SY26-3828-2
File No. S370-30

Systems

OS/VS2 MVS DADSM Logic

VS2 Release 3.8

Includes Selectable Units:

OS/VS2 MVS Data Management	**(VS2.03.808)**
VS2 MVS JES3 3850 MSS	**(5752-818)**
OS/VS2 MVS System Security Support	**(5752-832)**
OS/VS2 MVS Data Management Support	**(5752-860)**

IBM

Includes Selectable Units:

OS/VS2 MVS Data Management (VS2.03.808)
VS2 MVS JES3 3850 MSS (5752-818)
OS/VS2 MVS System Security Support (5752-832)
OS/VS2 MVS Data Management Support (5752-860)

Third Edition (August 1978)

This edition, as amended by technical newsletter SN26-0938, applies to Release 3.8 of OS/VS2 MVS and to any subsequent releases of that system unless otherwise indicated in new editions or technical newsletters.

Changes are continually made to the information herein; before using this publication in connection with the operation of IBM systems, consult the latest *IBM System/370 Bibliography,* GC20-0001, for the editions that are applicable and current. Significant system changes are summarized under "Summary of Amendments" following the list of illustrations. Because the technical changes in this edition are extensive and difficult to localize, they are not marked by vertical lines in the left margin; the entire edition should be carefully reviewed.

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

Forms for readers' comments are provided at the back of the publication. If the forms have been removed, comments may be addressed to IBM Corporation, P.O. Box 50020, Programming Publishing, San Jose, California U.S.A. 95150. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

PREFACE

The information in this book is intended for programming support representatives and system programmers who support the direct-access device space management (DADSM) routines.

The manual has six sections:

- "Introduction," which presents general information about the DADSM routines, the control blocks used (data set control blocks), and the relationship of the data set control blocks (DSCBs) in the volume table of contents (VTOC).
- "Method of Operation," which describes in detail how each DADSM function does its work. Diagrams are used to relate the input and output of each routine to the processing steps. The diagrams also help (1) teach the DADSM techniques to those who are not familiar with them and (2) provide a rapid means of refreshing your knowledge and the understanding of DADSM.
- "Program Organization," which presents module-level flowcharts and module descriptions.
- "Directory," which briefly describes each of the DADSM routines, tells how each is called, and provides cross-references to the module names that appear on the microfiche of the assembled source listings, the flowcharts, and method of operation diagrams.
- "Data Areas," which lists the control blocks used by the DADSM routines. This section also contains some of the work areas and tables used by DADSM. Other work areas and control blocks that are more widely used are contained in the *OS/VS2 System Programming Library: Debugging Handbook,* GC28-0708 and GC28-0709 (Volumes 1 and 2), or in *OS/VS2 Data Areas,* SYB8-0606.
- "Diagnostic Aids," which tells about some of the techniques that are used to find the sources of DADSM problems. A cross-reference to the messages issued by the scheduler component resulting from problems encountered in the DADSM Allocate routines is also included.

The book is intended to be used with listings (or microfiche) of the assembled source code.

How To Use This Book

This book is written for system programmers who support and alter the design of the VS2 direct-access device space management (DADSM) routines. It is intended to provide information at two levels:

1. General information about the DADSM routines and their relationship to the operating system.
2. Specific reference information concerning the operation of each of the routines and the control blocks they refer to and change.

The text of the "Introduction" and "Method of Operation" sections provide information at the first level for those who are not familiar with DADSM. The method of operation diagrams should be read in conjunction with the method of operation text by those readers who need information at the first level. Information about using the method of operation diagrams appears before the diagrams.

For readers who are familiar with DADSM processing, the method of operation diagrams are intended as quick reference and refresher material. The cross-references to module names and labels within these modules can be used with the "Program Organization" section to provide detailed reference material.

Terminology Notes

Non-ISAM data sets in this publication refers to data sets with sequential, partitioned, and direct (SAM, PAM, and DAM) organization and to data sets for which no access method is specified (EXCP access technique is assumed).

VSAM data space is an expression used to describe the physical space on a volume that is maintained for VSAM data sets by the catalog routines. One or more VSAM data sets may reside in a VSAM data space. The system catalog routines manage the extents allocated to a VSAM data space; the DADSM routines treat a request for allocation of a VSAM data space as if it were a request for a non-ISAM data set allocation.

Prerequisite Reading

For reference information on space allocation:

- *OS/VS Data Management Services Guide,* GC26-3783

For reference and "how-to" information on allocating, extending, and releasing space:

- *OS/VS2 JCL,* GC28-0692

For reference and "how-to" information on deleting data sets (SCRATCH macro instruction), renaming data sets (RENAME macro instruction), obtaining access to DSCBs (OBTAIN macro instruction), and protecting data sets (PROTECT macro instruction):

- *OS/VS2 System Programming Library: Data Management,* GC26-3830

For details about the System Management Facilities (SMF) records:

- *OS/VS2 System Programming Library: System Management Facilities (SMF),* GC28-0706

For reference and "how-to" information on using IEHLIST, IBCDASDI, and IEHDASDR:

- *OS/VS1 Utilities,* GC26-3901
- *OS/VS2 MVS Utilities,* GC26-3902

The O/C/EOV-DADSM service routine and the system recovery routines used by the DADSM component are described in *OS/VS2 Open/Close/EOV Logic*, SY26-3827.

For data area layouts of intercomponent control blocks (for example, the DSCBs, the DEB, UCB, and SVRB):

- *OS/VS2 System Programming Library: Debugging Handbook,* Volume 1, GC28-0708, and Volume 2, GC28-0709. (Both volumes can be ordered as GBOF-8211.)

Data area layouts of the more widely used DADSM work areas (the Allocate, Scratch, Extend, and Partial Release work areas) are included in *OS/VS2 Data Areas,* SYB8-0606, available only in microfiche.

For information concerning access method logic:

- *OS/VS2 BDAM Logic,* SY26-3831
- *OS/VS2 ISAM Logic,* SY26-3833
- *OS/VS2 SAM Logic,* SY26-3832
- *OS/VS2 VIO Logic*, SY26-3834
- *OS/VS2 Virtual Storage Access Method (VSAM) Logic,* SY26-3825

For messages returned by the system:

- *OS/VS Message Library: VS2 System Messages,* GC38-1002

CONTENTS

ILLUSTRATIONS

Figures

Diagrams

Charts

SUMMARY OF AMENDMENTS

Release 3.8

OS/VS2 MVS Data Management Support for Mass Storage System (MSS) Extensions Program Product

New Programming Support

MSS Extensions program product is supported with this programming support.

Revision

August, 1978

This revision incorporates information formerly contained in technical newsletters SN26-0819 and SN26-0862, (VS2.03.808) and System Supplement Newsletters SY26-3864 (5752-818), SY26-3869 (5752-832), and SD26-6003 (5752-860).

OS/VS2 MVS Data Management Support Selectable Unit (5752-860)

A common password/bypass function is provided in SCRATCH and RENAME. HSM will use this function while copying and restoring data sets to migration storage from public storage and vice versa.

OS/VS2 MVS System Security Support Selectable Unit (5752-832)

Resource Access Control Facility (RACF) Version 1, Release 2 Support

The Scratch Routine has been modified to check for authorization to a DASDVOL volume serial entity.

VS2 MVS JES3 3850 MSS Selectable Unit (5752-818)

New Programming Support

The Scratch and Rename routines of DADSM have been modified to provide for JES3 support of the IBM 3850 Mass Storage System.

VS2 MVS Data Management Selectable Unit (VS2.03.808)

Resource Access Control Facility (RACF) Support

The Allocate, Extend, Scratch, and Rename routines have been modified to define, redefine, and delete RACF-defined data sets. In addition, Rename and Scratch check RACF authorization to modify the definition. RACF provides support for the authorization checking of data sets defined to it.

Release 3.7

New Programming Support

The IBM 3344 Direct Access Storage and the IBM 3350 Direct Access Storage are supported in this release.

Release 3

New Programming Support

The IBM 3850 Mass Storage System (MSS) is supported with this release. The MSS virtual volumes are functionally equivalent to the 3330/3333 Disk Storage, Model 1.

Release 2

System Integrity Improvements

The Allocate, Extend, Scratch, Rename, Obtain, LSPACE, and Partial Release routines have been modified to improve system integrity:

- Allocate and Extend automatically request space reclamation to recover and correct a VTOC following system failures and I/O errors that occurred during DADSM processing.
- Scratch, Rename, Obtain, and LSPACE validate the addresses of work areas and lists of direct access volumes serial numbers passed to prevent inadvertent or malicious access to system resources. Before calling the Allocate routines, the supervisor component ensures that callers are either (1) executing in supervisor state or system storage protection key, or (2) are authorized under the Authorized Program Facility (APF).
- To prevent deleting or renaming a data set that is (1) currently open or (2) allocated to an address space other than the one requesting the deletion or renaming, the system supervisor component checks data set and task status before calling the Scratch, Rename, and Partial Release routines.

System Recovery Routines

Recovery routines have been added to ensure that the entire system control program does not fail when errors occur in the DADSM routines. These recovery routines are entered under all abnormal termination conditions encountered in the DADSM routines. Their function is to release all storage obtained and resources enqueued upon during DADSM processing. They also gather diagnostic information concerning the abnormal termination condition and make it available for inspection (1) by user STAE and ESTAE routines and (2) in diagnostic dumps of virtual storage. The recovery routines used by DADSM are documented in *OS/VS2 Open/Close/EOV Logic*.

Virtual Input/Output (VIO) Data Sets

VIO data sets can be allocated and scratched under VS2 release 2. Other DADSM services are not supported or are invalid for VIO data sets: VIO data sets cannot be renamed; LSPACE processing is not viable, since VIO data sets reside on the system paging data sets; only the *search* option of obtain processing is provided; requests for releasing unused space are ignored; and a VIO data set cannot be extended.

Suballocation and Split-Cylinder Data Set Support Eliminated

Two allocation options have been eliminated: the SUBALLOC and SPLIT DD statement parameters are no longer honored. The SUBALLOC and SPLIT values are converted by the system scheduler component to the equivalent values for a SPACE request and allocated as such. The code used to make allocations of these types has been removed from all of the DADSM routines, except for the code in the Scratch and VTOC Conversion routines.

The Scratch routine code is needed to delete existing split-cylinder data sets and the VTOC Conversion support allows conversion of VTOCs containing format-6 DSCBs.

Larger Load Modules

The DADSM modules are executed in the pageable link pack area (PLPA). To avoid paging, it is desirable to put the functions that are executed consecutively in the same load module. The DADSM routines consist of 12 load modules that comprise some 58 control sections (CSECTs), as indicated in the "Directory." A load module-to-object module cross-reference is provided.

Transfer of Control

The XCTL macro instruction, which was used in previous releases to transfer control among most of the DADSM object modules and to pass or return control to other system components, has been replaced by the IECRES macro instruction. This macro transfers control to the O/C/EOV-DADSM service routine, IFG019RA, which determines if the module containing the CSECT that is to receive control is currently loaded, and then transfers control by either loading the module containing the object module and branching to it or branching to it directly. The IECRES macro instruction is also used in the DADSM modules to get and free virtual storage and to establish the recovery routine address.

Storage Protection Keys

Storage protection has been enhanced in the DADSM routines, as in other system components, by the assumption of appropriate storage protection keys. Several DADSM routines, Partial Release, Extend, VTOC Conversion, and LSPACE, assume storage key 5 (data management key); however, user key 7 through 15 and key 0 are also assumed.

VSAM Support in DADSM

The addition of the virtual storage access method (VSAM) resulted in changes to the Extend and Obtain routines. VSAM data spaces are allocated and extended as if they were non-ISAM data sets. The catalog component allocates VSAM data sets from within these data spaces. The Obtain routine has been modified to read physical extent information for VSAM data sets from the master or user catalog. The Protect routine, SVC 98, is not used for data set protection for VSAM data sets.

RPS Support Simplified

The implementation of DADSM support for direct-access devices with the rotational position sensing (RPS) feature has been simplified for this release of OS/VS2. The RPS set-up module, IGG029R1, has been eliminated. The functions provided by IGG029R1 have been incorporated in the affected DADSM modules. Function and performance of the devices with RPS remains unchanged.

Partial Release Routine Modifications

In VS2, release 1, control was passed to the Partial Release routine from the Close and Checkpoint/Restart routines after the file mark had been written to denote the end of the data set for sequential and partitioned data sets. For this and subsequent VS2 releases, control is passed to the Partial Release routine before the file mark is written. In addition, the Partial Release routines will now write back the format-1 DSCB to the VTOC and dequeue the VTOC. This was previously done by the Close and Checkpoint/Restart routines after Partial Release returned control.

INTRODUCTION

Controlling Space on DASD Volumes

The DADSM routines control allocation of space on direct-access volumes through the volume table of contents (VTOC) of that volume. The VTOC is built when the volume is initialized by the direct-access storage device initialization (IEHDASDR or IBCDASDI) utility program. See "The Volume Table of Contents" in this section for more information about the VTOC.

The VTOC is a collection of data set control blocks (DSCBs). The different types of DSCBs are:

- Free VTOC record DSCB—format-0
- Identifier DSCB—format-1
- Index DSCB—format-2
- Extension DSCB—format-3
- VTOC DSCB—format-4
- Free space DSCB—format-5
- Shared extent DSCB—format-6

Each DSCB corresponds either to a data set or data space currently residing on the volume, or to contiguous, unassigned tracks on the volume. DSCBs are the data set labels, which contain characteristics of the data sets or data spaces and a description of the tracks on which the data sets resides. DSCBs for unassigned tracks indicate the location of unassigned, contiguous tracks.

The Allocate and Extend routines assign tracks and cylinders on direct-access volumes. The Allocate routines are used by the scheduler component to get space for new data sets. The Extend routine is called by the system catalog management and End-of-Volume components to get more space for a data set (or VSAM data space) that has already been allocated, but needs more space. Other DADSM routines (Scratch and Partial Release) are used to release space that is no longer needed on a direct-access volume.

When space is needed on a volume, the DADSM routines check the VTOC for enough contiguous, available tracks to satisfy the request. If there are not enough contiguous tracks, the request is filled using as many as five noncontiguous groups of free tracks. The appropriate DSCBs are modified to reflect the assignment of the tracks.

When space is released, the DADSM routines delete the DSCBs of the deleted data set or data space. A free space (format-5) DSCB is built, or modified if existant, to indicate that the tracks containing the affected data set or data space can be reallocated.

DADSM Routines

DADSM's space management routines are concerned with:

- Allocating primary space, which involves finding space for new data sets or for VSAM data spaces. These are the Allocate routines.
- Allocating secondary space, which involves finding additional space for data sets or VSAM data spaces that have exceeded their original, primary allocations. This is the Extend routine.
- Releasing space, which involves both deleting entire data sets or data spaces that are no longer needed, and freeing unused space in data sets that are being retained. These are the Scratch and Partial Release routines.

DADSM's VTOC-related service routines are concerned with:

- Changing the names of data sets. This is the Rename routine.
- Making control information available for examination. This is the Obtain routine.
- Determining the space available on a direct-access volume. This is the LSPACE routine.
- Maintaining the system PASSWORD data set, which controls access to data sets and their associated control information. This is the Protect routine.

Allocating and Releasing Space on Direct-Access Volumes

The DADSM routines which allocate space (Allocate and Extend), and release space (Scratch and Partial Release), add, delete, and modify records of the VTOC. These records are called data set control blocks (DSCBs). To make space available to a new data set or to increase the space allocated to a data set, the appropriate DSCBs are searched for available space; the space is allocated to the data set by writing the description of the space, called an extent, to the data set's DSCB and deleting the extent from the space available for allocation. To release space allocated to a data set, the allocate operation is reversed: the released extent is deleted from the data set's DSCB and added to the DSCB that describes available space.

Components of the operating system use the DADSM routines to allocate and release space in response to data definition (DD) statements. For example, job management (scheduler) routines call the Allocate routines to obtain space for a new data set. The End-of-Volume component of Open/Close/End-of-Volume (O/C/EOV) calls the Extend routine when an existing data set needs more space; the VS2 catalog management routines call the Extend routine to get more space for a VSAM data space; the VS2 catalog management routines call the Extend routine to get more space for a VSAM data space; and the OS catalog management routines call the Extend routine to allocate additional space for an OS catalog. Similarly, job management routines use the Scratch routine to delete data sets, and the catalog management routines use the Scratch routine to delete a data set when uncataloging involves deleting a data set of a generation data group. Utility programs (IEHPROGM, IEHMOVE, and IEBCOPY) use the Scratch and Allocate routines. Scratch processing is also available to the system programmer through the SCRATCH macro instruction.

The virtual storage access method (VSAM) allocates and releases space using the DADSM Allocate, Extend, and Scratch routines. These DADSM routines are called by the VS2 catalog management routines to allocate, extend, and delete VSAM data spaces.

The Partial Release routine is called by the Close routine of O/C/EOV to release unused space before a data set is closed. Partial Release is also called by the reposition-I/O routine of Checkpoint/Restart to release unused space.

VTOC-Related Service Routines

While Rename, Obtain, LSPACE, and Protect routines are used to read and change control information on the VTOC, none allocates or releases space. System macro instructions can be used to invoke the Rename, Obtain, and Protect routines (reference and "how-to" information for these macro instructions is provided in *OS/VS2 System Programming Library: Data Management*).

The Rename routine finds the DSCB for a specified data set and changes its name, after verifying that the requested name does not duplicate one already on the volume.

The Obtain routine finds the DSCB for a specified data set, then reads the DSCB into virtual storage. The Obtain routine is also used to get information about VSAM data sets from the VTOC, the VS2 master catalog, or a VSAM user catalog.

The LSPACE routine is called either (1) by routines issuing demount messages for direct-access volumes (for example, scheduler and O/C/EOV) when the operator has issued a "MONITOR SPACE" command or (2) by the System Management Facilities (SMF). The available space on the volume is calculated by searching and totaling the extents contained in the free space (format-5) DSCBs. At the same time, the largest available extent on the volume is located. If SMF information is required, an SMF type-19 record is gathered and written to the SMF data set.

