/VS-DOS Line Items	10.85
10.7.2 Product Function.	10.86
10.7.3 Product Use	10.88
Panel Description Table.	10.88
File/Database Support.	10.90
Terminal Support	10.95
Data Management Support.	10.96
User Written Routines.	10.98
DMS/CICS/VS Transactions	10.101
DMS/CICS/VS Offline Components	10.101
Minimum Configuration.	10.102
DMS DOS/VS Dependency Information Chart.	10.104
Supervisor Generation Options.	10.106
DMS/CICS/VS Terminal Support	10.107
10.7.4 DMS/VS-DOS Facilities in Detail	10.108
DMS/VS-DOS V1 M1 PTF0001	10.108
DMS/VS-DOS V1 M1 PTF0002	10.110
10.7.5 DMS/CICS/VS-DOS Facilities in Detail.	10.112
DMS/CICS/VS-DOS Release 1.	10.112
DMS/CICS/VS-DOS Release 2.	10.115
DMS/CICS/VS-DOS Release 3.	10.117
10.7.6 Product Reference Material.	10.120
DMS/VS References.	10.120
DMS/CICS/VS References	10.122
10.7.7 Appendix A.	10.123
RPG-II exit coding example	10.123
10.8 DOS/VS Sort/Merge Version 2 (5746-SM2)	10.126
10.8.1 Product Function.	10.127
ECPS:VSE Support	10.127
/370 Support	10.127
10.8.2 FBM Support	10.128
Input Files.	10.128
Output Files	10.128
Checkpoint Files	10.129
Work Files	10.129
SYSFIL Support	10.129
Record and Block Limitations	10.129
10.8.3 Variable-length Spanned Records	10.130
10.8.4 Usability Improvements.	10.131
Print File Routing Option.	10.131
Overlapping Control Fields	10.131
OUTREC Fields beyond Byte 4092	10.131

Improved Diagnostics10.131
 Improved Syntax and Error Handling10.131
 Improved File Handling10.132
 Control Statement Printout10.132
 The ANALYZE Statement.10.132
 10.8.5 Generation and Installation Options10.134
 10.8.6 Command Chaining for Input and Output10.134
 10.8.7 Hardware Configuration.10.135
 Input/output Devices10.135
 10.8.8 Software Configuration.10.135
 10.8.9 Storage Requirements.10.136
 FBM Support.10.136
 4300 processor Support10.136
 SVA Requirements10.136
 10.8.10 Performance Considerations10.137
 10.8.11 Compatibility.10.137
 10.8.12 Summary of SORT/MERGE Dependencies10.138
 10.8.13 Product Reference Material10.139

10.1 INTRODUCTION

This chapter contains a summary of details for the major programming facilities available to the DOS/VSE user. These consist of the basic language compilers (DOS/VS COBOL, DOS PL/I Optimizing Compiler and Libraries, DOS/VS RPG II and DOS FORTRAN IV Library) and the major productivity aid DMS/CICS/VS.

The information provided is as detailed and accurate as is the latest documentation regarding Functional Specifications and other literature received by WTSC Boeblingen. It is important to consider the schedules for the different products when using the information available in this guide. For instance, the most important DMS/CICS/VS-DOS enhancements to DMS/VS-DOS are incorporated in Release 2.0 planned to be available by end 1979. Then, detailed and stable information is not available for this product at this time frame.

Please notice that WTSC Boeblingen support for COBOL and PL/I has been defined as 'limited support'.

FORTRAN is not supported by WTSC Boeblingen.

A discussion of the new SORT/MERGE Version 2 product, will also be included here for convenience.

10.3 DOS/VS COBOL AND LIBRARY (5746-CB1)

Status : Program Product

Latest Release : 2.5

Announced : 1/79

Planned Availability : 4/79

Manuals Available : 4/79

SIPO's Available : DOS/VSE SIPO/E (Plan.Avail.3Q79)
VM/DOS/VS SIPO/E (Plan.Avail.4Q79)

Selected Product Line : yes

Installation License : N/A

Pre-Installation Test : 30 days

Program Services : Central Service until discontinued
upon 12 months written notice.

Local Service at no charge until
12/31/79 then Local Licensed
Program Support.

10.3.1 PRODUCT FUNCTION

This modification to Release 2 of the DOS/VS COBOL Compiler and Library will give users of DOS/VSE with VSE/Advanced Function Release 1 access to Fixed Block disk devices through DOS/VS COBOL. The rest of the functions have not changed from the Release 2.4. For instance, the VSAM support is equivalent to that available with DOS/VS Release 28.

The compiler minimum storage requirement is increased to 64K bytes of storage.

10.3.2 PRODUCT USE

The following important points should be noted :

- (i) DOS/VS COBOL 2.5 supports Fixed Block disk devices for SAM and VSAM files, and for compiler work files.

VSE/VSAM is required for VSAM file support. The specifications for VSAM files are not changed.

The specification for SAM files is as follows (notice the forced combination -UT-FBA1-S-):

```
SELECT filename ASSIGN TO SYSnnn-UT-FBA1-S-dblfilename.
```

- (ii) The compiler requires a minimum of 64k bytes of storage. The working files defined on FBA devices are specified by means of JCL statements.

- (iii) DOS/VS COBOL 2.5 does not support all of the functions included in VSE/VSAM. A list showing what is supported is illustrated in a section below .

- (iv) The VM/CMS user who has Display Management System/CMS (DMS/CMS) and VM/Interactive File Sharing (VM/IFS) facilities intalled on his system may also utilize the corresponding CALL interface in his COBOL program.

The formats of the CALL parameters and data areas used to invoke DMS/CMS and VM/IFS services are not described here. They have the same meanings and equivalent layout as described for RPG-II in this guide.

DOS/VS COBOL 2.5 DEPENDENCY INFORMATION CHART

DEPENDENCY ITEM	REMARKS
Program Prerequisites - SORT/MERGE	Required if application program requires the SORT facility - any supported DOS/VSE SORT/MERGE product.
	Unit Record Support. Any DOS/VS supported device. At least a card reader (or equivalent) and line printer (or equivalent) are required. Also supported are - 3504 with OMR, 3881 OMR, 3886 OCR, 3540.
Tape Support	Any DOS/VS supported device. SYS002 - 004 and SYS005 (optional) work files may be assigned to tape.
DASD Support - 2311 - 2314 - 2319 - 3330-1,2 - 3333 - 3330-11 - 3340 - 3344 - 3350 - 3310,3370	Native mode. Native mode. FBA disk types for SAM and VSAM files ONLY.
Data Management Support. - SAM - DAM - ISAM - VSAM	Not for FBA disk devices. Not for 3330-11, 3350 and FBA disk devices. See notes below for specific support.

SUPERVISOR GENERATION OPTIONS

The following is a breakdown of the mandatory and suggested Supervisor parameters within the Supervisor generation macros :

MACRO	PARAMETER	REMARKS
SUPVR	ASCII=YES	Only if ASCII support needed.
NOTE. The following FOPT parameters, recommended for DOS/VS, are standard under DOS/VSE:		
FOPT	AB=YES	Required
	GETVIS=YES	Forced by VSAM (if required) and required for 3886 OCR support
	PCIL=YES	Support for Private Core Image Lib. required
	RELLDR=YES	Required
	VSAM=YES	If VSAM support required
VSTAB	VSIZE=nK	As required by the system when in 370 mode
IOTAB	NRES=n	As required by VSE/VSAM (from NPARTS+2 to 255)

VSE/VSAM SUPPORT

This section briefly summarizes the VSE/VSAM functions which are available to the COBOL programmer :

(i) Data Set Organization

- VSAM KSDS. Keyed direct and sequential access are both supported, including READ, WRITE, UPDATE and DELETE functions.
- VSAM ESDS. Sequential access is the only supported. READ, WRITE and UPDATE same length record in place are available functions. No DELETE record function exists.

(ii) Password Protection

- READ
- UPDATE

(iii) No other VSE/VSAM functions are supported, but as already indicated, all VSAM Release 1 functions (DOS/VS Release28) are available.

10.3.3 PRODUCT REFERENCE MATERIAL

GC28-6473- General Information Manual
GC28-6487- Program Product Specifications
SC28-6478- Programmer's Guide
SC28-6479- Installation Reference Manual
SC28-6469- CMS Users Guide for COBOL
GC28-6394- DOS Full ANS COBOL
LY28-6423- Compiler Program Logic

10.4 DOS PL/I OPTIMIZING COMPILER AND LIBRARIES (5746-PL3)

Status	:	Program Product
Latest Release	:	5.0
Announced	:	1/79
Planned Availability	:	4/79
Manuals Available	:	4/79
SIPO's Available	:	DOS/VSE SIPO/E (Plan.Avail.3Q79) VM/DOS/VS SIPO/E (Plan.Avail.4Q79)
Selected Product Line	:	yes
Installation License	:	N/A
Pre-Installation Test	:	30 days
Program Services	:	Central Service until discontinued upon 12 months written notice. Local Service at no charge until 12/31/79 then Local Licensed Program Support.

10.4.1 PRODUCT FUNCTION

This enhancement to Release 5 of the DOS PL/I Optimizing Compiler and Libraries will give users of DOS/VSE with VSE/Advanced Function access to Fixed Block disk devices through DOS PL/I.

PL/I is a general purpose programming language which can be used to program both commercial and scientific applications. It is particularly useful for applications that require a combination of techniques to be used in a program.

PL/I support under DOS/VS is provided by the PL/I Optimizing Compiler and by two object-time libraries, the resident and transient libraries. The PL/I Optimizing Compiler is designed to provide optimized object programs from a comprehensive level of PL/I. It provides a high level of PL/I language diagnostics at both compile-time and object-time, and optimized object programs.

If optimization is specified, the compiler will process the PL/I source program, reorganizing it if necessary, so as to produce an efficient object program. If optimization is not specified, compilation time will be reduced.

A facility is provided by the compiler to allow communication between PL/I modules and modules produced by certain FORTRAN, RPG and COBOL compilers. The language level implemented by the optimizing compiler contains extensions beyond the PL/I D and F level subsets.

Two object-time libraries are required for the execution of object modules produced by the optimizing compiler. These libraries contain subroutines which must be combined with the object module to produce an executable program (the PL/I resident library), and other subroutines which are required dynamically as the object program is being executed (the PL/I transient library).

Both the resident and transient libraries are separate Program Products.

10.4.2 PRODUCT USE

The following important points should be noted :

- (i) DOS PL/I Optimizing Compiler 5.0 supports the use of FBA disk device types for SAM and VSAM files, and compiler workfiles. VSE/VSAM is required to access VSAM datasets. The specification for VSAM files has not changed.

The access to SAM files on FBA devices requires an FBA device type specification. This is accomplished by coding the three characters 'FBA' in the 'physical device type' in the MEDIUM option of the ENVIRONMENT attribute. As an example:

```
DCL SAMFILE FILE...ENV (RECZISE (80) MEDIUM (SYS007,FBA)...);
```

The specifications for working files on FBA devices is through JCL statements.

- (ii) The compiler minimum storage requirement is increased to 50K bytes of storage.
- (iii) DOS PL/I Optimizing Compiler 5.0 supports many of the functions available in VSE/VSAM. These are listed in a section below.
- (iv) The VM/CMS user who has Display Management System/CMS (DMS/CMS) and VM/Interactive File Sharing (VM/IFS) facilities intalled on his system may also utilize the corresponding CALL interface in his PL/I program.

The formats of the CALL parameters and data areas used to invoke DMS/CMS and VM/IFS services are not described here. They have the same meanings and equivalent layout as described for RPG-II in this guide.

PL/I DEPENDENCY INFORMATION CHART

DEPENDENCY ITEM	REMARKS
Program Prerequisites	
- SORT/MERGE	Required if application program requires the SORT facility - any supported DOS/VS SORT/MERGE product.
Unit Record Support. Any DOS/VS supported device. At least a card reader (or equivalent) and a line printer (or equivalent) are required. Also supported are - 3881 OMR, 3540.	
Tape Support	Any DOS/VS supported device.
DASD Support	Required for SYS001, SYS002 work files.
- 2311	
- 2314	
- 2319	
- 3330-1,2	
- 3333	
- 3330-11	Native mode
- 3340	
- 3344	
- 3350	Native mode.
- 3310,3370	FBA disk types for SAM and VSAM files ONLY.
Data Management Support.	
- SAM	
- DAM	Not for FBA disk devices.
- ISAM	Not for 3330-11, 3350 and FBA disk devices.
- VSAM	See notes below for specific support.

NOTE : The Floating Point feature MUST be installed on the CPU.
 The Decimal and Floating Point Instruction sets are required.

SUPERVISOR GENERATION OPTIONS

The following is a breakdown of the mandatory and suggested Supervisor parameters within the Supervisor generation macros :

MACRO	PARAMETER	REMARKS
CONFG	FP=YES	Required
NOTE. The following FOPT parameters, recommended for DOS/VS, are standard under DOS/VSE:		
FOPT	PCIL=YES	Support for Private Core Image Lib. required
	PC=YES	Required
	WAITM=YES	Required
	RELLDR=YES	Required
	TOD=YES	To use TIME built-in function
	IT=YES	To use DELAY built-in function
	VSAM=YES	If VSAM support required
	GETVIS=YES	Forced by VSAM
VSTAB	VSIZE=nK	As required by the system when in 370 mode
IOTAB	NRES=n	As required by VSE/VSAM (from NPARTS+2 to 255)

VSE/VSAM SUPPORT

The following section briefly describes the VSE/VSAM functions available to the PL/I programmer :

(i) Data Set Organization

- VSAM KSDS Keyed sequential and keyed direct access
- VSAM ESDS Addressed sequential and direct access via RBA
- VSAM RRDS Keyed sequential and keyed direct access via Relative Record Number

(ii) Alternate Index Support

- VSAM KSDS
- VSAM ESDS

(iii) Password Protection

- READ
- UPDATE

(iv) Skip Sequential Performance Option

- VSAM KSDS

(v) Backward Processing Option

- Get previous record

(vi) Re-Usable Data Set Option

- Eliminates need for VSAM Access Method Services DELETE/DEFINE operations for temporary/work files

(vii) Buffer Specifications

- Program is able to specify the number of INDEX and DATA buffers as well as the total buffer space to be allocated at execution time.

(viii) Alternate Index Duplicate Key READ/WRITE Built-In Function

- Simplifies detection of duplicate key status

10.4.3 PRODUCT REFERENCE MATERIAL

- GC33-0004- General Information Manual
- SC33-0008- Programmer's Guide
- SC33-0019- Execution Logic
- GC33-0005- Language Reference Manual
- SX33-6001- Reference Data
- SC33-0035- Transient Library Messages
- SC33-0020- Installation
- SC33-0051- CMS Users Guide for PL/I
- GC33-0016- Program Product Specifications - PL/I Opt. Compiler
- GC33-0017- Program Product Specifications - PL/I Resident Library
- GC33-0018- Program Product Specifications - PL/I Transient Library
- GC33-0046- Design Objectives
- SC33-0021- Compiler Messages

10.5 DOS/VS RPG II (5746-RG1)

Status : Program Product

Latest Release : 3.0

Announced : 1/79

Planned Availability : 4/79 (Initial limited availability)
6/79 (General availability)

Manuals Available : 4/79

SIPO's Available : DOS/VSE SIPO/E (Plan.Avail.3Q79)
VM/SIPO/E (plan.Avail.3Q79)
VM/DOS/VS SIPO/E (Plan.Avail.4Q79)

Selected Product Line : yes

Installation License : N/A

Pre-Installation Test : 30 days

Program Services : Central Service until discontinued
upon 12 months written notice.

Local Service at no charge until
12/31/79 then Local Licensed
Program Support.

10.5.1 SUMMARY OF RELEASES

Release 1.0 (GC33-6030-0 GIM manual)

- 3330 Model 11 Disk Storage
- 3350 Direct Access Storage Facility
- VSAM functions (KSDS, ESDS and RRDS)
- S/3 Batch equivalent functions
- Auto Report as part of the Licensed P.P.

Release 2.0 (GC33-6030-1 GIM manual)

- CICS/DOS/VS EXEC Command Level interface
- DL/I DOS/VS interface (RQDLI)
- CALL interface to COBOL, PL/1, Assembler and RPG-II
- Data Structures (binary mode, working areas, redefinitions, etc.)
- Revised DOS/VS RPG-II logic cycle (only Calculations (C) specifications required)
- Figurative constants (*BLANK/S, *ZERO/S)
- Symbolic dump
- Improved error handling (*PSSR subroutine takes control)

Release 3.0 (GC33-6030-2 GIM manual)
(Information about this release is contained in this chapter).

- Online Source Code Entry under VSE/ICCF and CMS/DOS
- Interactive compiler under VSE/ICCF and CMS/DOS
- Error Message Cross-Referencing
- Native Language Text Capability
- Display Management System/CMS (DMS/CMS)
- VM/Interactive File Sharing (VM/IFS)
- Support of IBM 4300 Processors
- Fixed Block Architecture (FBA) devices
- Dropping of IBM 2321 Data Cell Support

10.5.2 PRODUCT FUNCTION

DOS/VS RPGII is a programming language that can be used to create programs to perform a wide variety of commercial data processing jobs. Release 3 of DOS/VS RPGII is an enhancement of the current Release 2 compiler mainly for interactive usage.

DOS/VS RPGII Releases 2 and 3 run under the control of DOS/VSE with VSE/Advanced Functions or CMS in VM/370 Release 6 with VM/BSE or VM/SE Program Products.

The DOS/VS RPGII release 3 compiler supports all Fixed Block Mode devices which are supported by the data management of DOS/VSE on the 4331 and 4341 Processors for SAM and VSAM files, and compiler workfiles. VSE/VSAM is required for VSAM file support.

OBJECTIVES

The primary purpose of the RPGII Source Entry facility (RSEF) is to provide an easy-to-use, interactive, RPGII source program entry and modification facility which eliminates the need for column counting on the screen, and also provides line-by-line syntax checking. It is not intended to replace the standard editing facilities of the host system, and for this reason only those basic editorial facilities are provided which are necessary for positioning within the source.

Source language input and alteration are possible under control of the specification requirements of the RPGII language and of the Autoreport feature. This means, that the source being entered or modified is checked for correct syntax, as far as possible on a line-by-line basis, and that the user is informed immediately when syntactical errors are detected in the input, so that corrective action can be taken straight away, before the source is submitted to the compiler.

The new compiler features are aimed at providing the interactive user with facilities compatible with those available for other high level languages when compiling in an interactive environment.

Main objective of this release is to guide the batch oriented RPGII user to a terminal oriented programming environment. The features implemented for this purpose are:

(i) INTERACTIVE SOURCE ENTRY FACILITIES

- RPG Source Entry Facility (RSEF)

(ii) INTERACTIVE COMPILER FACILITIES

- Optional cataloging of object deck as CMS TEXT or VSE/ICCF library member
- Optional cataloging of listing as CMS LISTING file (this feature is not available under VSE/ICCF)
- Terminal output for error messages at compile time
- Cross Reference of all error messages
- Display Management System/CMS CALL capability for programs running under VM/CMS with the DMS/CMS facilities installed
- VM/Interactive File Sharing CALL support for programs running under VM/CMS with the VM/IFS facilities installed

(iii) FBA SUPPORT

- Support of compiler work files and, user SAM and VSAM files for all IBM DASD devices supported by DOS/VSE with VSE/Advanced Functions on the IBM 4300 Processors.

USER REQUIREMENTS

The general requirements of RPGII Release 3.0 are based on the need for convenient online terminal facilities for the column oriented RPG source language, since column counting is time consuming and error prone. A template for the chosen RPGII specification type will be displayed in a convenient layout allowing the correct positioning of single or multiple entries for a line of input.

In addition, the terminal user has options available in an interactive environment, which allow him to control the

storing of the compiler output (card deck or listing) on his host library facilities.

The VM/CMS user who has DMS/CMS and VM/IFS facilities installed on his system may also utilize the corresponding CALL interface in his RPGII program.

Types of User

Two groups of users are envisaged:

- VSE/ICCF users
- CMS/DOS users

GETFILE command; but no modification of a DOS/VS library member is possible since access to the DOS/VSE libraries is read-only. The GETFILE command searches the R sublibrary only.

Any display terminal supported by VM/CMS is required to execute RSEF.

All input is translated to upper case before processing.

VSE/ICCF

The user invokes RSEF by entering following at his terminal:

/RSEF

He will be prompted for the name of a library member to be created or modified, and for the password, if necessary. If this member exists, it will be the base for subsequent editing. Otherwise, a new member with the specified name will be created. VSE/ICCF rules and procedures for password protection will be followed.

The filename and password thus established will be the defaults for subsequent SAVE and FILE commands.

Members from the VSE/ICCF library, or from a private or system source statement library can be included via the GETFILE command. Members from a source statement library cannot be modified. Only the R sublibrary is searched.

ONLINE OPERATING MODES

Edit Mode

The user enters the edit mode at the start of an RSEF session, with either an empty workspace or a workspace containing the file specified when RSEF was invoked. The top line of the source displayed on the screen is known as the current line and is identified by the current line pointer, which is an implicit parameter for most of the RSEF commands, (e.g. DELETE). The location of the current line within the workspace may be controlled by the user by means of the RSEF commands (described later in this chapter). When a command has been completed, the screen is refreshed with a display of the workspace contents starting at the current line.

When the workspace has been updated to the user's satisfaction it may be stored in the program library, either under a new name, or it may be used to replace the existing version of the program.

Input/Alter Mode

The user may change from EDIT mode to input or alter mode by issuing the INPUT or ALTER commands. The purpose of these modes is to provide source entry and modification facilities with a subsequent syntax check.

When input or alter modes are entered, RSEF displays a template appropriate to the type of RPGII specification being handled, displays the present contents of the line being built or modified, and any error messages produced by the Syntax Checker.

When the user indicates to RSEF that he is satisfied with the contents of the line, by pressing the ENTER key, RSEF passes the source line to the Syntax Checker for compliance with the RPGII syntax rules.

If no errors are detected, the line is placed in the workspace, either by insertion (INPUT mode, new lines) or replacement (ALTER mode, modified lines). If errors are detected, the line is displayed again together with as many error messages from the syntax checker as will fit on the

screen.

Prompt submode of INPUT

The user may optionally request INPUT with prompt submode. This submode prompts the user for the fields of the chosen RPGII specification, in accordance with the RPGII syntax rules.

This mode has advantages for the inexperienced RPGII programmer, since the Syntax Checker ensures that mandatory fields are defined and that no conflicting fields are entered.

Data submode of INPUT

This submode is provided to allow compile time array data to be entered without the Syntax Checker being invoked.

The experienced RPGII programmer may wish to use this submode to enter bulk code rapidly without invocation of the Syntax Checker and then to use the SYNTAX command to syntax check the whole block of source entered in this way.

SCREEN FORMATS

To assist the user in entering or modifying his source text, different screen formats are displayed depending on the mode of operation.

1. Edit mode format:

This display format is presented to the user at the start of a RSEF session. It is used for all commands except for INPUT (not for DATA), ALTER, '?', and SYNTAX (in an error situation) which have their own special format. As many lines of RPGII source text are displayed as will fit on the screen, a line of message data, and a redisplay of the last command entered. If there are less lines of RPGII source than the screen allows, an end-of-file indicator will be displayed

***** END OF FILE *****

and the remaining lines blanked out.

2. Input/Alter mode format:

In this format RSEF displays a template for the type of RPGII specification currently being entered or modified. In addition up to 12 error messages are displayed, together with the source line being built or modified. This does not apply to the INPUT Data submode.

3. Command menu format:

RSEF displays a menu of the commands available in the current processing mode. This menu is displayed whenever the help command ' is entered.

NATIVE LANGUAGE TEXTS

All RSEF screen texts, messages, and prompting texts are modularized which means that World Trade countries wishing to provide native language texts for the various countries have only to modify these modules to the required language, and then link the RSEF anew.

10.5.3 PRODUCT USE

FBA SUPPORT

DOS/VS RPG-II Release 3.0 supports Fixed Block devices for SAM and VSAM files, and for compiler workfiles. The working files defined on FBA devices are specified by means of JCL statements. Block size will be 4000 bytes.

VSE/VSAM is a pre-requisite to provide access to VSAM datasets. The specifications for VSAM KSDS, ESDS and RRDS have not changed.

The specifications for a SAM file requires 'FBA' to be coded in columns 40 to 42 (Device field, columns 40-46), of the FILE (F) specification.

RPG-II SOURCE ENTRY FACILITY (RSEF) COMMANDS

All commands consist of a command verb followed by one or more parameters, some of which may be optional. The optional parameters may or may not have defaults.

Command verbs may be entered in two forms:

- (a) Full format, where all characters of the command are entered, (e.g. CHANGE)
- (b) Abbreviated format, where an abbreviation is entered (e.g. C, CH, CHA, CHAN or CHANG for CHANGE)

The minimal abbreviation for the command will be shown as alternative below the corresponding command verb.

Parameters are separated by one or more blanks.

Parameters with alternatives are indicated by means of vertical lists and default values for each parameter are also indicated.

ALTER Command

The ALTER command allows the user to modify the current line. When this command is used the altered line is passed to the Syntax Checker before it is stored in the work space (compare with CHANGE command).

The user will be shown a template which facilitates the input of the fields. By pressing the ENTER key he submits the line to syntax checking. The Syntax Checker will inform the user whether the source line is complete and syntactically correct, or not. To provide an optimum of information syntax checking of the source line may continue even after an error has already been found. However, subsequent error messages may be incorrect due to preceding errors.

Any error detected will be indicated by an error message, and by a '\$' above (VM/CMS) or below (VSE/ICCF) the beginning of the erroneous field. Any entry not required for a correct source statement will be indicated by an 'S'. If there are superfluous entries, the original statement and the syntactically correct statement purged from the superfluous items will be displayed. The user then has the option to choose either the syntactically correct line (by entering 'Y'), or his original line (by entering 'N').

If no syntax errors are detected and the line is complete, the new source statement will replace the current line in the work space.

The ALTER command processes only one line at a time - namely the current line. Control passes back to the edit mode display as soon as all alterations to the current line have been completed without a syntax error being detected.

If the escape characters '?Q' are entered in columns 1-2 of the input line the alter mode is terminated and no change is made to the current line.

```
ALTER  
A
```

Current line pointer: unchanged.

BACKWARD Command

The BACKWARD command allows the user to scroll 'n' (default is 1) screens backward in the workspace. No wrap-around is possible, and a BACKWARD command which scrolls beyond top-of-file points to the first active line in the workspace.

```
BACKWARD           1  
B                   n
```

Current line pointer: moved 1 or n screens backward in file.

BOTTOM Command

The BOTTOM command is used to set the current line pointer to the last line of the current workspace. If the workspace is empty this command is ignored.

```
BOTTOM  
BOT
```

Current line pointer: set to the last source line in the workspace.

CHANGE Command

Use the CHANGE command to change a specified string of characters to another specified string of characters of either the same or a different length. If the length of the new string is different, it will be either truncated or padded (with blanks). The command can only be used to change the current line.

No syntax checking is done for the resulting line. If syntax checking is required the ALTER command should be used, or the changed line should be syntax checked using the SYNTAX command.

```
CHANGE delimiter string1 delimiter string2 delimiter
C
```

The delimiter character is defined as the first non-blank character occurring after the command verb. Refer to the ZONE command for the search range of CHANGE.

Examples of valid change commands:

```
CHANGE /OLD/NEW/          'NEW'      replaces 'OLD'
C      *OLD*NEWNEW*      'NEW'      replaces 'OLD'
CHANGE XOLDOLDXNEWX      'NEW'      replaces 'OLDOLD'
```

Current line pointer: unchanged.

DELETE Command

The DELETE command is used to erase 'n' (default is 1) lines starting with the current line.

If '*' is specified, the current line and all lines after the current line will be erased. If the DELETE * command is used to erase all lines in the workspace the current line will be set to top-of-file.

DELETE	1
DEL	n
	*

Current line pointer: set to line preceding the first line deleted, or to top-of-file.

DOWN Command

Used to move the current line pointer to 'n' (default 1) lines after the current line. If the workspace is empty this command is ignored.

If moving down 'n' lines will move beyond end-of-file the command has the same effect as the BOTTOM command.

DOWN	1
D	n

Current line pointer: moved 1 or n lines forward.

FILE Command

Used to store the contents of the workspace in the VSE/ICCF library or onto a CMS disk. Resequencing will be done if the RESEQ command was issued before.

If no filename, filetype, filemode and/or password are specified, the values established when calling RSEF are assumed as defaults.

Under VSE/ICCF the file will only be stored in the user's private library. No write access to PUBLIC or COMMON ETSS libraries is provided.

If the workspace is empty, the FILE command is ignored.

VSE/ICCF format:

```
FILE filename password
```

where the defaults filename, password are determined as described above.

Password handling is according to VSE/ICCF rules.

CMS/DOS format:

```
FILE filename filetype filemode
```

where the defaults filename, filetype, filemode are determined as described above.

Current line pointer: not applicable.

FORWARD Command

The FORWARD command allows the user to scroll 'n' (default value is 1) screens forward in the workspace. No wrap-around is possible, and a FORWARD command which scrolls beyond end-of-file has the same effect as the BOTTOM command.

FORWARD	1
F	n

Current line pointer: moved 1 or n screens forward.

GETFILE Command

The GETFILE command is used to include, after the current line, sections of source programs which have been previously stored in either the DOS/VSE source library, the VSE/ICCF library, or the CMS libraries (plus their read-only extensions). The GETFILE command does not imply a syntax check of the included code.

Either the whole file or a section of the file can be included.

Under VSE/ICCF the member for inclusion may reside on a PRIVATE, PUBLIC or COMMON ETSS library or be stored in the private or system source statement libraries. The search occurs in this order.

Under CMS/DOS first the CMS libraries are searched, then the private or system source statement library, if attached.

VSE/ICCF format:

```
GETFILE filename password lo hi
GET
```

The decimal numbers 'lo' and 'hi' indicate the first and last lines, respectively, which are to be included. If both are omitted, the whole file is included. If one number only is specified, it is assumed to be 'lo' and the file from this line to end-of-file is included. If the specified 'lo' limit is outside the program to be included, the GETFILE command is ignored. If the 'hi' value is outside the program to be included, the program will be included up to the last line.

CMS/DOS format:

```
GETFILE filename filetype filemode (lo hi)
GET
```

The decimal numbers 'lo' and 'hi' indicate the first and last lines, respectively, which are to be included. If both are omitted, the whole file is included. If one number only is specified, it is assumed to be 'lo' and the file from this line to end-of-file is included. If the specified 'lo' limit is outside the program to be included, the GETFILE command is ignored. If the 'hi' value is outside the program to be included, the program will be included up to the last line. The righth parenthesis is optional.