The Protect routine adds, replaces, deletes, or lists entries in the PASSWORD data set. When the security protection status of a data set changes, the Protect routine also modifies the protection mode indicator field in the protected data set's DSCB.

The Volume Table of Contents

The volume table of contents (VTOC) is a data set consisting of 140-byte data set control blocks (DSCBs) that describe the contents of a direct-access storage device volume. The VTOC data set resides in a single extent (that is, it is a continuous data set); its address is located in the VOLVTOC field of the standard volume label (see Figure 1). There are seven different kinds of DSCBs. Each has a different purpose and is, consequently, given a different name and format number. Figure 2 lists each DSCB and its use.

The first record in every VTOC is the VTOC (format-4) DSCB that describes (1) the device that the volume resides on, (2) the attributes of the volume itself, and (3) the size and contents of the VTOC data set itself.

The format-4 DSCB is followed by a free-space (format-5) DSCB, which lists the extents on the volume that have not been allocated to a data set or VSAM data space. Each format-5 DSCB contains 26 extents. If there are more than 26 available extents on the volume, another format-5 DSCB will be built for every 26 extents. The format-5 DSCBs are chained using the last field of each format-5 DSCB. The third and subsequent DSCBs in the VTOC do not

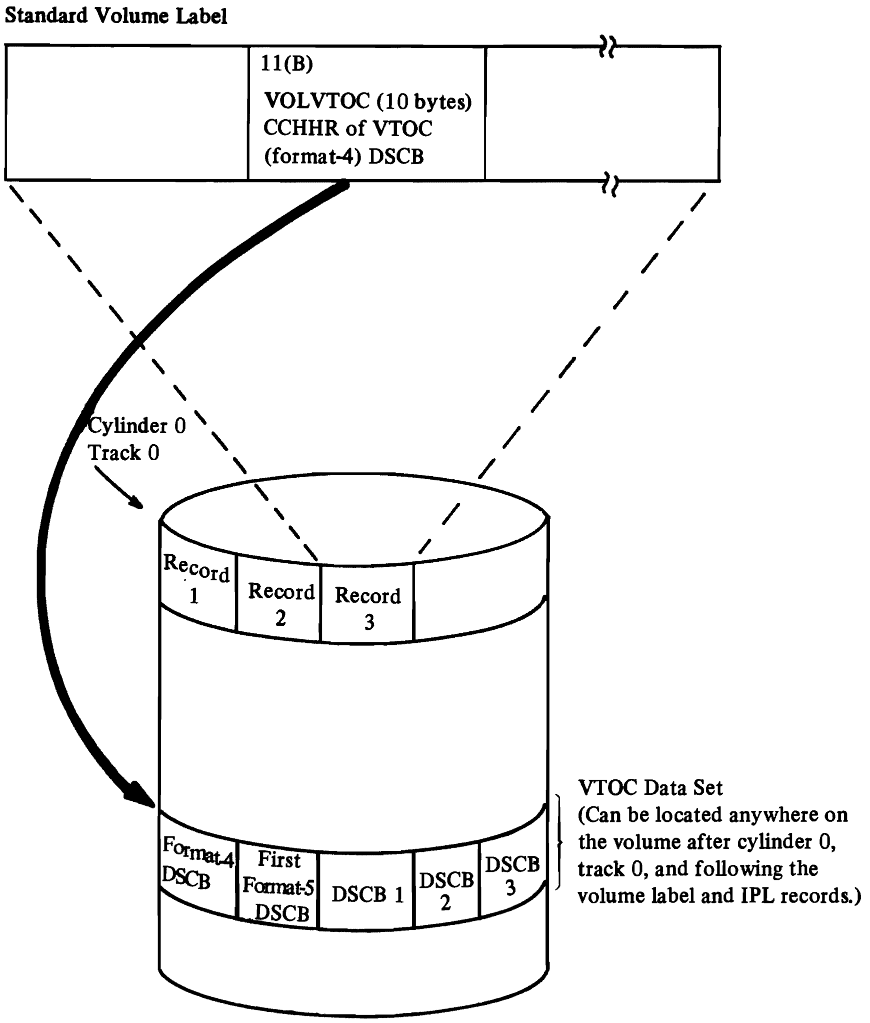


Figure 1. The volume table of contents (VTOC) and how it's located

necessarily occupy continuous space, nor do they have any prescribed sequence.

A data set or VSAM data space is defined by one, two, or three DSCBs in the VTOC of each volume on which it resides. The number of DSCBs needed to define a data set or VSAM data space is determined by (1) the organization of the data set (ISAM data sets need a format-2 DSCB to describe the index) and (2) the number of extents the data set or VSAM data space occupies (a format-3 DSCB is needed to describe the fourth through the sixteenth extents). Figure 3 shows the general makeup of a VTOC and the DSCBs needed to define two types of data sets (ISAM and non-ISAM).

Data set A (in Figure 3) is an ISAM data set; three DSCBs, a format-1, format-2, and format-3, are required. Data sets B, C, and D could be sequential, partitioned, or direct data sets or VSAM data spaces. Data set B has more than three extents and therefore requires both a format-1 and a format-3 DSCB.

Data sets C and D have three or fewer extents and need only a format-1 DSCB. The format-6 DSCB, pointed to by the format-4 DSCB, is used to keep track of the extents allocated in order to be shared by two or more data sets (split-cylinder data sets). For example, if data sets C and D share an

DSCB Name	DSCB Format Number	Function	How Many	How Found
Identifier	1	Describes a data set or VSAM data space and the first three extents.	One for every data set or VSAM data space on the volume, except the VTOC.	Search on key equal to the data set name.
Index	2	Describes the indexes of an ISAM data set.	See “ISAM Data Set Allocation.”	Chained from a format-1 DSCB that represents the data set.
Extension	3	Describes the 4th through 16th extents of a data set or VSAM data space. (Data sets and VSAM data spaces are restricted to 16 extents per volume.)	One for each data set or VSAM data space on the volume that has more than three extents.	Chained from a format-2 or a format-1 DSCB that represents the data set or VSAM data space.
VTOC	4	Describes the extent and contents of the VTOC and volume and device characteristics.	One on each volume.	VOLVTOC field of the standard volume label contains its address. It is always the first record in the VTOC.
Free Space	5	Describes the space on a volume that has not been allocated to a data set or to a VSAM data space (available space).	One for every 26 non-contiguous extents of available space on the volume.	The first format-5 DSCB on the volume is always the second record of the VTOC. If there is more than one format-5 DSCB, it will be chained from the first format-5 DSCB via the DS5PTRDS field of each format-5 DSCB.
Shared Extent	6	Describes the extents shared by two or more data sets (split-cylinder extents).	One for every 26 split-cylinder extents on the VTOC.	The address of the first format-6 DSCB is contained in the DS4F6PTR field of the format-4 DSCB. If there is more than one format-6 DSCB on the volume it will be chained to the first via the DS6PTRDS field of the format-6 DSCB.
Free VTOC Record	0	The unused records in the VTOC, which contain 140 bytes of binary zeros. To delete a DSCB from the VTOC, a format-0 DSCB is written over it.	One for every unused 140-byte record on the VTOC. The DS4DSREC field of the format-4 DSCB is a count of the number of format-0 DSCBs on the VTOC.	Search on key equal to X‘00’ (sometimes X‘00000000’).

Figure 2. Data set control block (DSCB) format types and their use

extent made up of one or more cylinders, this extent would be described in the format-6 DSCB. Note that split-cylinder data sets can no longer be allocated on VS2 systems, but existing split-cylinder data sets can still be processed.

To prepare a volume for use (to initialize it), the IBCDASDI or IEHDASDR utility is used. One of the things these utilities do is build the VTOC. After initialization, this VTOC will contain a format-4 DSCB and a format-5 DSCB. The format-5 DSCB contains an extent entry for all the free space on the volume; the initial number of extents in the format-5 DSCB is one or two, depending on where the VTOC is located on the volume. If the VTOC is located somewhere other than at the beginning or end of the volume, two extent entries are needed to describe the free space that precedes and follows it.

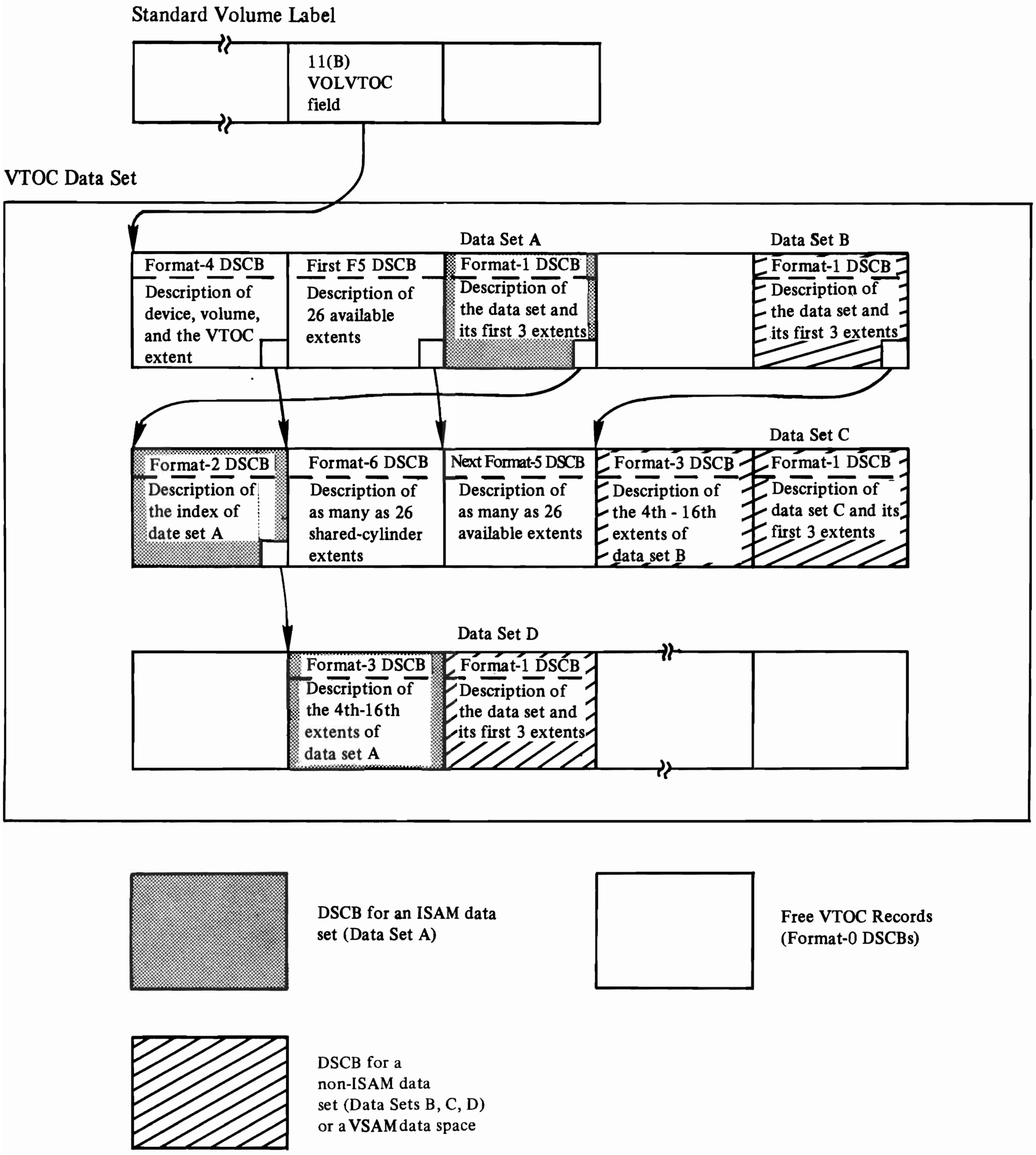

Standard Volume Label
11(B)
VOLVTOC
field
VTOC Data Set
Format-4 DSCB
Description of device, volume, and the VTOC extent
First F5 DSCB
Description of 26 available extents
Data Set A
Format-1 DSCB
Description of the data set and its first 3 extents
Data Set B
Format-1 DSCB
Description of the data set and its first 3 extents
Format-2 DSCB
Description of the index of date set A
Format-6 DSCB
Description of as many as 26 shared-cylinder extents
Next Format-5 DSCB
Description of as many as 26 available extents
Format-3 DSCB
Description of the 4th - 16th extents of data set B
Data Set C
Format-1 DSCB
Description of data set C and its first 3 extents
Format-3 DSCB
Description of the 4th-16th extents of data set A
Data Set D
Format-1 DSCB
Description of the data set and its first 3 extents
DSCB for an ISAM data set (Data Set A)
Free VTOC Records (Format-0 DSCBs)
DSCB for a non-ISAM data set (Data Sets B, C, D) or a VSAM data space

Figure 3. The contents of the VTOC—DSCBs describing data sets

Size of the Volume Table of Contents

The number of DSCBs in the VTOC determines the number of data sets or VSAM data spaces that can reside on a volume and is therefore essential information for the DADSM routines that allocate and release space.

The types of direct-access storage devices supported by this operating system and the number of DSCBs that will fit on a single track of each type, are:

Direct-Access Device Type	Number of DSCBs per Track
IBM 2305-1 Fixed Head Storage	18/track
IBM 2305-2 Fixed Head Storage	34/track
IBM 2314 Direct-Access Storage Facility	25/track
IBM 2319 Disk Storage	25/track
IBM 3330 Disk Storage, models 1 and 11	39/track
IBM 3333 Disk Storage and Control, models 1 and 11	39/track
IBM 3340 Direct Access Storage Facility	27/track
IBM 3344 Direct Access Storage	27/track
IBM 3350 Direct Access Storage	47/track

The DS4DSREC field of the format-4 DSCB contains a count of the number of free VTOC records (format-0 DSCBs) in the VTOC. This count is checked before each allocation. There must be enough free VTOC records for all the DSCBs required to define the data set or VSAM data space, as well as an extent or a combination of extents large enough to contain the data set or VSAM data space. The number of DSCBs needed to define a single data set or VSAM data space can be one, two, or three, depending on (1) whether it is an ISAM data set (a format-2 may be required) and (2) whether the data set has more than three extents (a format-3 DSCB is needed to list the fourth through the sixteenth extent). In addition, the DADSM Allocate routines make sure there is room for an additional format-5 in case it is necessary to create one during the allocation.

Volume Table of Contents Integrity

In an operating system with only one CPU, two or more tasks may require access to the same VTOC simultaneously for the purpose of reading or updating (that is, adding, deleting, or modifying DSCBs) that VTOC. If more than one CPU has access to the same device or devices, it becomes necessary to protect VTOCs from being accessed while the DADSM routines are in process.

To be sure that a VTOC is not changed while the DADSM routines are in process, the DADSM routines issue RESERVE, ENQ, and DEQ macro instructions. These macro instructions provide exclusive control of the VTOC for the task issuing the macro instruction. The RESERVE macro instruction is needed for systems in which two or more CPUs are processing concurrently, using the same data sets. These macro instructions provide exclusive control of the VTOC for the task issuing the macro instruction. Depending on the macro instruction, the "set-must-complete" option or the "release-must-complete" option may be specified in an operand of the macro instruction. The Allocate, Extend, Scratch, Rename, Partial Release, LSPACE, and Protect routines of DADSM issue these macro instructions. Of these routines, only Allocate, Scratch, and Partial Release use SMC=STEP in the ENQ and RESERVE macros, and RMC=STEP in the DEQ macro. The Extend routine links to the status routine (rather than issuing the ENQ macro) to obtain "step-must-complete" status, if the task that called Extend has not already done so.

The VS2 catalog management routines modify the DS4AMCAT and DS4AMTIM fields of the VTOC (format-4) DSCB. These routines also issue the RESERVE, ENQ, and DEQ macro instructions to maintain exclusive control while making modifications.

Note: When operating in an environment in which direct-access storage devices are not shared among systems, the RESERVE macro instruction defaults to (acts as) an ENQ macro instruction.

DADSM Interrupt Recording Facility (DIRF)

If a system fails or a permanent I/O error occurs during allocation of space or during a routine that updates the VTOC, the VTOC will probably be in error. To make sure the error is recorded, the DADSM routines use the DADSM interrupt-recording facility (DIRF). DIRF processing involves turning on a bit in the VTOC at entry to the DADSM function, and if no I/O errors occur during DADSM processing, turning it off again at exit from that function.

This bit is called the *DIRF* bit and is bit 5 of the DS4VTOCI field of the format-4 DSCB. The Scratch and Partial Release routines also turn on the DIRF bit if they encounter overlapping extents in a format-5 DSCB.

The next time an attempt is made to allocate space on a volume that has the DIRF bit set, the VTOC Conversion routine is invoked by Allocate or Extend, whichever is attempting to allocate more space on the volume. The VTOC Conversion routine builds new format-5 DSCBs to represent the available space on the volume, updates the format-4 DSCB, and returns to Allocate or Extend to continue the allocation. The "Diagnostic Aids" section tells how to deactivate the VTOC conversion by altering the DADSM routines.

DADSM Recovery Routines

To avert a complete system failure when an abnormal termination condition arises in DADSM processing, the DADSM routines include a recovery function. This function is given control by the recovery/termination manager (R/TM) when an abnormal termination condition is encountered in a DADSM routine. The same recovery routine is given control when an Open, Close, or EOV routine encounters an abnormal termination condition; the DADSM recovery function is therefore documented in *OS/VS2 Open/Close/EOV Logic*.

During the initialization of each DADSM function, the address of the recovery routine is established when virtual storage is obtained by the DADSM function. The O/C/EOV-DADSM recovery routine load module name is IFG0RR0A. Its address is established for a DADSM routine when the IECRES GET macro is issued for a work area. The functions of the IECRES macro, the expansion of which results in a branch to the O/C/EOV-DADSM service routine, IFG019RA, are documented in *OS/VS2 Open/Close/EOV Logic*.

The DADSM recovery routine saves diagnostic information in the SYS1.LOGREC, SYS1.DUMP, and GTF data sets, and then releases any resources obtained by the DADSM routine before it failed.