Current line pointer: set to the first line of the included text.

HELP ('?') command

The help command '?' may be used at any time to display the command menu on the screen. The workspace is not altered.

INPUT Command

The INPUT command is used to generate new RPGII source lines, and to automatically syntax check these lines, one at a time. The user indicates in the command which RPGII specification type he wants to generate. He is then shown a template which facilitates the input of the fields. By pressing the ENTER key he submits the line to syntax checking. The Syntax Checker will inform the user whether the source line is complete and syntactically correct, or not. To provide an optimum of information syntax checking of the source line may continue even after an error has already been found. However, subsequent error messages may be incorrect due to preceding errors.

Any error detected will be indicated by an error message, and by a '\$' above (VM/CMS) or below (VSE/ICCF) the beginning of the erroneous field. Any entry not required for a correct source statement will be indicated by an 'S'. If there are superfluous entries, the original statement and the syntactically correct statement purged from the superfluous items will will displayed. The user then has the option to choose either the syntactically correct line (by entering Y), or his original line (by entering N).

If no syntax errors are detected and the line is complete, the new source statement will be added to the work space following the current line.

Input mode is terminated by entering '?Q' in columns 1-2 of the input line, or by entering a blank line. The input line will be discarded.

If the DATA parameter is specified the user enters the DATA submode. In DATA submode, each line of input, delimited by the enter key, will be treated as data. No checks are performed since this mode is intended only for the purpose of defining input for compile time arrays. If this mode is used for entering RPGII specification statements, then any statement entered in this way should be checked for consistency by means of the SYNTAX command.

If the optional parameter '?P' is specified (for other than '*' and DATA) the user will enter prompt submode. The prompt mode allows free format input; alignments or leading blanks can be ignored. By entering '?R' in columns 1-2 of the input line the user can restart the prompting at the beginning of the specification. Previously entered fields will be redisplayed.

```
INPUT            H            ?P  
I                F  
                 E  
                 L  
                 I  
                 C  
                 O  
                 S  
                 U  
                 /  
                 *  
                 DATA
```

Current line pointer: set to last line created successfully by INPUT.

LOCATE Command

Use the LOCATE command to locate a specified string of characters in the source text after the current line.

```
LOCATE    delimiter    string    delimiter  
L
```

Current line pointer: set to line containing requested string, or will stay at the current line if the string was not found. Refer to the ZONE command for the search range of LOCATE. The delimiter is defined as the first non-blank character after LOCATE.

QUIT command

The QUIT command is used to terminate the current session without altering the source stored on the library. Control is returned to the host system. When the workspace is discarded by means of the QUIT command, no resequencing will take place. Refer to the RESEQ command.

QUIT

Current line pointer: not applicable.

RESEQ Command

Informs RSEF whether or not sequence numbers are to be inserted automatically in columns 1-5 of the source in the workspace before the file is stored in the library. The parameter 'n' is the sequencing increment from '0' to '51'. The default is 10.

If resequencing is specified, the sequence numbers in columns 1-5 will be updated before the file is stored (FILE, or SAVE commands). No resequencing will occur if the workspace is discarded by using the QUIT command.

```
RESEQ      10  
           n
```

Current line pointer: not applicable.

SAVE Command

Used to store the contents of the workspace in the library. The command does not imply the end of the current session and control will not be returned to the host system.

Resequencing will occur if the command RESEQ was issued before.

If no filename, filetype, filemode, and/or password are specified, then the values established when invoking RSEF are assumed as default.

Under VSE/ICCF the file will only be stored in the user's private library. No write access to PUBLIC or COMMON VSE/ICCF libraries is provided.

VSE/ICCF format:

SAVE filename password

Password handling is according to VSE/ICCF rules.

CMS/DOS format:

SAVE filename filetype filemode

Current line pointer: unchanged.

SYNTAX Command

This command enables the user to syntax check lines of code that may have been included by the GETFILE command or entered in DATA submode and therefore may have not yet been checked. The number of lines is given by the parameter 'n' (default is 1). If * is specified, the current line and all lines after the current one will be syntax checked.

If the compile time data indicator ('**' in columns 1-2) is encountered, RSEF will terminate the syntax check, and issue an information message.

If an erroneous line is detected this line will be treated in the same way as if it had been entered in input mode, with the exception that the line will remain unchanged in the work space if the user quits. The characters '?Q' in columns 1-2 of the input line will end the SYNTAX mode.

```
SYNTAX      1
            n
            *
```

Current line pointer: will point to the line following the last line syntax checked. If the user quits the SYNTAX mode, the pointer will point to the line from which the user quit.

TOP command

The TOP command is used to set the current line pointer to the top-of-file line. This command allows the user to add lines in front of the first line in the workspace.

```
TOP
T
```

Current line pointer: set to top-of-file.

UP Command

Used to move the current line pointer 'n' (default 1) lines before the current line.

If moving up 'n' lines will move beyond top-of-file, then the first line of the workspace will be displayed.

```
UP      1
U       n
```

Current line pointer: moved from 1 to n lines to the beginning of the file being edited.

ZONE Command

Used to restrict the search range of the CHANGE and LOCATE commands to the specified columns. Default values are 1 and 80, respectively.

```
ZONE    1      80
Z       n1    n2
```

Current line pointer: not applicable.

DOS/VS RPG-II COMPILERError Messages on Terminal

The compiler will generate an error cross-reference summary (Compiler Diagnostics Summary), which can be output to the user terminal (SYSLOG), and/or to the compiler listing (SYSLST), where it will replace the currently provided text of error messages.

For each error detected in the source code, the Compiler Diagnostic Summary will include:

- the text of the diagnostic message
- appropriate information to identify the source statement that originated the error

The terminal display is output to SYSLOG and it is invoked via the VSE/ICCF option LIST or CMS/DOS option TERM.

The output to SYSLST is requested via the VSE/ICCF option LIST or the CMS/DOS option ERRS.

In previous versions of the compiler the cross-reference information was provided by intermediate relational messages. These messages will be dropped in the new implementation, because their information is included in the Diagnostics Summary.

If the option NOERRS is specified, the SYSLST output is the same as in previous versions of the compiler, i.e. the relational messages are put out as soon as the errors are detected and, at the end of the listing, the compiler prints out only the text of error messages, without cross-reference information.

Error Cross-reference

The Release 2 compiler outputs the description and the identifier of detected errors, but no cross-reference information is provided: the programmer has to browse through the listing to find out where the errors occurred.

This is not an adequate support for on-line debugging, were a more concise and complete description of errors is required.

The new support, by providing error cross-reference information, facilitates, in an on-line environment, immediate correction and recompilation of a program without need to wait for the hardcopy output.

It also facilitates the programmer's task in a batch environment, by providing error information in a more condensed form.

The Compiler Diagnostics Summary provides, for each syntax error detected in the source code, the description of the error and at least one of the following types of cross-reference information:

- system assigned statement sequence number
- user statement sequence number
- other reference data like name of file, table, array, or field where the error is detected

Display of sequence numbers

The Release 2 compiler uses system generated sequence numbers as cross-reference between message number and program statement.

This system sequence number is not adequate for online debugging, because it is generated dynamically:

- it does not appear in the source code
- it does not match with the system sequence number of previous listings of the program if this has been updated

The user can provide his own statement sequence number, which is much more reliable as reference number because it is available with the source code and is not subject to change from one compilation to another.

This statement sequence number can also be generated automatically by using the RSEF online editor and therefore

it can be included in the source code with no programmer effort.

In the Release 2 compiler, this user sequence number is dropped after the initial reading of the source and is not available for display if the error is detected in a subsequent phase.

In the new release of RPGII this user sequence number will be maintained for the entire compilation run and, upon error detection, collected with other error information that will be displayed, at the end of compilation, in the Compiler Diagnostics Summary.

Formatting of terminal output

The messages written to SYSLOG will be formatted to fit the terminal display line.

The error message and cross-reference data will be displayed in the Compiler Diagnostics Summary, as shown in the following examples:

ILN099 PACKED ENTRY IN POSITION 43 IS NOT BLANK FOR AN ALPHA FIELD.
ASSUME BLANK.

00377 ALPFLD

ILN178 FACTOR 2 IS INVALID.SPEC IS DROPPED IF ANY ENTRY IS REQUIRED

D 00234 10-120 CALL
D 00288 11-023 PARM

Note the 'D' indicates that the two statements have been dropped.

ILN071 RAF/TABLE/ARRAY FILE MUST NOT BE A KSDS. SPEC IS DROPPED.

D 00330 01-022 ITAB OTAB

The error message is followed by a line of cross-reference data for each occurrence of the error. The cross-reference data displayed is the information available when the error is detected.

The cross-reference data is shown in four columns as follow:

- first column: system sequence number
- second column: user sequence number
- third and fourth columns: name of the elements (i.e. file, table, array, indicator, field, etc.) that caused the error

Compiling RPGII Programs Under VSE/ICCF

A VSE/ICCF CLIST procedure will be provided for compiling RPGII programs under VSE/ICCF. This CLIST procedure is similar to those available for other high level languages under VSE/ICCF. The parameters passed to this procedure define the compiler input and outputs.

```
RPGII filename1 (OBJ filename2 options)
*
```

where filename1 (required) is the file name of the source program. the other parameters are optional. If nothing or '(OBJ *' is entered, the compiled object program will be filed in the punch area under the name 'filename1'. 'filename2' must be used to file the compiled object program in the library under the name 'filename2'.

VSE/ICCF stores the object deck into a VSE/ICCF library if the parameter OBJ followed by a filename is passed to the procedure which initiates the RPGII compilation. If the filename is replaced with '*' the object deck is placed in the user's punch area.

Note: An object deck is only produced if the option DECK is in effect.

The SYSLST output under VSE/ICCF is directed to the user's terminal. The user can specify in addition the /HARDCOPY command to direct the output to a hardcopy printer.

There are several options available under VSE/ICCF. Refer to the /OPTION job entry statement. Some significant for the compiler are:

DECK/NODECK	Produces an object deck
LIST/NOLIST	Produces terminal display
LOAD/NOLOAD	Enables automatic execution of the compiled object deck
NOREST/RESET	Stacks more than one object deck in the punch area

Note: The options at the left side are the defaults.

Compiling an Auto Report Program Under CMS/DOS

A CMS EXEC procedure is provided to invoke the Auto Report feature:

RPGIAUTO filename (options)

were 'filename' (only parameter required) is the name of the source program to be compiled, which normally has a filetype of 'RPGIAUTO'. The options supported are those indicated for the compiler. Depending on the punch option in the Select/Sort Specification, the filetype can be 'RPGII'.

Invoking the DL/I Translator Under CMS/DOS

A CMS EXEC procedure is provided to invoke the DL/I translator:

RPGIXLTR filename

were 'filename' is the name of the source program to be translated, which must have a filetype of 'RPGII'. The output of this compilation is filed under 'filename RPGII' and is ready for normal compilation. The listing is filed under 'filename LISTING'.

DMS/CMS CALL INTERFACE SUPPORT

The Display Management System/CMS (DMS/CMS) is available to RPGII programs in the CMS/DOS environment. DMS/CMS allows the programmer to define (in CMS) terminal panels, and to use previously defined panels in RPGII programs: he can move data from data fields in the panel to data fields in the program, and vice versa, he can modify intensities, position the cursor, sound the audible alarm, and he can add comments to the bottom line of the panel before it is displayed.

The definition of a terminal panel is done by the CMS command CREATE. This is supported under VM/370 Release 6 with VM/BSE or VM/SE Program Products.

The RPGII programmer invokes the DMS/CMS panel manager using the High Level Language (HLL) interface of RPG-II through the CALL, PLIST, and PARM Calculation Specifications. A Calculation Specification with the operation code CALL provides the ability to link to another program. Parameter passing is specified by PLIST (parameter list) and PARM (parameter) Calculation Specifications.

For the convenience of the reader of this document the layout of the CALL, PLIST, and PARM Calculation Specifications is briefly summarized

CALL Operation Code

CALL is a Calculation Specification operation code which transfers control to the subprogram specified in Factor2 field. An associated parameter list specifying the data which are to be passed to the subprogram can appear in the Result field.

Col. 1-5	RPG Standard
Col. 6	C
Col. 7-17	RPG Standard
Col.18-27	Blank
Col.28-32 (Op.Code)	CALL
Col.33-42 (Factor2)	Required. Alphameric literal containing the name of the program to be called.
Col.43-48 (Result Fld.)	Optional.Name of PLIST to be passed. If blank, the CALL statement must be immediately followed by the PARM(s) to be passed. A CALL statement with blanks in the result field not fol- lowed by PARM(s) will be executed with no parameters passed.
Col.49-55	Blank
Col.56-57	Optional. An RPG Resulting Indicator may be specified to be set on for an an error return from the called program.
Col.58-59	Optional. An RPG Resulting Indicator may be specified to be set on if the called program returns with LR (Last Record Indicator) on.
Col.60-74	RPG Standard.
Col.75-80	RPG Standard.

A program can contain multiple CALL statements to the same subprogram with the same or different parameter lists

specified.

Upon return from a subprogram, execution continues at the first executable statement after the CALL statement.

Linkage will be by address, so that the called subprogram will be operating upon the parameters actually located in the calling program. Upon an abnormal return, the parameters will have been modified up to the point of the error in the called subprogram.

If the error indicator (Col 56-57) is not present in the CALL statement, an error return will cause execution of the user *PSSR subroutine, if present. If not, default will be the RPG-II system error handling convention.

If a Program Status Data Structure (PSDS) is specified in the calling program, it will contain a numeric return code upon return from a called program.

If the conditions for setting on the indicators in columns 56-59 are not met, they will be set off.

PLIST Operation Code

PLIST is a nonexecutable operation code which indicates a name by which a following list of parameters (PARM statements) may be referenced by a CALL statement.

Col. 1-5	RPG Standard.
Col. 6	C
Col. 7-17	Blank
Col.18-27 (Factor1)	Required. Name by which PLIST is referenced. Rules for RPG labels apply.
Col.28-32	PLIST
Col.33-59	Blank
Col.60-74	RPG Standard.
Col.75-80	RPG Standard.

The PLIST statement must be immediately followed by the parameters (PARM statements) which belong to it.

PARM Operation Code

PARM is a nonexecutable Calculation Specification operation code which indicates data elements which are to be used in communicating with another program.

Col. 1-5	RPG Standard.
Col. 6	C
Col. 7-27	Blank
Col.28-32	PARM
Col.33-42 (Factor2)	Blank
Col.43-48 (Result Fld.)	Required. A field name, data structure name, array element with constant subscript, array name which is to be the parameter.
Col.49-52 (Field Length and Decimal Positions)	RPG Standard.
Col.53-59	Blank
Col.60-74	RPG Standard.
Col.75-80	RPG Standard.

The Result Field of PARM statement may not contain a table name, a literal, a label.

Using The DMS/CMS Call Interface

To call the DMS/CMS panel manager the RPGII programmer has to prepare the following Calculation Specifications:

Col.6	Col.18	Col.28	Col.33	Col.43
C		CALL	'ILNEUDS'	parmlist
.		...		
.		...		
C	parmlist	PLIST		
C		PARM		control
C		PARM		load1
.	
C		PARM		loadn
C		PARM		unload1
.	
C		PARM		unloadn
C		PARM		datamask
C		PARM		textmask

where:

control is the name of a structure containing control information. The structure consists of the following subfields (in the order listed):

panel eighth character field containing the name of a previously defined panel using the DMS/CMS panel definition facilities.

request one byte field containing the literal 'D' if the panel is to be displayed, or the literal 'R' if the panel is to be released.

NOTE. The following parameters are applicable to the display 'D' request only:

rstatus is a one byte binary field which will indicate the key pressed in response to the display. In case of an I/O error it will be set to a value of 255.

dataflds is a halfword containing the number of data fields in the panel. This number must be provided if loadlist, unldlist,

or dflmask are set to 'Y' (yes).

textflds is a halfword containing the number of text fields in the panel, including the comment, if requested. This number must be provided if tfldmask is set to 'Y'.

loadlist one byte field containing the character 'Y' if data is to be moved to the panel, else it contains the character 'N'.

unldlist one byte field containing the character 'Y' if data is to be moved from the panel, else it contains a character 'N'.

dflmask one byte field which contains the character 'Y' if a data mask of attribute bytes is provided, else it contains the character 'N'.

tfldmask one byte field which contains the character 'Y' if a text mask of attribute bytes is provided, else it contains the character 'N'.

comment 79 byte area which will be treated as a comment and will be displayed on the last line of the screen.

signal one byte field. If this area contains the literal 'Y', the audible alarm will be sounded when the panel is displayed.

cursor halfword containing the number of the data field at whose beginning the cursor will be positioned. If the number is greater than the number of the data fields on the panel, the cursor will be positioned at the beginning of the first data field.

load1-loadn are areas in the program where data exists to be moved to the panel. The control parameter **loadlist** must be activated, and 'n' must equal the value of the parameter **dataflds**. If no load parameters are passed, the data fields will be displayed as nulls or the fill character defined under the DMS/CMS services.

unload1-unloadn are areas in the program where data is to be moved from the panel. The control parameter

unldlist must be activated, and 'n' must equal value in dataflds. A shorter data entry will be justified. If no unload parameters are passed, all data fields will be protected, and data cannot be entered at the screen.

datamask is an array of one byte fields, one for each data field of the panel. If a byte is set to '0' the corresponding data field will be displayed at its previously defined intensity; if the byte is set to '1', at normal intensity; if set to '2' at high intensity; if to '3', the field will not be displayed. The control parameter dfldmask must be activated if there is a datamask. If the datamask parameter is omitted, the data will be displayed at their defined intensities.

textmask is an array of one byte fields, one for each text field of the panel. The settings are the same as for the datamask array. That is '0', '1', '2' or '3' for default, normal, high or dark intensity respectively.

The control parameter tfldmask must be activated if there is a datamask. If the textmask parameter is omitted, the text fields will be displayed at their defined intensities.

In case of an error, the DMS/CMS displays an appropriate error message. Also, a RPG resulting indicator which was placed in columns 56-57 of the CALL Calculation Specification will be turned on, and the *STATUS field of a Program Status Data Structure (PSDS), if one was defined, will contain a return code.

VM/IFS CALL INTERFACE SUPPORT

The VM/Interactive File Sharing (VM/IFS) is available to RPGII programs in the CMS/DOS environment. It gives multiple transactions from interactive users the possibility to access and share common data stored in VSAM datasets.

Using The VM/IFS CALL Interface

The RPGII programmer invokes VM/IFS input/output services using the RPG-II High Level Languages (HLL) CALL interface which is summarized in the paragraph on VM/IFS. He has to prepare the following CALL and PARM Calculation Specifications

Col.6	Col.18	Col.28	Col.33	Col.43
C		CALL	'RPGTIFS'	plist
.		...		
.		...		
C	plist	PLIST		
C		PARM		count
C		PARM		function
C		PARM		pcb
C		PARM		ioarea
C		PARM		ssa

where:

count is a fullword which contains the value '3' if no SSA parameter is passed, else it contains the value '4'

function is a four byte field containing the literals:

- 'GU ' (Get Unique)
- 'GHU ' (Get Hold Unique)
- 'GN ' (Get Next)
- 'GHN ' (Get Hold Next)
- 'ISRT' (Insert)
- 'REPL' (Replace)
- 'DLET' (Delete)

indicating the action requested from VM/IFS.

pcb Program Control Block (PCB) for this file. The PCB is a declared structure in the application interface; its address is passed to the application. The layout of the PCB is compatible with the DL/I PCB format.

ioarea is an input/output area in the calling program. For variable length records, the first two bytes contain the record length.

ssa is an area containing the Segment Search Argument (SSA) which holds the key for the requested record. This parameter can be omitted for the function 'GN '. The layout of the SSA is compatible with the DL/I SSA format.

All numeric fields are in binary format.

If the VM/IFS detects an error condition it will set the status code of the PCB accordingly.

10.5.4 DEPENDENCIES

HARDWARE AND SOFTWARE

DOS/VSE environment

DOS/VS RPG-II Release 3 compiler and its object programs are designed to operate on any IBM System that is supported by DOS/VSE with VSE/Advanced Functions. A minimum DOS/VSE virtual partition must be available. For performance reasons, a minimum of 24K bytes of real storage is recommended.

For installation of DOS/VS RPG-II, the DOS/VSE Restore utility program and the DOS/VSE Maintain History Program (MSHP) are required.

VSAM users will require the VSE/VSAM Licensed Program.

The online component RSEF requires VSE/ICCF under DOS/VSE. It requires an interactive partition of 140K bytes of virtual storage.

DL/I DOS/VS (5746-XX1) Version 1 Release 4 is required for database (DB) requests. The DL/I DOS/VS Translator requires a minimum of 128K bytes in a virtual partition.

CICS/DOS/VS (5746-XX3) Version 1 Release 4 is required for datacommunication (DC) requests. The CICS/DOS/VS Translator requires a minimum of 128K bytes in a virtual partition.

CMS/DOS environment

DOS/VS RPG-II Release 3 compiler and its object programs are designed to operate on any IBM System that is supported by VM/BSE 1.2/CMS with a DOSLIB.

DMS/CMS and VM/IFS CALL interface users will require the DMS/CMS and VM/IFS Licensed Products, which in turn requires the VM/370 Release 6 plus VM/BSE or VM/SE Licensed Products.

The online component RSEF requires VM/BSE 1.2/CMS. The virtual storage requirements for RSEF under CMS are 140K

bytes.

The RPGII Compiler Release 3.0 under CMS/DOS does not require the presence of a DOS/VSE SYSRES since the compiler does not copy books from the System Source Library. However, the CMS/DOS DOSLIB must be available to support the RPGII Compiler and the RPGII object programs.

The RPGII Autoreport feature requires the presence of a DOS/VS SYSRES if the /COPY feature is used.

The DL/I DOS/VS and CICS/DOS/VS Translators requires the presence of SYSRES.

COMPATIBILITY WITH PREVIOUS RELEASES

Source programs written for the DOS RPGII Compiler or for the DOS/VS RPGII Compiler Releases 1 and 2 will be accepted without any modification by the DOS/VS RPGII Compiler Release 3.0.

Object programs using the Formatted Dump facility will run if the core image library contains the appropriate transient routines:

- Release 1.0 object programs require Release 1.0 Formatted Dump routines.
- Release 2.0 object programs require Release 2.0 Formatted Dump routines.
- Release 3.0 object programs require Release 3.0 Formatted Dump routines.

The DOS/VS RPG II Compiler accepts, without modification, source code programs written for the DOS RPG II Compiler (5736-RG1).

By means of a Conversion Pre-Processor (CPP) Release 1 (5735-CV1), batch source application programs written for the System/3 Model 10 Disk and Model 15 RPG II compilers (5702-RG1, 5704-RG1), are converted to a form acceptable to DOS/VS RPG-II.

The Auto Report feature is an integral part of the DOS/VS RPG II Program Product.

DOS/VS RPG II DEPENDENCY INFORMATION CHART

DEPENDENCY ITEM	REMARKS
Program Prerequisites	
- SORT/MERGE	(i) If the SELECT/SORT function of the AUTO REPORT program is required, SORT/MERGE 5746-SM1 or 5746-SM2, or an equivalent sort/merge program is required. (ii) If VSAM RRDS ADDR0UT processing is required, Sort/Merge Version 2 5746-SM2 is required.
UNIT RECORD Support.	
- 1442 CRP	At least a card reader (or equivalent) and a line printer (or equivalent) are required.
- 2501 CR	
- 2520 CRP	Also as a printer device
- 2540 CRP	
- 2560 MPCM	Also as a printer device
- 3504 CR	
- 3505 CR	Also as a printer device
- 3525 CRP	
- 5424 MFCU	Also as a printer device
- 5425 MFCU	
- 1403 Printer	
- 1443 Printer	
- 3203 Printer	
- 3800 Printer	
- 5203 Printer	
- PRT1 Printer devices	
- 3540 Diskette I/O Unit	
TAPE SUPPORT	Any DOS/VSE supported device (forward read only for device independent files). Work files may NOT use tape.

DOS/VS RPG II Dependency Information Chart (continuation)

DASD Support	Work files (SYS001,002) MUST be on SAME device type.
- 2311	
- 2314	
- 2319	
- 3330-1,2	
- 3333	
- 3330-11	
- 3340	
- 3344	
- 3350	
- 3310,3370	Fixed Block module (FBA) disk devices for SAM and VSAM files ONLY.
DATA MANAGEMENT Support	
- SAM	
- DAM	
- ISAM	
- VSAM	KSDS, ESDS and RRDS

NOTE : The Decimal Instruction Set is required.

SUPERVISOR GENERATION OPTIONS

The following is a breakdown of the mandatory and suggested Supervisor parameters within the Supervisor generation macros :

MACRO	PARAMETER	REMARKS
FOPT	SYSFIL=YES	If DASD or DISKETTE are used with device independence.
NOTE. The following FOPT parameters, recommended for DOS/VS, are standard under DOS/VSE:		
	PC=YES	Required
	TOD=YES	To use TIME built-in function
	VSAM=YES	If VSAM support required
	GETVIS=YES	Forced by VSAM
VSTAB	VSIZE=nK	As required by the system when in 370 mode
IOTAB	NRES=n	As required by VSE/VSAM (from NPARTS+2 to 255)

10.5.5 EXAMPLES OF USE

COMMAND MENU EXAMPLE

This screen is displayed when the HELP (?) command is entered from EDIT mode. EDIT mode is entered automatically when RSEF is invoked.

The screen displays are as shown below, with minor differences between VSE/ICCF and CMS/DOS. This one is for CMS/DOS.

A	ALTER	ALTER DATA IN SOURCE		L	LOCATE	LOCATE A GIVEN CHARACTER
B	BACKWARD	SCROLL BACK N PAGES				STRING
BOT	BOTTOM	MAKE LAST LINE INTO			QUIT	END SESSION WITHOUT
		CURRENT LINE				ALTERING LIBRARIES
C	CHANGE	ALTER DATA IN SOURCE			RESEQ	RESEQUENCE COLS 1-5
		WITHOUT CHECKING SYNTAX			SAVE	SAVE WORKSPACE IN LIBRARY
DEL	DELETE	DELETE ONE OR MORE LINES				AND CONTINUE
D	DOWN	ADVANCE ONE OR MORE LINES			SYNTAX	SYNTAX CHECK N SOURCE
		IN THE FILE				RECORDS
	FILE	WRITE SOURCE TO LIBRARY		T	TOP	GO BACK TO TOP OF FILE
		AND END SESSION		U	UP	GO BACK ONE OR MORE LINES
F	FORWARD	SCROLL FORWARD N PAGES				IN THE FILE
GET	GETFILE	INSERT SOURCE MEMBER FROM		Z	ZONE	SET UP THE SEARCH LIMITS
		LIBRARY				
?		DISPLAY COMMAND MENU				
I	INPUT	CREATE NEW SOURCE LINE				

.....+.....1.....+.....2.....+.....3.....+.....4.....+.....5.....+.....6.....+.....7.....+.....8

EDIT MODE DISPLAY

The screen displays are as shown below, with minor differences between VSE/ICCF and CMS/DOS. This display is for CMS/DOS.

```
-----  
ILNZ001 ENTER COMMAND (? WILL DISPLAY COMMAND MENU)  
00000H      1      J      B S      B  
00018FINPUT  IPE F  80  80      READ40 SYSIPT  
00030I      8  29 NAME  
00040C  01      INVAMT  ADD  TOTAL      TOTAL  72  
00050C  01      INVAMT  ADD  GRPTOT     GRPTOT  72  
00060COUTPUT H  181  1P  
* * * * * * * * * * END OF FILE * * * * * * * * * *
```

```
.....+.....1.....+.....2.....+.....3.....+.....4.....+.....5.....+.....6.....+.....7.....+.....  
input o
```

The top line is the 'message line'. For VSE/ICCF this line will be at the bottom of the screen. The message numbers have the format 'ILNxnnn' where:

- nnn is the message number suffix
- x is either 'Z' (indicating a general message) or a specification type (e.g. H,F,E,L,I,C,O) where the error has occurred

The first line after the 'message line' is the 'current line' to be handled by the RSEF commands.

The last two lines are the 'command line' and the 'scale line'. In the example the INPUT command is issued to enter an OUTPUT specification under control of the Syntax Checker. The proper template will be displayed next. For VSE/ICCF both lines will be at the top.