METHOD OF OPERATION

This section describes the functions of the DADSM routines, first in textual descriptions, followed by method of operation diagrams. The first diagram, Diagram 1, shows the contents of the other method of operation diagrams. Preceding Diagram 1 is a description of how to read method of operation diagrams.

The Allocate Routines

The Allocate routines find unused space on direct-access storage volumes and make that space available for new data sets. These routines create a format-1 DSCB (data set control block) for each new data set, and in the format-1 DSCB, write descriptions of the direct-access storage areas (these descriptions are called extents). Available extents are obtained by searching the format-5 DSCBs on the VTOC. Three routines are employed for making space allocations. Two of the sets of routines actually handle allocation of space (ISAM data set allocation routines and non-ISAM data set allocation routines); the third set (the VTOC Conversion routine) converts the VTOC of volumes on which allocations have been made under the Disk Operating System (DOS) to standard format before making an allocation. The VTOC Conversion routine is also called by the Allocate and Extend routines to reclaim VTOCs that have been invalidated by permanent I/O errors and system failures during DADSM processing. The non-ISAM Allocate routine handles the allocation of VSAM data spaces the same as allocation of non-ISAM data sets. All three sets of routines share the initialization procedures.

These initialization procedures include establishing work areas, building control blocks, and initializing channel programs to read from and write to the VTOC, enqueuing the VTOC using a RESERVE macro instruction, and searching the VTOC for a data set of the same name as the data set requesting allocation. DADSM Interrupt Recording Facility (DIRF) processing is also performed as a part of the initialization procedures. These functions are diagramed only once for all three sets of Allocate routines.

When DIRF processing is completed, it must next be determined whether (1) the data set being allocated is on a volume on which the last allocation was made by the Disk Operation System (DOS) or (2) the VTOC was invalidated during previous DADSM processing and must be reclaimed. In either case, the VTOC Conversion routine is given control to convert or reclaim the VTOC before the allocation is made. Control is passed to the VTOC Conversion routine and returned to the Allocate routine if the VTOC is converted or reclaimed without encountering any errors. Upon return, the channel program is reinitiated to search the VTOC for a data set of the same name as the data set requesting allocation.

The Allocate routines receive control via an SVC 32. As input, they receive (1) a job file control block or a partial format-1 DSCB and (2) the address of the unit control block (UCB) for the device on which the data set is to be allocated. With requests for ISAM data set allocations a pointer to a list of the addresses of data set association blocks (DSABs) is also passed.

Non-ISAM Data Set (and VSAM Data Space) Allocation

When initialization procedures have been completed and it has been verified that the non-ISAM data set or VSAM data space allocation request is not for a data set name already on the volume, this routine converts the space requested for data sets directories for partitioned data sets, and VSAM data spaces specified in cylinders and average records (block length) to the equivalent number of tracks. Note that when a VIO data set is being allocated, a format-1 DSCB is created in the scheduler work area. No additions to or modifications of a VTOC are necessary; therefore, none of the processing concerned with a real direct-access volume and its VTOC are necessary.

Finding Free Space on the Volume

The non-ISAM Allocate routine reads in the first free space (format-5) DSCB on the VTOC of the specified volume, converts its extent to the RTA-plus-the-number-of-tracks format and searches the extents for an extent with the number of tracks the new data set or VSAM data space requires. If the space is not found in the first format-5 DSCB, the search is continued through the balance of the format-5 DSCBs on the volume until it is found or until the format-5 DSCBs are exhausted. If the volume does not contain enough available space for the data set or VSAM data space, control is returned to the calling routine, which may then specify another volume to be used for the data set or VSAM data space, unless a specific volume serial number has been specified by the user.

When the space is found, the non-ISAM Allocate routine builds a table (see the extent table in the "Data Areas" section) of the format-5 extents used to fill the request. The extent table can have as many as five entries. Several different methods of building the extent table are used depending on the options specified:

- *User Labels*—If user labels are specified, the user-label track is allocated from the smallest extent in the first format-5 DSCB. When cylinder requests specify user labels, the user label extent will be taken from tracks that cannot be used to fill cylinder requests. The user label extent is the first entry in the extent table; when the extent table entry has been made, the format-5 DSCB in the allocate work area is updated so that the track assigned for user labels is no longer available for allocation.
- *Zero Quantity Requests*—For zero-quantity requests (requests that specify no space to be allocated), a format-1 DSCB is built, without extent entries, and written to the VTOC. If user labels are specified in a zero-quantity request, the format-1 DSCB will contain one extent, the user-label extent. Extent entries can be added later to the format-1 DSCB via secondary allocation (see the Extend routine).
- *Absolute Track Requests*—For absolute-track allocation requests (requests indicating that the data set is to start at a specific track), the extent containing the specified track is located in the format-5 DSCB. If this extent contains enough space following the specific track to fill the request it is entered in the extent table. Absolute-track allocation is the Disk Operating System (DOS) allocation method. It is supported in this operating system to ensure compatibility.
- *Cylinder and Track Allocation*—For normal, cylinder or track allocation requests, the request type (MXIG, CONTIG, ALX, or no-option-specified,

as defined in Figure 4) is determined, and the extents required to fill the request are entered in the extent table.

Option	Allocation Method
MXIG	Allocate all the storage space in the single largest area on the volume, if that area is equal to or larger than the request.
CONTIG	Allocate the exact amount of space requested from one area on the volume that is equal to or larger than the request.
ALX	Build a list of as many as five areas of available storage that are the largest on the volume. If a user label has been requested, a list of as many as four areas is built. Allocate all the space in as many of the areas found that are equal to or larger than the request.
No Option Specified (blank)	Build a list of as many as five areas that are smaller than but closest in size to the request. If a user label has been requested, a list of as many as four areas is built. If, while building the list, an area is found equal to the request, that area is allocated and the list is disregarded. If no areas are found equal to the request but areas are found larger than the request, the amount of space requested is allocated from the area that is larger than but closest in size to the request, and the list is disregarded. If all available areas are smaller than the request, and if enough space is available in the areas in the list, the exact amount of the request is allocated from among the largest areas found.

Note: For an ISAM data set only the CONTIG and the blank options are valid.

Figure 4. DASD space request options and their associated initial allocation methods

- *VIO Data Set Allocations*—If a VIO data set is being allocated, the space to be used for the data set will be in a system paging data set. A single extent (two extents if a user-label track is requested) will be allocated using a virtual DSCB, which represents space on the system paging data sets. The extent allocated will be equal to the primary extent plus 15 times the secondary extent quantity specified in the SPACE parameter of the DD statement.

Building Format-1 and Format-3 DSCBs

When enough available space has been found on a volume and the extent table created, identifier (format-1) and extension (format-3) DSCBs are built to describe the newly allocated data set or VSAM data space. Format-3 DSCBs are required only when more than three extents are needed to fill the space request. For zero-quantity requests, only a format-1 DSCB is created; no extent entries are included, unless user labels were specified, making a user-label extent necessary.

If the auto data set protection bit is set in the JFCB (JFCBADSP), then the RACF indicator is set in the format-1 DSCB and a RACDEF macro instruction is issued to define the data set to RACF.

Updating the Free Space (Format-5) and VTOC (Format-4) DSCBs

To complete the allocation of a non-ISAM data set or VSAM data space, it is necessary to update the format-5 and format-4 DSCBs to define the new status of the volume.

Updating Format-5 DSCBs: If an entire continuous area has been allocated to the data set or VSAM data space, the format-5 extent entry for that area is completely removed by moving forward by one location all subsequent format-5 extent entries in that DSCB and in other format-5 DSCBs that may be chained to it. If only part of an available area is allocated, the space is taken from the beginning of the area, and the track following the allocated area becomes the new lower boundary of the available area.

If the space to be allocated begins at a specified track address or on a cylinder boundary, the allocation may be made from within an available extent. In that case the allocation must create a new extent in the format-5, and the spaces preceding and following the allocated space will become two areas identified in two format-5 extents. This causes all subsequent format-5 extents in that and any chained format-5 DSCBs to be moved back one location.

Updating the VTOC (Format-4) DSCB: Two fields of the format-4 DSCB may be updated with each allocation. If a DSCB is added or removed (that is, if a free VTOC record (format-0 DSCB) is written over a format-5), the count of the number of format-0 DSCBs, which is in the DS4DSREC field of the format-4 DSCB, must be updated. This count is referred to as the *hole count.* The address of the last format-1 DSCB on the VTOC (the DS4HPCHR field of the format-4 DSCB) will have to be changed every time a new format-1 is added to the end of the VTOC. This address is referred to as the *high-water mark* and is used to terminate channel programs that search the VTOC. For VSAM data spaces, the VSAM catalog management updates the DS4AMTIM and DS4AMCAT fields of the format-4 DSCB.

Allocating a VIO Data Set

If a VIO data set is being allocated, virtual storage is obtained in the scheduler work area (SWA) for a virtual DSCB. The size of the extent to be allocated is calculated as the primary plus 15 times the secondary space requested and its beginning and ending address is entered in the virtual DSCB. A second extent, consisting of a single contiguous track will be allocated if the data set has user labels.

Formatting Directories of Partitioned Data Sets

If a partitioned data set is being allocated, the directory for the data set is formatted in the specified number of 256-byte blocks, each of which can contain from 3 to 21 data set member entries. For requests without directories, register 15 is loaded with a zero (successful) completion code and control is returned to the calling routine (that is, the scheduler or the utility program).

Handling Error Conditions

When error conditions are encountered in the non-ISAM Allocate routine, they load register 8 with the error code and give control to the error-handling routine, which dequeues from the VTOC, frees work area, loads the error code in register 15 and returns to the calling routine.

ISAM Data Set Allocation

The ISAM data set allocation routine shares the initialization procedures with the non-ISAM data set allocation routine. For an ISAM data set, the procedures include establishing and initializing work areas, building control blocks, reading the format-4 DSCB and the first free space (format-5) DSCB into the work area, and searching the VTOC for a format-1 DSCB with the same name as the data set requesting allocation.

Request Validation

The ISAM allocation routine receives control from the allocate initialization procedures. When control is received the following checks are made:

1. If a duplicate-name DSCB is indicated, the validity of the request must be verified. The duplicate-name request is valid (1) if it is the second or third of a sequence of two or three concatenated DD (data definition) statements that make up the ISAM allocation request and (2) if the same UCB is being used.
2. If a multiple DD statement request is being processed (that is, two or three area types—prime, index, or overflow), the first request must indicate a DD name and the second and subsequent requests must be concatenated with the first (that is, must not have DD names). This information is checked in the DD statement entries in the TIOT.
3. The space request must be in an integral number of cylinders or a number of tracks equivalent to an integral number of cylinders. (All space requests for a given data set must be specified in the same way, either as cylinders or absolute tracks.)
4. Cylinder allocations must be specified as either SPACE = CONTIG or unspecified (no SPACE designation) on the DD statement.
5. Only one of each type of area (index, prime, overflow) is allowed for a data set; they must appear in that order in the input stream.

If any of these conditions is not met, the error exit is taken (see "Error Handling").

If this is the first DD statement in the allocation request, the request validation routines set up the TIOT entry for this and subsequent DD statement entries.

Free Space (Format-5) DSCB Search

Before searching the format-5 DSCBs on the VTOC, the free-space-search routine makes sure there is enough room on the VTOC to write four new DSCBs (format-1, format-2, format-3, and format-5). It is possible that one of each of these DSCBs can be created during a single allocation.

The format-5 DSCB search routine searches all the format-5 DSCBs on a given volume to find enough space to satisfy the request.

The search involves comparing each format-5 DSCB extent in sequence with the requested quantity, which can be specified either as (1) an absolute track on a cylinder boundary and a number of tracks that ends on a cylinder boundary or (2) merely a number of contiguous or non-contiguous cylinders.

For absolute track requests, each extent of the format-5 DSCB is examined to see if the absolute track is in the extent and whether the extent is big enough to fill the request.

Searches for contiguous cylinders when looking for an extent, in the format-5 DSCB, equal in size to the request. Until an extent of equal size is found, the extent that is larger, but closest in size to the requested space, is saved. As each subsequent larger extent is encountered, its size is compared to the size of the extent that has been saved. The result of the comparison is always to save the smallest extent (of all these larger extents) that will satisfy the request.

If, during the search of the format-5 blocks, an extent exactly equal in size to the space requested is found, that extent is allocated as the requested space and the search ends. Otherwise, the search terminates after all format-5 DSCBs have been searched. If an extent larger than the request has not been found, an error exit is made. If a larger extent has been saved, the requested space is allocated from this extent. This method improves the allocation process in that space is taken from the smallest available extent that will fulfill the request, leaving larger extents intact to fill other requests.

Requests specifying that the allocated cylinders need not be contiguous are handled in the following manner. As the search for space progresses, a five-entry pushdown list is built and maintained. This list contains the largest extents encountered in the search that are not large enough to contain the total request. If an extent larger than the requested space is encountered, the pushdown list is abandoned and the larger extent is saved. The search is then continued as described for the contiguous cylinder requests.

The space requested is allocated either from the extent saved during the search, as in the case of a contiguous cylinder request, or from entries in the pushdown list. If the pushdown list is used, from two to five of the largest entries in the list are used as needed.

If the request cannot be fulfilled by using any of the preceding methods, an error exit is taken.

When enough space has been found to satisfy the request, an extent table is built containing at least one and no more than five extents in the form RTA1/RTA2. The extents are entered in ascending relative-track-address order.

Building Format-1, Format-2, and Format-3 DSCBs

Three different methods are used to build and write the format-1, format-2, and format-3 DSCBs for ISAM data sets, depending on (1) whether format-1 and format-2 DSCBs have been created during a previous pass on this volume and (2) whether an embedded index has been requested.

Creating Format-1 DSCBs: If there are no format-1 DSCBs on the volume for this data set, this routine builds a new format-1 DSCB using the extent table and the JFCB. If an embedded index has been requested, control is passed to the embedded index processing routine, described below, before the format-1 extents have been added. This format-1 build routine also builds a format-3 DSCB if more than three extents are required to satisfy the space request.

Figure 5 shows the conditions under which a format-2 DSCB is built.

If the auto data set protection bit is set in the JFCB, a RACDEF TYPE=DEFINE macro instruction is issued for the first volume and a RACDEF TYPE=ADDVOL is issued for subsequent volumes, to define the data set to RACF.

Format-1 DSCB Updating: When a format-1 DSCB for the data set being allocated already exists on the current volume, control is immediately passed to the format-1 DSCB updating routine. There is a format-1 DSCB in virtual storage when this routine receives control; the format-1 DSCB was read in by the Allocate initialization procedure or by the multivolume processing routine. Using the extent table as input, this routine adds extents to the identifier (format-1) DSCB or the extension (format-3) DSCB if there are extent locations left in these blocks. One format-3 DSCB is built if more than three extent locations are needed. The RTA1/RTA2 entries in the extent table are first converted to CCHH/CCHH format using the resident conversion routine.

If an extension DSCB (format-3) is added and there is already an index DSCB (format-2) on the VTOC, a pointer to the new format-3 must be added to the format-2. The last DSCB to be written to the VTOC is the format-1. It is delayed because a pointer to a new format-2 and/or format-3 may have to be added, and because the field in the format-1 DSCB that is a count of the number of extents in the data set must be updated.

Embedded Index Processing: Except for the extents, the format-1 DSCB is created by the standard request processing routine, before this routine is entered. The extent table has at least one entry.

This routine allocates space for the prime area and the index from the space available as indicated in the extent table. The space requested for the prime area is divided into two parts that are as nearly equal as cylinder boundary requirements permit. The entries in the extent table are then taken in order and assigned as prime extents as necessary to provide allocation for the first half of the prime area. The requested index area is then assigned from the next available track indicated in the extent table. Finally, the remaining space in the extent table is assigned to the other half of the prime area request.

The Complete Request for the Data Set (Given in terms of DD statement parameters)	**Current Request (Given in terms of DD statement parameters)**	**Format-2 DSCB Needs to be Built**	**Comments**
INDEX, PRIME, OVFLOW	INDEX	Yes	
	PRIME or OVFLOW (on different volume from INDEX)	No	A format-2 DSCB was built during the previous (index) pass. Both the index area and the format-2 DSCB are on a different volume from the one currently being searched. If the same volume had been specified for both the INDEX request and the current request, a valid duplicate-name format-1 DSCB would have been encountered.
INDEX, PRIME	INDEX	Yes	
	PRIME (on different volume from INDEX)	No	A format-2 DSCB was built during the previous (index) pass. If the same volume had been specified, a valid duplicate-name format-1 DSCB would have been encountered.
PRIME, OVFLOW	PRIME	Yes	For this type of request, an open executor module of the indexed sequential access method (ISAM) will construct an index in the overflow area.
	OVFLOW	No	A format-2 DSCB was built during previous (prime) pass. As previously described, a valid duplicate-name format-1 DSCB would have been encountered if the same volume had been specified.
PRIME	PRIME	Yes	An open executor module of the indexed sequential access method constructs the index area.

Note: If the DD statement associated with a given ISAM data set has its "type-of-space" parameter left blank, a request for PRIME is implied.

Figure 5. Conditions under which format-2 DSCBs are built

Depending on the requirements of the request and the sizes of the extent entries, it is possible to form as many as seven DSCB extents (two for the index and five for the prime area) from the entries in the extent table.

After the extents are created for the prime and the index areas, this routine adds as many as three extents to the incomplete format-1 DSCB. If more than three extents have been created, a format-3 DSCB is built to contain all those beyond the first three. As extents are placed in the format-1 (and possibly the format-3) DSCB, the index extents are entered first.

The extent table entries are converted from a relative track address (RTA1/RTA2) form to a cylinder and head (CCHH/CCHH) form as they are entered in the DSCBs. A resident conversion routine is given control to do the conversion.

Because an embedded index is always a first DD statement, it is necessary to construct an index DSCB (format-2). If a format-3 DSCB has been created, it is written to the VTOC. Then the format-2 DSCB is constructed, the address of the format-3 DSCB is placed in a pointer field, and the format-2 DSCB is then written to the VTOC. If a format-3 DSCB was not constructed, a format-2 DSCB is constructed and written out after the last extent entry is placed in the format-1 DSCB.

Once the format-2 DSCB is written, its address is placed in a pointer field of the format-1 DSCB, and the format-1 DSCB is written.

Updating the Free Space (Format-5) and VTOC (Format-4) DSCBs

When the format-1, format-2, and format-3 DSCBs have been written to the VTOC, the format-5 DSCBs must be updated to account for the space allocated and no longer available. The Allocate work area will still contain as many as five extent table entries. These are the extents that must be removed from the format-5 DSCBs, as they now represent extents allocated to the data set. Format-5 DSCBs are created and updated in the way previously described for non-ISAM data set allocation.

The VTOC (format-4) DSCB must be modified to indicate changes in the number of free VTOC records (format-0 DSCBs) in the VTOC. These changes result from adding format-1, format-2, and format-3 DSCBs and adding or deleting format-5 DSCBs. Allocation for the current DD statement entry in the TIOT is now complete. The ISAM allocate routine must determine whether (1) this is a multivolume request and (2) whether this is the last of a series of space-type requests (that is, prime, index, overflow).

Multivolume requests are handled as described in "Multivolume Processing." If the current request is the last of the DD statement entries for this allocation, the virtual storage work area is released, the TIOT fields are restored to their "pre-ISAM allocate" condition, and control is given to the job scheduler. If the current request is not the last of the DD statement entries, indicators are set to provide the necessary information for the next entry to be processed, virtual-storage areas are released, and control is returned to the job scheduler. Processing of the last volume of a multivolume prime request is the same as described for a single-volume request.

Multivolume Processing

This routine is entered to set up a loop in the ISAM allocate routine when an ISAM data set is to be allocated on more than one volume.

The VTOC of the most recently processed volume is dequeued before control is passed to this routine from the format-4 DSCB updating routine. It is necessary, in order to maintain information in virtual storage, to perform several of the functions without returning control to the scheduler or the allocate initialization routine. These, then, are the same operations performed in the allocate initialization routine: enqueuing the VTOC of the new volume using a RESERVE macro instruction, reading the format-4 and first format-5 DSCB into the allocate work area in virtual storage, and searching for a duplicate-name, format-1 DSCB (if found, it is read into virtual storage).

Processing of the multivolume request then continues through the same steps as for a single-volume request (that is, an extent table is built and the format-1, format-2, and format-3 are built as required, the format-5 DSCBs are built and/or modified, and the format-4 DSCB is updated).

Error Handling

When the ISAM data set allocation routine encounters errors, it sets an error code in register 5 and transfers control to the error handling routine. This routine dequeues the VTOC, frees the Allocate work area, loads the error code in register 15, and returns control to the scheduler. A cross-reference between messages issued and the error code and the modules that issue each code is in the "Diagnostic Aids" section.

VTOC Conversion

To provide a standard allocation procedure for the Disk Operating System (DOS) user and to provide the facility to reclaim VTOCs that have been invalidated as a result of permanent I/O errors and system failures during DADSM processing, the Allocate routines include a routine to convert and reclaim VTOCs. This reclamation and conversion involves rebuilding the format-5 and format-6 DSCBs and updating the format-4 DSCB.

Initial Conversion Processing

The VTOC Conversion routine receives control from the Allocate initialization routine and Extend routine when the DOS bit (bit 0 of the DS4VTOCI field) and/or the DIRF bit (bit 5 of the DS4VTOCI field) is set in the VTOC (format-4) DSCB. When control is passed, register 7 contains a pointer to the UCB defining the volume to be converted and register 13 points to the Allocate work area, which will contain the format-4 DSCB for the volume to be converted, along with an IOB, ECB, channel program, DEB/DCB, and the address of the first format-5 DSCB on the volume.

The VTOC Conversion routine first writes zeros over all shared extent (format-6) DSCBs on the volume. These DSCBs are deleted because they are invalid. If the DOS bit is on, a DOS allocation was made on the volume, and the DOS allocate routines do not keep track of (1) available space (2) shared-cylinder extents or (3) the high-water mark (the highest address of a format-1 DSCB, which appears in the DS4HPCHR field of the format-4 DSCB). If the DIRF bit is on, an error occurred that may have invalidated the format-4, -5, or -6 DSCBs.

After the format-6 DSCBs are deleted, a new format-5 DSCB is created and initialized to reflect the free space that was available when the volume was initialized, before any data sets were allocated. The highest track that can be allocated on the volume is calculated from the device characteristics table. The space for volume labels, IPL records if there are any on the volume, and the VTOC is subtracted from the new format-5 DSCB.

Next, the VTOC Conversion routine reads sequentially through the VTOC and establishes the highest address used for a format-1 DSCB (DS4HPCHR field). The nonsplit-cylinder data set extents in the format-1 and format-3 DSCBs are converted from CCHH/CCHH to RTA1/RTA2 form. Five extents are converted at a time and placed in the extent table. The converted extents are then subtracted from the free space.

Split-Cylinder Processing

If there are split-cylinder data sets on the volume, each format-1 DSCB on the VTOC is read again, and the split-cylinder extents are identified and converted from CCHH/CCHH to RTA1/RTA2 format. As each split-cylinder data set is encountered, its extent is entered in a format-6 DSCB, or, if the extent has already been entered in a format-6 DSCB, the count of number of data sets sharing the extent (the Z in the XXYYZ entry) is increased by one. A test is made to determine whether the split-cylinder data set extent in the extent table overlaps two format-6 extent entries. If there is an overlap, the two format-6 extent entries are merged into a single extent and their data set counts are added together.

When all of the format-6 DSCBs have been built, chained, and written to the VTOC, they are read in again. The format-6 extents are entered, five at a time, in the extent table, then subtracted from the free space (format-5) DSCBs. When all the extents have been subtracted, they are again read into virtual storage. The format-6 extents are converted to final form (from RTA/number-of-cylinders/number-of-data-sets format to XXYYZ) and written back to the VTOC.

Completion of Conversion Processing

When the split-cylinder processing is complete or when there are no split-cylinder data sets on the volume, the conversion processing is completed by converting the format-5 DSCBs to the final XXYYZ format. The new format-5 DSCBs are then written to the VTOC over the old format-5 DSCBs. This routine then turns off the DOS bit (bit 0 of the DS4VTOCI field) and turns on the "DOS-converted" bit (bit 4 of the DS4VTOCI field) and/or the "DIRF reclaimed" bit (bit 6 of DS4VTOCI), and writes the format-4 DSCB back to the VTOC. Control is then returned to the calling routine, Allocate initialization or Extend.

Conversion Limitations

The VTOC Conversion routines cannot build accurate format-5 and format-6 DSCBs if:

- A split-cylinder data set is located on cylinder zero.
- A split-cylinder data set is located on the same cylinder as the VTOC.
- A split-cylinder data set is located on the same cylinder as a non-split-cylinder data set.
- Two data sets have overlapping extents.

Extending Data Set and VSAM Data Space Allocations (Secondary Allocation)

The Extend routine dynamically allocates additional space to sequential data sets or VSAM data spaces that exceed the space allocated to them. Data sets with indexed sequential organization cannot be extended. Non-ISAM data sets (including the OS system catalog) can either be extended on the last volume a data set currently occupies, or they can be extended on a new volume. A VSAM data space and the VS2 master catalog can only be extended on the volume they currently occupy. The data set or VSAM data space can be extended by as many as five noncontiguous extents at a time, but no data set or VSAM data space is allowed to occupy more than 16 extents on a single volume.

The Extend routine uses the same methods as the Allocate routine when extending on a new volume: it takes space from the format-5 DSCBs, builds a list of extents to be allocated, builds a format-1 DSCB (and a format-3 DSCB if required), then updates the format-5 and format-4 DSCBs.

When extending on the last volume that the data set or VSAM data space currently occupies (also referred to as the *current* volume), the Extend processing is the same as non-ISAM allocate processing, except that the format-1 DSCB is merely modified or a format-3 DSCB may be built if there isn't one for the data set or VSAM data space on the VTOC. If there is a format-3 DSCB, it is modified.

Although new split-cylinder data sets cannot be allocated under VS2, existing split-cylinder data sets can still be extended. Split-cylinder data sets are extended differently from other data sets. The space for the extension is allocated on new cylinders, and only the one extended data set can reside on these new cylinders. The cylinders containing the extension are not regarded as part of the original split-cylinder extent, even though they may be contiguous to the original extent. No entry is made in the format-6 DSCB for split-cylinder data set extensions. (VSAM data spaces cannot reside on split cylinders.)

Input to the Extend routine includes pointers to (1) the UCB and DEB of the data set or VSAM data space to be extended and (2) the O/C/EOV work area defined by the IECDSECT mapping macro, which contains the JFCB for the data set or VSAM data space to be extended and various control blocks for reading and writing the DSCBs on the VTOC. Register 13 contains a code to identify the caller (EOV, VS2 catalog management, or OS catalog management) and to indicate whether the extension is to be made on the current volume of the data set or VSAM data space or on a new volume.

Extend Initialization

Extend processing begins by enqueuing the VTOC of the volume on which the data set or VSAM data space is to be extended. A RESERVE macro instruction is issued for the VTOC. The VTOC (format-4) DSCB is then read in, the DADSM interrupt recording facility (DIRF) indicator is set, and the format-4 DSCB may be written back to the VTOC, so that it can be determined if an interrupt that could invalidate the VTOC took place during Extend processing.

Next, it must be determined whether (1) the data set or VSAM data space being extended is on a device on which the last allocation was made by the Disk Operating System (DOS), or (2) the VTOC was invalidated during previous DADSM processing and must be reclaimed. In either case, the VTOC Conversion routine is given control to convert or reclaim the VTOC before the data set (data space) is extended. The Extend routine (1) gets a work area for saving registers and for use by the VTOC Conversion routine, (2) builds the control blocks needed by the I/O supervisor (the DEB/DCB, IOB, and ECB) in the work area, and (3) moves the format-4 DSCB into the work area. Control is passed to the VTOC Conversion routine and will be returned when the VTOC has been converted or reclaimed. If no errors are encountered during the conversion (indicated by a return code in register 8), Extend processing is continued.

The Extend initialization routine determines whether the extension is to be allocated on the current volume or on a new volume by the code passed in register 13 at entry to this routine. If the extension is to be made on a new volume, it is necessary to search the identifier (format-1) DSCBs on the VTOC for a duplicate name, since a duplicate name would be invalid. No search is made when the extension is to be made on the current volume, but it is necessary to determine the *preferred track* (the first track following the end of the data set or VSAM data space being extended).

The Extend initialization routine then converts requests in cylinders and average records (blocks) to tracks and accounts for a user label track, if user labels are requested.

Building the Table of Extents to be Allocated

The search for available space on the volume begins with the Extend routine attempting to allocate space contiguous to the extent of the data set (this is the *preferred* extent). If there is not enough contiguous space to fill the request, the routine builds a pushdown list of extents of as many as five entries representing non-contiguous areas that are smaller than, but closest in size to the requested space. If, while the pushdown list is being built, an extent equal to the request is found, that extent is allocated. If no extent in the format-5 DSCBs equals the request, the extension is allocated from the extent that is larger than and closest in size to the request. If no extent is found that is equal to or larger than the request, the extension is allocated from the pushdown list of extents. As few as two and as many as five of the extents in the pushdown list may be required to fill the request. The request is always filled using the fewest possible number of extents. The extent, or extents, required to fill the request are entered in the extent table.

Building and/or Modifying Format-1 and Format-3 DSCBs

When the extent table has been successfully completed and contains an entry or entries representing the extents to be allocated, the format-1 and/or format-3 DSCBs must be built or modified to indicate the extension.

If the extension is to be allocated on the current volume, the format-1 DSCB is read into the Extend work area. If the format-1 DSCB is full (that is, it has three extent entries), the format-3 DSCB is read into the work area, if a format-3 DSCB exists for the data set or VSAM data space. If no format-3 DSCB exists for the data set or VSAM data space, one is built.

Then, using the extent table entries, the format-1 and/or format-3 DSCBs are completed and written back to the VTOC; the format-3 is written first, then its address is entered in the format-1 DSCB, which is then written to the VTOC.

For extensions on a new volume, a format-1 DSCB must be built using the information from the JFCB. If there are more than three extent entries in the extent table, a format-3 DSCB must also be built. When completed, the format-3 DSCB is written to the VTOC, its address put in the format-1 DSCB, and the format-1 is written.

When a new format-1 DSCB is created, a RACDEF TYPE=ADDVOL macro instruction is issued to add the new volume to the data set if the previous volume was RACF-defined.

Updating the Format-5 DSCBs

The first format-5 DSCB is read into the work area and, again using the extent table entries, format-5 DSCBs may be updated, added, or deleted.

If an entire contiguous area has been allocated to the data set or VSAM data space during the Extend processing, the format-5 extent entry for that area is completely removed from the format-5 DSCB. The vacated extent in the format-5 DSCB is removed by moving forward by one location all subsequent format-5 extent entries in that DSCB and in other format-5 DSCBs that may be chained to it. If only part of an available area is allocated, the space is taken from the beginning of the area, and the track following the allocated area becomes the lower boundary of the available area.

If the space to be allocated begins on a cylinder boundary, the allocation may be made from within an available area. In that case, the allocation must create a new extent entry in the format-5 DSCB, and the space preceding and following the allocated space becomes two areas identified in two format-5 extents. This causes all subsequent format-5 extents in that and any chained format-5 DSCBs to be moved back one location.

If a free VTOC record (format-0 DSCB) must be written as a result of a format-5 DSCB being removed, the format-0 DSCB is written before the format-4 DSCB is modified.

Updating the Format-4 DSCB

Since this is the end of Extend processing, this routine tests and resets the DIRF indicator in preparation for writing the format-4 back to the VTOC. This routine also handles error conditions encountered in the previous Extend processing (such as too many extents on the volume or not enough free records left on the VTOC) by setting an error return code and returning to the caller.

If entered with no previous errors, this routine writes a format-0 DSCB if requested by the format-5 updating routine. It then updates the format-0 count and the highest extent used for a format-1 DSCB and writes the modified format-4 DSCB to the VTOC, dequeues the VTOC, and frees work areas.

This routine then returns control to OS catalog management or VS2 catalog management using the XCTL macro. Control is returned to EOV using the IECRES LOAD macro, which results in a branch to EOV.

Deleting Data Sets and VSAM Data Spaces

The Scratch routine is used to reverse the allocation process for all types of data sets. When a data definition (DD) statement specifies that a data set or VSAM data space is to be deleted, the Scratch routine is entered from the scheduler at task termination. The Scratch routine is also given control as a result of the issuance of the SCRATCH macro instruction by a problem program, a data set utility program, or the catalog management routines. The extents in the data set's or VSAM data space's identifier (format-1) DSCB and extension (format-3) DSCB are added to a format-5 DSCB and a free VTOC record (format-0 DSCB) is written over each of the data set's or VSAM data space's format-1, format-2, and format-3 DSCBs. Only ISAM data sets have index (format-2) DSCBs and only data sets or VSAM data spaces with more than three extents on a volume have format-3 DSCBs.

Extents allocated to split-cylinder data sets are not released until all of the data sets sharing a given cylinder extent are deleted.

Scratch Initialization, Volume Verification, and Volume Mounting

When the Scratch routine is entered, register 0 contains either a pointer to a unit control block or all zeros. Register 1 contains a pointer to a parameter list that (1) indicates whether the purge date is to be overridden, (2) points to the data set name of the data set or VSAM data space to be deleted, and (3) points to a list of one or more volume entries, one for each volume on which the data set resides. Requests to scratch a VSAM data space are made one volume at a time; the parameter list passed by catalog management contains only one entry in the volume list. The volume list consists of a 2-byte count of the number of volumes in the data set or VSAM data space to be scratched and a 12-byte entry for each volume of the data set or VSAM data space. The 12-byte entry is used to specify the device code and the volume serial number. Only one data set or VSAM data space can be deleted with each request.

When a UCB pointer is passed in register 0, the Scratch routine verifies that the device represented by the UCB is allocated to the task and that it is on-line and, in case a volume containing the data set or VSAM data space is not mounted, that the volume currently mounted is eligible for demounting (permanently mounted, reserved, and shared volumes cannot be demounted). When (1) no UCB pointer is passed or (2) the device represented by the UCB address passed is not eligible for demounting, the Scratch routine searches the TIOT DD entries (via the DSAB chain) for the address of the UCB with the volume serial number that matches the volume serial number passed in the volume list (by way of the Scratch parameter list). When a matching volume serial number is found, the UCB is tested (1) to verify that the device it represents is on-line and (2) to determine whether the volume it represents is eligible for demounting. This routine also enqueues on the TIOT (resource name is SYSZTIOT) and the data set name (resource name is SYSDSN) to ensure that the data set is not open.

At the conclusion of initialization, after enqueuing on the TIOT and data set name, Scratch determines if the next volume in the volume list is mounted. If the volume containing the data set or VSAM data space to be scratched is not mounted, a mounted volume can be demounted. If the volume is not an MSS virtual volume, a mount message is issued, to which the operator can respond either by replying "U" and mounting the volume, or by replying "M". If the volume is an MSS virtual volume and is on a JES-3-managed unit, the IEFSSREQ macro is issued to JES3 for volume control prior to issuing the mount request to MSS. If the volume is an MSS virtual volume, a mount request is issued to MSS via SVC 126 and no message is issued to the operator unless there is an error. If he mounts the volume, its label is read and the volume serial number is verified. SMF information about the volume being demounted will be recorded (1) for systems with the SMF feature and (2) if the "MONITOR SPACE" command has been issued by the operator. If the operator replies "M", the next volume in the volume list is processed.

Upon verification that the volume containing the data set to be scratched is mounted, or after having mounted the required volume, the Scratch routine builds the control blocks required by the I/O supervisor to read from and write to the VTOC. Next, the VTOC is enqueued using a RESERVE macro instruction with the "set-must-complete" option, and the addresses of the format-4 and the first format-5 DSCBs on the VTOC are calculated.