EXTENSION Specification

```

=====
ILNZ052 INSERT A RPG II LINE OR ENTER ?Q (COL 1-2) TO QUIT INPUT MODE
=====
A = PAGE/LINE                K = P/B/L/R/BLANK
E = E                        L = DECIMAL POSITIONS (0-9)
B = REC.SEQ.OF THE CHAINING FILE  M = SEQUENCE (A,D,BLANK)
C = NUMBER OF CHAINING FIELD (C1-C9)  N = TAB/ARRAY NAME (ALTERN.FORMAT)
D = FROM FILE NAME          O = LENGTH OF ENTRY (1-256)
F = TO FILE NAME            P = P/B/L/R/BLANK
G = TABLE OR ARRAY NAME    Q = DECIMAL POSITIONS (0-9,BLANK)
H = NUMB.OF ENTR.PER RECORD (1-999,BLANK)  R = SEQUENCE (A,D,BLANK)
I = NUMB.OF ENTR.PER TAB./ARRAY (1-9999)  S = COMMENTS
J = LENGTH OF ENTRY (1-256)  T = PROGRAM IDENTIFICATION
=====
      E  CC      FFFFFFFF      HHH      JJJ L NNNNNN      P R      TTTTTT
AAAAA BB  DDDDDDDD      GGGGGG      IIII      K M      000 Q SSSSSSSSSSSSSSSSS
      E      intable outable tab1      10 200      5 0atab2      5 2 tab1.of standards
.....+.....1.....+.....2.....+.....3.....+.....4.....+.....5.....+.....6.....+.....7.....+.....8
    
```

10.5.6 PRODUCT REFERENCE MATERIAL

- GC33-6028- Design Objectives
- GC33-6029-1 Licensed Program Specifications
- GC33-6030-2 General Information
- GC21-7514-2 Introduction to RPG-II Report Writing
Recomended as introduction. It is application oriented.
- SC33-6032-2 Installation Reference
- GC33-6074-0 User's Guide
- SC33-0085-0 CICS/DOS/VS Application Programmer's Reference Manual
for RPG-II
- SH12-5411-4 DL/I DOS/VS Application Programming Reference
- SC33-6031-2 DOS/VS RPG II Language
plus TNL SN33-9256
- SC33-6034-1 Auto Report Feature
- SC33-6033-1 Messages
plus TNL SN33-9257
- SC33-6035-0 System/3 Conversion Preprocessor Installation and
Reference Manual
plus TNL SN33-9258
- GX21-9090-3 Output Specifications
- GX21-9091-2 Extension and Line Counter Specifications
- GX21-9092- Control and File Description Specifications
- GX21-9093-2 Calculation Specifications
- GX21-9094- Input Specifications
- GX21-9095- Indicator Summary
- GX21-9129- Debugging Template
- GX21-9139- Auto Report Specifications
- LY33-9062-2 Logic Manual
- SC33-6068- VSE/Interactive Computing and Control Facility

Terminal User's Guide

- GC20-1818- IBM Virtual Machine Facility/370
CMS Command and Macro Reference
- GC20-1819- IBM Virtual Machine Facility/370
CMS User's Guide
- GC24-5197-0 IBM Virtual Machine Facility/370
Display Management System for CMS
General Information
- SC24-5198-0 IBM Virtual Machine Facility/370
Display Management System for CMS
Guide and Reference
- LY24-5206-0 IBM Virtual Machine Facility/370
Display Management System for CMS
Logic
- GC24-5195-0 IBM Virtual Machine Facility/370
VM/Interactive File Sharing for CMS
General Information
- SC24-5196-0 IBM Virtual Machine Facility/370
VM/Interactive File Sharing for CMS
Guide and Reference
- LY24-5205-0 IBM Virtual Machine Facility/370
VM/Interactive File Sharing for CMS
Logic
- BO-03-78-01 RPG-II and VSAM-KSDS Variable Length Records (WTSC
Boeblingen April 17, 1978)
- BO-03-78-03 RPG-II and VSAM Sample (WTSC Boeblingen April 17, 1978)
- BO-07-78-03 RPG-II Batch to Online Conversion Guide (WTSC Boeblingen
September 29, 1978)

10.6 FORTRAN IV LIBRARY OPTION 1 (5746-LM3)

Status	: Program Product
Latest Release	: 3.0
Announced	: 1/79
Planned Availability	: 1/80
Manuals Available	: Licensed publications will be updated. Current user will automatically receive them plus a memo to users.
SIPO's Available	: No
Selected Product Line	: No
Installation License	: N/A
Pre-Installation Test	: 30 days
Program Services	: Central Service until discontinued upon 12 months written notice. Local Service at no charge until 12/31/79 then Local Licensed Program Support.

10.6.1 PRODUCT FUNCTION

This modification to Release 2 of the DOS FORTRAN IV Library Option 1 Release 3.0 will give users of DOS/VSE with VSE/Advanced Function Release 2 access to Fixed Block disk devices through DOS FORTRAN IV. The rest of the functions have not changed from the Release 2. For instance, the I/O support is equivalent to that available with Library Option 1 Release 2:

- 2560 Multi-Function Card Machine
- 5425 Multi-Function Card Unit
- 3505 Card Reader
- 3525 CardPunch

- 3410 Magnetic Tape Subsystem
- 3420 Magnetic Tape Subsystem

- 3330 Disk Storage Facility
- 3340 Disk Storage Facility
- 3330-11 Disk Storage Facility
- 3350 Disk Storage Facility

- Improved tape and disk utilization with larger block sizes

- ASCII magnetic tape standards for interchange of data

10.6.2 PRODUCT USE

Four important points should be noted :

- (i) Library Option 1 is required for object time support of the 3330, 3340, 3350, 3330-11, and Fixed Block devices.

Support of the Fixed Block disk devices requires the device support provided by the DOS/VSE with VSE/Advanced Functions Release 2.

- (ii) Fixed Block device support provides support for user sequential data sets. Direct access data sets are not supported.

- (iii) DOS FORTRAN IV Library Option 1 Release 3 will provide the same support for I/O devices provided by Release 2 on DOS/VS Release 34 and will run on DOS/VS Release 34. The Library Option 1 can be used to

The Library Option 1 can be used to provide ASCII support of System/360 and System/370.

- (iv) The Library Option 1 requires:

- DOS FORTRAN IV Compiler (360N-FO-479)
- DOS FORTRAN IV Library (360N-LM-480)

These Type 1 components must be ordered as part of the DOS Type 1 Tape (5731-AA1).

DOS FORTRAN IV LIBRARY OPTION 1

DEPENDENCY ITEM	REMARKS
Program Prerequisites - DOS FORTRAN IV Compiler (360-FO-479) - DOS FORTRAN IV Library (360-LM-480)	
Unit Record Support. At least a card reader (or equivalent) and line printer (or equivalent) are required. - 3505 CR - 3525 CP - 2526 MCM - 5425 MCU - 3211 Printer	
Tape Support - 2400 series - 3410 - 3420	
DASD Support - 2311 - 2314 - 3330-1,2 - 3333 - 3330-11 - 3340 - 3344 - 3350 - 3310,3370 FBA disk types for SAM ONLY.	
Data Management Support. - SAM - DAM Not for FBA disk devices.	

NOTE : The Floating Point feature MUST be installed on the CPU.
 The Decimal and Floating Point Instruction sets are required.

SUPERVISOR GENERATION OPTIONS

The following is a breakdown of the mandatory and suggested Supervisor parameters within the Supervisor generation macros :

MACRO	PARAMETER	REMARKS
SUPVR	ASCII=YES	Only if ASCII support needed.
CONFIG	FP=YES	Required
VSTAB	VSIZE=nK	As required by the system when in 370 mode

10.6.3 PRODUCT REFERENCE MATERIAL

- SC28-6883- DOS FORTRAN IV Library Option 1
Programmer's Guide Supplement
- GC28-6882- DOS FORTRAN IV Library Option 1
Program Product Specifications
- GC28-6596- DOS FORTRAN IV Library Subprograms
- GC28-6397- DOS FORTRAN IV Programmer's Guide
- GC28-6515- IBM S/360 and s/370 FORTRAN IV Language

10.7.1 SUMMARY OF RELEASES

Development Management System/CICS/VS-DOS replaces DMS/VS-DOS for DOS/VS and DOS/VSE. This program product simplifies application development using the 3270 Information Display System (IBM 3270 IDS). It aids in the design and generation of online application programs by providing a simple, still powerfull, interface to the user.

The simplified application development and increased programmer productivity can be achived through the facilities already included in the DMS/VS-DOS Version 1 Modification 1 and the new support and capabilities offered by the new DMS/CICS/VS-DOS Releases 1.0, 2.0 and 3.0. An exception to this is the support of DL/I-Entry wich is not incorporated in the new Program Product.

DMS/CICS/VS-DOS is written in S/370 Language. The online components operates under control of CICS/DOS/VS (5746-XX3) Version 1.4 and subsequent releases unless otherwise noted. Thus, support of S/370 and 4300 Processors, and devices under DOS/VS and DOS/VSE is provided through CICS/DOS/VS 1.4 and its Independent Component Release (ICR).

For the database user DL/I-DOS/VS (5746-XX1) Version 1.4 and its ICR will provide the proper support under DOS/VS and DOS/VSE.

Note that the ICR's mentioned above permits the access to Fixed Block type of devices, under DOS/VSE, for the FBA devices attached to a 4300 Processor.

Since the batch components use VSAM, SAM and SYSIPT/SYSPCH/SYSLST device access, there is no incompatibility to process them under DOS/VS and DOS/VSE accessing the supported input/output devices.

DMS/CICS/VS-OS (5740-XC4) operating under control of CICS/OS/VS Version 1.4 (5740-XX1) plus the optional IMS/VS-DB product, offer to the OS user the same DB/DC capabilities available to the DOS user. The very few exceptions to this rule will be noted at the proper places in the discussion that follows.

The following line items highlight the facilities provided by both, DMS/VS-DOS and DMS/CICS/VS-DOS.

DMS/VS-DOS LINE ITEMS

DMS/VS-DOS Version 1 Modification 0 (delivered by end 1976).

- DMS-II (DOS/VS) functions are preserved.
- Formatted Terminal Data facilities for Post-Response and Pre-Display program exits written in COBOL, PL/I and Assembler languages.
- DLI-DOS/VS and DL/I-Entry enhancements for inquiry, update/delete and add single or multiple segments (not repeated) from/to up to four databases from a panel. Windowing through a \$RPT function to inquiry repeated segments of one type from one database only.
- Secondary Index access for DL/I-DOS/VS and DL/I-Entry databases, and Alternate Index access to VSAM datasets.
- Program Function keys correspondences to Function Identifiers coded on the same panel.
- Global Target (&) panel exit for added security and independence in the design and code of panel hierarchy.

DMS/VS-DOS V1 M1 PTF0001 (delivered by end 1978).

- Performance enhancements reducing the number of CICS calls made by DMS/VS:
 - . panel loads and deletes are optimized.
 - . CICS LINK, XCTL and RETURN macros are replaced by Branch-and-Link/Branch instructions when control is transferred to/from the majority of the DMS/VS online modules.

DMS/VS-DOS V1 M1 PTF0002 (delivered beginning 1979).

- Support of the additional Program Function keys and the various screen sizes available for 3276 and 3278 display terminals.

DMS/CICS/VS-DOS LINE ITEMS

DMS/CICS/VS-DOS Release 1.

- DMS/VS-DOS features with the exception of the DL/I-Entry support. Compatibility is preserved.
- DMS/CICS/VS is divided into two components charged separately:
 - . Base Product
 - . Application Generation Feature
- Central and Local Services provided by IBM Customer Engineering. Local Service at no charge till December 31, 1979. Then, Local Licensed Program Support is available to be provided by IBM Customer Engineering.

DMS/CICS/VS-DOS Release 2.

- Calculations and edit capabilities to perform simple arithmetics (add, subtract, multiply and divide); to validate fields (move, compare/test, table lookup, de-editing and setting of attribute bytes); and to control operations (execute new function identifier, write-no-erase, terminate a transaction, etc.).
- DL/I enhancements to display repeated under repeated segments, and variable length segments.
- Add and Update files and/or data bases across multiple panels. This is provided for DMS/CICS/VS-OS under Release 1.0.

DMS/CICS/VS-DOS Release 3.

- File and data base support through CICS/VS Inter System Communication (ISC) facility.
- RPG-II Command Level support of FDF (Formatted Data Facility).
- DMS/DBD generation from input supplied by the DB/DC Data Dictionary Program Product.
- Interactive Application Definition facility to allow for Data Definition, Panel Definition and Process Definition in an interactive, online environment.

10.7.2 PRODUCT FUNCTION

Development Management System/CICS/VS-DOS is an Application Management Offering which simplifies the implementation of online applications, using the 3270 Information Display System. Use of DMS/CICS/VS can dramatically improve the productivity of application and system programmers responsible for implementing and maintaining online applications. This may reduce application design and implementation time from months if using BTAM or VTAM, or weeks if using CICS/VS Basic Mapping Support and Command Level, to days when using the DMS/CICS/VS approach. Ease of use of the DMS/CICS/VS precoded CICS/VS, , DL/I, VSAM and other functions makes this possible.

DMS/CICS/VS facilitates the generation of an application, via the Application Generation Feature. This consists of two stages. An Application Definition Facility and a Batch Application Generation. The first one helps the user to specify the DMS/CICS/VS online and offline requirements of the application and puts these specifications into the Application Specification Library. From there, the second will generate the tables to be interpreted by the DMS/CICS/VS online and offline components of the Base Product when the actual execution of the application takes place.

The Application Definition facility comes in two versions, offline and online. The Offline Application Definition takes place through use of supplied fill-in-the-blanks forms. The Interactive Application Definition provides an online prompting capability to specify the application to DMS/CICS/VS.

Pre-programmed facilities are provided for application selection, screen paging and basic file and database operations such as data entry, inquiry, update, search and browsing.

One of the most important components of DMS/CICS/VS is the Display Management Transaction, a set of Assembler modules that constitute an online application running under control of CICS/VS. This application creates panel images and sends them to the terminals, receives responses and data from terminals and performs requested functions. These functions can be input/output operations against files/databases followed by the display of required information on terminals. They may also cause the routing of messages or

screen contents to other terminals or, they may provide for some system services, invocation of user provided routines, and so on.

10.7.3 PRODUCT USE

PANEL DESCRIPTION TABLE

To perform its functions, the Display Management components need tables which specify the data required and functions to be performed.

One of the most important tables is the Panel Description Table. This contains information that describes :

- (i) Layout and contents of a 3270 IDS panel screen.
- (ii) The files and/or databases to be accessed and the record and/or segment fields used.
- (iii) The functions that the Display Management Transaction has to perform, dependent on different operator responses.
- (iv) Use of user written routines for special cases, etc.
- (v) Calculations, validation/editting, control of flow based on special conditions detected.

To build this type of table DMS/CICS/VS provides an offline Batch Application Generator utility called the Panel Table Description generator and a form called the Panel Description Form to be filled-out by the user. An alternate to this filling-in-the-blanks approach is offered by the online Interactive Application Definition

To code an online application using DMS/CICS/VS, the user must fill out this form with the specifications of his application design, have the form converted to 80 column card images and then process it with a Batch Application Generator utility, which will convert it into a Panel Description Table. Once a set of panels has been processed in this way, the application (or part of it) is ready to test. To do this, the user signs on under CICS/VS, then he keys in the transaction ID (for the DMS/CICS/VS Display Management transaction) followed, by the panel ID of the first panel in the application sequence and one of the 24 security key values (01 to 24) associated with the user sign-on for CICS/VS.

This first (Main Menu) panel is displayed, and the operator at the terminal is able to work with a new application coded in a short period of time. Modifications during testing and maintenance along the life of the application, follow the same productive approach. Documentation for the application is done mainly during filling out the forms and thus it becomes an automatic by-product of the DMS/CICS/VS application design/specification/code/test cycle. For a list of DMS/CICS/VS precoded functions, refer to the General Information Manual.

An alternative to this filling-in-the-blanks approach is offered by the online Interactive Application Definition facility.

FILE/DATABASE SUPPORT

Up to four files/databases may be accessed per panel. All the files supported are fixed length records within blocked or unblocked data sets. Organizations and file processing functions supported, are :

(i) SAM, Sequential files

Online : - supported as extra-partition data sets for input or output but not simultaneously.

- INQUIRY, retrieving next record. In sequential processing is possible to display one or more records in a panel and also to produce several panels to display these records.

- DATA ENTRY, entering next record.

Offline : - INPUT, for updating or creating other file organizations (DMXUPDTE utility).

- OUTPUT, for storing selected records from other file organizations (DMXPURGE utility).

(ii) VSAM-KSDS, VSAM Key Sequenced Data Set files

Online : - INQUIRY, retrieving selected records by key or retrieving next record once sequential processing is set (BROWSING). When in sequential processing, it is possible to display one or more records in a panel and also to produce several panels to display these records.

- UPDATE, of individual records.

- DELETE, of individual records.

- DATA ENTRY, of individual records.

Offline : - INPUT, for selection of records to be logged to a SAM file (DMXPURGE

utility). The selected records may optionally be deleted and logged.

- OUTPUT, for updates and additions of records from a SAM file (DMXUPDTE utility).

(iii) ISAM Indexed Sequential Access Method files

Online : - INQUIRY, retrieving selected records by key or retrieving next record once sequential processing is set (BROWSING). In sequential processing is possible to display one or more records in a panel and also to produce several panels to display these records.

- UPDATE, of individual records.
- DELETE, of individual records. This is not supported but it is possible to flag a record for deletion by use of the UPDATE function or via a user routine in an exit. The actual DELETE is accomplished through reorganizing the ISAM file. The OS/VS users may utilize the IEBISAM utility.
- DATA ENTRY, of individual records.

Offline (only for DMS/CICS/VS-OS):

- INPUT, for selection of records to be logged to a SAM file (DMXPURGE utility). The selected records may optionally be flagged for deletion and logged. Actual deletion can be accomplished through reorganizing the ISAM file using the IEBISAM utility. DMS/CICS/VS-OS only.
- OUTPUT, for updates and additions of records from a SAM file (DMXUPDTE utility). DMS/CICS/VS-OS only.

(iv) DMS/CICS/VS INDEXED files

This is a special organization supported by DMS/CICS/VS, to provide rapid access and search capability for small, unordered, active files. In DOS, it consists of two VSAM ESDS (VSAM Entry Sequenced Data Set) files. One is the Index File and contains the key and optionally, selected fields of the data record which can be used as search arguments instead of the primary key during inquiry. The second is the Reference File which contains the data records. In OS, both files are implemented using Basic Direct Access Method (BDAM).

Records in both files are in corresponding relative positions. Data records are always accessed through the index file. Once the index record is found, the corresponding data record is accessed from the reference file by means of the VSAM RBA (Relative Byte Address) calculated by DMS/CICS/VS-DOS or, the Relative Record Number calculated by DMS/CICS/VS-OS. It is important to realize that each access to the file is through the Index File, beginning at the first index record, until the required one is found. Then the Reference File can be accessed.

Keys are unique but unordered in the file, and additions are made always at the end of the file. This type of file with a highly blocked index may be used for :

- (a) Maintaining records that are not ready to go to a master file, suspense files, and
- (b) Highly active daily work files, exception files and other temporary files.

The file processing function supported are:

- Online : - INQUIRY, retrieving of selected record by key.
- SEARCH, retrieving of lists of keys or records using a search argument (may be generic) to be compared with the data in the Index File.
- UPDATE, of individual records.
- DATA ENTRY, of individual records.

Offline : - Same functions as for VSAM-KSDS and,

- INITIALIZATION, preformatting of Index and Reference files (DMXBI utility).

(v) USER files.

These consist of any file/database organization the user desires to access. The user has to provide a program to perform all input/output file/database and terminal online operations. He must provide for the offline support also.

(vi) DMS/CICS/VS GROUP files

This is a special grouping of DMS/CICS/VS Indexed files. One logical Indexed file may be divided into two or more physical sub-files. A table containing the highest key values for every sub-file (Group File Table) is kept in virtual storage acting as a higher level index to speed access online.

The availability of the file is increased because the failure of one sub-file does not affect the other sub-files. The online and offline functions supported are the same as those for DMS/CICS/VS Indexed files.

The SEARCH function can be used online to inquiry into only one sub-file at a time.

USER files can be grouped.

The OS user may group DMS/CICS/VS Indexed, USER, VSAM-KSDS and ISAM files.

(vii) Data Base support

DL/I data bases are supported by DMS/CICS/VS through DL/I DOS/VS. Logical relationships and secondary indexes are supported. Up to four data bases and/or files may be accessed per panel.

The OS user has the option to use the IMS/VS-Data Base Program Product to implement the corresponding DL/I database support under CICS/OS/VS.

Online : - INQUIRY, for a single segment or for multiple segments of different types and/or for repeated occurrences of segments of the same type, per panel.

Repeated occurrences of segments may be displayed more than once using the same panel image which can then be paged FORWARD/BACKWARD by the operator. (This is called WINDOWING in the DMS/CICS/VS terminology).

Only one repeated segment type can be WINDOWED by panel in Release 1.0. The retrieval of repeated segment types, including repeated under repeated segment types, is supported under Release 2.0.

- SEARCH acts like INQUIRY to display repeated segments of the same types but, a search argument is provided for selection of segments. The output can be limited to a certain amount of segments selected or number of pages to be displayed.
- UPDATE, a single segment or multiple segments of different types, Release 1.0. Under Release 2.0 the update of repeated segments is supported as well.
- DELETE, a single segment or multiple segments of different types.
- DATA ENTRY, of a single segment or multiple segments of different types, Release 1.0. Under Release 2.0 the addition of repeated segments is supported as well.

Offline : - No support is provided.

In summary, file/data base support is as described above, and dependent upon the SAM, VSAM, DL/I and CICS/VS support for the type of devices required.

TERMINAL SUPPORT

The IBM 3270 Information Display System devices are the only type of terminals supported by DMS/VS. A table, later in this section, summarizes the available support.

DATA MANAGEMENT SUPPORT

Apart from the input/output support indicated for files and data bases, the following data manipulation is handled by DMS/CICS/VS :

(i) Online :

- automatic conversion of file/database field mode to terminal character field mode and vice versa. File/database modes may be :

- . 'C', character (no conversion needed)
- . 'P', packed decimal
- . 'F', fixed binary
- . 'X', hexadecimal

- data editing and de-editing of numeric and character mode fields based on a mask selected from a table of masks. One table is provided with DMS/CICS/VS. The user may extend/change this and/or create new ones.

- data decoding. Through tables created by the user, the binary contents of a one byte data field, can be displayed on the screen of a terminal as a more comprehensive text (EXCLUSIVE TRANSLATE). A similar facility permits the decoding of one/two byte/s data fields into one to eight/sixteen lines of text, depending on which bits are on in the data field (INCLUSIVE TRANSLATE). The opposite, that is the screen to file encoding is not supported.

(ii)Offline :

- automatic conversion of file field mode, when using the DMXUPDTE utility for creation, addition or updating of a DMS/CICS/VS file from a SAM input file.

- adding/subtracting numeric fields in a SAM file to/from the corresponding fields of a DMS/CICS/VS file (via DMXUPDTE utility). Replacement of the corresponding fields by the specified ones in the SAM file is also possible.

- no conversion is provided by the DMXPURGE utility since this produces only copies of the selected records from a DMS/CICS/VS file to a SAM file. No conversion is provided by the DMXBI utility since this preformats the Index (with X'FF') and the Reference (with X'00') files of a DMS/CICS/VS Indexed file.

USER WRITTEN ROUTINES

For facilities not provided by DMS/CICS/VS, like data validation, special en-coding/de-coding, editing/de-editing and calculations, DMS/CICS/VS provides interfaces and linking facilities to eassy the task of coding user routines. This user coding should be substancially réduced when in Release 2.0 CALCULATION OPERATIONS are used instead. The user routines may be coded in Assembler, COBOL or PL/I languages, using a common and standard approach for including CICS/VS and DMS/CICS/VS areas, record/segment and terminal input/output areas as dummy sections, and providing for addressability. In this way, although almost any CICS/VS Macro/Command can be used, the Application Programmer only needs to use the DFHPC TYPE=RETURN macro or the RETURN command of CICS/VS to return control to DMS/CICS/VS at the logical end of the routine.

For RPG-II the facility to generate the Formatted Terminal Data Area will be included in Release 3.0. Nevertheless, there is no present restriction to code user routines using RPG-II CICS/DOS/VS Command Level interface provided the proper RPG-II DATA STRUCTURES and addressability are set. There is one example of this in the chapter 'Coding Examples' below.

The different types of routines, in order of increasing difficulty to code, are :

(i) FIELD LEVEL exit. When a field is moved from a file/database to the screen or vice versa, a user routine can be invoked to examine or modify the field.

(ii) PANEL and DL/I exits. User routines can be invoked at different points in time during the DMS/CICS/VS applications flow, like :

(a) (post-response) After a response from the terminal operator. At this point in time, the information entered at the terminal is available, and the fields from the panel may be addressed by name in an area formatted by DMS/CICS/VS, instead of using the complex terminal I/O area format. The area is called Formatted Terminal Data Area (FTDA) and the functions are called Formatted Data Facility (FDF). The user may check data, make calculations, modify

data entered at a terminal, and decide further action like redisplay the panel with a message, perform a different function (as if the operator had given a different response), or continue normally, etc.

(b) (pre-display) Prior to displaying the panel. The FDF facility may also be used to make calculations and modify or check data that will be displayed. As in the post-response exit, access to the Terminal Input/Output Area (TIOA) or FTDA and decision on further action are possible.

(c) (pre-read) Prior to a read from a file/database. At this point the key and, for a database, the segment search arguments (SSA's), may be accessed for checking or modification, before the read takes place. The user may code this exit to perform his own read access from a non-supported file, and let DMS/CICS/VS to edit the data into the panel to be displayed. Further action as indicated to DMS/CICS/VS by the routine.

(d) (post-read) After a read from a file/database. The user may access the file/database work/input-output area containing the data which was read, and may address file/database fields by name and perform any desired calculations and editing of data, before the data is displayed. Further action as indicated to DMS/CICS/VS by the routine.

(e) (pre-write) Before writing to a file/database. Data is available in the work/input-output area for further checking, calculation and modification and may be addressed by name as in the post-read exit. File organizations not supported by DMS/CICS/VS may be processed by the routine at this time, but the complex movement of data from the terminal area to the file work area can be performed by DMS/CICS/VS. Further action as indicated to DMS/CICS/VS by the routine.

(f) (post-write) After writing to a file. At this point the user may check the status of the input/output operation. This is not available for DL/I DOS/VS databases. Further action as indicated to DMS/CICS/VS by the routine.

(iii) USER FILE PROCESSOR program. Standard CICS/VS interface is provided for the user to write a complete I/O module to handle special non-supported file organizations. The recommendation is to use the module DMXVP (VSAM-KSDS File Processor Model) as a model.

(iv) CALL USER program. A complete CICS/VS transaction may be coded to be invoked by the operator at response time, to perform any desired function.

(v) USER SUPERVISOR FUNCTION interface. A complete new module may be coded by the user and linked with DMXSM (DMS/CICS/VS Supervisor Module), to extend the functions provided.

In spite of the many alternatives offered, a normal installation can be implemented using only Panel and DL/I user exits and furthermore, the majority of the application requirements can be fulfilled by post-response, pre-write and post-read exit types.

DMS/CICS/VS TRANSACTIONS

Apart from the Display Management transaction mentioned earlier, and supported by DMS/CICS/VS as well, DMS/VS provides two other sets of modules associated with two more transactions that operate under CICS/VS. They are :

(i) Message Switching transaction, which allows the routing of messages, panel data and panel images among terminals. This is also provided by DMS/CICS/VS.

(ii) File Rebuild transaction, which permits the reconstruction of files based on backup copies of files and sequential files containing changes logged by CICS/DOS/VS Automatic Journalling Facility or under OS by a facility provided by DMS/VS-OS. Based on the current information available we may conclude that this function will be not provided by DMS/CICS/VS.

No rebuild facilities are provided for databases. For them, the standard CICS/VS and, DL/I-DOS/VS or IMS/VS-DB procedures must be used.

DMS/CICS/VS OFFLINE COMPONENTS

Apart from the various modules associated with the online transactions, DMS/CICS/VS provides six offline programs. Three of them are the Offline Application Definition and Batch Application Generation facility of the Releases 1.0 and 2.0. (The Offline Application Definition will be enhanced by the Interactive Application Definition in Release 3.0. The other three are the Batch Utilities DMXBI, DMXPURGE and DMXUPDTE that were addressed before.

The first ones are used to create tables which will direct the online activities of the DMS/CICS/VS Transactions and the offline activities of the Batch Utilities.

The other three are used to create, maintain and update DMS/CICS/VS files. They operate with VSAM KSDS, ISAM (for DMS/CICS/VS-OS only) and DMS/CICS/VS Indexed files, but are particularly oriented to the latter files, because of their unique DMS/CICS/VS format.

MINIMUM CONFIGURATION

Any minimum DOS/VS or DOS/VSE in a S/370 or 4300 configuration supporting a CICS/DOS/VS partition, can be used to install, test and maintain DMS/VS. At least one 3275/6/7/8 Display Station (local or remote) is needed to validate the installation.

The following figures may be used for planning purposes for a typical small configuration using 10 terminals and one transmission line. They do not include resources needed for other components like access methods, CICS/DOS/VS, DL/I DOS/VS, etc.

DMS/CICS/VS Component	Virtual Storage Requirements Kilobytes	Additional Kilobytes per unit
DMS/CICS/VS Online Supervisor	6	
DMS/CICS/VS loadable modules	47.5	
Transmission line buffer	2	2 per line
DMS/CICS/VS terminal working storage	15	1.5 per terminal
DMS/CICS/VS transaction work areas	9	.9 per terminal
Total, in Kilobytes	79.5	

If the DMS/VS-DOS File Rebuild module is used to reconstruct files using CICS/DOS/VS automatic journalling facilities, it occupies 6.7K bytes of virtual storage. The DMS/VS-OS File Rebuild module occupies 2K bytes of virtual storage. Remember, it seems that the File Rebuild facility is not going to be implemented under DMS/CICS/VS.

When a panel is accessing a DL/I database, the following dynamic storage is also needed (bytes) :

$$500 + \frac{\text{sum of segment lengths in the DMS/DBD}}{\text{number of segments accessed}} + 178 \times$$

NOTE : DMS/DBD is the Data Base Definition block, similar to a DL/I DBD but, it is used to describe to DMS/CICS/VS the logical database record structure containing detailed description of fields within segments.

For planning purposes, the following figures for real storage (working set) may be used (Kbytes) :

DMS/CICS/VS online supervisor	4.6
DMS/CICS/VS loadable modules	26
Transmission line buffers	2 per line

Additional real storage requirements are imposed by DMS/CICS/VS terminal working storage and transaction work areas, the access methods, CICS/VS modules and tables and DL/I modules, input/output buffers. Other applications and user written routines (as used in DMS/CICS/VS exits) must be considered as well.

For more detailed information on storage requirements, the reader is referred to the DMS/CICS/VS literature listed in the references, and also to the appropriate CICS/VS, DL/I and DOS/VSE manuals.