When a data set residing on more than one volume is to be deleted, the Scratch routine maintains control until all volumes have been processed. It is possible, however, that, because of a permanent I/O error or because a volume is mounted, the data set may not be deleted from all volumes. The success or failure (and reason for failure) of the scratch processing is communicated in the last byte of the volume list entry (the scratch status code). Success or failure is also indicated by the return code passed in register 15 when control is returned to the caller.

DSCB Deletion and Split-Cylinder Processing

When it has been verified that the volume containing the data set or VSAM data space to be deleted is mounted, the Scratch routine reads in the format-4 DSCB and the format-1 DSCB for the data set or VSAM data space. If the data set is password-protected, control is transferred to the SECLOADA module of the Open/EOV password-protection function. SECLOADA verifies the password and returns control. If the data set is RACF-indicated, a RACHECK macro instruction is issued to check the user's authority to delete the data set. If he is authorized, a RACDEF TYPE=DELETE macro instruction is issued to undefine the data set to RACF; the macro instruction is bypassed for RACF-defined data sets if the JSCBPASS is set to 1 and the SCRATCH parameter list, second byte, second bit, is set to 1. If the data set is RACF-defined, password checking is bypassed.

Next, the purge date for the data set or VSAM data space is checked, if it is not overridden. The format-4 DSCB is processed for DIRF. The extents of the data set's or VSAM data space's format-1 DSCB (and format-3 DSCBs, if there are any) are entered in the extent table (this table can contain as many as 16 extent entries); when the extents have been converted from CCHH/CCHH to RTA1/RTA2 and entered in the table, a format-0 DSCB is written over the format-1 and format-3 DSCBs.

Note: Format-2 DSCBs for ISAM data sets are also written over with format-0 DSCBs during this processing.

When the format-1, ISAM format-2, and format-3 DSCBs have been deleted, this routine reads in either the format-6 DSCB if a split-cylinder data set is being deleted, or the first format-5 DSCB if no shared extents are involved.

Whether split-cylinder data sets are involved or not, the Scratch routine now sorts the extent table into ascending relative track address sequence. However, if the data set is on split cylinders, each extent in the format-6 DSCB is compared with the extent table entries until the appropriate format-6 extent is found. When found, the number of data sets sharing the extent (the

Z of XXYYZ) is decreased by one. If the number of data sets goes to zero, the extent is removed from the format-6 and each subsequent format-6 DSCB extent must be moved up one location to close the vacated space on the VTOC. If the number of users does not go to zero, the format-6 DSCB is written back to the VTOC with only the number of users changed. But the deleted data set's extent must be removed from the extent table, because no space is made available until all data sets have been deleted from the shared extent. At the conclusion of Split cylinder processing, the first format-5 DSCB is read into virtual storage.

Updating the Free Space on the Volume (Format-5 DSCB) and the VTOC (Format-4) DSCB

The format-5 DSCBs are updated by merging the extent table entries with the existing format-5 DSCBs in sequence. If the deleted data set shares cylinders with another data set and no space is released, the merge is bypassed. As the format-5 DSCBs are read, their extents are converted from XXYYZ to RTA1/RTA2; before the updated format-5 DSCBs are written back to the VTOC, the extent entries are again converted from RTA1/RTA2 to XXYYZ format.

When the format-5 DSCBs have been updated, the number of free VTOC records (format-0 DSCBs) now on the VTOC must be updated if the number has changed. The format-4 DSCB is then written back to the VTOC and the VTOC is dequeued. If the data set is on multiple volumes, the Scratch routine can now begin to process the next volume. If processing is complete, the TIOT and data set name are dequeued, work areas are freed, the return code is loaded in register 15, and control is returned to the caller. If the system has the SMF feature and data set activity records are requested, a type-17 SMF record may be written before control is returned.

Freeing Unused Space (Partial Release)

The Partial Release routine deletes extents (area identifications) from the format-1 and/or format-3 DSCBs of a data set, beginning at a specified track or cylinder and continuing to the end of the data set's allocated extent. For example, suppose a data set was allocated three extents at the beginning of a job. During the job, a program writes records up to the second track in the third extent of the data set. If the job specified release of the unused space in the data set, the Close routine of O/C/EOV would request the Partial Release routine to release the tracks following the last used track of the third extent. If either the original request is a request in cylinders or the ROUND option was specified, the area released begins at the first cylinder boundary following the last track written. The tracks released are returned to available space for future allocation, as is done by the Scratch routine.

Note: Partial Release processing of ISAM data sets is not possible because information in overflow areas may be deleted.

The Partial Release routine is entered from the Reposition I/O module of Checkpoint/Restart and from Close processing.

When the Partial Release routine is entered, a work area pointed to by register 4 will contain (1) the format-1 DSCB and JFCB for the data set, and (2) the control blocks required to read from and write to the VTOC. (This work area is referred to as the O/C/EOV work area and defined by the IECDSECT mapping macro.) Register 2 contains a pointer to the user's DCB and register 10 contains a pointer to the UCB.

Updating Format-1 and Format-3 DSCBs

When entered, the Partial Release routine first gets a work area for reading in the format-4 (VTOC) DSCB and format-5 (free space) DSCB, enqueues the VTOC, and reads in the format-4 DSCB using the VTOC address in the unit control block (UCB). After DIRF processing, this routine determines whether the last track written to on the data set (obtained from the user's DCB) is in an extent of the format-1 DSCB or an extent of the format-3 DSCB.

If this extent is in the format-1 DSCB, its location is calculated. The address of the first track to be released is calculated; the identification of the released extent in CCHH/CCHH is converted to RTA1/RTA2 and entered in the extent table. Any subsequent extents in the format-1 DSCB are also converted and entered in the extent table, and their identifications are removed from the format-1 DSCB. Next, it is determined whether there is a format-3 DSCB on the VTOC for this data set. If so, it is read in, its extents are converted and entered in the extent table, and a free VTOC record (format-0 DSCB) is written over the format-3 DSCB on the VTOC.

If the last track to be written on the data set is contained in the format-3 DSCB, the format-3 DSCB is read in and, again, the location of the first track to be released is calculated. The extents following this track are converted to RTA1/RTA2 format and entered in the extent table. Then the extents are written over with zeros and the format-3 DSCB is written back to the VTOC.

When SMF information is required, the task's TCB contains a pointer to the timer control table (TCT), and the CVT contains a pointer to the system management control area (SMCA). If the fourth bit of the SMCAOPT is set to one, data set information is required. Using the extent table, this routine adds the number of released extents and the amount of space released into the appropriate fields of the data set entry in the TCT I/O table of SMF (the Close and EOV routines use this information when they construct a type-14 or type-15 SMF record).

This routine then writes back the format-1 DSCB whether or not any space has been released. If space is being released this routine sorts the extent table into ascending RTA sequence, and reads in the first format-5 DSCB.

Updating the Format-5 (Free Space) and Format-4 (VTOC) DSCBs

When the format-1 and format-3 DSCB updating is complete, this routine merges the extent table entries with converted format-5 extents, reconverts the merged extents from RTA1/RTA2 format to XXYYZ, and writes the format-5 DSCBs back to the VTOC.

Format-4 DSCB updating involves first processing for DIRF, then accounting for the number of new free VTOC records (format-0 DSCBs) on the VTOC and writing the format-4 DSCB back to the VTOC.

At the conclusion of Partial Release processing the VTOC is dequeued and the Partial Release work area is freed.

A completion code is returned to the caller in register 1.

The VTOC-Related Services Routines

Renaming Data Sets

Data sets on direct-access volumes are renamed by using the RENAME macro instruction (SVC 30), which invokes the Rename routine. VSAM can use the Rename routine if a data space is established for only one VSAM data set (UNIQUE). The format-1 DSCB of the data set or data space to be renamed is read in (after it is verified that the new name is not already on the volume), the 44-byte name is changed, and the format-1 DSCB is written back to the VTOC. Only a single data set may be renamed by each issuance of the RENAME macro; however, multivolume data sets may be renamed, if either all volumes are mounted simultaneously, or one of the mounted volumes can be demounted (physically and logically).

When the Rename routine is entered, register 0 must contain either a pointer to a unit control block or all zeros (no UCB address specified). Register 1 contains the address of a parameter list that contains pointers to (1) the name of the data set to be renamed, (2) the new name of the data set, and (3) a volume list. The volume list contains a 2-byte count of the number of volumes in the data set to be renamed and a 12-byte entry for each volume of the data set. The 12-byte entry is used to specify the device code and the volume serial number.

The Rename routine then goes through the same initialization and volume verification logic and then, if needed, the same volume demounting/mounting routine that are described previously for the Scratch routine under "Scratch Initialization, Volume Verification, and Volume Mounting." Upon verification that a volume to be renamed is mounted or after having mounted the required volume, the Rename routine builds the control blocks and channel programs required by IOS to read from and write to the VTOC.

When a data set residing on more than one volume is to be renamed, the Rename routine maintains control until all volumes have been processed. It is possible, however, that, because of a permanent I/O error or because a volume cannot be mounted, the data set may not be renamed on all volumes. The success or failure (and reason for failure) of the Rename processing is communicated in the last byte of the volume list entry (the Rename status code). Success or failure is also indicated by the completion code passed in register 15 when control is returned to the caller.

When a volume containing a data set to be renamed is mounted, the VTOC is searched for a data set name that is the same as the new name in the Rename parameter list (a duplicate name is not allowed on the VTOC). Next, the VTOC is enqueued, and the identifier (format-1) DSCB of the data set to be renamed is read in. If the data set to be renamed is RACF-indicated, password checking is bypassed and a RACHECK macro instruction is issued to check authorization to rename, followed by a "RACDEF NEWNAME" to change the name in RACF. The RACDEF NEWNAME is bypassed for RACF-defined data sets when the caller is authorized and the JSCBPASS bit as well as the RENAME parameter list, second byte, second bit, is set to 1. If the data set is password protected, an exit is taken to the SECLOADA module of the Open/EOV password-protection function, which verifies the password by polling the operator. (*OS/VS2 System Programming Library: Data Management* describes password protection procedures.) When the password has been verified, control is returned to Rename; the name in the format-1 DSCB is changed, the format-1 DSCB is written back to the VTOC, and the VTOC is dequeued. If there are additional volumes that contain the data set to be renamed, processing is repeated. If the system includes SMF and data set activity records are requested, type-18 records will be written when processing is completed. The TIOT and data set name are dequeued, the work areas are freed, the completion code is loaded, and control is returned to the calling routine.

Reading a DSCB from the VTOC (Obtain)

The Obtain routine is entered when the OBTAIN macro instruction (SVC 27) is issued. The function of the Obtain routine is to read a DSCB into a work area provided by the caller.

The caller has two options when he issues an OBTAIN macro: (1) he may specify the data set name and receive the data portion (last 96 bytes) of the format-1 or format-4 DSCB for that data set or data space and the absolute address of the DSCB (search option), or (2) he may specify the absolute track (CCHHR) of the DSCB he wants (seek option) and receive the complete DSCB (140 bytes). Note that for VSAM and VIO data sets an absolute track address of zero is returned because no DSCB exists. Only one DSCB can be obtained with each issuance of the OBTAIN macro instruction. The volume on which the DSCB resides must be mounted. TSO uses the Obtain routine to get information about a VSAM data set or index. However, only the "search" option is available. The data set name for a suballocated VSAM data set appears only in the VS2 master catalog or in a VSAM user catalog, not in the VTOC. The data set name of a unique VSAM data set does appear in the VTOC. Therefore, to determine the extents of a suballocated VSAM data set, the Obtain routine must issue a VSAM LOCATE command to retrieve physical extent information from the VSAM catalog.

When the Obtain routine receives control, register 1 points to a parameter list that contains (1) a code specifying the search or seek option, (2) the address of a data set name (search option) or a pointer to the absolute track address (CCHHR) of the DSCB he wants (seek option), (3) the address of the volume serial number of the volume on which the DSCB resides, and (4) the address of a 140-byte work area, into which the DSCB and its absolute track address are read.

Obtain processing begins with getting a work area used for control blocks and channel programs. If the caller is in problem state, but not key 0-7, Obtain branches to the resident validity-checking routine to verify that the caller's work area has a protection key that matches the key in the current task's TCB. Next, a search of the UCBs is conducted to find the UCB with the volume serial number that matches the volume serial number passed in the Obtain parameter list.

If the search option has been specified, the 96-byte data portion of the format-1 DSCB (or format-4 DSCB if the data set name is 44 characters of X'04') of the named data set is read into the Obtain work area, along with its absolute track address (CCHHR). This information is moved to the caller's work area. If the Obtain request is for a suballocated VSAM data set, the search of the VTOC will fail because a VSAM data set name appears only in a VSAM catalog. The VSAM data space name, which may be the name of a unique VSAM data set, appears in the format-1 DSCB. Before taking an error exit (no data set name found), the Obtain routine checks the format-4 DSCB to determine whether there are VSAM data sets on the volume. If so, the Obtain routine issues a VSAM LOCATE command to get extent information from the VS2 master catalog. This information is moved into the caller's work area.

If the seek option is in effect, the format-4 (VTOC) DSCB is read to verify that the absolute track address of the DSCB specified is in the VTOC extent. Next the complete DSCB is read and moved to the caller's work area.

Obtain processing is now complete. The Obtain work area is freed, the completion code is loaded in register 15, and control is returned to the caller.

VIO Data Sets: No DSCB exists for VIO data sets. However, a virtual DSCB resides in the scheduler work area. The Obtain routine finds this virtual DSCB by searching the DSAB chain for a TIOT DD entry containing a virtual UCB whose virtual DSCB contains the data set name of the data set for which the OBTAIN macro was issued. When located, the virtual DSCB is moved to the caller's work area. Only the search option of OBTAIN can be issued for VIO data sets.

Obtaining Information About Free Space on a Volume (LSPACE)

The LSPACE routine accumulates information about free space on a volume. This information is used by the System Management Facility and the routines that issue demount messages (when the "MONITOR SPACE" command has been issued by the operator). LSPACE is invoked by the issuance of an SVC 78.

At entry to LSPACE, register 0 points to a UCB and register 1 indicates the functions to be performed, as follows:

- If the three low-order bytes of the register have a nonzero value, space information is to be obtained for a routine that issues demount messages when the "MONITOR SPACE" command has been issued by the operator. The value is a pointer to a 30-byte message area that will receive the space information.
- If the high-order byte of the register is 0, no space information is required for SMF.
- If bit one of the register is set to 1, space information may be required for SMF, depending upon other conditions to be tested.
- If bit zero of the register is set to 1, space information is required for SMF.

The LSPACE routine checks the UCB passed to ensure that it identifies a direct-access device and checks the message area to be sure it is in the user's area (if the user is in program state).

Next, a work area is obtained, the I/O supervisor control blocks are built in the work area, and a channel program for reading the format-4 and first format-5 DSCBs is relocated to the work area.

When the channel program and the control blocks are established, the VTOC is enqueued using a RESERVE macro instruction, and the channel program is executed to read in the format-4 and first format-5 DSCB. The channel program is modified to read in the subsequent format-5 DSCBs on the volume.

The extent, cylinder, and track information is accumulated from the format-5 DSCBs. If SMF information is required, the volume label is read in and a type-19 record is constructed and written to the SMF data set.

If the "MONITOR SPACE" command is in effect, the accumulated extent, cylinder, and track information is converted to EBCDIC and moved to the 30-byte message area provided by the caller. The message looks like this:

SPACE=CCCC,TTTT,AAAA/*cccc,tttt*

Where:

CCCC	= Total number of free cylinders
TTTT	= Total number of free tracks
AAAA	= Total number of free extents
cccc,tttt	= Largest contiguous free area in cylinders and tracks

LSPACE processing is now completed by dequeuing the VTOC, freeing the LSPACE work area, loading the completion code in register 15 and the message address in register 1, and returning control to the caller.

Processing PASSWORD Data Set Requests

The Protect routine (SVC 98) is used to add, replace, delete, or list entries in the system PASSWORD data set.

Note: Another protection scheme is used for VSAM data spaces. See *OS/VS2 Virtual Storage Access Method (VSAM) Logic* for a description of the protection provided for VSAM data spaces.

Validity Checking

Input to the Protect routine is the address of a parameter list. The size and contents of the parameter list depend on whether a Protect routine is going to add, replace, delete, or list an entry in the PASSWORD data set. See *OS/VS2 System Programming Library: Data Management,* for an explanation of the parameter lists and the functions of the PROTECT macro instruction.

If (1) the parameter list is incomplete, (2) the password name specified in the parameter list is not valid, or (3) there is no PASSWORD data set, an error code is set. Also, if a buffer address is specified in the parameter list, the buffer address is tested by the validity-checking routine that resides in the system nucleus. The routine determines whether the buffer location has a protection key equal to the protection key in the task control block (TCB) of the routine that issued the PROTECT macro instruction.

If the parameter list information is valid, the Protect routine enqueues the system-residence volume, which contains the PASSWORD data set. An OBTAIN macro instruction is issued to read the format-1 DSCB for the PASSWORD data set. Next the control blocks needed by IOS are built, and a channel program to search and update the PASSWORD data set is initialized.

Processing the PASSWORD Data Set

The request is processed by one of the following four routines:

- The add routine—if the requested function is to add a record to the PASSWORD data set
- The replace routine—if the requested function is to replace a record in the PASSWORD data set
- The delete routine—if the requested function is to delete a record in the PASSWORD data set

- The list routine—if the requested function is to list a record in the PASSWORD data set

Updating the Format-1 DSCBs

If the function requested is to add an entry to the end of the PASSWORD data set, the DS1LSTAR field (address of the last block written to the data set) of the format-1 DSCB for the PASSWORD data set must be updated and written back to the VTOC.