DMS DOS/VS DEPENDENCY INFORMATION CHART

DEPENDENCY ITEM	REMARKS
Program Prerequisites	
- CICS/VS	Required with support for 3270.
- DL/I DOS/VS	Required to provide database support
- DOS/VS COBOL	Required to provide user exits in COBOL
- DOS PL/I OPT	Required to provide user exits in PL/I
- Assembler	Required to provide maintenance and Assembler exits
- DOS/VS RPGII	Required to provide user exits in RPG-II
UNIT RECORD Support	
	Any DOS/VS and DOS/VSE supported device (offline). Any CICS/VS supported device (online).
TAPE Support	
	Any DOS/VS and DOS/VSE supported device (offline). Any CICS/VS supported device (online).
DASD Support	
- 2311	Any DOS/VS and DOS/VSE supported device (offline). Any CICS/VS supported device (online).
- 2314	
- 2319	
- 3330-1,2	
- 3333	
- 3330-11	
- 3340	
- 3344	
- 3350	
- 3310, 3370	Fixed Block Module devices for VSAM and SAM files and for DL/I databases.
DATA MANAGEMENT Support	
- SAM	Supported as CICS/VS extrapartition data sets online and as I/O for the Batch Utilities offline.
- ISAM	Supported as CICS/VS ISAM data set online.
- VSAM	KSDS for inquiry, browse, update, delete, data entry online and I/O for the Batch Utilities offline.
- DMS INDEXED	Unique to DMS/CICS/VS consisting of one VSAM ESDS for INDEX and other ESDS for REFERENCE data. They may be used for inquiry, search, update and data entry online and, Input/Output for the Batch Utilities offline.
- GROUP FILE	Special grouping of DMS INDEXED with same support.
- DL/I DOS/VS	For data base support of inquiry, search, update, delete and data entry online. No offline functions provided.
CURRENT RELEASES	
1.0	Providing similar support as DMS/VS V1 M1 PTF0002

Compatibility Information

- (i) CICS/VS DMS/CICS/VS will be pregenerated under CICS/VS 1.4 versions and is supported by follow on versions unless otherwise noted.
- (ii) DL/I DOS/VS DMS/CICS/VS has been requested to be validated DL/I DOS/VS Version 1.4 and subsequent ones unless otherwise noted.
- (iii) DOS/VS and DOS/VSE support is preserved as long as the online components of DMS/CICS/VS are accessing CICS/VS datasets. The offline compatibility is preserved since the offline components of DMS/CICS/VS are accessing VSAM, SAM and SYSIPT/PCH/LST datasets.
- (iv) DMS/VS No change is required for operation under the new DMS/CICS/VS environment. Any existing DMS/VS user routine must be recompiled. Notice that the large screen support provided by DMS/VS-DOS V1 M1 PTF0002 forced a recompilation of any existing DMS/VS panel.
- (v) DMS/DPPX, DMS/3770, DMS/3790, DMS/CMS and DMS/CICS/VS are very similar in concept. However, because of the unique requirements of the individual systems, there are incompatibilities among the DMS products.

SUPERVISOR GENERATION OPTIONS

The following is a breakdown of the mandatory and suggested Supervisor parameters within the Supervisor generation macros :

MACRO	PARAMETER	REMARKS
SUPVR	TP=accs.method	BTAM or VTAM
NOTE. The following FOPT parameters, recommended for DOS/VS, are standard under DOS/VSE:		
FOPT	PFIX=YES GETVIS=YES PCIL=YES RELLDR=YES VSAM=YES	If BTAM runs in V=V mode. Forced by TP=VTAM. Forced by VSAM. Support for Private Core Image Library. Required Required for VSAM support
VSTAB	VSIZE=nK	As required by the system when in 370 mode
IOTAB	NRES=n	As required by VSE/VSAM (from NPARTS+2 to 255)

NOTE : See Chapter CICS/VS for additional CICS/VS requirements.

DMS/CICS/VS TERMINAL SUPPORT

DMS/CICS/VS supports only the IBM 3270 Information Display System, via local or remote attachments. This is supported via normal CICS/VS Terminal Control. The following table summarizes the supported 3270 components and features :

TERMINAL	REMARKS
3790	Data Stream Compatibility (DSC) mode. Configuration Support Number 9165 and 9169).
3271 3272 3274	Remote Control Unit Models 1, 2, 11, 12. Local Control Unit Models 1, 2. L & R Control Unit Models 1A, 1B, 1C (SNA or BSC).
3275 3277 3276 3278	Display Station Models 1, 2, 11, 12. Display Station Models 1, 2. CU Display Station Models 1, 2, 3, 4, 11, 12, 13, 14. Display Station Models 1, 2, 3, 4.
3284 3286 3287 3289	Printer Models 1, 2. Printer Models 1, 2. Printer Models 1, 2. Printer Models 1, 2.
3270	Options as follows: - EBCDIC Transmission Code (#9761) - ASCII Transmission Code (#9762) - Keyboard Numeric Lock (#4690) - Selector Light Pen (#6350) - Operator ID Card Reader (#4600) - Audible Alarm (#1090)

NOTE: Remote attachments via BSC or SDLC disciplines and provided through CICS/VS using the proper BTAM/VTAM Program Products. Refer to the CICS/VS chapter.

10.7.4 DMS/VS-DOS FACILITIES IN DETAIL

DMS/VS-DOS V1 M1 PTF0001

Changes have been made to reduce the number of CICS calls made by DMS/VS. These have reduced the path lengths in two general categories

- . Panel loads and deletes have been optimized. For instance, there was the case where a Panel Table was deleted and reloaded immediately. This has been eliminated.

- . New DMS/VS macros have replaced many CICS/VS CALL macros. Logic has also been incorporated in the DMS/VS modules that reference them. There are three exceptions:
 - No change to the CALL's to user written programs.
 - No change to transfer of control between DMS/VS and CICS/VS.
 - Same few CALL's between DMS/VS modules have not been changed.

These changes require that DMS/VS has access to the online module addresses and new register save areas to link/return to/from the modules. The additional space is provided from the Common System Work Area (CSWA) and the Transaction Work Area (TWA) as follows:

- CSWA size = 480 bytes. The first 256 bytes can be shared with other programs. The last 224 bytes are of exclusive use of DMS/VS and must not be overlaid. This change may force some users to respecify the CSWA size and recompile the online programs that uses the same area than DMS/VS. The new DMS/VS modules are precompiled.

- CWA size = 980 bytes. The users have to modify and recompile the Program Control Table (PCT) to respecify the new CWA size. The DMS/VS user written routines must be reassembled.

The new DMS/VS macros replacing CICS/VS LINK/XCTL/RETURN macros put consistent entries to the CICS/VS Trace Table as follows:

Word 1	byte 1	trace identification code	'D4'
"	bytes 2 to 4	general register 14	value
" 2			not used by DMS/VS
" 3		DMS/VS module name not including DMX prefix	
" 4		function: LINK, XCTL or RTRN	

When the first DMS/VS transaction is submitted, it will invoke either DMXTI (Transaction Initialization) or DMXST (Message Switching). Either of these modules performs a one time load of all the DMS/VS online modules and builds a table with their names and addresses in the CSWA. To insure that these modules remain in storage, the parameter LOADLST=NO is used. Then, since the modules are not loaded again during the CICS/VS run, the user must not delete a module nor he can use the Master Terminal transaction CSMT NEW,PGRMID=DMX... for a DMS/VS module.

If a module can not be loaded, the message 'DMS/VS MODULE DMX... NOT LOADED' is sent to destination 'CSMT' and, at the end of the loading process, the message 'ONE OR MORE DMS/VS ON-LINE MODULES NOT LOADED' is sent to the initiating terminal. Since not all the modules are required for operation by different user environments, the lack of some modules and the issuing of the correspondin warning messages, do not preclude the use of DMS/VS.

DMS/VS-DOS V1 M1 PTF0002

Changes have been made to support the additional Program Function Keys and screen sizes available with 3276/3278 display terminals. Once this PTF is installed, all existing panel descriptions must be regenerated/recompiled. No changes are required to existing panels unless they are going to make use of the new support.

Extensions made to the DMS/VS Panel Description Form are the following:

Display Description Card (type B):

- Column 1, model of the 3276/78 display as follows,

480	character	model	(compatibility mode)	'1'
1920	"	"	(compatibility mode)	'2'
960	"	"		'A'
1920	"	"		'B'
2560	"	"		'C'
3440	"	"		'D'
- Column 76 should contain a '1' if a second Display Description Card follows this one.

Second Display Description Card (type B):

- 12 Columns 60-71, space for the Function Identifier to be associated with Program Function keys Nos. 13 to 24.
- Column 76 must contain a '2'.

Display Line Cards (no card type):

- Up to 19 additional lines may be coded following line 24th ('0' in column 1). The code in column 1 can be 'P' to 'Z' or, '.', '(', '*', ')', ';', '/', ',', or '?' according to the screen size of the 3276/78 model to be used:

Lines 1-9 and A-C for	480/960 character
" 1-9 " A-O "	1920 "
" 1-9 " A-W "	2560 "
" 1-9 " A-Z and .(*);/,? for	3440 "

When specifying the CICS/VS Terminal Control Table (TCT) and the Program Control Table, the ALTSCRN (TCT) and SCRNSZE (PCT) operands must be coded for proper use of the 2560 and 3440 character displays.

This PTF have been pregenerated/tested using CICS/VS 1.4. The number and importance of pre-applied fixes is another reason for the users to install this tape.

10.7.5 DMS/CICS/VS-DOS FACILITIES IN DETAIL

DMS/CICS/VS-DOS RELEASE 1.

(i) DMS/VS-DOS features with the exception of the DL/I-Entry support. Compatibility is preserved. The precoded Supervisor Functions implemented are: (This functions are explained in the DMS/VS and DMS/CICS/VS General Information manuals indicated in the literature)

Display Control	Panel Overlay	'A'
	Panel Link	'B'
	Page Forward	'C'
	Page Backward	'D'
	Redisplay	'E'
	Panel Return	'F'

Database Operations	Inquiry	'G'
	Inquiry for Update	'G3'
	Segment Add	'H'
	Segment Update	'I'
	Segment Delete	'I1'
	Cursor Key Select (Inquiry)	'L'
	Cursor Key Select (for Update)	'L3'
End Inquiry	'P'	

File Operations	Inquiry	'G'
	Browse (ISAM and VSAM-KSDS only)	'G1'
	Seq. Not Record Found (not for SAM)	'G2'
	Inquiry for Update (not for SAM)	'G3'
	Record Add	'H'
	Record Update (not for SAM)	'I'
	Record Delete (VSAM-KSDS only)	'I1'
	Search (DMS/CICS/VS INDEXED only)	'J'
	Set Sequential (ISAM, KSDS only)	'K'
	End Sequential (ISAM, KSDS only)	'K1'
	Cursor Key Select (Inquiry)	'L'
	Cursor Key Select (for Update)	'L3'
	Sel. Pen Key Select (Inquiry)	'M'
	(DMS/CICS/VS INDEXED only)	
	Sel. Pen Key Select (for Update)	'M3'
	(DMS/CICS/VS INDEXED only)	
Copy (not for SAM)	'N'	
End Inquiry	'P'	

Data Routing	Data Route	'Q'
	Send Message	'R'
	Get Message	'R1'
	Send Image	'S'
	Get Image	'S1'
Program Control	Call User Program	'T'
	Return to Calling Program	'U'
	Reset Communication Fields (all)	'V'
	Delete Communication Fields	'V1'
	Reset Comm. Fields (Selectively)	'V2'
	End Transaction	'W'
System Services	Load PF Key Table	'X'
	Load Edit Mask Table	'X1'
	Set Time, Date, Operator ID	'Z'
	User Supervisor Function	'>'

NOTE: The File Rebuild Supervisor Function 'Y' of DMS/VS-OS is not supported under DMS/CICS/VS.

(ii) DMS/CICS/VS is divided into two components charged separately: a Base Product and an Application Generation Feature.

- Base Product. Contains the online functions and the offline maintenance facilities. That is, the online modules corresponding to the Display Management and Message Switching transactions, the Batch Utilities for creation/maintenance of files (mainly DMS/CICS/VS INDEXED files), and the macros to tailor the online DMS/CICS/VS environment.
- Application Generation Feature. Contains the offline modules to create the user applications from using the specifications coded on the DMS/CICS/VS forms. That is the batch modules that produces the different tables required by the operation of the Base Product.

This allows users with multiple locations to generate and test applications at a central site and run in an execution-only mode at other locations. That is, the development site will use the Base Product and the Application Generation Feature, while the other locations need only the Base Product for execution.

(iii) Central Service including the IBM Support Center, will be available until discontinued by IBM upon twelve month's written notice. Local Service, provided by IBM Customer Engineering, will be available at no charge till December 31, 1979. Then, Local Licensed Program Support is available (provided by IBM Customer Engineering) until discontinued by IBM upon twelve month's written notice.

DMS/CICS/VS-OS (5740-XC5) Central and Local Services provided until discontinued by IBM upon six month's notice. Local Service will be provided by IBM Customer Engineering.

This Program Service will dramatically impact the quality of the product and the service to the users compared to the Class B Program Service of DMS/VS.

The name of the product has been changed from Display to Development Management System/CICS/VS to better reflect its broad capabilities and full Service Support.

DMS/CICS/VS-DOS RELEASE 2.

(i) Calculations and edit capabilities to perform simple arithmetics (add, subtract, multiply and divide); to validate fields (move, compare/test, table lookup, de-editing and setting of attribute bytes); and to control operations (execute new function identifier, write-no-erase, terminate a transaction, etc.).

A new form for Calculation/Edit card type 'K' is provided to allow the specifications of the new functions to take place at predetermined Cycle Points. These Cycle Points are the same pre/post exit points used for the execution of user written subroutines. The new functions provided will suffice for the majority of the situations where a user written subroutine is needed. Nevertheless, should a special processing requirement appears, the Calculation/Edit functions allow for exit and return from/to the line sequence of function being executed at the corresponding Cycle Point.

The Calculation Operations available are (This functions are explained in the DMS/CICS/VS General Information manual indicated in the literature):

Arithmetic Operations	Add	'+'
	Subtract	'-'
	Multiply	'*'
	Divide	'/'
Move Operations	Move Remainder	MVR
	Move	MOVER
	Move Left	MOVEL
	Merge Terminal Data	MERGE
Testing/Compare Operatns.	Compare	COMP
	Test Display Field	TEST
Setting Field Attributes	Set a Field Attribute	SET
Table Lookup Operation	Search a Table	FIND

Control Operations	End Processing Cycle	EXIT
	Execute New Function ID	NEWFI
	Terminate a Transaction	TERM
	Terminate With a Dump	TERMD
	Write Error Message	WMSG
	Redisplay	RDISP
	Write no Erase	WNOER
Conversion Operations	De-edit	DEDIT

(ii) DL/I enhancements to display repeated under repeated segments. One of the benefits of this ability is simplification of the retrieval and updating of segments present in a Bill of Material database.

The ability to Add and Update repeated dependent segments.

Removal of the former restriction that precluded the intersperse of segments on a screen for Add and Update operations. Now the designer may decide whatever arrangement of segment fields on the screen.

Variable length segments will be supported.

The Inquiry 'windowing' operation can be stopped after a certain number of segments have been found or a certain number of pages have been built by DMS/CICS/VS. The operator can dynamically select the number to be used during the Inquiry.

(iii) Add and Update files and/or data bases across multiple panels. This provides the capability to capture data from a set (up to sixteen logically connected panels) of displays and to construct from this data a single logical file record or a single data base path for each one up four files or databases. The actual Add or Update occurs after the last panel of the set is processed. This is provided for DMS/CICS/VS-OS under Release 1.0.

DMS/CICS/VS-DOS RELEASE 3.

(i) File and data base support through CICS/VS Inter System Communication (ISC) facility. The ISC support provides the capability for connecting CICS/VS systems together. This allows an application running in one system to access files and DL/I databases, and initiate the processing of applications or queue messages on connected CICS/VS systems. These facilities give installations the ability to distribute system resources such as application programs, files, and databases in a network without the need for resources to be attached to the system that is requesting access to them.

(ii) RPG-II Command Level support of the Formatted Data Facility (FDF) as was provided in former Releases for the users of COBOL, PL/I and assembler languages to help in the code of post-response and pre-display exits. The FDF allows for symbolic access to data from the Terminal Input Output Area (TIOA) at the indicated exit or cycle points.

(iii) DMS/DED generation from input supplied by the DB/DC Data Dictionary Program Product. The user will not be required to code DMS/CICS/VS Assembler macro instructions for DMS/DBD generation. This should simplify definitios of DL/I applications. This optional facility requires the DB/DC Data Dictionary Program Product (DOS/VS: 5746-XXC, OS/VS: 5740-XXF).

(iv) Interactive Application Definition facility to allow for Data Definition, Panel Definition and Process Definition in an interactive, online environment. These facilities are incorporated in the Application Generation Feature of DMS/CICS/VS. They will allow for interactive (online under CICS/VS) specification of the application replacing the coding of forms and its batch process. The Application Generation Feature is composed of the Interactive Application Definition Facility (online) and the Batch Application Generation (offline). The first maintains the specifications in the a library called Application Specification Library. From there, the tables required by the Base Product are created by the Batch Application Generation.

Data Definition

Data Definition allows the user to describe the file organization, data structure, and field characteristics of files and databases like names, data attributes and layout.

All characteristics of the data structures are stored in the Application Specification Library from where they can be displayed/updated online.

The specifications are referenced during Panel Definition and Process Definition.

Panel Definition

Panel Definition allows the user to describe the general characteristics of the map and display device to be used. Then, the map format can be defined by keying the fixed and variable fields onto a model screen. This is done by keying the data in the same positions desired for actual display. Editing facilities are provided to insert/delete lines and to shift data.

Once the map is formatted, the user is prompted to describe the field attributes in the order they appear on the map. Default information is provided from the Data Definition stored in the Application Specification Library. The completed maps are stored in this library as well.

Process Definition

Process Definition allows the user to describe the DMS/CICS/VS functions to be performed for a particular map when is displayed. The user specifies the operator responses and associated Supervisor Functions, Calculation/Editing to be performed and exits (if still necessary).

HELP Facility

Additional information about the fields displayed at any stage of Interactive Application Definition can be requested by the user. Return to the current stage is automatic.

During the Definition process, the user may refer to other stages of the process and return to the current stage.

Recovery/Restart

The last Data, Panel or Process Definition is saved and automatically retrieved when the corresponding task is restarted after an interruption suffered either by the user or the system.

Completed new specifications can be saved as a test version until the user decides to replace the already existing version with the new one.

10.7.6 PRODUCT REFERENCE MATERIAL

DMS/VS REFERENCES

- GH20-4298- DMS/VS Program Product Design Objectives
- GV20-0658-0 DMS/VS Slide Set
- GH20-4516-0 DMS/VS Program Product Specifications
- GH20-1863-1 DMS/VS General Information Manual
- SH20-1865-2 DMS/VS Program Reference Manual
- SH20-1864-0 DMS/VS Operations Guide
- SH20-2024-0 DMS/VS Application Program Design Guide
- LY20-2266-0 DMS/VS Logic Manual
- 5746XC200 DMS/VS SECOM entries. Some tips and early information is also available. When pulling out entries use date instead of 'pin' number since some entries are re-updated on the same place.
- GX20-2000-1 DMS/VS Panel Description Form
- GX20-1999-0 DMS/VS File Description Form
- GX20-1998-0 DMS/VS Data Transfer Specification Form
- I76-002 DMS/VS Highlights, published by Central Region Support Intermediate Systems of Canada on November 22, 1976. Distributed by WTSC Boeblingen on publication B77-01, March 30, 1977
- B77-01 DMS/VS Applications and Systems Programming Workshop, Handouts published by WTSC Boeblingen March 30, 1977
- B77-06 DMS/VS Questions and Answers, published by the End User Product Center, Canada and distributed by WTCS Boeblingen September 9, 1977
- B77-07 DMS/VS & 3270 SDLC=YES, End Users Product Center, Canada. WTSC Boeblingen distribution November 10, 1977
- B77-07 DMS/DOS/VS and 3790 (3270 CM), End User Product Center, Canada. WTSC Boeblingen distribution November 10, 1977
- B78-01 DMS/VS and DL/I-Entry Secondary Indexes, working example

using secondary access to a database. Published by WTSC Boeblingen March 15, 1978

- B78-01 DMS/VS Application Design Guide, pre-release published by WTSC Boeblingen March 15, 1978
- B78-03 DMS/VS Papers, 14 papers from W. Brent Lingenfelter (DPD), published by WTSC Boeblingen April 17, 1978
- B79-01 DMS/VS. Standard System Design, Theory and Practice, paper from W. Brent Lingenfelter (DPD), published by WTSC Boeblingen February 9, 1979.
- B79-01 DMS/VS DOS/VS V1 M1 PTF0001 Tape, Performance Enhancements published by WTSC Boeblingen February 9, 1979.
- B79-03 DMS/VS DOS/VS V1 M1 PTF0002 Tape, Large Screen Support published by WTSC Boeblingen March 16, 1979.
- B79-03 DMS/VS Education Guide developed in the Dallas Education Center (DPD), published by WTSC Boeblingen March 16, 1979.

DMS/CICS/VS REFERENCES

- GH20-4510- DMS/CICS/VS Licenced Program Product Design Objectives
- GH20-2195-0 General Information Manual
- GE19-5246 DMS/CICS/VS Foil Presentation. Developed to introduce
DMS/CICS/VS to DP Managers and technicians

10.7.7 APPENDIX A

RPG-II EXIT CODING EXAMPLE

The following example shows how to code an RPG-II Command Level exit. The example performs the same function as the Assembler Sample in the DMS/VS Program Reference Manual (APPENDIX D and panel DMSSP120, figure 59), form SH20-1865-2.

```
// UPSI 01                                TRANSLATOR OUTPUT INTO SYS002
// EXEC DFHERP1$,SIZE=128K                COMMAND LANGUAGE TRANSLATOR FOR RPG-II
```

```
0101 H                                     C
01011H*
01012H* ----- CICS/VS EXEC COMMAND CALLS ONLY, NO DL/I -----
01013H* -----
```

```
0201 IDFHDM          DS
0202 I               1   4 SELPTR
0203 I               5   8 TWAPTR
0204 I               9  12 USRPTR
0205 I              13  16 FWAPRT
02051I*
02052I* ----- BASE LOCATOR FOR LINKAGE. POINTERS FOR THE DATA STRUCTURES (DS) -----
02053I* -----
```

```
0206 IDMXTWA        DS
0207 I               417 420 TUEPLA
02071I*
02072I* ----- DMS TRANSACTION WORK AREA DATA STRUCTURE. ONLY TWAUEPLA NEEDED -----
02073I* -----
```

```
0208 IDMXUSP        DS
0209 I               1   8 CICSAA
0210 I               9  16 USRPGM
0211 I              17  24 USRFNM
0212 I               25  25 USRRC
0213 I               26  26 USROPT
0214 I               27  27 USOPTM
0215 I               28  28 USRMSG
0216 I               29  29 USRFI
0217 I               30  30 RESRV1
```

0218 I		31	31	USRDME
0219 I		32	32	USREXT
0220 I		33	36	UFTDID
0221 I		33	34	USRDL E
0222 I		37	40	USRDA D
0223 I		37	40	USRACD
0224 I		41	72	USRRKY
0225 I		37	38	USRPLE
0226 I		39	40	USRFLE
0227 I		41	140	USRFLD
0228 I		141	144	USPBLK

02281I* -----
 02282I* COMPLETE DMS USER PARAMETER LIST DATA STRUCTURE
 02283I* -----

0229	IDHFWA	DS		
0230 I			B 65	680SLTAX
0231 I			B 69	720SLAMT
0232 I			B 101	1040SLAMT1
0233 I			B 133	1360SLAMT2
0234 I			B 165	1680SLAMT3
0235 I			B 197	2000SLAMT4
0236 I			B 229	2320SLAMT5
0237 I			B 261	2640SLAMT6
0238 I			B 293	2960SLAMT7

02381I* -----
 02382I* FILE WORK AREA DS. ONLY FIELDS OF INTEREST MUST BE SPECIFIED
 02383I* -----

0301 C	*ENTRY	PLIST		
0302 C		PARM	DFHEIE	
0303 C		PARM	DFHCOM	1
0304 C		PARM	DFHDUM	
0305 C		PARM	DMXTWA	
0306 C		PARM	DMXUSP	
0307 C		PARM	DFHFWA	

03071C* -----
 03072C* PARAMETER LIST TO ALLOW DMXTWA, DMXUSP AND DFHFWA ADDRESSABILITY
 03073C* -----

0308 C	ADDRESS	EXEC		13
0309 C	TWA	ELEM	TWAPTR	
0310 C		MOVE TUEPLA	USRPTR	
0311 C		CALL 'ILNSAD'		13
0312 C		MOVE USRDAD	FWAPTR	
0313 C		CALL 'ILNSAD'		13

03131C* -----
 03132C* STEPS TO MAKE DMXTWA, DMXUSP AND DFHFWA ADDRESSABLE
 03133C* -----

```

0314 C          SLAMT1    ADD  SLAMT2    WORKAR  90
0315 C          SLAMT3    ADD  WORKAR    WORKAR
0316 C          SLAMT4    ADD  WORKAR    WORKAR
0317 C          SLAMT5    ADD  WORKAR    WORKAR
0318 C          SLAMT6    ADD  WORKAR    WORKAR
0319 C          SLAMT7    ADD  WORKAR    WORKAR
0320 C          0.05      MULT WORKAR    SLTAX
0321 C          SLTAX     ADD  WORKAR    SLAMT
03211C*
03212C* ----- ARITMETIC OPRERATION TO ADD SALE AMMOUNTS ADDING 5 % TAXES -----
03213C* -----
    
```

```

0322 C          BITOF'11111111'USRRC
0323 C          RETURN    EXEC          13
03231C*
03232C* ----- USER RETURN CODE IS SET TO X'00' AND RETURN TO DMS -----
03233C* -----
    
```

```

/*
// UPSI 01          COMPILER INPUT FROM SYS002
// OPTION CATAL
// PHASE DMSSPUX1,* REPLACES SAMPLE EXIT PRM PANEL120 AND APPENDIX D
// INCLUDE DFHERI   INCLUDE EXEC INTERFACE STUB FOR RPG-II
// EXEC RPGII,SIZE=80K
// EXEC LNKEDT
    
```

10.8 DOS/VS SORT/MERGE VERSION 2 (5746-SM2)

Status	:	Program Product
Latest Release	:	2.0
Announced	:	1/79
Planned Availability	:	4/79
Manuals Available	:	4/79
SIPO's Available	:	DOS/VSE SIPO/E (Plan.Avail.3Q79) VM/DOS/VS SIPO/E (Plan.Avail.4Q79)
Selected Product Line	:	yes
Installation License	:	N/A
Pre-Installation Test	:	30 days
Program Services	:	Central Service until discontinued upon 12 months written notice. Local Service at no charge until 12/31/79 then Local Licensed Program Support.

10.8.1 PRODUCT FUNCTION

SORT/MERGE Release 2 will run under DOS/VS Release 33 and 34, or under DOS/VSE on a /370 machine and under DOS/VSE on 4300 processors, either in ECPS:VSE mode or /370 Compatibility mode.

The program can detect which release is controlling it and whether or not the system is in ECPS:VSE mode.

Release 2 provides an SVA load list (\$SVASORT) containing the Release 2 phases which are eligible for the SVA, together with conditions for loading them. The SVA load list is included as a load module in the distribution file. The current SVA procedure will still be enhanced and maintained for users of Releases 33 and 34, but will contain only those phases which may execute under control of these releases.

ECPS:VSE SUPPORT

Release 2 uses IORBs for its I/O requests. Page fixing takes place if a page fixable area of sufficient size for the SORT/MERGE I/O areas was defined with the SIZE command. This is not done, however, if the SORT/MERGE 'VIRT' option is specified.

/370 SUPPORT

Release 2 continues to use IDAL CCW addressing, but uses new system interface facilities (EXTRACT, GETVCE, etc.), in parallel with the old interfaces, in order to permit execution under DOS/VS Releases 33 and 34.

10.8.2 FBM SUPPORT

SORT/MERGE V2R2 (from here on referred to as SM2) can use FBM devices when running in ECPS:VSE mode on 4300 processors. It uses EXCPS when writing to or reading from input, output, and work files on FBM or CKD devices.

INPUT FILES

FBM devices can be used as input to both SORT and MERGE.

SAM files on FBM devices may be mixed with SAM files on non-FBM devices, but not with VSAM files. VSAM files on FBM may be mixed with other VSAM files. The following functions are restricted for SAM files on FBM devices:

- ASCII - not allowed
- ADDROUT - not allowed
- BYPASS - applies only to wrong-length, variable-length records.
- KEYLEN - not allowed
- Split Cylinder Files - not allowed

Because of the increased size of the input module, MERGE-only applications will need more virtual storage (or SVA space) in all cases (whether FBM or not).

OUTPUT FILES

FBM devices can be used for output from both SORT and MERGE. SAM and VSAM files are supported.

The VERIFY option may also be used for FBM devices.

The control interval size should be specified in the SORTOUT DLBL statement. If it is not specified, the CISIZE will be set to a value large enough to contain the blocksize.

CHECKPOINT FILES

SORT checkpoints can be written to FBM files. Only one extent can be defined. If SORT is called by another program, or if user-written routines are used, this extent must be large enough to checkpoint the whole partition; otherwise, it must be large enough to checkpoint only the SORT storage, as defined by SIZE, STORAGE, or default calculations.

WORK FILES

SORT allows the work files to be allocated to FBM devices. It is also possible to mix one FBM device type with one CKD device type.

WORK=DA may be used with FBM work files.

The ERASE option may be specified. The erase function writes zeroes into all extents occupied by work files.

SYSFIL SUPPORT

FBM support for SYSIN, SYSLST, and the debug files is obtained by recompiling all SORT/MERGE modules which contain DTFCP, CPMOD, or DTFSL.

RECORD AND BLOCK LIMITATIONS

The maximum input and output block size is 32,761 bytes when FBM devices are used. If the input or output file is declared as fixed unblocked, the block size value represents the maximum record length. The limiting factors are the amount of virtual storage allocated and the type of work device. If CKD work devices are used, the maximum record length is limited by the track capacity. If FBM devices are used, the maximum record length of 32,761 bytes may be used. Note that block descriptor words and control field specifications have the same effect on the maximum record length as in Release 1.

10.8.3 VARIABLE-LENGTH SPANNED RECORDS

Variable-length spanned records can be used for input to SORT or MERGE or may be requested as output. A new parameter 'SPAN' is added to the INPFIL and OUTFIL statements for this purpose. It is valid only for records described as TYPE=V in the RECORD statement; it is not valid for VSAM files, as spanned VSAM records are already supported. ADDR0UT cannot be combined with variable-length spanned input. If BYPASS is requested and a block is found to be in error, all records which are completely or partially contained in that block will be skipped.

The limiting factor for the maximum record length is the track capacity of the work device, if it is a CKD device. If it is an FBM device, the record length may be up to 32,749 bytes, but may sometimes be less, depending on the SORT control field specifications.

SM2 can handle "null segments", i.e. segments containing only a segment descriptor.

IMPROVED FILE HANDLING

SM2 will execute diagnosis and terminate if extent specifications are incorrect.

Incorrect specifications are:

- Overlap of output files with the work file(s).
- Overlap of one work file with another in the same SORT application.

Checking is done after the OPEN for a file or additional extent.

CONTROL STATEMENT PRINTOUT

If SM2 is dynamically invoked by another program, the control statement list will be printed if the SORT/MERGE PRINT=ALL option is in effect. The statements will be printed after syntax checking and after any error messages. The alternative, collating sequence table (for the AQTT statement) will not be printed.

Printing of each statement starts on a new line. If a statement will not fit on a single line, it is continued on the following line(s). A maximum of 72 characters of statement text is printed on each line.

THE ANALYZE STATEMENT

This new statement, with the format -

(label) ANALYZE CALC

is available for both invoked and standalone SORTs and can be used to retrieve information on how SM2 will use the environmental specifications and to debug new SORT/MERGE applications.

This function provides the following information:

- Storage allocation,
- SM2 capacity,
- Storage requirements, if too small,
- Buffer allocation, etc.

When SM2 recognizes the CALC option, the following additional options will be forced, regardless of how the user had specified them: DIAG, ROUTE=LST, PRINT=ALL, and (on termination) NODUMP. Furthermore, SM2 will assume a correct job set-up with correct file assignments. When it has completed Phase 0, and if it found no severe errors during syntax checking, SM2 will terminate with the message 7C19I ANALYZE END. This is a critical message and will cause the jobstream to be flushed or the invoking program to receive a return code of 16.

10.8.5 GENERATION AND INSTALLATION OPTIONS

SORT/MERGE permits the user to define default specifications for the following SORT/MERGE parameters: EQUALS, PRINT, ROUTE, STORAGE, SORTOUT, SORTIN, SORTWK, VERIFY, ERASE, DIAG, and DUMP. All of these (except EQUALS, which appears in the SORT statement) appear in the SORT/MERGE OPTION statement.

By changing the defaults to match his specific needs, the user can avoid the necessity of specifying the parameters each time he uses SORT/MERGE; specification is required only to override the defaults. The user defaults may be defined after installation of the program and may be changed at any time. All keyword parameters which have a variable number of values should be defined with the maximum number of such values which will be needed. SORT/MERGE will then, each time it is executed, ignore the superfluous default values. If, for example, the user wishes to use up to five work files, the default should be something like SORTWK=(15,16,17,18,19). If only two work files are used at execution time (WORK=2), SORT/MERGE will check and use only the work files on SYS015 and SYS016, and will ignore the others.

The generation and installation options are distributed as a separate assembler macro on the distribution file. This macro assembles into a jobstream which replaces a dummy module with a module containing the default specifications.

If a specification contradicts other options at execution time, VERIFY will always be used if the output is on tape and EQUALS will be forced if SUM is specified. In addition, an information message will be issued.

The following new SM2 statement parameters are provided to permit the default specification to be overridden: NOEQUALS, NOVERIFY, NOERASE, NODIAG, and STORAGE=(,NOVIRT).

10.8.6 COMMAND CHAINING FOR INPUT AND OUTPUT

The program will automatically try to use command chaining when reading input from, or writing output to, tape or CKD files (SORT only). Command chaining on input can be suppressed by specifying NOCHAIN in the INPFIL statement.

The BYPASS option will prevent command chaining on input; the VERIFY option suppresses it on output. SM2 does not use command chaining across track boundaries. The effect of

command chaining is to reduce the number of EXCPS, thus reducing CPU time and elapsed time.

10.8.7 HARDWARE CONFIGURATION

As already mentioned, SM2 will operate with DOS/VS on both 4300 processors and /370 machines.

INPUT/OUTPUT DEVICES

The following devices can be used for input, output, and checkpoint operations, as far as they are supported by DOS/VS or DOS/VSE : 2311, 2314, 3310, 3330 Models 1, 2, and 11, 3340/44, 3350, 3370, 2400, 3400, 8809.

Work files can be on: 2314, 3310, 3330 Models 1, 2, and 11, 3340/44, 3350, and 3370.

Notes:

- 8809 streaming mode is not supported
- Tapes may not be used as work file devices

10.8.8 SOFTWARE CONFIGURATION

SM2 has the same supervisor requirements as Release 1 when executed under DOS/VS Release 33 and 34, or under DOS/VSE in /370 Compatibility mode (FOPT macro parameter 'RELLDR' is standard for the DOS/VSE supervisor). If DOS/VSE is used in ECPS:VSE mode, FOPT RPS=YES is the only required parameter (for non-FBM DASD) if RPS support is desired; all other supervisor options are standard or have been made obsolete.

10.8.9 STORAGE REQUIREMENTS

FBM SUPPORT

SM2 may need more virtual storage for execution of a given application since the input and output buffer allocations now depend on the specified Control Interval size.

The input and output code for MERGE-only jobs has also increased in size and additional virtual storage may be required, if SORT/MERGE is not loaded in the SVA, when executing applications which do not use FBM devices.

The minimum allocation per work extent has been set to 64 blocks in order to allow for a meaningful SORT work file block size.

The maximum block size for a work file on an FBM device is 32K bytes.

4300 PROCESSOR SUPPORT

If neither STORAGE nor SIZE is specified in ECPS:VSE mode, SM2 will use all available storage for execution.

SVA REQUIREMENTS

For all code, SM2 requires approximately 275K bytes of storage in the SVA.

10.8.10 PERFORMANCE CONSIDERATIONS

SORT performance is expected to be better in environments where SM2 can choose command chaining, i.e. when sufficient storage is available. The elapsed time improvements will be large, while the CPU time improvements will be moderate. No improvement will occur in environments where the input and output block sizes are already optimal.

10.8.11 COMPATIBILITY

Release 2 of SM2 is compatible with Release 1 and runs under DOS/VS Releases 33 and 34, as well as under DOS/VSE.

Certain functions (ADDROUT and KEYLEN if the input file is on an FBM device) are not available in DOS/VSE. Any attempt to use these functions will be diagnosed by SM2.

VSE/VSAM is required for VSAM file processing under DOS/VSE.

Note: All functions described in the first edition of this Guide (DOS/VS R34 Planning Guide ZZ10-9822-00), but not mentioned here, are still available.

10.8.12 SUMMARY OF SORT/MERGE DEPENDENCIES

Dependency Item	Remarks
PROGRAM SUPPORT	Programs that can dynamically invoke SORT/MERGE:
- ASSEMBLER	
- COBOL/VS	
- PL/I	
- DOS RPG II	via Auto Report Feature
- DOS/VS RPG II	via inherent Auto Report Feature
UR SUPPORT	Any unit record device supported by DOS/VS
TAPE SUPPORT	Any tape supported by DOS/VS (NOT for work files)
DASD SUPPORT	
- 2311	input/output files only
- 2314	
- 2319	
- 3310	FBM
- 3330-1,2,11	
- 3333-1,11	
- 3340	
- 3344	
- 3350	
- 3370	FBM
DATA MANAGEMENT SUPPORT	
- SAM	I/O files (incl. spanned) and work files
- DAM	Work files only
- VSAM	I/O files only (KSDS, ESDS, RRDS)
CURRENT RELEASES	
- 2.0	

Notes:

- (1) Work files may be mixed on any TWO supported DASD device types (including FBM).
- (2) Input files for a SORT or MERGE job may reside on mixed tape and DASD types.
- (3) Input records may be fixed or variable length, blocked or unblocked SAM or VSAM files (but not mixed) or may be a mixture of FBM and CKD VSAM files.
- (4) No special effort has been made to support 8809 streaming mode.

10.8.13 PRODUCT REFERENCE MATERIAL

GC33-4043-	General Information Manual	(GIM)
GC33-4042-	Licensed Program Design Objectives	(LPDO)
GC33-4047-	Licensed Program Specifications	(LPS)
SC33-4045-	Installation Reference Manual	(IRM)
SC33-4044-	Programmer's Guide	(PG)
GX33-8002-	Reference Card	
LY33-8044-	Program Logic	(PL)

CHAPTER 11

VSE/INTERACTIVE COMPUTING AND CONTROL FACILITY

RELEASE 1

(5746-TS1)

04/05/79

IBM INTERNAL USE

CONTENTS

CHAPTER 11 VSE/ICCF. 11.1

11.1 INTRODUCTION. 11.1

11.2 VSE/ICCF (5746-TS1) 11.2

11.3 PRODUCT FUNCTION AND ENHANCEMENTS 11.3

11.3.1 General Description 11.3

11.3.2 Equipment Supported 11.6

 Terminal Support 11.6

 DASD Support 11.6

 Tape Support 11.6

11.3.3 Highlights of New Functions 11.7

11.3.4 Detailed Description of New Functions 11.8

 Conversion from FDP to PP and Adaptation to DOS/VSE. . 11.8

 Dynamic Label Information Handling 11.8

 Dynamic Disk Space Allocation. 11.9

 Supervisor Generation Option for VSE/ICCF. 11.12

 Monitor Call Class 4 11.14

 Intercept of Changed Supervisor Services 11.15

 Operator Communication with VSE/ICCF 11.18

 Device Type Specification for Library File 11.19

 Cancel of Interactive Partitions 11.19

 Fixed Block Mode (FBM) Support 11.19

 Support of TTF 11.27

 Support of New Display Terminals 3278. 11.29

 VSE/VSAM/AMS Support 11.30

 Interactive Usability Aids 11.31

 Introduction 11.31

 Implementation 11.34

 Distribution 11.36

11.3.5 Interfaces. 11.37

11.3.6 Security. 11.38

11.3.7 Summary of FDP-PP Changes 11.39

 VSE/ICCF Generation Options. 11.39

 External Initialization Changes. 11.43

 Utility and Control Statement Changes. 11.44

 Display of VSE/POWER Reader and Punch Queue. 11.46

11.3.8 Performance 11.47

11.3.9 Dependencies. 11.48

 Products (Equipment) 11.48

 Programs 11.48

11.4 INSTALLATION PLANNING 11.49

11.4.1 Shipment Package Layout 11.49

11.4.2 VSE/ICCF Generation 11.50

11.4.3	Pregenerated Version of VSE/ICCF.	11.51
11.4.4	Disk Space Requirements	11.52
	User Profile and Library File Considerations	11.52
	Interactive Partition Workfiles.	11.54
11.4.5	Terminal Control.	11.55
	Terminal Control Considerations.	11.55
	CICS/DOS/VS.	11.55
	TTF.	11.55
	Control Unit/Terminal Support.	11.56
11.5	Migration/Coexistence	11.59
11.5.1	Migration Path From SPM to VSE/ICCF	11.59
	The Library File	11.59
	Educate the User	11.62
	Discover New Features.	11.62
	Features Different or not Available.	11.62
	Technical Considerations	11.63
11.5.2	Migration Path from ETSS/II FDP	11.63
11.6	REFERENCE MATERIAL.	11.64

11.1 INTRODUCTION

This chapter contains a summary of details for the new timesharing program product VSE/ICCF. It is not the intention of the guide to document every function, but where necessary details will be provided. The following topics will be discussed:

- (I) PRODUCT FUNCTION
- (II) MIGRATION AND INSTALLATION CONSIDERATIONS
- (III) PRODUCT REFERENCE MATERIAL

11.2 VSE/ICCF (5746-TS1)

Status	: SPL
Product Latest Release	: 1.0
SIPO available	: yes
Installation Licence	: N/A
Pre-Installation Test	: 1 month
Points value	: 60

11.3 PRODUCT FUNCTION AND ENHANCEMENTS

11.3.1 GENERAL DESCRIPTION

VSE/ICCF is a follow on product of the FDP ETSS/II (5798 - CLR). ETSS has been documented previously in the PLANNING GUIDE in the WTSC Special Distribution EO-SE01-78. As this document gave a description of the ETSS FDP it will be assumed the reader is familiar with the FDP. The ETSS S.E. GUIDE distributed via DAPS Form Number ZZ10-9824 also contains descriptions of the FDP, as of course do the FDP manuals. The main aim of this section is to document the new program product features.

The VSE/INTERACTIVE COMPUTING AND CONTROL FACILITY Program Product Release 1.0 (VSE/ICCF R1.0) is an adaptation of the widely used DOS/VS Entry Time Sharing System (FDP 5798-CLR) to DOS/VSE. VSE/ICCF uses either CICS/DOS/VS or TTF as its terminal control facility, thus allowing users to establish a single, unified terminal environment supporting both timesharing and transaction oriented applications (CICS/DOS/VS only).

The VSE/ICCF is an interactive system designed to extend the power of the computer to multiple terminal users concurrently. The functional capabilities brought to the terminal users can provide significant productivity benefits in many different user environments. The timeshared computer concept, while having nearly limitless applications, has proven to be of particular benefit in the areas of individual problem solving, programming development and maintenance, and education.

For the general problem solving environment, the system provides the framework for complex problem solving utilizing the user's various high level language application programs. The professional programmer may use VSE/ICCF not only for program development, but for library maintenance and creation of more complex job streams. High level language

programs as well as Assembler may be compiled and tested directly from the terminal with the results received back at the terminal. A submit-to-batch capability via VSE/POWER is also provided for the execution in a time-shared environment. One can write, edit, compile and execute programs without leaving the terminal. Interactive usability aids will be provided assisting the user to easily create various DOS/VSE jobs without having to deal with DOS/VSE job control or job entry control cards.

The system may be viewed as a collection of tools: an online program development system, an online library maintenance system, a set of interactive usability aids, a powerful context editor, a text manipulation facility, a job entry system, a time-sharing/job-execution monitor and a facility for development and execution of interactive problem programs and DOS/VSE Access Control-Logging and Reporting.

The ETSS/II (FDP 5798-CLR) is the base for the program product VSE/ICCF. It is intended that VSE/ICCF is based on the DOS/VSE System Control Program (SCP) and VSE/ADVANCED FUNCTIONS package. No support of releases prior to DOS/VSE is provided.

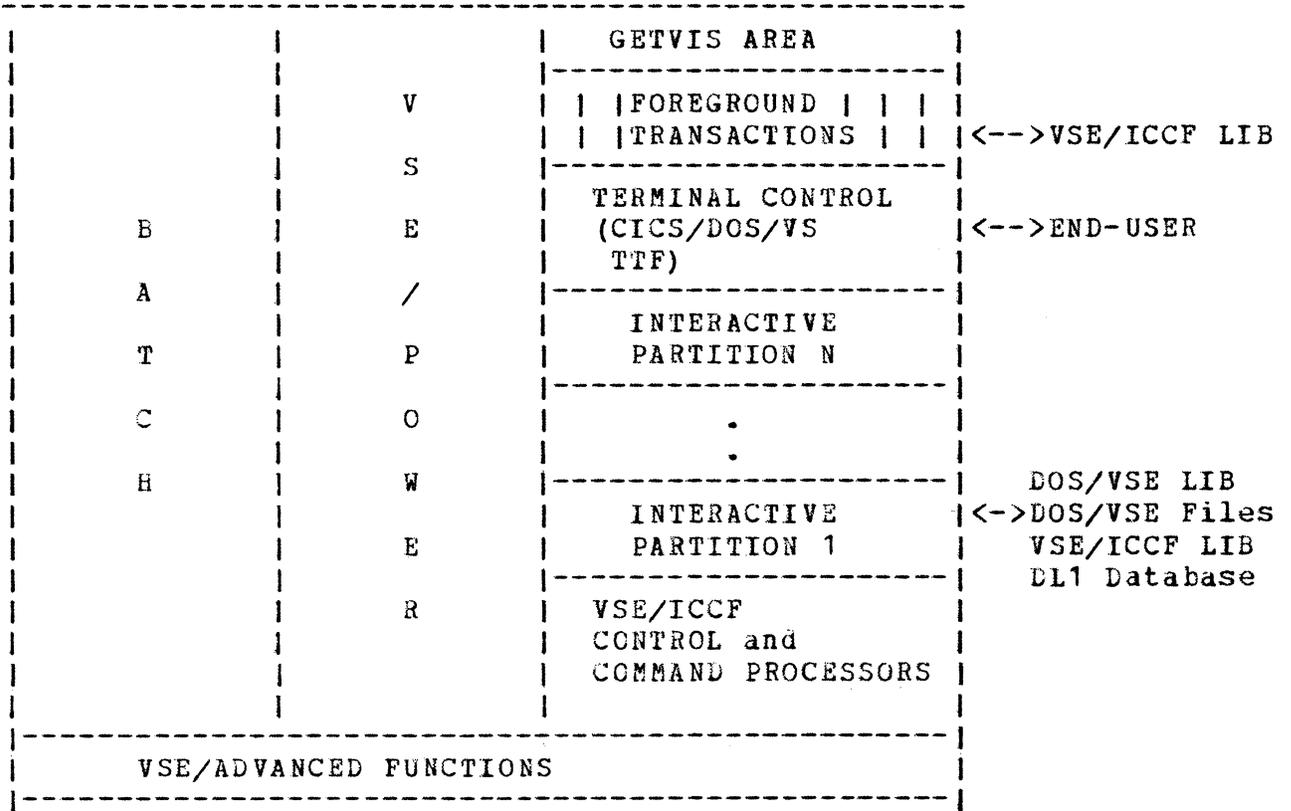


Figure 1: System Overview

11.3.2 EQUIPMENT SUPPORTED

Both 4300 processors and /370-machines are supported in VSE/ICCF. Fixed Block Mode (FBM) DASD device is supported as an VSE/ICCF library, as SYSRES, and for user files accessed by programs from the interactive partitions.

TERMINAL SUPPORT

The terminal access method, BTAM-ES for TTF and BTAM-ES or ACF/VTAM for CICS/DOS/VS determines the terminal support available to the VSE/ICCF user. See the appropriate section in this chapter for support of display stations and printers.

Note: The prompts are written for display terminals. They will not be tested on typewriter terminals.

DASD SUPPORT

At least two spindles should be present on the system. RPS is an VSE/ICCF generation option. The DASD support is provided for the VSE/ICCF library, dynamic space area, and SYSRES on 2314, 2319, 3330 Model 1 and Model 2, 3330 Model 11, 3340, 3350, and FBM devices.

TAPE SUPPORT

For an installation with non-removeable DASD, a tape drive can be used for the installation of the system, but it is not required for operation of VSE/ICCF.

11.3.3 HIGHLIGHTS OF NEW FUNCTIONS

Conversion from FDP to PP and Adaptation to DOS/VSE

The VSE/ICCF Program Product is adapted to DOS/VSE (/370 and E-mode) in the following areas:

The new label area concept of DOS/VSE is used for dynamic label information handling; dummy labels are no longer required.

Files which are permanently allocated from the dynamic disk space area may be date protected as normal DOS/VSE files.

A new supervisor generation option for VSE/ICCF support is introduced.

Various generation options have been removed in VSE/ICCF. These functions are now standard.

Monitor Call class 4 is reserved for VSE/ICCF in order to allow VSE/ICCF to intercept services which do not result in SVC's (e.g. LABEL macro, JOBCOM macro).

Intercept of new and changed Supervisor services (COMRG, EXTRACT).

VSE/ICCF operator commands are routed to the VSE/ICCF operator communication routine via the attention task directly and not via the CC STXIT.

The device type specification via UPSI-bits has been deleted.

Support of TTF (Terminal Transaction Facility) and continued Support of CICS/DOS/VS terminal control

Support of the New Display Terminals 3278

VSE/VSAM/AMS Support

Interactive Usability Aids (prompters)

11.3.4 DETAILED DESCRIPTION OF NEW FUNCTIONS

CONVERSION FROM FDP TO PP AND ADAPTATION TO DOS/VSE

Aside from changes necessary to convert the FDP to a Program product and to adapt to the new 4300 processors architecture and to the fixed block mode, DOS/VSE provides a series of new centralized services. These services, such as the common VTOC handler and the Symbolic Label Access (SLA) facility, enables VSE/ICCF to be less dependent upon DOS/VSE changes and to improve certain VSE/ICCF functions and the maintenance of VSE/ICCF. The VSE/ICCF Program Product adaptation to DOS/VSE only includes those changes necessary to run on the new release, but also the usage of the new centralized services wherever possible.

The following sections describe how VSE/ICCF appears to the user.

DYNAMIC LABEL INFORMATION HANDLING

The dynamic label information handling of ETSS/II (/FILE..job entry statement) must be adapted to the new label area concept of DOS/VSE. In order to support VSE/ICCF, the label area will be extended by a new subarea, called free-usage label subarea (to be used by IBM components only). In addition, a search of the interactive partition label area will be made by SLA for all label requests from interactive partitions. The search sequence for retrieving a label is changed to:

1. Interactive partition's label subarea
2. Partition's temporary label subarea
3. Partition's permanent label subarea
4. System label subarea

The dynamic file space label requests will no longer be handled by an VSE/ICCF transient, but will be read directly from their storage area in the interactive partition by the SLA routines through an interface to VSE/ICCF. This search will occur after the VSE/ICCF partition temporary label area is searched.

By making use of the new label area concept, the dummy file labels DTSDUMMY, DTSDUM1, DTSDUM2,...which were required at ETSS/II initialization time are now unnecessary.

In order to support the definition of data sets on FBM devices, the /FILE job entry statement must be adjusted. The LOC=strttrk,ntrks parameter has the following meaning for a data set on an FBM device:

strttrk For FBM the parameter specifies the address of the physical block with which the extent starts.

ntrks For FBM the parameter specifies the number of logical tracks the extent consist of. A logical track is a unit of FBM space currently implemented as 16 blocks.

Moreover, a new parameter called CFSIZE=cinv, which corresponds to the new CFSIZE-parameter in the DOS/VSE DLBL job control statement, is added to the /FILE job entry statement. With CFSIZE= the user specifies the control interval size for the SAM files on a FBM device. The control interval size must be a multiple of the physical blocksize of the FBM device. CFSIZE= is optional and may only be specified in conjunction with TYPE=SEQ.

DYNAMIC DISK SPACE ALLOCATION

General

As with ETSS/II, disk space can be dynamically allocated for jobs in interactive partitions from a dynamic disk space area which may consist of one or more extents on one or more disks of any supported type (i.e. also a combination of FBM and CKD devices). CKD and FBM devices are supported. The number of extents the dynamic space area may consist of equals the number of labels specifications given at VSE/ICCF initialisation time.

Unlike in ETSS/II where the dynamic disk space area had to be defined via two direct access data sets per extent, the dynamic space area is now one or two multi-extent direct

access data sets with the filenames DTSDYNC or DTSDYNW. The date protection fields of the labels are ignored. The absence of both labels indicates that VSE/ICCF dynamic file space is not to be initialized. The normal DOS/VSE conventions are valid for defining the file via // DLBL and // EXTENT job control statements.

Here an example for FBM

```
// DLBL DTSDYNC,,0,DA
// EXTENT SYSxxx,nnnnnn,1,0,block-address,number-of-blocks
// DLBL DTSDYNW,,0,DA
// EXTENT SYSxxx,nnnnnn,1,0,block-address,number-of-blocks
```

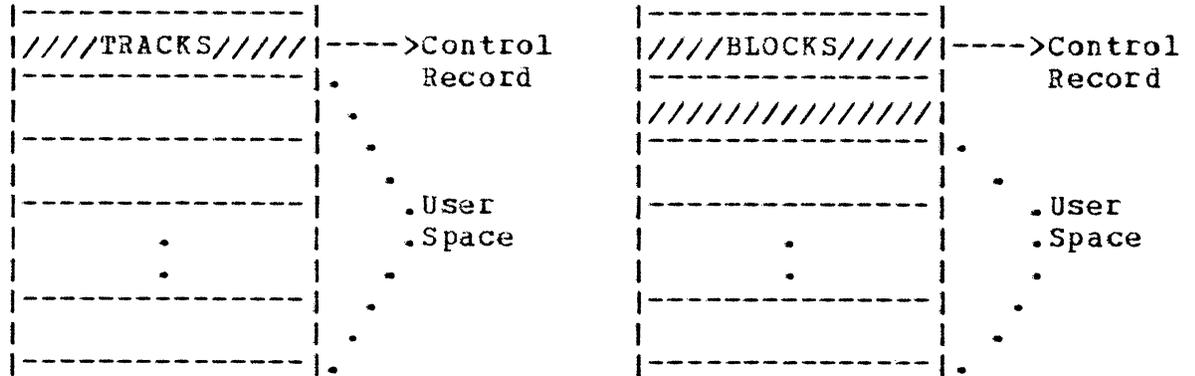
DTSDYNC denotes that the extents following are to be cold started unless the operator specifically requests warm start. DTSDYNW denotes a similar concept for warm extents.

In an interactive partition the user can use the dynamic disk space allocation as follows: SAM and DAM are supported for CKD devices and only FBM-SAM for FBM devices.

Organization of DTSDYNC and DTSDYNW

Each extent of the file consists of a control record and user space. For CKD devices the first track of each extent is reserved for holding the control record. In the case of FBM devices the control record is located in the first physical blocks of each extent. Dependent on the size of the extent, 1 to 16 physical blocks may be used for the control record. The control record contains information about the status of allocation of the user space. The user space may be allocated to user files in multiples of one track in case of CKD devices or in multiples of 16 physical blocks in case of FBM devices. The maximum size of each extent of the dynamic disk space area is 7200 tracks for CKD devices or 7200*16 physical blocks for FBM devices.

Aside from some general control information a control record contains one byte for each unit of allocation, that is, one byte per track for CKD devices or one byte per 16 physical blocks for FBM devices. There are as many allocation bytes in a control record of an extent as units of allocation exist in the extent.



Extent on CKD Device

Extent on FBM Device
Example where two physical blocks
are needed for the control record

Figure 2: Example of Dynamic Disk Space Allocation

An allocation byte describes the interactive partition which owns the allocated space and the disposition as specified via the /FILE job entry statement. Each control record will be date protected via a format 1 label in the vtoc. The file-ids of the control record are user specified. If cold start of the dynamic disk space area is requested the initialisation routine of VSE/ICCF creates the format 1 labels for the control records and writes them to the VTOC. After that it builds the control records and writes them to the beginning of the corresponding extents. If warm start of the dynamic disk space area is requested the initialization routine of VSE/ICCF reads the format 1 labels for the control records from the VTOC and checks the extent information in the control records against the extents specified in the label area. If an extent doesn't match with the information in the control record a cold start of that volume is performed. The operator has to specify the option at VSE/ICCF initialisation time that all extents (DTFDYNC and DTFDTNW) are to be cold or warm started.

Usage of the Dynamic Disk Space Area

As with ETSS/II the user specifies a request for space to be allocated from the dynamic disk space area via the /FILE job entry statement. However, the parameter TRACKS=n for specifying the size of the space is replaced by the new parameter SPACE=n. If the dynamic disk space area is on CKD devices n specifies the number of tracks, whereas, in case it is located on FBM devices n specifies the number of

groups of 16 blocks desired. This group of 16 blocks is much like requesting tracks and allows the same /FILE statement to hold for CKD or FBM without change.

Unlike in ETSS/II, the user may specify expiration information via the DATE= parameter for a permanent file (DISP=KEEP). As with ETSS/II, date protection is always assumed to be off for temporary files (disp=delete or DISP=PASS).

Note: Files with the DATE= parameter will not be available anymore after a cold start.

SUPERVISOR GENERATION OPTION FOR VSE/ICCF

A new supervisor generation option is provided in the SUPVR macro. The new keyword parameter is entitled ICCF.

If ICCF=YES has been specified, a vector table is generated which is pointed to by a new 4-byte field added to the system communication region. The presence or absence of the address in the new field indicates whether or not the supervisor has been generated with ICCF=YES. The vector table serves as a communication area between the supervisor and VSE/ICCF. After VSE/ICCF initialization has verified that the supervisor has been generated with the required ICCF option, it places addresses of VSE/ICCF routines into the vector table for later use by the supervisor. At the end of the initialization process, VSE/ICCF places its PIK into byte 0 of the new field in the system communication region to indicate that VSE/ICCF is active.

During VSE/ICCF processing the vector table is in addition used for transferring status information from the supervisor to VSE/ICCF. Moreover the vector table contains information for the VSE/ICCF termination process which is now part of the DOS/VSE EOJ processing.

ICCF=YES also causes routines to be generated in the supervisor which support storage protection of interactive partitions. These routines replace the special supervisor generation macros DTSPROTA and DTSPROTB of ETSS/II. This means that the storage protect feature for interactive partitions is standard for VSE/ICCF and the ETSS/II generation option PROTECT= is obsolete.

In addition a new SVC-service routine is generated which allows VSE/ICCF to initialize the pages of a interactive partition and to set its storage protect key without

accessing and changing the page tables in the supervisor directly.

Also if ICCF=YES has been specified and VSE/ICCF is active, the supervisor checks if an SVC-interrupt came from a program in the VSE/ICCF-partition. If the interrupt belongs to the VSE/ICCF-partition, the supervisor transfers control to the VSE/ICCF SVC-interrupt routine addressed via the vector table. This mechanism makes the change of the SVC New PSW at VSE/ICCF initialization time obsolete.

Moreover, by specifying ICCF=YES, a new SVC-service routine is generated which is called via SVC 82. SVC 82 replaces the SVC number used in ETSS/II which is either 175 or a number specified via the VSE/ICCF generation parameter SVCNUM. The SVC-service routine transfers control to an VSE/ICCF internal service routine via the vector table. The VSE/ICCF generation option SVCNUM is then obsolete.

Finally with ICCF=YES a routine is generated in the program check handler of the supervisor which checks if a monitor call interrupt of class 4 came from a program in the VSE/ICCF-partition. If the interrupt belongs to the VSE/ICCF-partition, the supervisor transfers control to the VSE/ICCF MC-interrupt routine addressed via the vector table.

Monitor call class 4 is enabled at VSE/ICCF initialization time and disabled at VSE/ICCF shutdown time. The SVC 82 routine contains functions to enable and disable the Monitor Call class 4.

If ICCF=YES has been specified the following supervisor generation options are forced:

<u>MACRO</u>	<u>Parameter</u>
SUPVR	NPARTS=3 if NPARTS=1 has been specified or the parameter has been omitted.
	AP=YES
FOPT	IT=YES OC=YES AB=YES PFIY=YES
CONFG	FP=YES

Note: **The following supervisor generation options which are required by VSE/ICCF are standard starting with DOS/VSE

<u>Macro</u>	<u>Parameter</u>
FOPT	PC=YES RELLDR=YES GETVIS=YES
PIOCS	BLKMPX=YES

In order to be consistent with DOS/VSE the GETVIS-area support for interactive partitions is now standard. Therefore the VSE/ICCF generation option GETVIS= is obsolete. Moreover, the minimum size of a interactive partition and its GETVIS-area are adjusted to DOS/VSE partition sizes. That is, the minimum size of a interactive partition is now 128k and the minimum size of its GETVIS-area is 48K.

As the floating point support in the supervisor is forced with ICCF=YES, the VSE/ICCF floating point generation option FP= becomes obsolete.

Aside from what has been described above various hooks have been introduced into the DOS/VSE control program, primarily, in order to cause control program services to access and update interactive partition control blocks in case the request came from VSE/ICCF. Services changed for that purpose are FETCH, LOAD, GETVIS, FREEVIS, etc.