When a data set has been newly protected, it is necessary to change the protection status of the data set. This is done by reading in the format-1 DSCB, modifying the password protection bits of the DS1DSIND field, and writing the format-1 DSCB back to the VTOC. (The VTOC is enqueued during the updating.) The volume that contains the data set is located either (1) by issuing a LOCATE macro instruction if the data set is cataloged or (2) using a user-supplied list of volume serial numbers that is provided as part of the input. The UCBs are searched for a volume serial number that matches the number in the volume list. If the newly protected data set resides on multiple volumes, the format-1 DSCB is modified on each volume. When all volumes have been processed, the work areas are freed, a completion code is loaded into register 15 (see *OS/VS2 System Programming Library: Data Management*), and control is returned to the calling routine.

How to Use the Method of Operation Diagrams

Diagram 1 is an index to help you find the diagrams or group of diagrams that you're interested in by name.

Overview Diagrams

Diagrams 2, 3, 7, 11, 14, 17, and 21 relate input and output (on the left and right, respectively, of each diagram), to the process steps in the center block of the overview diagram. The numbers on the process blocks of these diagrams are the diagram numbers of subordinate diagrams. The process blocks in the overview diagrams summarize the processing that is described in the subordinate (bottom-level) diagrams. The overview diagrams also contain a legend.

Reading the Bottom-level Diagrams

These diagrams relate processing steps to the input and output they use. The input—control blocks, register contents, and records read from I/O devices—is on the left. Input is what the processing steps use. The processing steps in the center block describe the function of a routine or series of routines in the source code. Each processing step is preceded by a number that is keyed to the extended description on the facing page. The processing block also shows flow of control using the on-page and off-page connectors described in the legend. The output of the processing steps consists of modified control blocks, register contents, and records written to I/O devices. The nature of the use of the input—for reference to virtual storage moved from one location to another, or records read into virtual storage from I/O devices—is determined from the type of data arrow used. The arrows and their meanings are described in the legend. Similarly, control block modification, register settings, and records written to I/O devices in the output block as determined from the arrow type associated with the output.

Reading the Extended Description

The extended descriptions provide more detail about a processing step. They also identify a processing step with the CSECTs and labels within these CSECTs where you can find the source code. Each processing step has a number that corresponds with a number in the extended description.

The bold characters preceding the paragraphs of the extended description—for example, **IGG0325A**—are the CSECT/object module names (and microfiche names) of the source code modules in which you can find the implementation of the functions. The characters in brackets—for example, [CONVNEXT]—following the paragraphs of the extended description are labels in the source code.

Work Areas in the Input and Output

The DADSM routines use work areas for reading from and writing to the VTOC and for communication among control sections. The conventions used in obtaining work areas and the work area chaining procedure is discussed in "DADSM Data Area Chaining." In the diagrams that follow, the base prefix of these work areas and the audit trail used by the recovery routines are not shown; only the extended prefix and WTG table are shown.

The contents of these work area prefixes, the WTG table, and the audit trail are included in:

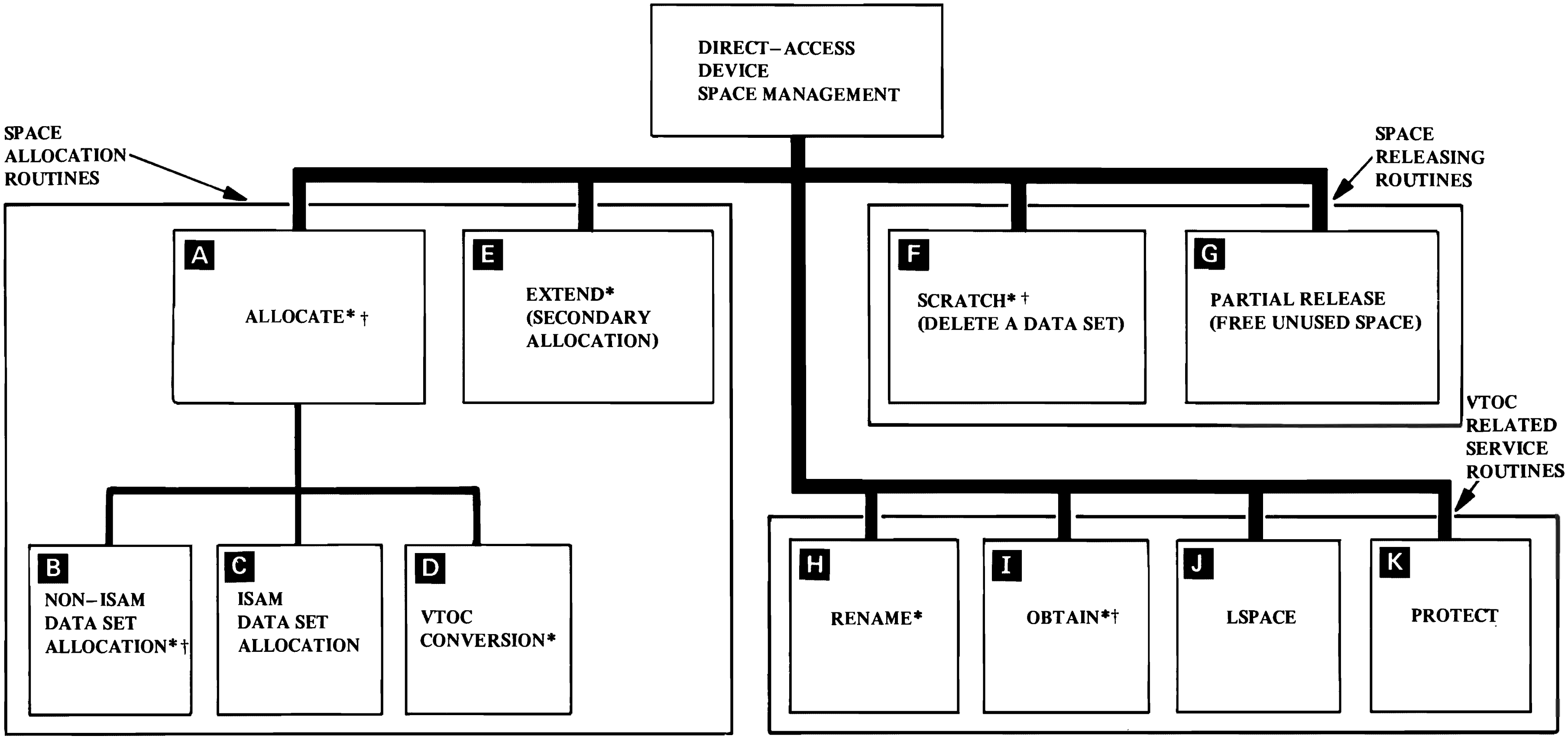
- *OS/VS2 Data Areas*

  Allocate work area
  Extend work area
  O/C/EOV main work area
  Partial Release work area
  Scratch work area

- the "Data Areas" section of this book

  LSPACE work area
  Obtain work area
  Protect work area
  Rename work area
  VTOC Conversion work area

Diagram 01. DADSM Overview



*DADSM Routines used to process VSAM data spaces
†DADSM routines used to process VIO data sets

		Diagram
A	ALLOCATE OVERVIEW	2
B	NON–ISAM ALLOCATION	3 – 6
C	ISAM ALLOCATION	7 – 10
D	VTOC CONVERSION	11 – 13
E	EXTEND	14 – 16

		Diagram
F	SCRATCH	17 – 20
G	PARTIAL RELEASE	21 – 23
H	RENAME	24
I	OBTAIN	25
J	LSPACE	26
K	PROTECT	27

Diagram 02. Allocate Overview

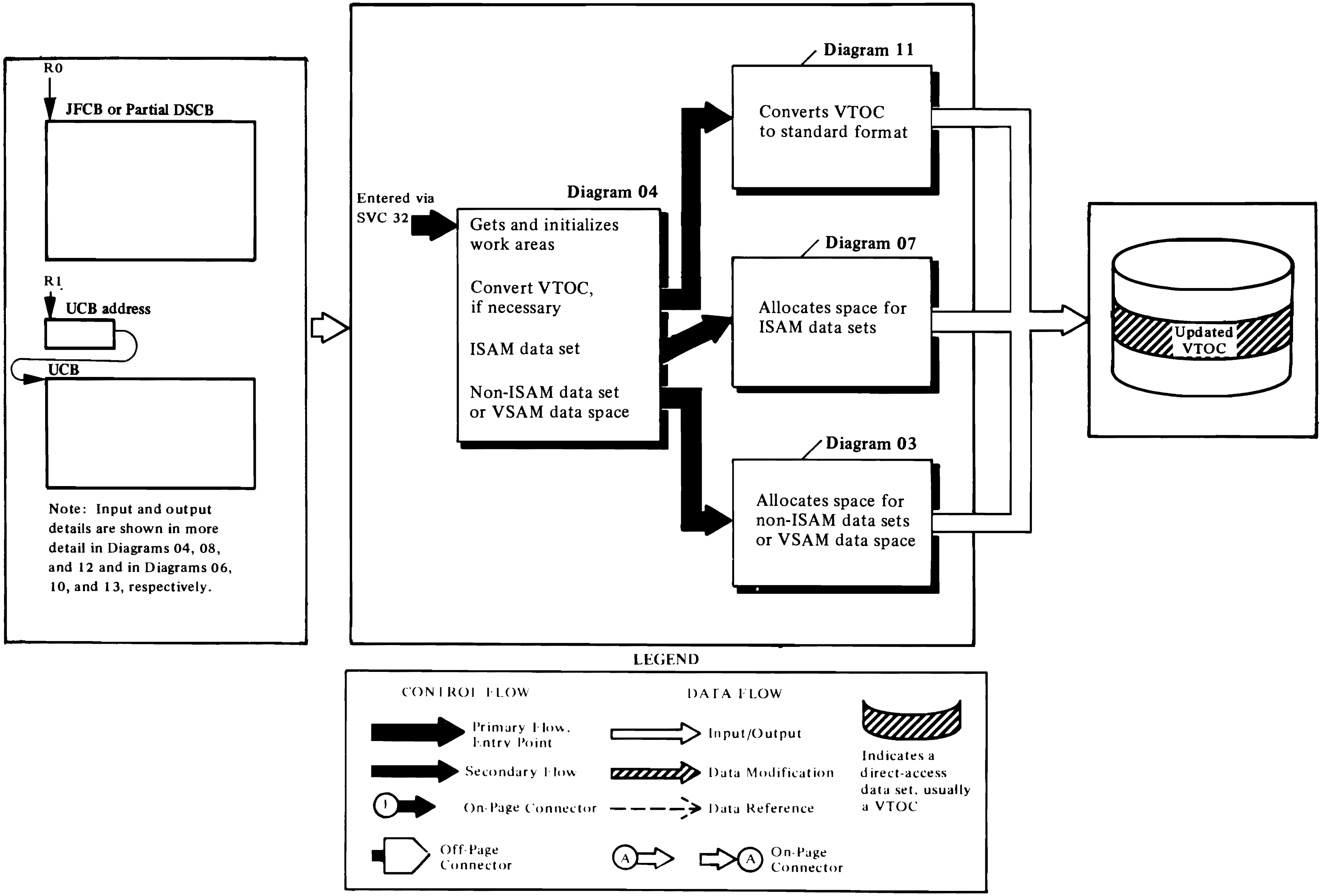
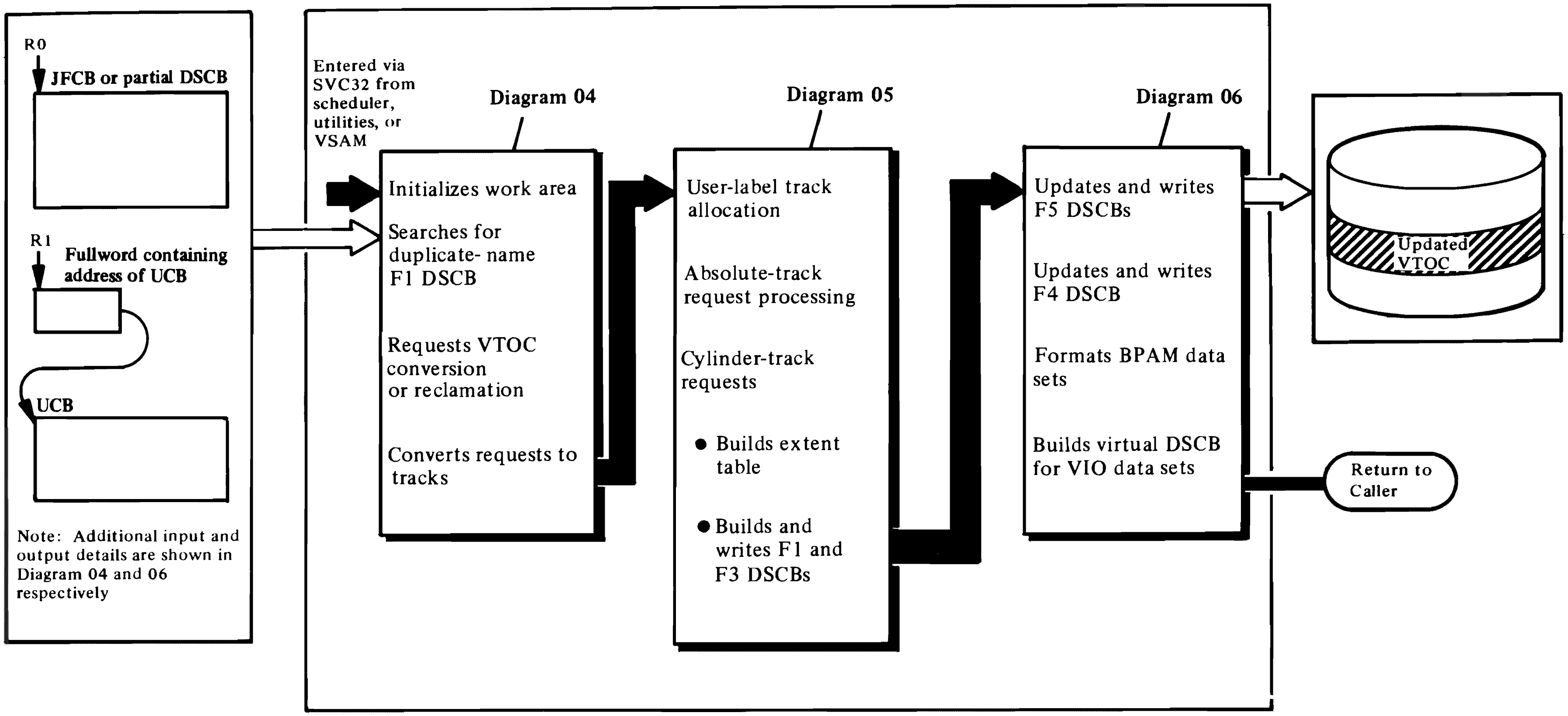

R0
JFCB or Partial DSCB
R1
UCB address
UCB
Note: Input and output details are shown in more detail in Diagrams 04, 08, and 12 and in Diagrams 06, 10, and 13, respectively.
Entered via SVC 32
Diagram 04
Gets and initializes work areas
Convert VTOC, if necessary
ISAM data set
Non-ISAM data set or VSAM data space
Diagram 11
Converts VTOC to standard format
Diagram 07
Allocates space for ISAM data sets
Diagram 03
Allocates space for non-ISAM data sets or VSAM data space
Updated VTOC
LEGEND
CONTROL FLOW
Primary Flow, Entry Point
Secondary Flow
1
On-Page Connector
Off-Page Connector
DATA FLOW
Input/Output
Data Modification
Data Reference
A
A
On-Page Connector
Indicates a direct-access data set, usually a VTOC

Diagram 03. Non-ISAM Data Set or VSAM Data Space Allocation—Overview


R0
JFCB or partial DSCB
R1
Fullword containing address of UCB
UCB
Note: Additional input and output details are shown in Diagram 04 and 06 respectively
Entered via SVC32 from scheduler, utilities, or VSAM
Diagram 04
Initializes work area
Searches for duplicate- name F1 DSCB
Requests VTOC conversion or reclamation
Converts requests to tracks
Diagram 05
User-label track allocation
Absolute-track request processing
Cylinder-track requests
● Builds extent table
● Builds and writes F1 and F3 DSCBs
Diagram 06
Updates and writes F5 DSCBs
Updates and writes F4 DSCB
Formats BPAM data sets
Builds virtual DSCB for VIO data sets
Updated VTOC
Return to Caller