MONITOR CALL CLASS 4

The Monitor Call class 4 has been reserved for VSE/ICCF. It has been introduced as a general mechanism which allows VSE/ICCF to intercept DOS/VSE control program services which do not result in a SVC interrupt. Such services are usually those which are called from a user program by invoking a macro which does a branch and link to an SVA-routine. In case VSE/ICCF has to intercept such a service in order to do something special for interactive partitions the invoking macro or the service routine itself has to issue a Monitor Call instruction of class 4. AS the Monitor Call class 4 is enabled while VSE/ICCF is active a program check interrupt occurs as soon as such a service is invoked. In case a interactive partition issued such a service request the supervisor transfers control to the VSE/ICCF MC-interrupt routine which performs the special handling for that service.

Control program services of DOS/VSE which are intercepted

via that mechanism are the file label service (to be invoked by the LABEL macro) and the job-to-job communication service (to be invoked by the JOBCOM macro).

The file label service has to be intercepted by VSE/ICCF in order to adapt the protect key such that the service works for interactive partitions as well. Also the labels have to be read into the label service's own buffers and then moved into the interactive partition (CCB and I/O areas have to have the same storage key). If the label service's buffer is not large enough to contain a label, then VSE/ICCF is informed via a monitor call and cancels the job. The job-to-job communication service is not allowed to be used in interactive partitions. VSE/ICCF causes a program in a interactive partition which issues JOBCOM to be canceled.

DL/I when running in a interactive partition needs to know the interactive partition communication region address. Since the SVC 33 is costly DL/I loads the partition communication region address from low core. In an VSE/ICCF environment it will retrieve the interactive partition communication region by a monitor call.

INTERCEPT OF CHANGED SUPERVISOR SERVICES

The intercept of the COMRG macro is extended such that the address of the interactive partition communication region is returned in the register the user has specified with the COMRG macro.

The EXTRACT macro is intercepted by VSE/ICCF for all those services which cause partition related information to be returned to a program. VSE/ICCF returns the corresponding information of a interactive partition.

The LOAD/FETCH/GETVIS/FREEVIS/CDLOAD SVC intercepts are changed in such a way that they repeat the SVC and pass the interactive partitions communication region address to the supervisor SVC routines.

The USE/RELEASE SVC's are intercepted and recorded. As long as a interactive partition holds any resource, it may not loose its associated DOS/VSE task in order to be able to release it again.

In general all SVCs which may not be used by a job executing in an VSE/ICCF interactive partition, and which are not already rejected by DOS/VSE are rejected by the VSE/ICCF SVC intercept routine. The job is cancelled with the DOS/VSE 'ILLEGAL' SVC message.

SVC	macro	SUPPORTED	NOT SUPPORTED	IGNORED
0	EXCP	*		
1	FETCH	*		
2		*		
3		*		
4	LOAD	*		
5	MVCOM	*		
6	CANCEL	*		
7	WAIT	*		
8		*		
9	LBRET	*		
10	SETIME		*	
11		*		
12				*
13				*
14	EOJ	*		
15	SYSIO		*	
16	STXIT (PC)	*		
17	EXIT (PC)	*		
18	STXIT (IT)		*	
19	EXIT (IT)		*	
20	STXIT (OC)	*		
21	EXIT (OC)	*		
22	SEIZE		*	
23			*	
24	SETIME		*	
25			*	
26		*		
27			*	
28	EXIT (MR)		*	
29	WAITM		*	
30	QWAIT		*	
31	QPOST		*	
32	-			
33	COMRG	*		
34	GEMME	*		
35	HOLD		*	
36	FREE		*	
37	STXIT (AB)	*		
38	ATTACH		*	
39	DETACH		*	
40	POST		*	
41	DEQ		*	
42	ENQ		*	
43	-			
44		*		
45			*	
46			*	
47	WAITF		*	

48			*	
49			*	
50			*	
51		*		
52	TTIMER		*	
53			*	
54			*	
55			*	
56		*		
57		*		
58			*	
59			*	
60		*		
61	GETVIS	*		
62	FREEVIS	*		
63		*		
64		*		
65	CDLOAD	*		
66	RUNMODE	*		
67	PFIX		*	
68	PFREE		*	
69	REALAD		*	
70	VIRTAD		*	
71	SETPFA		*	
72			*	
73			*	
74			*	
75	SECTVAL	*		
76			*	
77		*		
78	CHAP	*		
79			*	
80	SETT		*	
81	TESTT		*	
82	ICCF SVC	*		
83			*	
84			*	
85	RELPAG			*
86	FCEPGOUT		*	
87	PAGEIN		*	
88	TPIN		*	
89	TPOUT		*	
90			*	
91			*	
92	XECBTAB		*	
93	XPOST		*	
94	XWAIT		*	
95	EXIT (AB)		*	
96	STIXT (TT)		*	
97	EXIT (TT)		*	
98		*		
99		*		
100			*	

101		*			
102				*	
103				*	
104		*			
105		*			
106		*			
107				*	
108		*			
109		*			
110	LOCK/UNLOCK		*		

- Notes:
1. CANCEL ALL is changed by VSE/ICCF to CANCEL
 2. Fast CCW-translation is skipped for interactive partitions
 3. EXCP with the parameter REAL is not supported in interactive partitions
 4. Programs which use DOS/VSE service or access supervisor tables which are not described as official interfaces in the 'DOS/VSE Macro Reference' manual may cause problems if they are executed in interactive partitions, although they run without problems in DOS/VSE batch partitions.

Figure 3: List of Legal and Illegal SVCs

OPERATOR COMMUNICATION WITH VSE/ICCF

For consistency and to simplify the communication with VSE/POWER and ACF/VTAM, the VSE/ICCF operator commands are routed to the VSE/ICCF operator communication routine via the attention task and no longer via the OC STXIT routine as in ETSS/II. It is no longer necessary for the operator to press the interrupt key or to enter the MSG command, wait for an ETSS/II message, and finally enter the operator command. The operator can simply enter the command in attention mode. The attention task transfers control to the VSE/ICCF operator communication routine addressed via the new vector table.

The VSE/ICCF routine recognizes and accepts any VSE/ICCF command. If the command does not belong to VSE/ICCF control goes back to attention for further investigation of the

command.

DEVICE TYPE SPECIFICATION FOR LIBRARY FILE

The UPSI switch setting for the device type of the VSE/ICCF library will no longer be required nor supported. The device type will be dynamically determined by reading the label and extent information for the fixed name DLBL DTSTFILE via the DOS/VSE SLA facility. A message will be issued and processing terminated if the DTSTFILE was not supplied and/or the label access was unsuccessful. The DOS/VSE GETVCE facility will then be used to determine the device type for the given logical units. All device types must be the same. The device related characteristics will be calculated and the DTF updated per the logical-to-physical assignment.

CANCEL OF INTERACTIVE PARTITIONS

Currently jobs running in a interactive partition go through DOS/VSE end-of-job processing only if they end with an EOJ request, or if they are cancelled by DOS/VSE. If ETSS discovers an error situation the jobs just don't receive control again. In this case user specified PC or AB exit routines do not get executed.

In order to have any abnormal exit routine get control, to have DOS/VSE resources freed, to have the address space occupied invalidated, in an error situation detected by VSE/ICCF control program, the jobs are forced through DOS/VSE end-of-job processing. this means, that now before a VSE/ICCF cancel message appears on the screen, a DOS/VSE cancel message will appear first.

FIXED BLOCK MODE (FBM) SUPPORT

The VSE/ICCF Program Product will utilize the Fixed Block Mode (FBM) of the E-Series machines for the VSE/ICCF library, SYSRES, and for the dynamic disk space area when on FBM devices. Data is stored and retrieved to and from FBM peripheral storage devices in fixed blocks. The addressing is by relative block number. VSE/ICCF will not be dependent upon the current 512-byte block size and will

use DOS/VSE GETVCE facility to retrieve such device related data as the block size, device block capacity, etc. By the use of FBM devices, VSE/ICCF is less dependent upon the physical device characteristics and can use the linear addressing scheme for any FBM device regardless of its block size. The dynamic disk space area which allows the user to dynamically allocate disk space to his program is now supported on FBM devices as well.

It is also necessary for VSE/ICCF to adapt to an FBM SYSRES since VSE/ICCF accesses the system residence volume for loader processing.

Internal VSE/ICCF Library File Organization

The current ETSS/II library file is a single data set consisting of one or more extents residing on one or more devices. All logically related data records are chained together with backward and forward relative record pointers as shown:

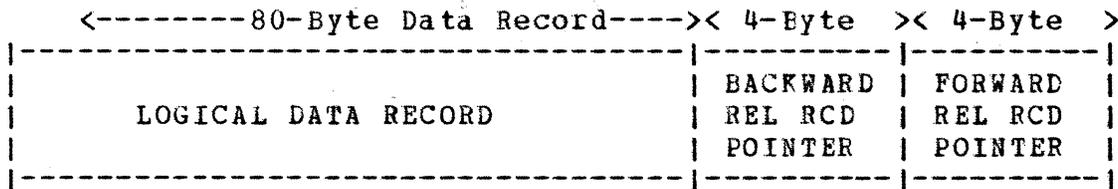


Figure 4: VSE/ICCF Library Logical Record Format

The first record in the VSE/ICCF Library file is the system record which contains information such as the number of user profiles defined within the file, number of sub-libraries within the file, and the start address of the broadcast records. Free area pointers are also contained in the system record. The free records are either allocated from the free area at the end of the used portion of the file or from the free area chain of deleted records within the file area.

The user profile records which contain the user ID, password, security level, library code, and statistics,

follow the system record. This is followed by the library header records which point to the first directory record for the specific library. The directory records are allocated as library members are created. They contain information relative to the library member and a pointer to the first data record within the member.

Each member consists of one or more 80-column data records with the 8-byte backward and forward pointers as shown above.

The VSE/ICCF library on a FBM device will maintain the same internal logical record format. For FBM, the number of records per block is calculated for the given control interval size and assigned to the NRECS= parameter. the NRECS= generation parameter of VSE/ICCF which results in the physical blocking factor and buffer size for CKD processing is still required. The FBM physical block size is fixed at 512-bytes and the implementation will be independent of this blocksize. The control interval size is specified on a new VSE/ICCF generation parameter CISIZE. The control interval is the unit of data transfer between storage and the FBM device, that is, the I/O buffer. The control interval consists of an integral number of physical blocks. The control information of the control interval is a Control Interval Definition Field(CIDF) and two record Definition Fields(RDF). The CIDF contains a 2-byte offset to the beginning of free space in the control interval followed by 2-byte free space length. The first RDF contains a control flag byte followed by 2-byte length field and the second RDF contains the number of records in the control interval. Since the VSE/ICCF library records are fixed at 88-bytes, there will be just two RDF's and one CIDF per control interval as shown below:

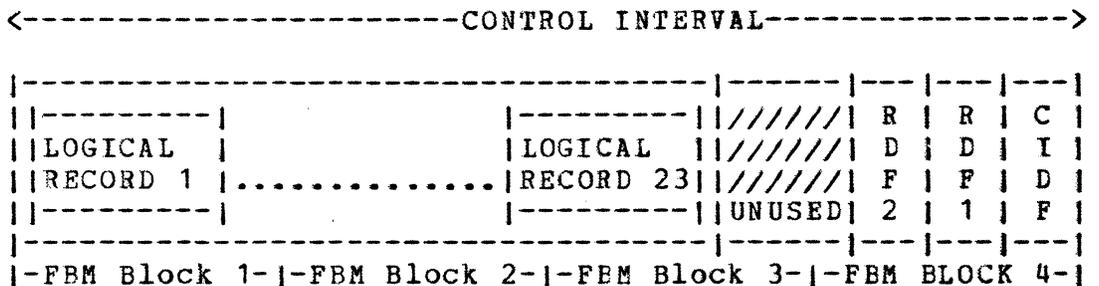


Figure 5: Example of an VSE/ICCF FBM Library C.I.

The characteristics of this example follows:

Control Interval Size (Buffer Size)	2048 Bytes
FBM Physical Block Size	512 Bytes
VSE/ICCF Logical Record Size	88 Bytes

In this example, the control interval consists of 23 VSE/ICCF logical records (80 bytes of data and 8 bytes of control information), 14 bytes unused and 10 bytes of control information. No logical records can span control intervals.

The control fields of the control interval in the above example would have the following values:

Record Definition Field (RDF-2)	
Replicated Record Count	23
Record Definition Field (RDF-1)	
Record Length	88
Control Interval Definition Field (CIDF)	
Freespace Offset	2024
Freespace Length	14

Specification of VSE/ICCF Library on FBM

The control interval size to be used for processing is specified by a new CISIZE= keyword of the DTSOPTNS VSE/ICCF generation macro. A preliminary check of the CISIZE value is done at macro generation time. The CISIZE given must be a multiple of the default 512-byte physical block size and a multiple of 2k if greater than 8k. The maximum value is 32,768 and the default is 2048. This value also determines the size of the I/O buffers allocated; the number of buffers is determined by the NBUFS= value specified on the DTSOPTNS macro.

A primary prerequisite necessary for FBM processing is the presence of certain FBM DOS/VSE supervisor generation options. FBA must be given on the DISK= keyword of the generation macro PIOC and via the IPL ADD command. The DFBA= keyword of IOTAB specifies the number of FBM drives. VSE/ICCF OPEN processing will determine the presence of FBM in the system by interrogating an option byte in the System Communications region.

FBM Library Management

The VSE/ICCF library on FBM is to be created and initialized by a batch utility as it is done on current releases for count-key-data devices. With the assignment of the library to an FBM device and the corresponding DLBL and EXTENT statements

```
( // DLBL DTSFILE,'ICCFFILE',xx/xxx,DA
  // EXTENT SYS012,nnnnnn,1,0,block-address,#-of-blocks )
```

giving the extents in FBM-block ranges, including the VSE/ICCF and DOS/VSE system generation options discussed above, the VSE/ICCF library file creation processing will be done.

The device characteristics for the VSE/ICCF library will be obtained by using the GETVCE DOS/VSE Supervisor service facility to dynamically retrieve the physical block size. The control interval size given must be a multiple of the physical blocksize. If the control interval size does not meet this basic prerequisite, a message is written and processing stopped.

The VSE/ICCF CISIZE generated is placed in the DTFPH CISIZE field and an FBM OPEN for output issued. OPEN processing will place the CISIZE in the FORMAT 1 label. LBRET is issued until all extents are processed. After returning from LBRET, it is verified that the CISIZE in the DTF is the same as the one given for VSE/ICCF generation. If different, it's likely that an inconsistent CISIZE override was specified on the DLBL statement. A message is written and processing stopped. Otherwise, all FBM extents are preformatted with formatted control intervals. It's not allowed to have the library span CKD and FBM devices. After all extents are successfully preformatted and the file CLOSED, the FBM library has been initialized.

Opening, Accessing, and Closing by On-line

VSE/ICCF Processing During VSE/ICCF initialization or bring-up time the VSE/ICCF library is OPENED for processing. The FBM block EXTENTS and the CISIZE which is on the FORMAT 1 label as a result of the utility creation process is returned by LBRET processing. If the returned CISIZE does not match the VSE/ICCF generated size, an error message is written and VSE/ICCF initialization is terminated.

Since the control interval is the unit of data transfer, it is always a multiple of physical blocks and the extent is rounded by OPEN to contain a number of blocks that is an integer multiple of the number of blocks per control interval. The upper physical block number of the extent will be rounded down to the last physical block of the last control interval of that extent. VSE/ICCF will construct an Extent Definition Block to map the logical block numbers to the physical block numbers of each extent. The forward and backward relative record chaining concept of the CKD VSE/ICCF library is continued in order to minimize logic changes to VSE/ICCF. VSE/ICCF FBM library file characteristics such as control interval size, number of physical blocks per control interval, block size, logical records per control interval, and logical relationship to physical device as contained in the VSE/ICCF constructed Extent Definition Block, are necessary to calculate relative record number-to-logical block number and logical block number-to-relative record number.

The VSE/ICCF FBM Library file characteristics in the example below are a 512-byte physical block size, 1024 control interval size, and eleven 88-byte VSE/ICCF Library logical records per control interval. (Logical records per control interval=

$(\text{control interval size} - (2 * \text{length}(\text{rdf}) + \text{length}(\text{cidf})) / 88)$) The library consists of three extents spanning two FBM volumes.

and de-blocking for FBM processing, and the usage of the new FBM mode CCW's at the EXCP level. Write verification will be done as per user specification.

The batch utilities are to be adapted to process FBM devices in a similar manner as count-key-data devices.

SYSRES on FBM Device

To OPEN or CLOSE a data set successfully, it's necessary that the labels for the data set are available in the label area. When a program running in an interactive partition issues an OPEN/CLOSE, VSE/ICCF receives control via an exit from the SLA routines. This VSE/ICCF routine reads the interactive partition labels, which were constructed by VSE/ICCF from the VSE/ICCF Job Entry /FILE statement, and returns a 00 return code and a file-id in LPL and label buffer to the requestor if the label is found. If the label was not found a non zero return code is set. Control is passed to SLA which goes to the user or keeps searching the disk area as required. A Standard Label Access(SLA) facility is provided by DOS/VSE as a standard interface to the label areas, which is device independent to the user. VSE/ICCF uses this SLA facility and thus remains independent of the label area format and whether it's on a CKD or FBM device.

The access to the system residence directory of a FBM SYSRES is changed. The VSE/ICCF LINKNGO facility is adapted to the DOS/VSE librarian directory format for accessing the relocatable library.

SUPPORT OF TTF

The two terminal control systems supported by VSE/ICCF are CICS/DOS/VS and TTF. TTF is derived from the existing FDP ETSS/II TCS which is a subset of MTCS/VS. TTF is tailored to the needs of VSE/ICCF only. Except for VSE/ICCF no user application can run on TTF. TTF is dedicated to support the low-end user who has no needs for the complete set of data communication facilities offered by CICS/DOS/VS.

TTF will be part of the VSE/ICCF program product whereas CICS/DOS/VS has to be ordered separately.

Characteristics of TTF

TTF provides a terminal control facility for VSE/ICCF only. The functions supported are in general the same as those of ETSS/II TCS. TTF handles all terminal control functions thru use of BTAM-ES. For support of hardcopy terminals it provides a message routing facility.

TTF is a single-thread transaction system. That means, transactions are executed serially one after the other. For VSE/ICCF that means, processing of commands entered from various terminals is serialized. This approach provides good response for a limited number of terminals.

Because of the limited functions of TTF it has a much smaller working set than CICS/DOS/VS and it is simpler to install and to use than CICS/DOS/VS.

Terminal Support in TTF

TTF supports only local terminals. It supports the display terminals 3278 in 3272 Model 2 mode, that means, screen sizes with 1920 characters are supported.

For a more detailed list of terminals supported by TTF see 11.4.5.

VSE/ICCF on TTF

In order to run with TTF VSE/ICCF can be generated with the parameter SYSTEM=TTF. If omitted TTF is default. The parameter SYSTEM=MTCS which had to be specified to run ETSS/II on MTCS or on ETSS/II TCS will not be supported with the program product. If VSE/ICCF uses CICS/DOS/VS, SYSTEM=CICS must be specified.

The DOS/VSE supervisor must be generated with BTAM-ES support.

SUPPORT OF NEW DISPLAY TERMINALS 3278

VSE/ICCF supports the large screen sizes of new display terminals 3278 via CICS/DOS/VS only. Via TTF only the default size of 1920 characters of the new display terminals 3278 stations is supported.

Support for the new display terminals 3278 is provided only as far as new number of lines per screen is concerned, the linesize is always 80, the number of PF-keys remains 12, and the max. number of input lines is still restricted to 4.

Generation of Large Screen Support

If the VSE/ICCF user wants to make use of the large screen sizes of the 3278 Model 3 and Model 4 display stations, two tables of CICS/DOS/VS have to be generated in the following manner:

The corresponding terminal control table entries have to be generated with the new parameter ALTSCRN=(lines,columns) which defines the large screen. The program control table entries for all VSE/ICCF transactions except for I\$5 and I\$6 have to be specified with the new parameter SCRNSZE=ALTERNATE.

Usage of Large Screen Sizes

The /SET SCREEN and SET SCREEN commands are extended with respect to the 'd1' and 'd2' parameters in order to support the large screen sizes.

If the user works with the large screen size of the 3278 Model 3 display station the parameter 'd1', which indicates the starting physical line number of the output area of the screen, may range from 3 to 28. The parameter 'd2', which indicates the number of lines the output area consists of, may range from 5 to 30. The parameter 'r1', which indicates the logical begin of the output area on the screen rather than the physical begin may range from 1 to 26. The parameter 'r2', which indicates the logical end of the output area may range from 5 to 30.

If the user works with the large screen size of the 3278 Model 4 display station, the parameter 'd1' may range from 3 to 39 and the parameter 'd2' may range from 5 to 41, parameter 'r1' from 1 to 37, and parameter 'r2' from 5 to 41.

The VERIFY command is extended with respect to the 'n2' parameter in order to support the large screen sizes. 'n2' which is the total number of lines to be displayed on the 3270 screen when LONG verify mode is set may range from 5 to 28 for the large screen size of the 3278 Model 3 display station and from 5 to 39 for the large screen size of the 3278 Model 4 display station. The default value for 'n2' is 28 for the large screen size of 3278 model 3 and 39 for the large screen size of 3278 model 4.

VSE/VSAM/AMS SUPPORT

VSE/VSAM files (except share option 4) may be shared for concurrent processing between interactive partitions or between interactive partitions and DOS/VSE partitions. AMS is supported to run in a interactive partition.

INTERACTIVE USABILITY AIDSINTRODUCTION

In the following the term prompters will be used instead of interactive usability aids. The prompters are intended to be an interactive and easy to use facility with which even an inexperienced user can utilize the resources of an operating system.

Depending on the kind of job he wants to do the user of the prompters needs to have only little knowledge of DOS/VSE. Especially he does not have to know details on the various components of DOS/VSE, e.g. the format and sequence of required control statements. He also should not have to deal with the characteristics of a certain device he wants to use, e.g. number of bytes per track on a 3340.

The prompters will free the user from looking up those details in the corresponding manuals.

The prompters are application programs or CLISTS running in a VSE/ICCF interactive partition. They act as an interface between the user and DOS/VSE. The user enters his input from the terminal. The prompter then generates a DOS/VSE job stream which may be saved as a member in the VSE/ICCF library and/or submitted to a DOS/VSE partition via VSE/POWER. The saved member may be password protected.

The job streams generated by the prompters are syntactically correct but nevertheless the job may not run due to semantical errors which the prompters are not able to detect. In this case the user will receive a DOS/VSE error message.

Prompter Structure

All prompters are part of a general prompter structure. This structure may be invoked at each of its levels. The levels of the structure represent a certain level of detail. Thus the top level gives a short introduction into the prompters and outlines the areas (installation or daily work) for which prompters exist. The next lower level indicates for which components of an area prompters are available. The next level lists the prompters available for a certain

component. The last level then is the actual prompter doing a special function.

Users

The prompters are intended for users with some DOS/VSE knowledge, they are not intended for the completely inexperienced end user.

It is assumed that the user knows which components exist in DOS/VSE and what their functions are. He does not need to know which control statements are required to execute the component function. He also should not have to deal with the characteristics of an I/O device (e.g. track capacity) but he has to know which devices exist, what their addresses are and which device he wants to use.

IMPLEMENTATION

The prompters may be implemented by using any combination of the following facilities

- Assembler
- PLS
- VSE/ICCF CLIST facility

depending on what fits best for a specific implementation.

Prompters written via the CLIST facility reside as members in the VSE/ICCF library. Prompters written in assembler or PLS reside in the DOS/VSE core image library.

Prompter Macros

Experience has shown that the VSE/ICCF CLIST language is not suitable for long, complex or performance critical prompters. For this reason a set of assembler macros has been developed. Using these macros the prompter programmer will be able to:

- Write the prompter in assembler giving improved performance and greater flexibility.

Keep all language dependent output messages and replies together in a message table and a reply table. This will facilitate the translation into other languages.

Specify for each output message a 'normal message' and a 'help message'. The macros will determine if the terminal user is in 'normal' or 'help' mode and transmit the required message.

Permit the user to 'page back' to the previous prompting conversation.

Redisplay cleared screen.

Display all the input parameters to the user who will then be given the option of either changing selected parameters, or saving the input in an VSE/ICCF library before continuing. The saved input may be used as a model when next using the prompter.

Reload saved member.

Execute VSE/ICCF commands.

Prompter to prompter communication.

Prompter Description

This section summarizes which prompters will be provided for the supported components.

AMS

There will be 16 prompters which supports a subset of full AMS.

DOS/VSE System Utilities

Backup and Restore System (subset)

Copy and Restore Diskette (subset)

Fast Copy Disk (CKD and FBM) (subset)

VTOC Display (subset)

Librarian

A subset of the librarian functions will be supported.

Job Control

The prompters for Job Control make the DOS/VSE JCL, JECL and the VSE/ICCF JES transparent to the user. There will be one prompter.

Terminal Support

The prompts are designed and written for usage on display terminals. They will not be tested on typewriter terminals. The prompts also are designed for a full screen (using a model 2 size screen (1920 char.)), i.e. the user should not modify the size of his screen via /SET SCREEN when using a prompter.

DISTRIBUTION

The prompts are distributed with VSE/ICCF. Prompts written in assembler language and the prompter macros are distributed in the same file as the VSE/ICCF. The names of these file members all start with DTS. This tape is in DOS/VSE backup and restore format.

11.3.5 INTERFACES

A problem program running in an ETSS/II FDP pseduo-partition can run unchanged in the VSE/ICCF program product version with the exception of the difference between CKD and FBM device characteristics, such as track versus block allocation.

VSE/ICCF will support the Interactive Debug FDP 5798-CKF.

11.3.6 SECURITY

In general, all security features provided by ETSS/II are also available with VSE/ICCF, but there is one difference: if VSE/Access Control - Logging and Reporting is installed VSE/ICCF does not use its own data security tables but rather those of the DOS/VSE system. The storage protect feature for interactive partitions is now standard. A permanent file for which the space has been allocated from the dynamic disk space area may be date protected as with normal DOS/VSE files.

11.3.7 SUMMARY OF FDP-PP CHANGESVSE/ICCF GENERATION OPTIONS

The VSE/ICCF generation options are shown below in tabular form summarizing any differences between the FDP and the Program Product. Those denoted unchanged will generally use the same specifications as used in the base FDP. A detailed explanation for the changed, deleted and new parameters are included in this Section.

<u>DTSOPTNS</u>	<u>UNCHANGED</u> (N/C)	<u>CHANGED</u> (C)	<u>DELETED</u> (D)	<u>NEW</u> (N)
ALTSEC	N/C			
ALN2741	N/C			
CANKEY	N/C			
CISIZE	N/C			N
COMLIB	N/C			
CRJE	N/C			
DEVICE			D	
DISPKEY	N/C			
EDFLAG	N/C			
EDEND	N/C			
FP			D	
GETVIS			D	
HCLINE				N
INTCOMP	N/C			
INTDBG			D	
INTRVAL	N/C			
LOADPRT	N/C			
MAXINP			D	
MULTEX			D	
NAME			D	
NDISKS			D	
NBUFS		C		
NRECS		C		
NXTNTS	N/C			
NUSRS	N/C			
NPARTS	N/C			
NTASKS	N/C			
PATCH			D	
PFKEY			D	
PROTECT			D	

PGMRINP	N/C		
PGMRLST	N/C		
PGMRPCH	N/C		
PGMRPIN	N/C		
PGMRLOG	N/C		
POWRPCH	N/C		
PRTMAXD			D
PRTMAXH			D
PRTY			D
PSIZE		C	
PWORK			D
PARTN		C	
RLDBUFF			D
RDR	N/C		
RDR2	N/C		
PCH	N/C		
PRT			D
RPS			D
SPOOL	N/C		
SVCNUM			D
SYSTEM		C	
TCNAME			D
TCUPSI	N/C		
TCTOFS			N
TIOA00	N/C		
TIOA40	N/C		
TIOA70			D
T2740			D
T2741			D
T3767			D
T3270L			D
T3270R			D

Changed VSE/ICCF Generation Options

nbufs

The meaning of NBUFS has not changed, but the user must be aware that when using FBM devices the new CISIZE parameter determines the buffer size used by VSE/ICCF. Since the CISIZE is generally larger than the CKD buffer size, care must be used when applying the NBUFS multiplier in order that the desired working set is maintained. (see CISIZE)

nrecs

The meaning of NRECS for CKD processing has not changed. Its specification for an FBM device will be ignored because the CISIZE parameter determines the size the the

I/O area for FBM devices (see CISIZE). Since a specific device type is no longer specified (see DEVICE), the value must correspond to the most likely device type characteristics in a range of 7-24.

psize

The minimum size of an interactive partition is now 128K virtual memory.

partn

The operand indicating the size of a interactive partition must not be less than 128K.

system

Specify CICS for a CICS/DOS/VS system or TTF for TTF. MTCS/VS is no longer supported. The default is TTF.

Deleted VSE/ICCF Generation Options

DEVICE	
FP	No longer required.
GETVIS	Standard to be compatible with supervisor.
INTDBG	
MAXINP	
MULTEX	
NAME	Fixed name to be assigned by VSE/ICCF.
NDISKS	
PATCH	Maintenance area to be standard.
PFKEY	
PROTECT	Now standard with ICCF=yes in supervisor.
PRTMAXD	
PRTMAXH	
PRTY	
PWORK	
RLDBUF	
PRT	
RPS	
SVCNUM	No longer required.
TCNAME	No longer an option.
TIOA70	
T2740	
T2741	
T3767	
T3270L	
T3270R	

New VSE/ICCF Generation Options

CISIZE =

The control interval size is required to process the VSE/ICCF time sharing library residing on an FBM device. The CISIZE decimal value must be a multiple of the device physical block size which is 512 bytes, and a multiple of 2k if greater than 8k. The maximum value is 32,768 and the default is 2048. This VSE/ICCF generation CISIZE is placed in the FORMAT 1 label area by OPEN and it cannot be altered since VSE/ICCF library record calculations and buffer size allocations are based on this value. The number of buffers allocated via GETVIS is determined by the NBUFS= specification. The user must be cognizant of the CISIZE specification relative to its affect on performance and the VSE/ICCF working set. Namely, the CISIZE in combination with the NBUFS parameters and the related buffer control block areas determine the I/O area portion of the working set. The number of VSE/ICCF logical records per control interval can be determined by the following formula: Records per CI = $(\text{CISIZE} - 10) / 88$ where 88 is the VSE/ICCF logical record size, and the variable is the CISIZE.

For example, using the 2048 default CISIZE:
 $(2048 - 10) / 88 = 23$ Records per CI
with 14 bytes unused.

In the above example, a minimum of 23 logical records will be read or written per physical I/O for the VSE/ICCF library on a FBM device. This corresponds to the NRECS= value used for for a CKD device which is ignored when CISIZE is given.

HCLINE=

In order to avoid that the 328X printers enter an additional blank line in case a line with the length of the printer's physical line size has to be written the physical line size has to be specified via the HCLINE parameter. If the installation has printers with various physical line sizes or if terminals like 2740's are used as hardcopy printers HCLINE=NO should be specified.

TCTOFS =

The CICS/DOS/VS user who has other programs which use the User Area in the TCT can specify the particular part of the area which VSE/ICCF is to use through this parameter. The

valid range is 0 through 255 and the value must be numeric. Several other programs which run under CICS/DOS/VS use this area for their own purposes. If any of these programs expect to pass values from one transaction to another where a VSE/ICCF terminal session could intervene, it is necessary to make sure that these applications do not use the same area as VSE/ICCF. Examples of such applications might include TEP (CICS/DOS/VS supplied or user's own), performance statistics, or page copy functions.

EXTERNAL INITIALIZATION CHANGES

UPSI Switch Setting

The UPSI switch setting for the device type of the VSE/ICCF library will no longer be required nor supported. (see Section 3.1.1.5)

Dummy Label Track Entries

The dummy DLBL statements DTSDUMY, DTSDUM1, DTSDUM2... are no longer required.

Dynamic Disk Space Area

Unlike in ETSS/II where the dynamic disk space area had to be defined via two direct access data sets per extent, the dynamic space area is now two multi-extent direct access data sets with the filenames DTSDYNC or DTSDYNW

Initialisation Option Replies

The facility of predefined initialisation option responses has been extended. The replies are now read via DTFDI from SYSIPT and can therefore reside in a procedure or a SYSIN data stream on disk. In addition if a mistake is made in the predefined reply, the error is typed on the console, the

prompt repeated, and the answer read from the console. In addition, when replies are made via the SYSIPT facility only the important initialisation messages are logged on SYSLOG along with any error or warning messages.

UTILITY AND CONTROL STATEMENT CHANGES

Generally, all control statements referencing devices will no longer require the device operand. Wherever the default device would have been the device specified by DEVICE= on ETSS/II, VSE/ICCF will use the device type per the assignment of the DTSXXXX DLBL/EXTENT statements where applicable. It's recommended that these DLBL/EXTENT statements be permanently located on DOS/VSE standard label area along with the permanent SYS assignment to avoid processing errors when VSE/ICCF determines the DASD device type.

Removal of Submit to Batch Facility without VSE/POWER

A facility is provided in ETSS/II to flag library members for eventual batch execution. Then during a session, the computer operator or system administrator executes the DTSSELCT utility via the master terminal to punch all submitted jobs into the VSE/POWER internal reader queue. This facility was only retained in ETSS/II to be compatible with the previous versions and support will be removed from the program product. The program product will support the job submission and retrieval of printed output via the VSE/POWER PUTSPOOL/GETSPOOL /CTLSPool interface and via SEGMENT which is also included in ETSS/II.

Control Statement Removal

/SUBMIT no longer supported

Utility Removal

Utility to select jobs for batch processing (DTSSELCT) is no longer supported. Utility to clear submit to batch flag bit (DTSCLR) is no longer supported.

Enter VSE/ICCF Commands via Batch UtilityControl Statement Change

\$\$FBA accepted on /INSERT or GETFILE statement for file name of an input file created on an FBM device via SAM-FBM
\$\$DISK will read data from the type of disk specified for the fixed name DLBL statement DTSBTCH and the corresponding assignment

File Maintenance Batch UtilityUPSI Setting for Device Type.

The UPSI switch setting for the VSE/ICCF library is no longer supported.

Control Statement Changes. All device related control statements such as BACKUP, MERGE, RESTORE, INPUT, etc. will no longer require the device operand for the backup or restore file.

VSE/ICCF Library Conversion. To convert an VSE/ICCF library from a CKD to a FBM device it is necessary to use the batch library utility program (DTSUTIL). First, the CKD library is BACKUP'd to tape in one job. Then in a separate DOS/VSE job with the appropriate FB< DLBL and EXTENT statements for the new VSE/ICCF library, a DTSUTIL FORMAT and RESTORE are necessary to format the FBM library and to restore the CKD library to a FBM device. The reverse process also applies when converting from FBM to CKD.

Link and Go Facility Name Change

The VSE/ICCF utility which provides the capability for execution of object card images (called LOADER in the FDP) will be referred to as LINKNGO in the PP, i.e. /LOAD LINKNGO instead of /LOAD LOADER.

Submit to Batch Facility

Changes have been made to the submit to batch facility to allow DOS/VSE JCL and VSE/POWER JECL to be used in a member containing VSE/ICCF JES statements. This facility is

primarily for prompters but can be used by anyone. In the FDP, when a member contained ETSS JES, the first non-ETSS JES card image indicated the start of data. In the PP the first non-VSE/ICCF and non-DOS/VSE (beg. with // or * \$\$) indicates start of data. DOS/VSE type cards are passed unchanged to the output job stream.

The submit facility has also been changed to place an // ASSGN statement in the output stream after each // DLBL and // EXTENT which was converted from /FILE. This // ASSGN statement will generically assign the sysnumber on the EXTENT to the volume on the EXTENT. The submit facility will also output the CISIZE operand on // DLBL statements whenever it is encountered on the /FILE statement.

11.3.8 PERFORMANCE

For performance figures please consult WTSC BOEBLINGEN
publication DOS/VSE PERFORMANCE GUIDE which will be
distributed via DAPS (ZZ10-9823).

11.3.9 DEPENDENCIES

PRODUCTS (EQUIPMENT)

The VSE/ICCF program product supports the same hardware as ETSS/II and in addition the 4300 Processors and FBM DASD devices supported by DOS/VSE and the 3278 terminals.

PROGRAMS

The 'For VSE/ICCF' column is a required prerequisite for VSE/ICCF and the 'For VSE/ICCF User' column indicates programs available to the user of VSE/ICCF.

<u>Control Program</u>	<u>For VSE/ICCF</u>	<u>For VSE/ICCF User</u>
DOS/VSE SCP	X	
Advanced Functions 1.1	X	X
POWER/VS 1.0		X 1
VSAM/AMS 1.0		X

Language/SORT

RPG II 2.0		X
COBOL 2.4		X
PL/I Optimizer		X
FORTRAN		X
VS/BASIC		X
Assembler	X 2	X
SORT/MERGE SM2 2.0		X

Telecommunications

BTAM-ES (CICS/DOS/VS-TTF)	X 3
ACF/VTAM (CICS/DOS/VS)	X 3

Data Base/Data Communications

CICS/DOS/VS 1.4	X
CICS/DOS/VS 1.4.1 (FBM DASD ICR)	

Notes:

- 1 Required for the submit-to-DOS/VSE facility
- 2 Required for installation purposes only
- 3 Either CICS/DOS/VS or TTF is required
- 4 DL/I DOS/VS MPS is not supported
DL/I DOS/VS batch is supported in interactive partitions

11.4 INSTALLATION PLANNING11.4.1 SHIPMENT PACKAGE LAYOUT

VSE/ICCF is distributed in 9 track tape or disk format. The shipment package has the following format:

1. Tape Layout

File 1: Header Record
 File 2: VSE/ICCF Library File
 File 3: Traylor Label
 File 4: Installation Sample Jobs (blocked 3440 bytes)
 File 5: Dummy
 File 6: Dummy
 File 7: Private Libraries (PCL, PRL, and PSL)
 File 8: Dummy
 File 9: Shipment History File

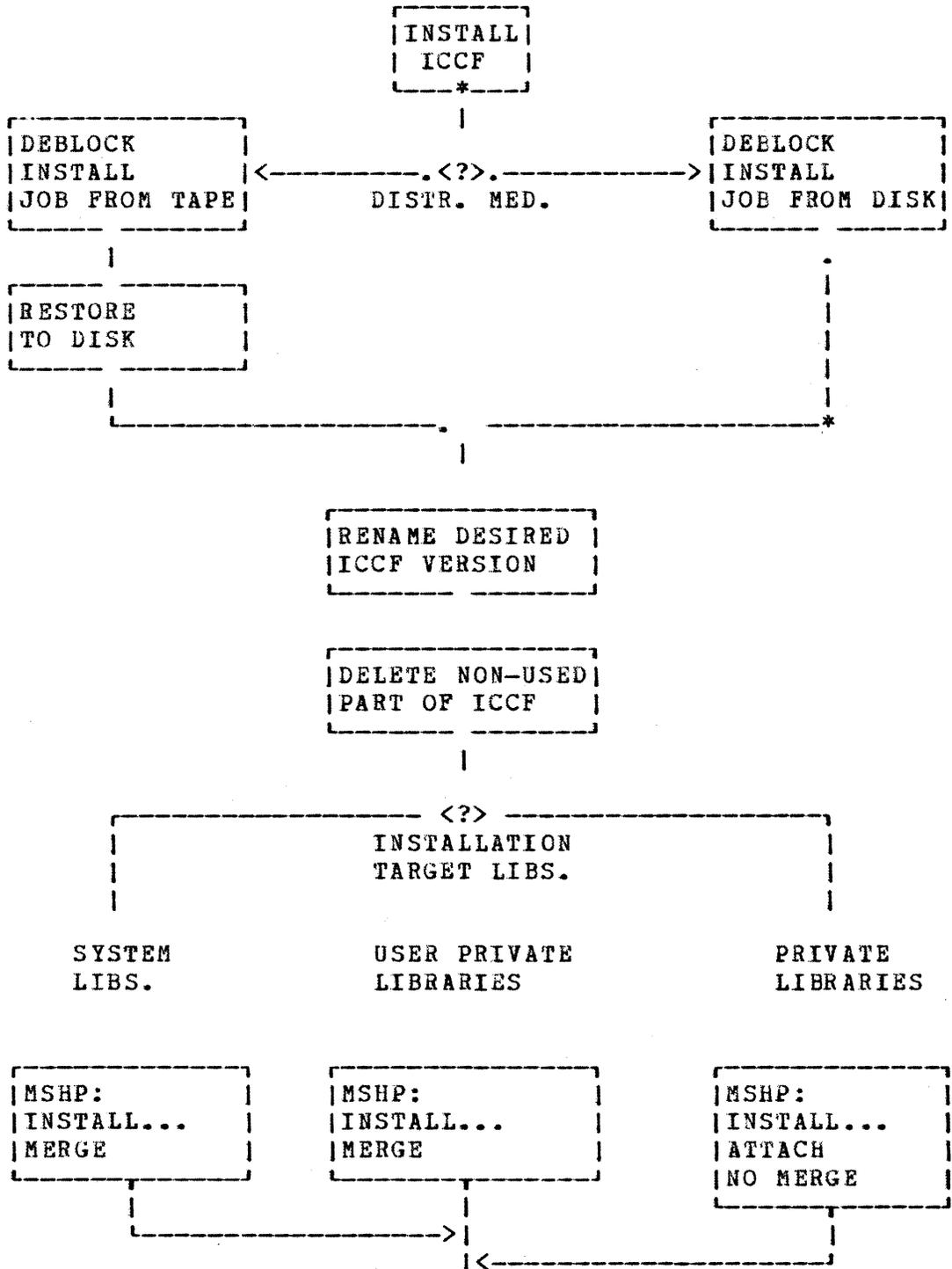
2. Disk Layout

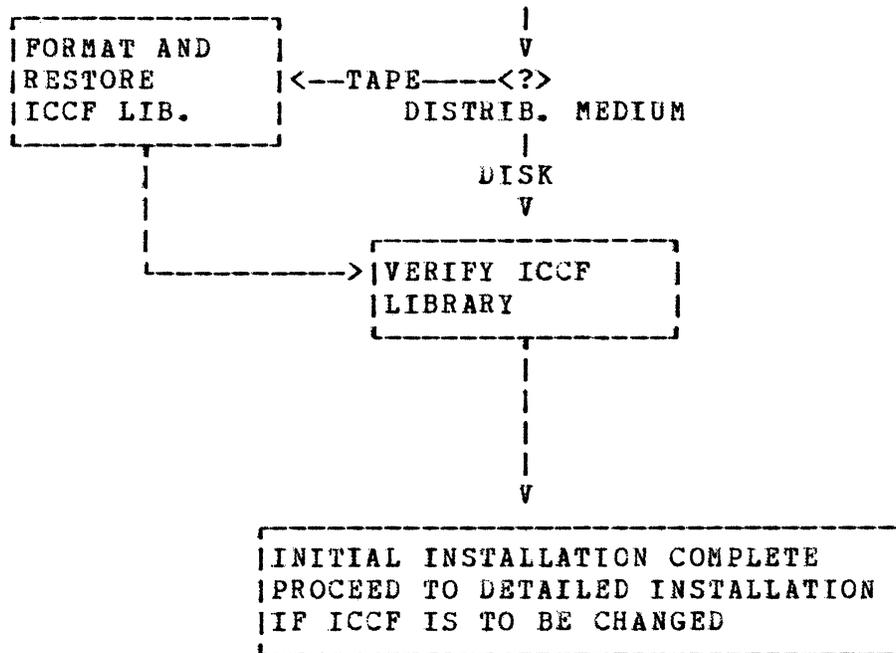
FILE-ID	CONTENTS
A5746TS1.SYSCLB	Private Core Image Library
A5746TS1.SYSRLB	Private Relocatable Library
A5746TS1.SYSSLB	Private Source Statement Library
A5746TS1.HISTORY.FILE	Shipment History File
A5746TS1.ICCF.FILE	VSE/ICCF Library file
A5746TS1.INSTJOB	VSE/ICCF Installation File

The tape or disk contains all VSE/ICCF source components. In addition pregenerated components are available. Source and pregenerated material are stored on tape as DOS/VSE private libraries in RESTORE utility format. Jobstreams for assembling and linking VSE/ICCF components are also available.

11.4.2 VSE/ICCF GENERATION

INSTALLATION FLOWCHART





11.4.3 PREGENERATED VERSION OF VSE/ICCF

A pregenerated version of VSE/ICCF will be available for the CICS/DOS/VS and TTF environment.

If options given by the pregenerated system do not meet the users requirements he should alter the pregenerated system instead of completely reassembling a new VSE/ICCF system, since this will save much assembly time.

A VSE/ICCF system generation macro DTSSG has been developed to support the user in this partial or complete regeneration.

Before starting the regeneration it should be considered if the installation will allow a reconfiguration at VSE/ICCF initialisation time.

Since a lot of the VSE/ICCF generation options may be altered at initialisation time a regeneration might be unnessary.

The following steps have to be performed to regenerate VSE/ICCF:

1. List VSE/ICCF options to be different from the ones in the

- pregenerated version.
2. Also list the options where the default differs from the specification in the pregenerated version.
 3. Specify the wanted options in the macro DTSSG and assemble it. This will produce job control for the jobs to be performed.
 4. Execute the job stream build in the previous step.

11.4.4 DISK SPACE REQUIREMENTS

USER PROFILE AND LIBRARY FILE CONSIDERATIONS

The following questions have to be answered to build up the VSE/ICCF library file.

1. How much space should be allocated?
2. Where should the file be placed in relationship to the page data set and to work files?
3. How many potential library records should be allocated?
4. How many actual libraries should be defined?
5. How many potential user profile records should be allocated?
6. How many actual user profiles should be defined?
7. How are the users to be named?
8. What passwords and profile characteristics should be given to individual users?

Define the VSE/ICCF File

How much Space to allocate

The amount of space to allocate is calculated in terms of 88-byte records. The number of 88-byte records to be allocated may be approximated using the following formula:

$$1000 + (N \times D \times M) + (U \times 5000)$$

where

- N = the number of user profile records which have been defined and
- D = the average number of members which a given user will have in his library and
- M = the average length of a library member and

U = the number of users which may be logged on at one point in time.

Example: For an installation with 4 terminals, 10 user profile records, an average number of 100 members in each users library of an average length of 150 records, the estimated file size would be 171,000 bytes.

Where should the file be placed?

It is important that the timesharing system library file be placed in an optimum location in relation to the page data set and the compiler workfiles.

How many potential Library Header Records?

When the timesharing library file is initially formatted, the user must specify the number of library header records to be allocated. This number represents a pool of records from which actual library header records will be allocated. The specifications should be larger than the anticipated number of libraries to allow for future library expansion without having to reformat or restore the file.

How many actual Library Header Records?

The actual number of library records to be defined depends on need. If 10 user profiles will be entered into the system, no more than 10 actual libraries would be required unless all users had their own libraries and some users owned more than one library.

How many potential User Profile Records?

When the timesharing library file is initially formatted, a pool of potential user profile records must be defined. Actual user profiles need not be specified for all potential profile record allocations. The profile record positions allocated but not defined represent room for expansion.

How many actual User Profile Records?

It is entirely up to the installation how libraries and user profile records are allocated.

How are the users named?

What password and profile characteristics should be given to individual users? Each user whose profile will be added to the VSE/ICCF library file will be represented by a four character identification code which must be specified when logging on. The codes are defined by the installation. They may relate to the user in a mnemonic or functional manner but this relationship is not required. In addition a log on password must be defined for each user identification. Also user characteristics have to be specified which may restrict users in their actions.

The system administrator should have the lowest level of restrictions.

Once these questions have been answered, the installation of the timesharing system should be able to specify the control cards necessary in order to define and initialize the library file.

INTERACTIVE PARTITION WORKFILES

In addition to the space required for the timesharing file itself there is some disk space needed for the interactive partition workfiles. For each interactive partition in which compilations are to be done a set of compiler workfiles is required. The number of workfiles depends on the compilers used. Workfiles can be placed in the dynamic disk space area.

11.4.5 TERMINAL CONTROLTERMINAL CONTROL CONSIDERATIONS

VSE/ICCF makes use of either CICS/DOS/VS or TTF for its terminal control requirements. The generation of the terminal control system (CICS/DOS/VS or TTF) is completely independent of the VSE/ICCF generation. Thus the terminal control system you generate for use with VSE/ICCF will also run in a standalone manner in a DOS/VSE partition. However certain functions must be included in the terminal control system generation in order to allow VSE/ICCF to function properly.

CICS/DOS/VS

Some of the CICS/DOS/VS tables are affected when installing VSE/ICCF. PPT and PCT must be reassembled. DCT has to be reassembled if using hardcopy devices. TCT, TCP and TDT may have to be reassembled. The necessary specifications are described in the manuals. The following CICS/VS modules must be present in the system: DCP, ICP, KCP, PCP, SCP, TCP, TDP and TSP.

TTF

The macros of TTF MTCSG and MTCTT and MTCFT have to be changed. The necessary entries are described in the manuals.

CONTROL UNIT/TERMINAL SUPPORT

CONTROL UNITS/TERMINALS	TTF BTAM	CICS/VS BTAM or VTAM
CHANNEL:		
3272 Model 2 with -3277 Model 2 Display: 1920 -3284 Model 2 -3286 Model 2 -3287 Model 1 and Model 2 -3288 Model 2	X	X
3274 Model 1A (SNA) with -3277 Model 2 Display: 1920 -3278 Model 2 Display: 1920 -3278 Model 3 Display: 1920/2560 -3278 Model 4 Display: 1920/3440 -3284 Model 2 -3286 Model 2 -3287 Model 1 and Model 2 -3288 Model 2 -3289 Model 1 and Model 2		X
3274 Model 1B with -3277 Model 2 Display: 1920 -3278 Model 2 Display: 1920 -3278 Model 3 Display: 1920/2560 -3278 Model 4 Display: 1920/3440 -3284 Model 2 -3286 Model 2 -3287 Model 1 and Model 2 -3288 Model 2 -3289 Model 1 and Model 2	Dspl:1920	X
DOS/VS Operator Console		X
SS:		
2740 Model 1 and Model 2		X
2741		X
3767 in 2740 or 2741 Mode		X

CONTROL UNITS/TERMINAL TYPES	TTF BTAM	CICS/VS BTAM or VTAM
BSC:		
3271 Model 2 with		X
-3277 Model 2 Display: 1920		
-3284 Model 2		
-3286 Model 2		
-3287 Model 1 and Model 2		
-3288 Model 2		
3274 Model 1C with		X
-3277 Model 2 Display: 1920		
-3278 Model 2 Display: 1920		
-3278 Model 3 Display: 1920/2560		
-3278 Model 4 Display: 1920/3440		
-3284 Model 2		
-3286 Model 2		
-3287 Model 1 and Model 2		
-3288 Model 2		
-3289 Model 1 and Model 2		
3275 Model 2 w/Display: 1920		X
-3284 Model 3		
3276 Model 2 with		X
-3278 Model 2 Display: 1920		
-3287 Model 1 and Model 2		
3276 Model 3 with		X
-3278 Model 3 Display: 1920/2560		
-3287 Model 1 and Model 2		
3276 Model 4 with		X
-3278 Model 4 Display: 1920/3440		
-3287 Model 1 and Model 2		

CONTROL UNITS/TERMINAL TYPES	TTF BTAM	CICS/VS VTAM
SDLC:		
3271 Model 12 with -3277 Model 2 Display: 1920 -3284 Model 2 -3286 Model 2 -3287 Model 1 and Model 2 -3288 Model 2		X
3274 Model 1C with -3277 Model 2 Display: 1920 -3278 Model 2 Display: 1920 -3278 Model 3 Display: 1920/2560 -3278 Model 4 Display: 1920/3440 -3284 Model 2 -3286 Model 2 -3287 Model 1 and Model 2 -3288 Model 2 -3289 Model 1 and Model 2		X
3275 Model 12 w/Display: 1920 -3284 Model 3		X
3276 Model 12 with -3278 Model 2 Display: 1920 -3287 Model 1 and Model 2		X
3276 Model 13 with -3278 Model 3 Display: 1920/2560 -3287 Model 1 and Model 2		X
3276 Model 14 with -3278 Model 4 Display: 1920/3440 -3287 Model 1 and Model 2		X

11.5 MIGRATION/COEXISTENCE

The VSE/ICCF Program Product will run on DOS/VSE only. VSE/ICCF as the 'target system' will have several migration 'sources':

1. CICS Source Program Maintenance On-line II (FDP)
2. ETSS/II Field Development Program

11.5.1 MIGRATION PATH FROM SPM TO VSE/ICCF

THE LIBRARY FILE

The first difference when converting from SPM to VSE/ICCF is the library file. Formats, structure, blocking factors are different; therefore VSE/ICCF cannot access a SPM library file without modification. So a conversion will take place. In order to do it smoothly, some points are to be considered.

General Concept

The first approach is to read an SPM Back-up tape and modify it in order to get a tape which will be a valid input for the DTSUTIL program. The DTSUTIL utility program is a VSE/ICCF program which loads new members into the VSE/ICCF library file. Information to do this is documented in the DOS/VS Interactive ETSS/II SE-Guide(DAPS-# ZZ10-9824).

The second approach is to use VSE/ICCF procedures in order to select conversationally the existing SPM programs to be transferred as VSE/ICCF members.

An example of such a procedure is given in the 'ICCF Reference and Installation Manual'.

Library Protection

The only way to have separate libraries in SPM is the facility to use different DOS files together with varying the first letter of the transaction identifier.

In VSE/ICCF, the library file is logically divided in a number of libraries defined by the system administrator. These libraries (Public, Private, Common) have different level of protection and the user's access may be restricted by user profile. In addition to these restrictions an alternate security feature is available which gives even stronger facilities.

Member Protection

In SPM, an 8-character-password is available for member-level protection. In VSE/ICCF, a member may be protected by a 4-character-password. In addition, members may be public or private and the alternate security option has a direct effect on this protection.

Multiple Copies of the Same Member

In SPM, the member name is 8 characters long. Version and level allows you to have a lot of members with the same 8-character name. In VSE/ICCF there is no version and level capability. However, the generation member group offers a similar facility: when the member name has up to 6 characters, it is possible to concatenate with it an hyphen and a 1 digit number. It is so possible to have a maximum of ten versions with the same 6-character programname. The youngest one will be the -0 and oldest one, at least -1 and up to -9.

Compressed and Extended Members

In SPM, compressed and extended members are in different logical libraries so that the same (or the supposed same) member may exist in both libraries, even with same version and level. In VSE/ICCF, a member may exist in compressed or expanded format, even intermixed in the same generation member group, but is always unique. If you want to save an extended format of a compressed member, you have to do it explicitly and save it under a new name (or new generation).

Remark:*It is possible to display a VSE/ICCF compressed member without expanding it.

Member Manipulation

In SPM, include and select, sequence and copy are the two manipulation functions.

In VSE/ICCF we have to make a difference between the input mode, the edit mode, and the execution mode.

In input mode, the command is /INSERT membername. In edit mode, the command is GETFILE membername. In execution mode, the command is /INCLUDE membername which has the same effect as the <INCLUDE> of SPM.

Back up and Restore

In SPM both are done by SPMBATCH with appropriate control and parameter cards.

VSE/ICCF provides the same facility through the DTSUTIL Program. DTSUTIL performs archive and new file format functions, too. Its capabilities also include addition and deletion of libraries, library members, users, and broadcast records; printing and punching of members or libraries; displaying directory lists; display and clear accounting information.

EDUCATE THE USER

Most SPM commands have VSE/ICCF equivalents, but the format is generally different. A first general remark is that SPM works with fixed format commands where VSE/ICCF allows variable format commands and a lot of abbreviations. Some VSE/ICCF commands are much more powerful than the related SPM commands.

DISCOVER NEW FEATURES

The edit facility will easily be understood and used by the new VSE/ICCF user. Execution in interactive partitionns needs some briefing and counseling. Use of tab- and PF-key-setting, macro writing and procedures execution are other new valuable facilities.

Dynamic space allocation within the VSE/ICCF work space via the /FILE statement must not be neglected.

FEATURES DIFFERENT OR NOT AVAILABLE

Automatic Numbering

Automatic numbering of entered programs is not performed in VSE/ICCF except in prompt mode for BASIC and in linemode input in the editor. It is of course still possible to have numbered cards in programs but this is the result of a specific edit command (RENUM). Linemode editing will number all new input but will not renumber existing lines to squeeze in new sequence numbers.

LOD Command

The LOD facility of SPM has no equivalent in VSE/ICCF. The LOD retrieves output from the VSE/POWER punch queue. This is possible because of a modification of VSE/POWER modules provided by SPM.

In VSE/ICCF Rel. 1 it is only possible to display the VSE/POWER reader and punch queue.

Sep. Display of Comp. and Non Compr. Directories

In SPM, there are two different commands (X01 and Z01). In VSE/ICCF, the compression is considered as an attribute of the member with the result that a /LIBRARY FULL display will show all the library directories with all their attributes, included the compressed attributes whenever existing.

TECHNICAL CONSIDERATIONS

Both VSE/ICCF and SPM use dummy devices. Any conflict by specifying these devices will give unpredictable results. One of these experimented results is the inability to submit jobs to VSE/POWER neither in SPM nor in VSE/ICCF. The system administrator must know that SPM scans PUBS in ascending order looking for a specific reader type. VSE/ICCF just knows by generation parameters which are "it's" dummy devices. So the conflict is easily avoided by generating a SPM valid dummy reader with address lower than the VSE/ICCF reserved reader addresses.

11.5.2 MIGRATION PATH FROM ETSS/II FDP

Libraries created under ETSS/II are usable also under VSE/ICCF. For changes to generation options, initialization, commands, control statements and utilities please refer to the applicable sections of this chapter.

11.6 REFERENCE MATERIAL

The following manuals and documentation will be available:

FORM NO.	SUBJECT	DESCRIPTION
SB21-2145	ETSS/II	ETSS TERMINAL USER'S GUIDE
SB21-2331	ETSS/II	ETSS FULL SCREEN EDITOR TERM. USER'S GUIDE
SB21-2146	ETSS/II	ETSS TERMINAL USER'S SUMMARY
SB21-1982	ETSS/II	ETSS PROGRAM DESCRIPTION AND OPERATIONS
GC33-6066	VSE/ICCF	VSE/ICCF G.I.M.
SC33-6067	VSE/ICCF	VSE/ICCF Installation Reference
SC33-6068	VSE/ICCF	VSE/ICCF Terminal Users Guide
SC33-6069	VSE/ICCF	VSE/ICCF Messages
LY33-9081	VSE/ICCF	VSE/ICCF PLM

ZZ10-9826

DOS/VSE PLANNING GUIDE
Volume II

Chapter 11.6
VSE/ICCF

Chapter 11.6
VSE/ICCF

DOS/VSE PLANNING GUIDE
Volume II

ZZ10-9826

CHAPTER 12

DOS/VSE ACCESS CONTROL FACILITY

April 1979

04/02/79

IBM INTERNAL USE

CONTENTS

CHAPTER 12 DOS/VSE Access Control Facility12.1

12.1 INTRODUCTION.12.1

12.2 System Requirements for DOS/VSE Access Control Facility.12.2

12.3 DOS/VSE Access Control-Logging and Reporting R.1 (5746-XE7).12.3

12.3.1 Highlights.12.3

12.3.2 Access Control Logging.12.4

12.3.3 Access Control Reporting.12.5

12.4 Affected components in DOS/VSE.12.7

12.4.1 Access Control hooks in CSC12.7

 Supervisor Hits.12.7

 IPL Hits12.7

 JCL Hits12.8

 Librarian Hits12.9

12.4.2 Security Hooks in BAM12.9

 Tape OPEN for Labelled and Unlabelled Tapes.12.9

 Hits in the Master OPEN Transients12.10

12.5 VSE/ICCF Considerations12.11

12.5.1 Attach Logging Task12.11

12.5.2 Access Control Table.12.11

12.5.3 DOS/VSE Access Control Function Versus VSE/ICCF Security.12.14

12.6 VSE/VSAM Implications12.16

12.7 Equipment supported12.17

12.8 Performance12.18

12.8.1 LOGGER.12.18

12.8.2 Reporting Program12.18

12.8 Reference Material.12.19

12.1 INTRODUCTION

It is the aim of data security to protect certain areas of an installation from unauthorized access. Therefore it is necessary to identify the resources and the users authorized. This means that one has to:

- * define user profiles and to
- * define resources to be protected.

Now with a certain authorization method verify one against the other:

- * verification

After a verification has taken place security events have to be recorded and reports to be printed:

- * logging
- * reporting

The different points reflect different components in the DOS/VSE system:

The definition of user profiles and resources is done via a VSE/ICCF table -DTSECTAB- see 12.5.2.

Verification is done by DOS/VSE for instance at OPEN time of files - see 12.4.

Logging and reporting is done via the new program product DOS/VSE Access Control-Logging and Reporting - see 12.3.