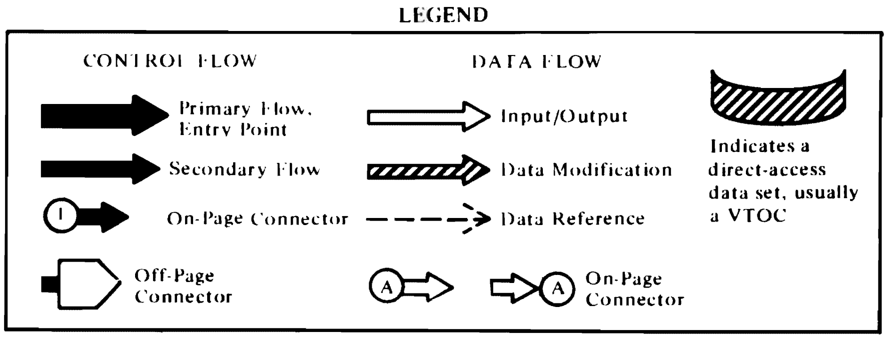
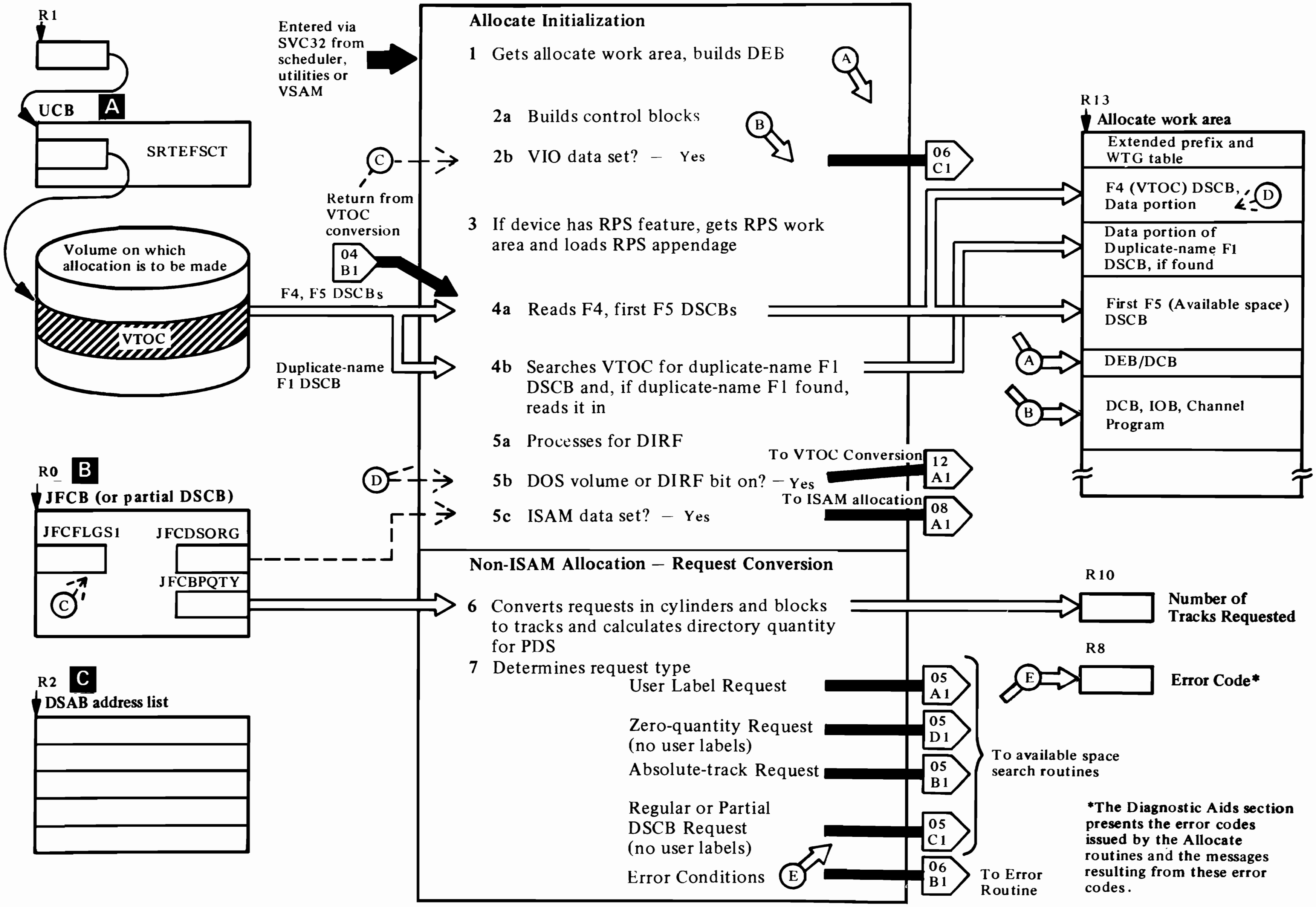


Diagram 04. Allocate Initialization—Duplicate-Name Search and Non-ISAM Allocate Request Conversion



Allocate Initialization

1 Gets allocate work area, builds DEB

2a Builds control blocks

2b VIO data set? — Yes

3 If device has RPS feature, gets RPS work area and loads RPS appendage

4a Reads F4, first F5 DSCBs

4b Searches VTOC for duplicate-name F1 DSCB and, if duplicate-name F1 found, reads it in

5a Processes for DIRF

5b DOS volume or DIRF bit on? — Yes

5c ISAM data set? — Yes

Non-ISAM Allocation — Request Conversion

6 Converts requests in cylinders and blocks to tracks and calculates directory quantity for PDS

7 Determines request type

User Label Request

Zero-quantity Request (no user labels)

Absolute-track Request

Regular or Partial DSCB Request (no user labels)

Error Conditions

*The Diagnostic Aids section presents the error codes issued by the Allocate routines and the messages resulting from these error codes.

R1
UCB
A
SRTEFSCT
Volume on which allocation is to be made
VTOC
R0
B
JFCB (or partial DSCB)
JFCFLGS1
JFCDSORG
JFCBPQTY
C
R2
C
DSAB address list
Entered via SVC32 from scheduler, utilities or VSAM
Return from VTOC conversion
04 B1
F4, F5 DSCBs
Duplicate-name F1 DSCB
A
B
C
D
E
06 C1
To VTOC Conversion
12 A1
To ISAM allocation
08 A1
05 A1
05 D1
05 B1
05 C1
To available space search routines
06 B1
To Error Routine
R13
Allocate work area
Extended prefix and WTG table
F4 (VTOC) DSCB, Data portion
D
Data portion of Duplicate-name F1 DSCB, if found
First F5 (Available space) DSCB
A
DEB/DCB
B
DCB, IOB, Channel Program
R10
Number of Tracks Requested
R8
E
Error Code*

Extended Description for Diagram 04

This function is entered from the SVC SLIH via SVC 32 (Allocate) issued by the scheduler, utilities, and VSAM components of the system control program.

IGC0003B

1 Gets work area used by all Allocate routines. [BEGINA]

2a Builds DEB/DCB, IOB, and FCB in the Allocate work area. Tests JFCFLGS1 field for VIO data set indicator. If a VIO data set is being allocated, transfers control to IGG0325I to build virtual DSCB. [MAINUCB]

2b If a VIO data set is to be allocated, an exit is taken to the VIO allocation routine, IGG0325I (see Diagram 06).

3 Tests for RPS feature. If present, gets 128-byte RPS work area and copies the appendage vector table (AVT) into it. Loads module IGG019EK and enters its address in the AVT. Modifies the DEB in the Allocate work area to point to the RPS AVT. [RPSTEST]

IGG0325A

4a Converts VTOC address from TTR to MBBCCHHR using the resident conversion routine (IECPCNVT). [BEGINA]

Relocates a channel program to the Allocate work area to read the F4 and first F5 DSCB into virtual storage and to search the VTOC for a duplicate name F1 DSCB. Enqueues the VTOC using a RESERVE macro instruction. [MOVECHPG]

4b Reads the F4 and first F5 DSCB and searches all F1 DSCBs on the VTOC for a duplicate name (valid only for ISAM data sets). [EXECUTET]

5a Processes for DADSM interrupt-recording facility (DIRF): sets/resets the DIRF bit. If no previous interrupt has occurred, the F4 is written back to the VTOC with the DIRF bit set. If a previous interrupt has occurred, the F4 is not written back. When a DOS volume or a volume on which the DIRF bit was set (system failed during VTOC updating, a permanent I/O error occurred during the reading or writing of a DSCB, or overlapping extents were found in a DSCB) is encountered, control is transferred to the VTOC conversion routine to convert the VTOC to standard format or to reclaim a VTOC that was invalidated by system failure or permanent I/O error during DADSM processing. The DOS and DIRF bits are bits 0 and 5 of the DS4VTOCI field of the F4 DSCB. [DIRFTEST]

5b If the DOS or DIRF bit is on in the DS4VTOCI field of the F4 DSCB, the VTOC must be converted or reclaimed before an allocation can be made. The VTOC Conversion routine is given control to convert or reclaim the VTOC.

5c When a duplicate name F1 DSCB is found on the VTOC, this section, verifies that an ISAM data set is being allocated. For ISAM data sets, the data portion of the duplicate name DSCB is relocated in the Allocate work area and control is transferred to the ISAM Allocate routine (Diagram 08). [SAMENAME]

IGG0325B

6 Determines what kind of allocation request is being made: in cylinders, in tracks, or in average records. For zero-quantity requests specifying user labels, transfers control to IGG0325K. For zero-quantity requests, transfers control to IGG0325E. [CYLTRKRC]

Converts cylinder requests to tracks. [MULT]

Converts average record requests to tracks. [NOCAL]

Calculates the space required for BPAM directories and converts the quantity to tracks. [DIRCAL]

7 For absolute-track requests, transfers control to IGG0325C. [CLEANUP]

For all allocation requests in tracks (except absolute-track requests), cylinders, or average records, transfers control to IGG0325D, except when user labels are specified (goes to IGG0325K). [REGULAR]

A R1 points to a fullword containing a UCB address. If, however, a VIO data set is being allocated, R1 points to a fullword that contains a device code.

B If the high order bit of R0 is zero, Allocate was called by the task scheduler or VSAM and R0 contains the address of a JFCB. If the high-order byte of R0 is set to one, Allocate was called by a utility and R0 points to a partial DSCB.

C If an ISAM data set is being allocated, R2 points to a list of DSAB addresses. The contents of the list depend on the number of DD statements associated with the data set being allocated:

Word 1—address of DSAB associated with the DD statement being allocated

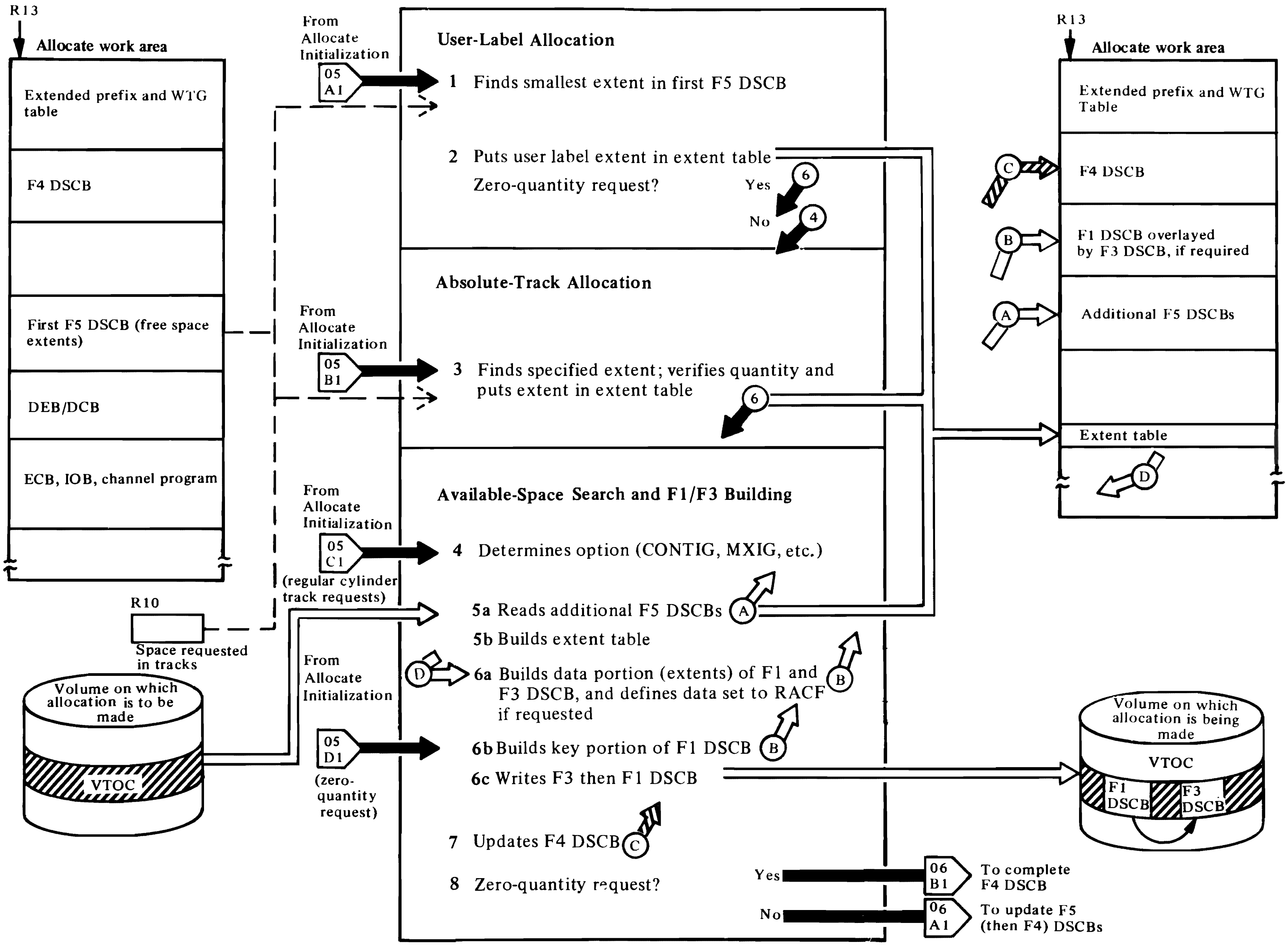
Word 2—address of DSAB that represents the first DD statement associated with the data set (the DD statement containing the DD name)

Word 3—address of the DSAB that represents the second DD statement (zero if there is only one DD statement)

Word 4—address of the DSAB that represents the third DD statement (zero if there are only two DD statements and absent if there is only one DD statement)

Word 5—zero if there are three DD statements, otherwise absent.

Diagram 05. Non-ISAM Allocation—Free Space Search and F1/ F3 DSCB Building



User-Label Allocation

1 Finds smallest extent in first F5 DSCB

2 Puts user label extent in extent table

Zero-quantity request?

Absolute-Track Allocation

3 Finds specified extent; verifies quantity and puts extent in extent table

Available-Space Search and F1/F3 Building

4 Determines option (CONTIG, MXIG, etc.)

5a Reads additional F5 DSCBs

5b Builds extent table

6a Builds data portion (extents) of F1 and F3 DSCB, and defines data set to RACF if requested

6b Builds key portion of F1 DSCB

6c Writes F3 then F1 DSCB

7 Updates F4 DSCB

8 Zero-quantity request?

R13
Allocate work area
Extended prefix and WTG table
F4 DSCB
First F5 DSCB (free space extents)
DEB/DCB
ECB, IOB, channel program
R10
Space requested in tracks
Volume on which allocation is to be made
VTOC
From Allocate Initialization
05 A1
From Allocate Initialization
05 B1
From Allocate Initialization
05 C1
(regular cylinder track requests)
From Allocate Initialization
05 D1
(zero-quantity request)
Yes
6
No
4
6
A
D
B
B
C
Yes
06 B1
To complete F4 DSCB
No
06 A1
To update F5 (then F4) DSCBs
R13
Allocate work area
Extended prefix and WTG Table
C
F4 DSCB
B
F1 DSCB overlayed by F3 DSCB, if required
A
Additional F5 DSCBs
Extent table
D
Volume on which allocation is being made
VTOC
F1 DSCB
F3 DSCB

Extended Description for Diagram 05

This function is entered from the Allocate Initialization function (Diagram 04) when work areas and I/O control blocks have been established and the type of allocation has been determined: step 1 is entered if a user label track is required; step 3 is entered if the allocation is specified at a specific track (absolute track allocation); step 4 is entered for a normal request; or step 6 is entered if a zero-quantity request is being processed.

IGG0325K

1 Converts first F5 DSCB extents from XXYYZ format to RTA and the number of tracks. Finds smallest extent that has the number of tracks needed. [START]

2 Determines request type (in cylinders, tracks, or average records) and makes an entry in the extent table for the user-label track. Modifies the F5 extent from which the user-label track was taken. [TESTYPE]

Transfers control according to request type: zero-quantity request (to IGG0325E), or regular cylinder/track request (IGG0325D). [EXIT]

IGG0325C

3 Searches the F5 DSCBs until the extent containing the absolute track requested is found; when found, determines if there is enough available space at the specified location to fill the request. [RESET]

Moves the F5 extent that fills the request to the extent table. Transfers control to IGG0325E. [INSERTLO]

IGG0325D

4 Converts extents of the first F5 DSCB from XXYYZ to RTA and the number of tracks format, when not entered from the user-label routine (IGG0325K). Determines request option: CONTIG, MXIG, ALX, or no option specified. [START]

5 Searches F5 DSCBs and builds entries for the extent table according to the option specified:

- CONTIG request [CONTIG]
- No option specified [BLANK]
- MXIG or ALX [ALX]

Builds a pushdown list of as many as five extents (four if user labels were specified) when more than one extent is required to fill the request. [PSHDOWN]

IGG0325E

Sorts extent table into ascending RTA sequence. [FIRSTPAS]

6a Branches to subroutine that builds an F1 DSCB and an F3 DSCB, if required. Branches to the resident conversion routine to convert extent table entries from RTA1/RTA2 to CCHHCCHH and enters them in the F1 or F3 DSCB. [START1]

Sets RACF indicator in DSCB [DS1DSIND=X'40'] if auto data set indicator [JFCFLGS1=JFCBADSP] is set. [BLDF1A]

Passes control to RACF to define the data set to RACF if the RACF indicator in the DSCB is set. (RACDEF TYPE=DEFINE passing DSNAME and VOLSER) [NOUSER]

If RACF indicates duplicate (return code=4) or not authorized to define (return code=8), sets an error code of X'A8' or X'AC' respectively. [NOUSER]