12.2 SYSTEM REQUIREMENTS FOR DOS/VSE ACCESS CONTROL FACILITY

- * DOS/VSE
- * VSE/ADVANCED FUNCTIONS
- * VSE/ICCF
- * DOS/VSE Access Control-Logging and Reporting
- * (Sort/Merge 5746-SM2 if wanted for sorting reports)

12.3 DOS/VSE ACCESS CONTROL-LOGGING AND REPORTING R.1
(5746-XE7)

The DOS/VSE Access Control-Logging and Reporting PP is part of the DOS/VSE Access Control Function offered to all users of DOS/VSE and designed to assist in auditing a DOS/VSE data processing installation.

It requires the 'Compare and Swap' hardware feature.

Status	:SPL
Product Latest Release	:1.0
SIPO available	:Yes (Optional Feature)
Installation License	:N/A
Pre Installation Test	:30 Days
Points Value	:35

12.3.1 HIGHLIGHTS

- * Logging of information about access related events to protect resources (i.e. data sets, libraries and programs) to a sequential data set called the log data set file.
- * Logging is optional.
- * Logging is performed asynchronously by a DOS/VSE subtask attached to VSE/ICCF.
- * Reporting provides a printout of selected data from the log data set for auditing purposes.
- * Selection criteria are provided to the reporting module during program execution by means of control statements.
- * Several reports corresponding to various selection criteria may be obtained in one program run.
- * Access event data may be archived on a tape and restored from tape for deferred reporting.
- * The log data set consists of two parts used alternatively: while one part is used for logging the other one may be used for reporting.

12.3.2 ACCESS CONTROL LOGGING

The Logger is responsible for writing all security related events out to the log data set. This data set consists of two parts used in flip/flop mode, and its size can be predefined by the user. Input to the Logger will consist of a queue, called log queue, residing in main storage acquired by the Logger using a GETVIS request. A pointer to the header of the logging queue is contained in the Security Vector Table. For the maximum number of entries in the logging queue see 12.4.1. This number is also contained in the Security Vector Table. Elements will be added to this queue by all access control routines detecting a security event, and elements will be removed from the queue by the Logger after a log record has been written to the log data set.

The following types of events may be logged:

- * data set access
- * library access
- * program access

The type of event actually logged depends on the logging options in effect at the time when the event occurs. Furthermore, it depends on the logging options whether each of the specified events will be logged or only security violations. For more information see 12.5.2.

The Logger runs as a subtask attached automatically by the VSE/ICCF maintask if SEC=xx is generated in the supervisor. Precautions are taken so that the Logger will not be attached several times.

When performing a shut-down operation for VSE/ICCF, this will also mean a shut-down for the Logger. In a system with SEC=xx, VSE/ICCF will place a special request for the Logger into the logging queue and after termination of the Logger proceed with VSE/ICCF shut-down

12.3.3 ACCESS CONTROL REPORTING

While the Logger logs on one part of the log data set the reporting module may process the other part simultaneously.

The reporting module runs in a DOS/VSE batch partition. When the log data set the Logger is currently using is full, it will be closed and the operator will be notified. The message displayed will contain the name of the log data set now full. The operator now has to initiate the reporting module, so that after processing of the log data set it will again be available to the Logger.

The Reporting module performs four functions:

- * Print out selectively the content of the log data set giving a formatted list which will be used for auditing purpose. The selection criteria are specified in a control statement provided at execution time and checked for validity by the program. Several reports corresponding to various selection criteria can be obtained in one program run. Since the data is not destroyed the program can be run several times.
- * Dump the log data set from the disk onto a tape or disk, if needed for later processing or for archiving.
- * Restore the log data set from tape or disk onto disk, if needed for deferred processing.
- * Clear the log data set by using a control statement. This will make the data set ready for further use by the Logger.

The SELECT Control Statement

In the SELECT control statement selection criteria for the reports can be specified. If a specification is not given it is not taken into consideration for the selection.

SELECT	BEGIN=date1,time1	date as specified in the DOS/VSE system and time hh:mm:ss
	END=date2,time2	where selection is to start date and time where selection is to stop
	USERID=name1,...	user-id causing the event
	EVENT=event1,...	PROGRAM, DATASET or LIBRARY can be specified
	DSNAME=name1,...	data set name
	PGMNAME=name1,...	names of phases from CL
	VIOLATION=YES NO	security event or violation
	SORT=arg1,...	USERID, TIME, EVENT, RNAME

and VIOLATION can be
specified. (RNAME=resource name)
PRINT=ALL|SUMMARY complete list or summary page

Access Control Reports

The output is printed on SYSLSST in form of one or several reports, depending on the number of SELECT statements. Each SELECT statement corresponds to one report. Each report contains all the records corresponding to the criteria specified. The records are then printed in the order specified in the SORT parameter, or in the order they were recorded in the log data set, if no SORT parameter was specified. Each line contains the elements provided in the log data set:

- * time at which the security event occurred
- * date at which the security event occurred
- * event description indicator (security violation or not)
- * user identification, if supplied
- * user name, if supplied
- * logging reason
- * data set name or program name

At the end a summary report will be printed.

12.4 AFFECTED COMPONENTS IN DOS/VSE12.4.1 ACCESS CONTROL HOOKS IN CSCSUPERVISOR HITS

The supervisor will have a new generation option, which will generate the corresponding supervisor code.

FOPT SEC=xx

where xx is a value between 10 and 32k-1.

This value specifies the number of entries in the logging queue. Each entry in the logging queue requires 84 bytes of storage. In installations with high logging activity (many protected resources and all accesses to be logged) a larger value should be specified than in installations with low logging activity.

IPL HITS

If a system is started with SEC=xx then IPL sets up storage in the system GETVIS area for security related control blocks.

The new load list \$SVASEC contains the phases to be loaded into the SVA as Security Routine, Authorization Routine, Open Appendix Routine and the Access Control Table.

IPL sets up pointers to the phases in the Security Vector Table.

All SVA phases must be available during IPL. For the case that not all phases are available It is advisable to also generate a supervisor without access control.

JCL HITS

// ID Statement

A new JCL statement is introduced which contains the user identification and his password. This statement must immediately follow the // JOB statement.

```
// ID USER=XXXX,PWD=XXXXXX
```

where USER is the user identifier and PWD a 3 to 6 alphameric character password. Both parameters will not be displayed on SYSLOG or SYSLSLST.

An access control routine checks this information against the contents of the Access Control Table (DTSECTAB) and, depending on the result of this check, the job is allowed to run or will be cancelled.

It is possible to run jobs without an ID statement (for compatibility reasons), but the job is canceled as soon as an access to a protected resource is attempted.

// EXEC Statement

Before starting an execution, job control has to check, if the current user-id is allowed to work with the PCL currently assigned (if there is one), maybe from a previous job under another user-id.

This check will be done by issuing an OPEN for the PCL.

If a access control check occurs, JCL will issue a message and allow the operator either to assign a new PCL or to unassign the existing one and issue a new // EXEC.

NOTE: This check at // EXEC time means, that the label information for the PCL must be available, else an OPEN label error will occur.

Also a FETCH/LOAD Appendix Routine is part of a new SVC routine. After authorization checking during OPEN control is passed to this routine.

ASSGN SYSCLB,... Statement

When assigning a PCL, OPEN is done by job control. The used DTF has a DTF extension containing security related information.

If a security violation occurs, then JCL issues a message and the operator has the chance either to assign another library, unassign the existing one or enter a new // ID statement.

LIBRARIAN HITS

Whenever a private library is to be accessed the corresponding librarian program issues an OPEN. An exception are PCL's; they are opened at ASSGN time. Before issuing the OPEN, a DTFCP or DTFPH is set up. For access control checking the librarian DTF's contain DTF extensions with pointers to the Authorization Parameter List containing for instance the READ-only or UPDATE indicator for the library access. This is transparent to the user.

NOTE: As system libraries are not opened they are not candidates for access control checking.

12.4.2 SECURITY HOOKS IN BAMTAPE OPEN FOR LABELLED AND UNLABELLED TAPES

Labelled tape OPEN currently allows the operator to skip over label errors, or even process a labelled input tape as 'UNLABELLED'. In a system with SEC=xx, this will not be allowed. The operator response 'IGNORE' will not be accepted as a valid response to the following messages: 4111A, 4112A, 4113D, 4115A, 4118D, 4119A, 4123D, 4125D, 4132D, 4133D.

No logging will be performed for such attempted tape label check violations.

HITS IN THE MASTER OPEN TRANSIENTS

The \$\$BOPEN and \$\$BOPENR transients have to check, if they are running in a system with SEC=xx. If so they have to branch to the OPEN Appendix Routine.

This routine collects user and data set related information which then will be checked by the Authorization Routine against the contents of the Access Control Table.

12.5 VSE/ICCF CONSIDERATIONS

12.5.1 ATTACH LOGGING TASK

The logging task is attached as a subtask by the main task of VSE/ICCF.

12.5.2 ACCESS CONTROL TABLE

Resources to be protected must be defined in the access control table DTSECTAB. The following types of resources can be protected:

* Files

- all DASD files and labeled tape files defined by a file description macro DTFxx, or by appropriate file definition statements of the installations IBM compiler(s)
- VSAM user files and VSAM catalogs

* Private Libraries

* Phases

- phases in the system core image library
- phases in a private core image library

Resources may also be protected by specifying a group name. Then all resources whose initial characters match that group name are protected, which is especially valuable for resources that have just been created and do not have an own table entry.

NOTE: DTSECTAB itself must be protected by specifying an entry in the access control table for the phase name DTSECTAB. This entry should allow only the installation's security administrator to access that phase.

In addition to the resources also the users must be specified in the access control table.

The macro DTSECTAB serves to specify and to generate the

access control table. The macro has to be assembled, link edited, and cataloged to the system core image library.

```
DTSECTAB TYPE=USER|GROUP|LIB|FILE|PHASE,  
          NAME=username|groupname|libname|filename|phasename  
          (,PASSWRD=password)  
          (,AUTH=YES|NO)  
          (,ACC=(class,class,...U))  
          (,LOG=(class,class,...))  
          (,SUBTYPE=INITIAL|FINAL)
```

TYPE defines the type of entry: user or resource

NAME defines the name of user or resource

for USER a 4 character user id

for GROUP a 1 to 7 character group name

for LIB a 1 to 44 byte file identification

for FILE a 1 to 44 byte file identification

for PHASE a 1 to 8 byte phase name

PASSWORD 3 to 6 characters if TYPE=USER

AUTH=YES if TYPE=USER and security administrator;
can only be specified once

ACC 1 to 32 access control classes; U is update
and can only be specified for TYPE=USER

LOG 1 to 32 logging classes; if not specified only
access violations are recorded

SUBTYPE first macro must contain INITIAL, last
macro FINAL

Here some examples:

```
DTSECTAB TYPE=USER,  
          NAME=PAUL,  
          ACC=(1-32,U),  
          PASSWRD=XB3L25,  
          AUTH=YES,  
          SUBTYPE=INITIAL
```

This user is the security administrator. Only one such entry is possible in the whole system. The ACC parameter is only specified for clarification, since the security administrator indication (AUTH=YES) bypasses all access authorization checks.

```
DTSECTAB TYPE=USER,  
          NAME=OTTO,  
          ACC=(1-8,25-32,U),  
          PASSWRD=7BMC3U
```

This user can access only resources defined with access control classes 1-8 and 25-32; he has read/update access to those with classes 25-32 and only read access for classes 1-8.

```
DTSECTAB TYPE=GROUP,  
      NAME=USER,  
      ACC=(1-8,25,32),  
      LOG=(1-8,25)
```

This group entry allows users with matching access control classes to access files, libraries and phases that begin with the character string 'USER'. The LOG specification causes all accesses for classes 1-8 and 25 to be logged; only access violations are logged for class 32.

```
DTSECTAB TYPE=LIB,  
      NAME=USER.PRIVATE.CIL,  
      ACC=(1-8,32),  
      LOG=(1-8)
```

Note that this entry begins with the characters 'USER' which is also defined as a group entry. The access and log specification is a subset of that group entry. However, for library access validation the library entries are searched first before the group entries. This means that users with access class 25 cannot access the library 'USER.PRIVATE.CIL', even though access is permitted at the higher 'USER' group level.

```
DTSECTAB TYPE=PHASE,  
      NAME=DITTO,  
      ACC=(25-32),  
      LOG=(25-31),  
      SUBTYPE=FINAL
```

The SUBTYPE specification has to be made for the first and last macro specified.

Now for access control checking the user information and the resource information, both specified in DTSECTAB, is compared.

This check is done in two steps:

- A check for the match of the access control class
If there is a match for the access control class for the user and the resource, processing can continue. Otherwise a security violation is indicated. This check is done for all resources.
- A check for a match of the READ/UPDATE indication.
This check depends on the fact if the file was opened for input or update. If there is a match for the READ/UPDATE indication for the user and the resource, processing can continue. Otherwise a security violation is indicated.

Note: access control distinguishes between read or update authorization for VSAM files only.
For all other files or libraries update

authorization is required.

The access control table is a SVA resident phase, loaded into the SVA at IPL time. It is possible to update and catalog DTSECTAB during processing and the system will always use the latest version. No re-IPL is necessary, as long as the SVA has sufficient space available for a new copy of the table. After a successful link editing of DTSECTAB in the BG partition, the processing of the next ID statement makes the new version available.

Sensitive information in the table will be scrambled to provide additional protection.

When creating the table with the macro DTSECTAB, the password specified in the 'USER' entries is scrambled, and will be displayed in the assembly listing only in the scrambled format.

When issuing the first // ID statement after creating the table DTSECTAB, or after loading a new version into the SVA, a special routine will automatically be invoked which will scramble all specified USER-IDs and (a second time) all PASSWORDS in the SVA.

12.5.3 DOS/VSE ACCESS CONTROL FUNCTION VERSUS VSE/ICCF SECURITY

If DOS/VSE has been generated with SEC=NO, VSE/ICCF uses its system program, load protection, and system file table for access authorization checking, if VSE/ICCF users access DOS/VSE resources defined in those tables for processing in an interactive partition.

If DOS/VSE has been generated with SEC=xx, the three tables are ignored for access authorization checking since the access authorization checking for DOS/VSE resources is done by using the Access Control Table (DTSECTAB) in DOS/VSE (the access request may originate from VSE/ICCF or DOS/VSE). Therefore, for a system with SEC=xx, all DOS/VSE resources to be protected must be specified in DTSECTAB.

For VSE/ICCF users a user profile and password has to be specified under VSE/ICCF. If users working under VSE/ICCF in interactive partitions are also allowed to submit jobs to a DOS/VSE batch partition they have also to be defined in DTSECTAB.

Jobs submitted to the DOS/VSE batch can consist of:

1. DOS/VSE job control
2. VSE/ICCF job entry statements.

In the first case the job is put into the VSE/POWER reader queue as it is, and therefore must contain a valid ID statement.

In the second case DOS/VSE JCL is created by VSE/ICCF and therefore the ID statement is also created by VSE/ICCF, which is possible through the user profile information available to VSE/ICCF. A verification check of that information was already done at VSE/ICCF log-on time.

12.6 VSE/VSAM IMPLICATIONS

In a system that has the DOS/VSE Access Control Function installed, the REWIND option must be specified in the following VSE/VSAM commands: EXPORT, EXPORTRA, IMPORT, IMPORTRA, PRINT and REPRO. This means that no multifile volumes on unlabeled tapes are supported.

12.7 EQUIPMENT SUPPORTED

- * Any CPU supported by DOS/VSE
- * Disk devices:
 - 2314
 - 3330 mod. 1 and mod. 11
 - 3340/44
 - 3350
 - FBM
- * Any device supported by DOS/VSE DTFMT may be used for backout the log data set for archiving purposes.

12.8 PERFORMANCE

12.8.1 LOGGER

Performance of the logging facility is dependent on the following facts:

- * number of requests for logging issued per second
- * priority of the logging task (i.e. priority of the main task to which the logger is attached)
- * size of the logging queue.

If the logger has low priority and there are many requests for logging, the log queue may be frequently full, thus forming a bottleneck for the system.

12.8.2 REPORTING PROGRAM

The reporting program is run offline in a batch partition and thus does not have critical performance requirements.

12.8 REFERENCE MATERIAL

GH12-5045 Licensed Program Design Objectives
GH12-5130 General Information Manual
SH12-5536 Program Reference and Operations Manual
GC33-6077 Data Security Guide under DOS-VSE
LY12-5032 PLM

CHAPTER 13

SYSTEM INSTALLATION PRODUCTIVITY OPTION/ EXTENDED

- SYSTEM IPO/E -

CONTENTS

Chapter 13 13.1
SYSTEM INSTALLATION PRODUCTIVITY OPTION/EXTENDED
(SYSTEM IPO/E). 13.1

13.1 Introduction. 13.1
13.1.1 What is System IPO/E. 13.1

13.2 System IPO/E Highlights 13.2

13.3 System IPO/E Facilities 13.3
13.3.1 System IPO/E Options. 13.4

13.4 System IPO/E Contents 13.5
13.4.1 Base 13.5
13.4.2 Optional Features. 13.5
13.4.3 Details of base and Optional features 13.6
13.4.4 Data Security:. 13.7
13.4.5 SNA Support:. 13.7

13.5 Functional Details of the facilities. 13.8

13. 6 Supported Configurations : 13.9
13.6.1 Hardware. 13.10
Prerequisite configuration 13.10
Preconfigured System 13.10
Operating Environments 13.10
DOS/VSE System IPO/E - Example 13.11
13.6.2 Software Configuration. 13.12

13.7 Compatibility of Component Products 13.13

13.8 Programming Service and Update Information. 13.14
Programming Service / Currency 13.14
Documentation. 13.14
Updates. 13.14
Retain 13.14

13.9 License Agreement, Terms and Conditions 13.15
Availability 13.15
Terms and Conditions 13.15

13.10 Bibliography 13.16

13.1 INTRODUCTION

This chapter describes briefly the System IPO/E Release 1. In particular the DOS/VSE System IPO/E is covered. The VM based System IPO/E's are not addressed in this guide.

SYSTEM INSTALLATION PRODUCTIVITY OPTION/EXTENDED (SYSTEM IPO/E)

DOS/VSE System IPO/E	Option 1	5750-AAA
	Option 2	5750-AAB
	Option 3	5750-AAC
VM System IPO/E		5750-AAD
VM/DOS/VSE System IPO/E	Option 1	5750-AAE
	Option 2	5750-AAF

Status	:	SPL program product
Release	:	1
System IPO/E Available	b:	This product is System IPO/E
Installation Licence	:	N/A
Pre-Installation Test	:	N/A

Planned Availability	:	DOS/VSE System IPO/E	-	3Q79
		VM System IPO/E	-	3Q79
		VM/DOS/VSE System IPO/E	-	4Q79

13.1.1 WHAT IS SYSTEM IPO/E

The System IPO/Es are designed to reduce the time and skill level required to install, service and use DOS/VSE, VM/370 and other selected licensed programs. System IPO/Es are a new generation of software productivity options which expand installation assistance into both system management and usage. They consist of preconfigured bases of pregenerated products with preapplied service, ready to use in specific operating environments. System IPO/Es are an extension and replacement of the successful and proven DOS/VS and VM/370 System IPOs and provide a simplified, more productive interactive interface to assist customers in managing, servicing and using their system more effectively.

13.2 SYSTEM IPO/E HIGHLIGHTS

HIGHLIGHTS

System IPO/Es enhance user productivity, system serviceability and application growth by providing:

- Preconfigured systems
 - for immediate installation and use.
- Predefined procedures
 - for installation of features and application of preventive service across System IPO/E base and selected featured components.
- Interactive full-screen display and prompting
 - for easier and more productive communication with the system.
- Online information
 - designed to reduce user prerequisite education and assist the new or infrequent user.
- Executable procedures
 - that reduce the need for detailed knowledge of VM/370 system commands and DOS/VSE control statements.
- Enhanced problem reporting facilities
 - for effective management and control of system/user errors.
- Communication procedures
 - to assist block data transfer with remote systems.

13.3 SYSTEM IPO/E FACILITIES

The System IPO/Es are operational interactive systems, which utilize the Interactive Productivity Facility (IPF) in conjunction with VSE/ICCF in the DOS/VSE System IPO/E and CMS in the VM and VM/DOS/VSE System IPO/Es.

System IPO/Es comprise:

- Pregenerated systems
 - with preapplied service that have been designed for operation in specific environments.
- Interactive procedures
 - or documentation with a resulting simplification of the installation of licensed program features and the servicing of DOS/VSE and VM/370 systems, utilizing the MSHP/PUT and PLC processes.
- Full-screen display, prompts and aids
 - that minimize the potential for errors in installing, servicing and using the system.
- Online information
 - in the form of explain facilities and a first use tutorial to increase or refresh knowledge of System IPO/E.
- User oriented procedures
 - that provide system guidance and direction based upon CP/CMS commands and DOS/VSE JCL.
- Structured problem reporting system
 - based on IPCS to assist in APAR generation and easier problem management.

- System communication procedures
 - implemented under DOS/VSE and VM to control/direct the flow of user data to remote systems.

13.3.1 SYSTEM IPO/E OPTIONS

The DOS/VS System IPO/E is available in three options :

OPTION	
1	Batch/Interactive/DB/DC
2	Batch/Interactive/DC
3	Batch/Interactive

13.4 SYSTEM IPO/E CONTENTS

Each of the System IPO/Es consist of a preconfigured base and a series of optional features.

13.4.1 BASE

The base is an integrated set of software components, pregenerated, preallocated with preapplied service and configured to a specific operating environment. A System IPO/E order results in a minimum shipment of the integrated base components.

13.4.2 OPTIONAL FEATURES

Besides the base other components are available as optional features. Base and featured components receive synchronized service. Cross-component system validation via Installation Verification Procedures is performed on a system which includes both base and all optional features.

13.4.3 DETAILS OF BASE AND OPTICNAL FEATURES

DOS/VSE System IPO/E: 3 Options

PROGRAM NAME	NUMBER	REL LVL	Option 1	Option 2	Option 3
			5750-AAA B/I/ DB/DC	5750-AAB B/I/DC	5750-AAC B/I
<u>BASE CONTENT</u>					
DOS/VSE	5745020		X	X	X
VSE/ADVANCED FUNCTIONS	5746XE8	1.0	X	X	X
VSE/IPCS	5745SA1	1.0	X	X	X
VSE/ICCP	5746TS1	1.0	X	X	X
VSE/POWER	5746XE3	1.0	X	X	X
BTAM-ES	5746RC5	1.0	X	X	X
VSE/VSAM	5746AM2	1.0	X	X	X
INTERACTIVE PRODUCTIVITY FACILITY	5748MS1	1.0	X	X	X
CICS/VS	5746XX3	1.4	X	X	
DL/I DOS/VS	5746XX1		X		
<u>FEATURES</u>					
DATA DICTIONARY	5746XXC		X		
DMS/CICS/VS-DOS	5746XC4	1.0	X	X	
VSE/PT	5796PLQ		X	X	X
CICS/VS PA II	5798CFP		X	X	
VSE/POWER/RJE	5746XE3	1.0	X	X	X
DOS/VSE RJE WORKSTATION	5746RG9		X	X	X
IIS	5748XX6		X	X	
VSE/DITTO	5746UT3		X	X	X
COBOL	5746CB1		X	X	X
RPG II	5746RG1	3.0	X	X	X
PL/I OPT./LIBS.	5736PL3		X	X	X
PL/I OPT.	5736PL1		X	X	X
PL/I RES. LIB.	5736LM4		X	X	X
PL/I TRANS. LIB.	5736LM5		X	X	X
SORT/VS	5746SM2		X	X	X
EP/VS (3705)	5747AG1		X	X	X
VSE/ACCESS CONTROL - LOGGING AND REPORTING	5746XE7		X	X	X
VSAPL	5748AP1		X	X	

Note: Where a particular release level is not indicated, the release information will be specified at System IPO/E availability.

13.4.4 DATA SECURITY:

Customers using System IPO/Es can use the security facilities provided by VSE/ICCF in DOS/VSE and the security facilities provided by VM/CMS in VM/370. Customer management is responsible for the selection, implementation and adequacy of these facilities.

13.4.5 SNA SUPPORT:

IBM will include ACF/VTAM and ACF/VTAME as part of DOS/VSE System IPO/E and VM/DOS/VSE System IPO/E future releases. The first System IPO/E SNA support is planned for 1Q80.

13.5 FUNCTIONAL DETAILS OF THE FACILITIES

Detailed information will be provided in the System IPO/E General information manual and the Planning Guide.

13. 6 SUPPORTED CONFIGURATIONS :

A series of user selectable configuration environments based on the DOS/VSE System IPO/E will be available to support the new IBM 4300 Processors and existing S/370 systems.

- DOS/VSE System IPO/Es are designed for DOS/VSE users to support commercial interactive applications in both stand-alone and remote configurations. Supporting a wide range of hardware including the 4331 Processor, 4341 Processor and comparable S/370. DOS/VS System IPO/Es are available in three options:

OPTION		MINIMUM SYSTEM SIZE
1	Batch/Interactive/DB/DC	1 Megabyte
2	Batch/Interactive/DC	1 Megabyte
3	Batch/Interactive	.5 Megabyte

13.6.1 HARDWARE

PREREQUISITE CONFIGURATION

The prerequisite configuration is defined as the minimum required to install and service the System IPO/E

PRECONFIGURED SYSTEM

This is defined as the maximum number of devices supported without system regeneration.

OPERATING ENVIRONMENTS

The System IPO/Es will be validated in a number of operating environments. System IPO/E prerequisite and a series of preconfigured operating environments exist for 4300 Processors and S/370s, details of which will be available in the System IPO/E General Information Manual (GC20-1873). The customer is responsible for installing machine configurations of sufficient capacity to permit use of the products selected.

An example of prerequisite and preconfigured environment for 4300 Processors is given below:

DOS/VSE SYSTEM IPO/E - EXAMPLE

	Hardware	Prerequisite	Preconfigured
CPU	4331	1	1
	B/I	.5 MB	.5 MB
	DC	1.0 MB	1.0 MB
	DB/DC	1.0 MB	1.0 MB
DASD	3310 Spindles		
	B/I	4	8
	DC	5	8
	DB/DC	5	8
Console	3278-2A	1	1
Printer	3203-5 or	1	1
	3289-4		
Card	3505	1	1
Reader or	or		
Diskette	#3401		
Card Punch	3525	0	1
Tape	8809	1	6
Terminals	3278-2	1	14
	3287	0	1
	Remote Display Stations	0	8
CCU	CA or 370X	0	1

Notes:

1. Comparable S/370 configurations and DASD requirements are contained in the System IPO/E General Information Manual (GC20-1873).
2. Prerequisite hardware configurations are the minimum required to install and service the System IPO/E.
3. Preconfigured defines the maximum number of devices supported without regeneration. System IPO/E supports expansion of system parameters up to the component product maximum.
4. Card reader or diskette is included in the prerequisite configuration where it is required for hardware service.
5. The individual programming components contained in the System IPO/Es are being provided in an unmodified form. Therefore, the preconfigured systems and/or the individual programming components may be tailored by the customer to obtain support for devices, e.g. 3262-1, 3370, which are otherwise supported by the programming components.

13.6.2 SOFTWARE CONFIGURATION

See section 13.4.3 for details of the software supported in the base and optional features configurations.

13.7 COMPATIBILITY OF COMPONENT PRODUCTS

The individual programming components contained in the System IPO/Es are being provided in an unmodified form. Therefore, the preconfigured systems and/or the individual programming components may be tailored by the customer to obtain support for devices, e.g. 3262-1, 3370, which are otherwise supported by the programming components.

13.8 PROGRAMMING SERVICE AND UPDATE INFORMATION

PROGRAMMING SERVICE / CURRENCY

The Programming Services or Programming Service Classification and Currency of each individual component contained in the System IPO/E is unchanged from that stated in the availability notice of the individual component.

DOCUMENTATION

Detailed information on the System IPO/Es is contained in the System IPO/E General Information Manual (GC20- 1873) which will be available in March 1979 from EPC, Copenhagen. In addition, System IPO/E Planning Guides will be available prior to first customer ship.

UPDATES

The System IPO/Es will be updated at EPL with service and new releases of its component products on a periodic basis. System IPO/Es are intended to ease installation of new function. Thus, as new versions of products contained in System IPO/Es become available they may replace previous versions. Service will then continue for these products as stated in the individual announcement letters for these products.

RETAIN

RETAIN entries will be available. The account PSR will obtain RETAIN entries which should be reviewed by the account SE and PSR before installation of the System IPO/E and its associated licensed programs.

13.9 LICENSE AGREEMENT, TERMS AND CONDITIONS

AVAILABILITY

The System IPO/Es are available only to users of IBM CPUs. Questions concerning the separate availability of the licensed program features of the System IPO/Es should be directed to Country Business Practices.

TERMS AND CONDITIONS

Each System IPO/E has the following terms and conditions

- Prior to or concurrent with the ordering of a System IPO/E, each of the licensed programs in the System IPO/E base and the desired System IPO/E features must either be or have been separately ordered using the same customer numbers used to order the System IPO/E. System IPO/E orders that do not meet this requirement will be rejected by EPL. The customer must have also signed the License Agreement for IBM Program Products and be specifically licensed for the appropriate licensed programs prior to ordering a System IPO/E.
- Should a customer wish to discontinue any product that is within a System IPO/E base, normal license program discontinuance procedures apply. The user will not receive updated System IPO/Es.

13.10 BIBLIOGRAPHY

GC20-1873 System IPO/E General Information Manual
XXXX-XXXX DOS/VSE System IPO/E Planning Guide
XXXX-XXXX DOS/VSE System IPO/E Program Directory
XXXX-XXXX IPF DOS/VSE Feature Program Directory
XXXX-XXXX IPF DOS/VSE Feature User Guide

(IPF stands for Interactive Productive Facility)

DOS/VSE PLANNING GUIDE VOL. 2 - ZZ10-9826-0

Your comments Please . . .

Reply requested

Yes

No

Name _____

Job Title _____

Address _____

Thank you for your cooperation

YOUR COMMENTS PLEASE . . .

Your comments on the other side of this form will help us improve future editions of this publication.
Each reply will be carefully reviewed by the persons responsible for writing and publishing this material.

fold

fold

**IBM Laboratory/IBM Deutschland GMBH
World Trade Systems Center
Department 3222
Building 44
Schoenaicherstrasse 220**

**703 Boelingen
Germany**

fold

fold

IBM

IBM World Trade Corporation
360 Hamilton Avenue, White Plains, New York 10601
(International)

DOS/VSE PLANNING GUIDE VOLUME 2