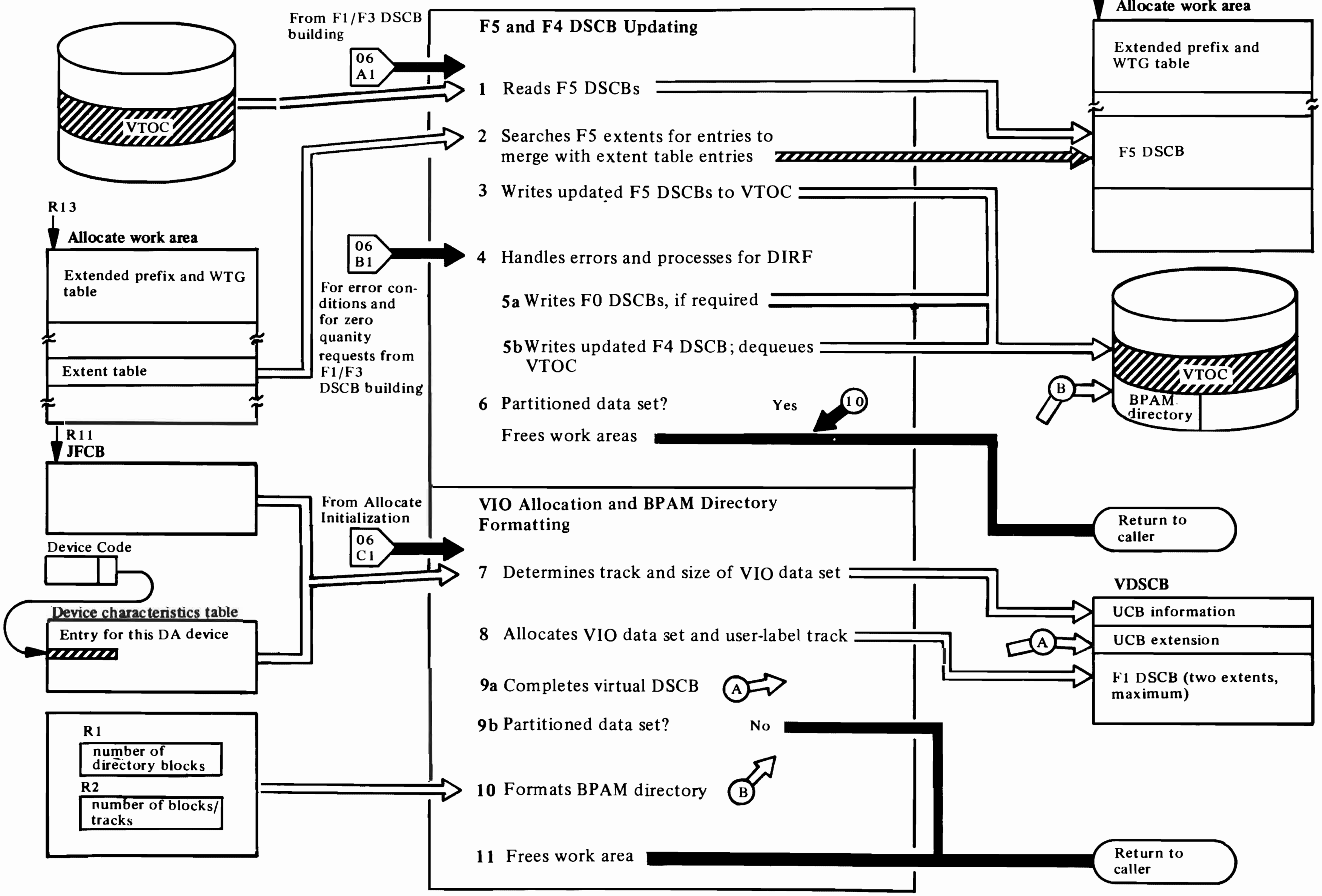
Builds an F3 DSCB and chains it to the F1 DSCB. [BLDF3]

6b When a partial F1 DSCB has been supplied, builds the F1 using the partial DSCB as a model. [BLDF1B]

Builds an F1 DSCB from the JFCB information. [BLDF1A]

7 Updates the hole count (DS4DSREC) and high-water mark (DS4HPCHR) in the F4 DSCB, if necessary. [OLDHIWAT]

Diagram 06. Non-ISAM Allocation–F4 and F5 DSCB Updating, VIO Allocation and BPAM Directory Formatting



F5 and F4 DSCB Updating

1 Reads F5 DSCBs

2 Searches F5 extents for entries to merge with extent table entries

3 Writes updated F5 DSCBs to VTOC

4 Handles errors and processes for DIRF

5a Writes F0 DSCBs, if required

5b Writes updated F4 DSCB; dequeues VTOC

6 Partitioned data set?

Frees work areas

VIO Allocation and BPAM Directory Formatting

7 Determines track and size of VIO data set

8 Allocates VIO data set and user-label track

9a Completes virtual DSCB

9b Partitioned data set?

10 Formats BPAM directory

11 Frees work area

Extended Description for Diagram 06

This function is entered at step 1 to update the F4 and F5 DSCBs, after the allocation has been entered in the F1 and F3 DSCBs. Step 4 is entered when errors have occurred in previous allocation processing. Step 7 is entered to allocate a VIO data set, which does not involve VTOC modification. Step 9 is entered to format BPAM data sets.

IGG0325G

1 Initializes pointers and counters, clears F5 DSCB output area. If the first F5 DSCB is not in virtual storage, branches to a subroutine that reads it in. Branches to a subroutine that converts the F5 extents from XXYYZ format to RTA1/RTA2. [DADSMSTR]

2 Determines when and where the extent table entries should be merged with the F5 DSCB extents. [DUELOOP]

Merges extent table entries with the F5 DSCB extents if the space is allocated from the beginning or middle of an F5 extent. [ALTEREX]

Merges extent table entries with the F5 DSCB extents if the space is allocated from the end of an F5 extent or if it takes all of the extent. [TOTALL]

3 When a new F5 DSCB has been created, searches for a hole (F0 DSCB) on the VTOC over which the new F5 can be written. [SEARCH]

Converts RTA1/RTA2 entries to XXYYZ format in the updated/new F5 DSCB. Chains the F5 DSCB, completes the fields and branches to a subroutine that writes the F5 to the VTOC.

When the second and subsequent F5 DSCBs must be read into virtual storage, links to a subroutine that reads in the next F5 and then converts its extents from XXYYZ to RTA1/RTA2. [SAVEID]

Branches to a subroutine that writes the last F5 DSCB to the VTOC; transfers control to IGG0325H. [WRITEFNL]

IGG0325H

4 Sets/resets DIRF bit. Tests DIRF indicator; if there was a previous interrupt or if a permanent I/O error has been encountered, returns to caller without writing the F4 back to the VTOC. [START]

Calculates the TTR or converts the CCHHR of each data set to TTR format and stores it in the JFCB. [CONVERT]

5a If an F0 DSCB is to be written (when an F5 DSCB is to be removed), branches to a subroutine that writes the F0 DSCB. [TSTZERO]

5b Modifies the channel program and links to a subroutine that writes the modified F4 DSCB to the VTOC. [WRITEF4]

6 When directory processing is required, links to a subroutine that dequeues the VTOC. Loads registers required by CVOL and transfers control to IGG0CLF2. [TSTDIR]

Entered for non-directory requests, dequeues the VTOC, releases the Allocate work area, loads the return code, and returns control to the caller. [GOODEND]

IGG0325I

7 This routine receives control from IGC0003B (Allocate initialization) when space is to allocated for a VIO data set. VIO data sets can be of any data set organization except indexed sequential or the VSAM organization. After storage is obtained for the virtual DSCB, initializes the fields of the UCB using constants, the volume serial number from the JFCB, and information from the device characteristics table. [BLDUCB]

Determines the CCHH of the first track to be allocated to the data set. This depends on whether the request is for an absolute track, normal track or cylinder, or average block allocation. [DETSCCHH]

Initializes the last-track-written and track-balance fields in the F1 DSCB. [INITLSTR]

Calculates the number of tracks to be allocated, based on whether the request is for a specific (absolute) track or is in the from of a normal cylinder or track request. The number of tracks is equal to the primary number requested for the data set plus as much as 15 times the secondary quantity. If the primary quantity plus 15 times the secondary quantity is greater than the volume capacity, the complete available area on the volume (containing system paging data sets) is allocated. [ALLOCTRK]

For requests in average blocks, calculates the total number of tracks. [BIGABL]

8 If user labels are requested, allocates a track for user labels. [ALLOCULT]

Uses information from the JFCB to complete the F1 DSCB. [MRGJFCB]

Puts the extents in the F1 DSCB. [BLDDSEXT]

9a Puts the address of the first track allocated, and the number of tracks allocated, in the VIO work area of the VDSCB. [VIOWA]

9b For partitioned data sets, passes the directory quantity and directory blocks per track to IGG0CLF2, the BPAM directory formatting routine. For sequential and direct data sets, this routine returns to the caller. [FREEDSCB]

IGG0CLF2

10 Writes 256-byte blocks with 8-byte binary zero keys in the directory of BPAM data sets. [BPLOOP1]

If the data set is a VIO data set, the first issuance of the EXCP macro instruction to write a directory block causes the VIO routines to build control blocks (the VBPPL, and the WICB) and obtain storage for the virtual track. The control blocks are used to translate the EXCP request (see *OS/VS2 VIO Logic* for details). At conclusion of processing, issues a WICLOSE macro instruction to release these control blocks and close the VIO data set. [BPNLST]

11 Frees work areas and returns to caller. [RETURN]

Diagram 07. ISAM Allocation—Overview

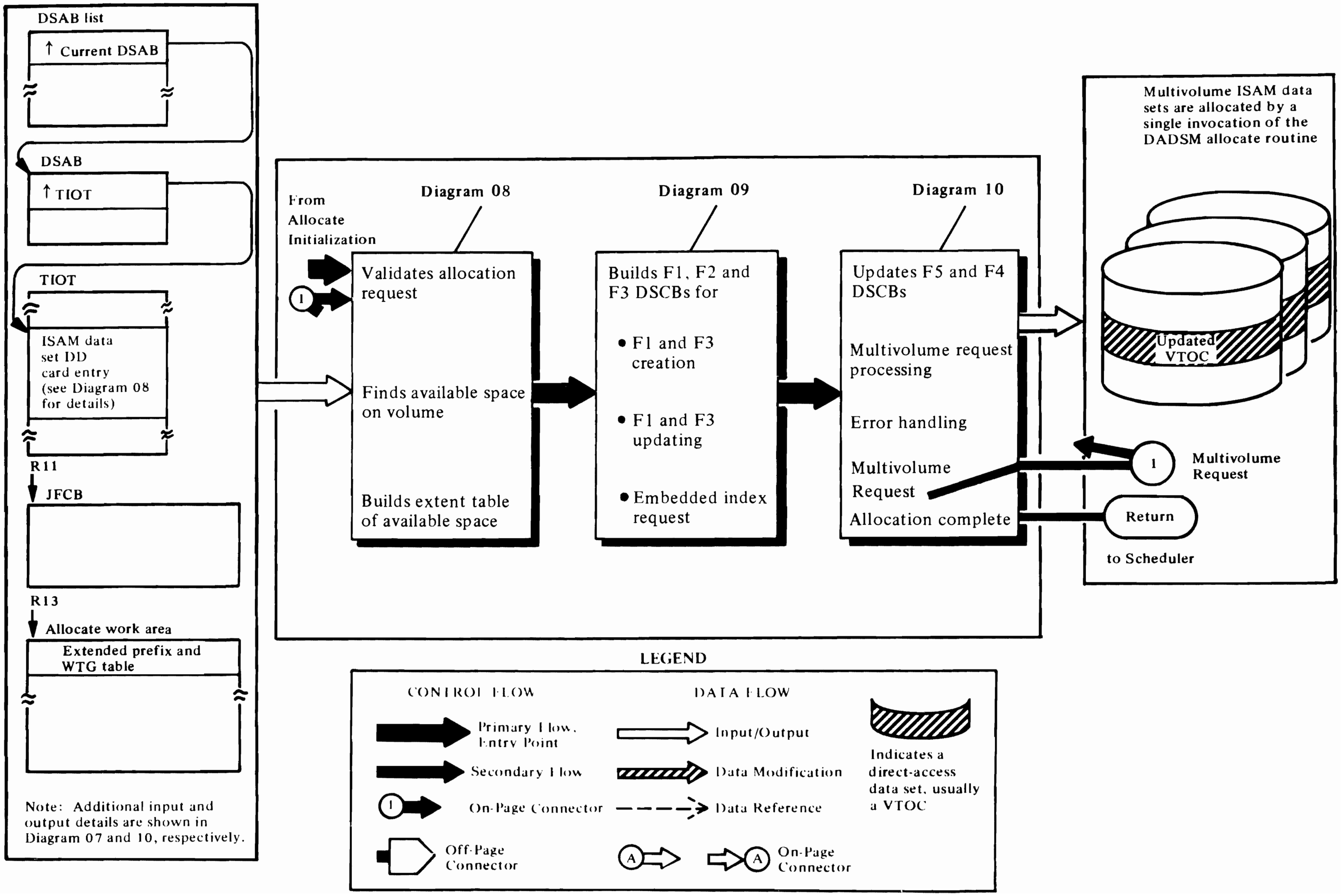
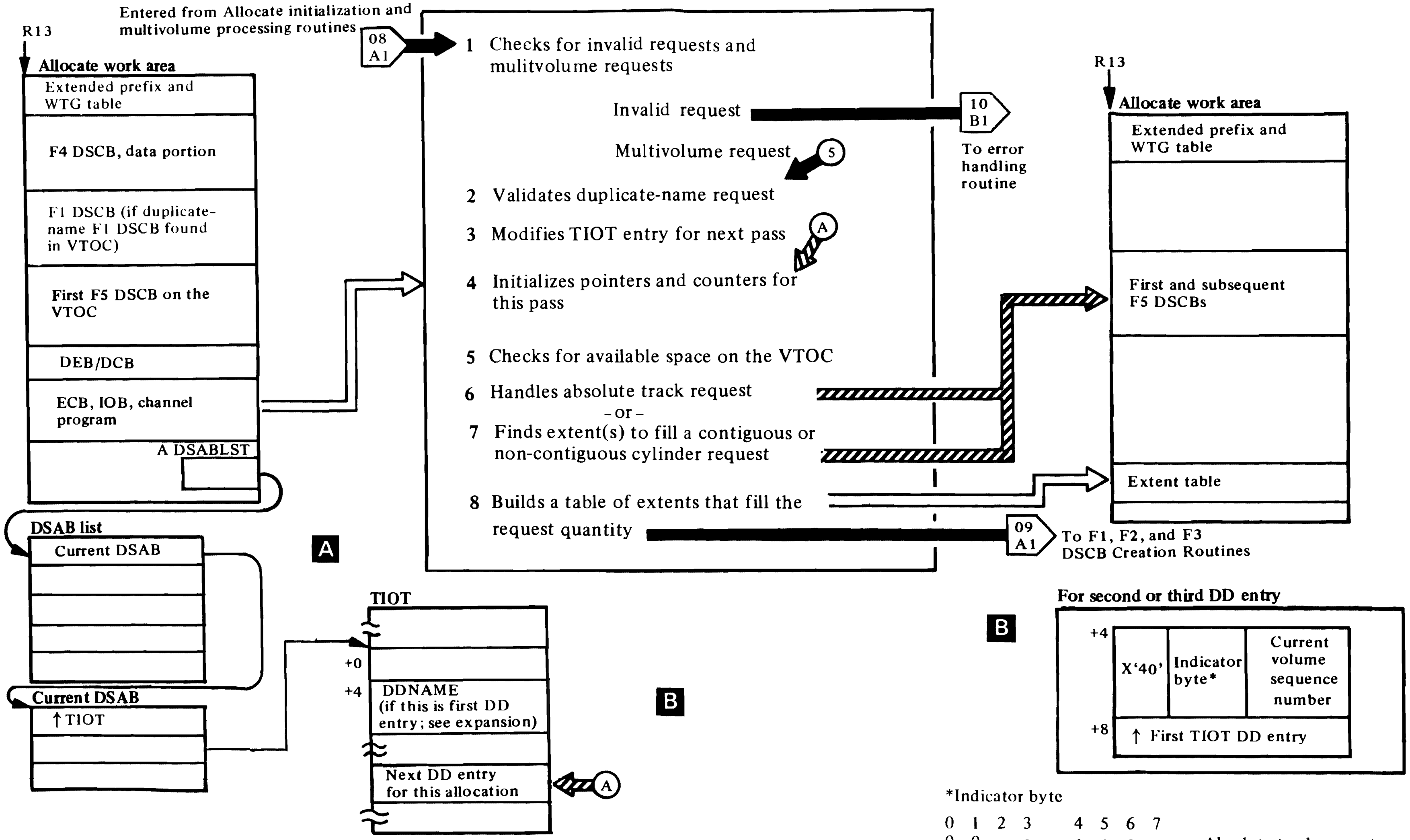

DSAB list
Current DSAB
DSAB
TIOT
TIOT
ISAM data set DD card entry (see Diagram 08 for details)
R11
JFCB
R13
Allocate work area
Extended prefix and WTG table
Note: Additional input and output details are shown in Diagram 07 and 10, respectively.
From Allocate Initialization
1
Diagram 08
Validates allocation request
Finds available space on volume
Builds extent table of available space
Diagram 09
Builds F1, F2 and F3 DSCBs for
• F1 and F3 creation
• F1 and F3 updating
• Embedded index request
Diagram 10
Updates F5 and F4 DSCBs
Multivolume request processing
Error handling
Multivolume Request
Allocation complete
Multivolume ISAM data sets are allocated by a single invocation of the DADSM allocate routine
Updated VTOC
1
Multivolume Request
Return
to Scheduler
LEGEND
CONTROL FLOW
Primary Flow, Entry Point
Secondary Flow
1
On-Page Connector
Off-Page Connector
DATA FLOW
Input/Output
Data Modification
Data Reference
A
A
On-Page Connector
Indicates a direct-access data set, usually a VTOC

Diagram 08. ISAM Allocation—Request Validation and Free Space Search



Entered from Allocate initialization and multivolume processing routines

1 Checks for invalid requests and mulitvolume requests

Invalid request

Multivolume request

2 Validates duplicate-name request

3 Modifies TIOT entry for next pass

4 Initializes pointers and counters for this pass

5 Checks for available space on the VTOC

6 Handles absolute track request

-or-

7 Finds extent(s) to fill a contiguous or non-contiguous cylinder request

8 Builds a table of extents that fill the request quantity

To error handling routine

To F1, F2, and F3 DSCB Creation Routines

08
A1
10
B1
09
A1
5
A
A
B
B
R13
Allocate work area
Extended prefix and WTG table
F4 DSCB, data portion
F1 DSCB (if duplicate-name F1 DSCB found in VTOC)
First F5 DSCB on the VTOC
DEB/DCB
ECB, IOB, channel program
A DSABLST
DSAB list
Current DSAB
Current DSAB
↑TIOT
TIOT
+0
+4
DDNAME (if this is first DD entry; see expansion)
Next DD entry for this allocation
R13
Allocate work area
Extended prefix and WTG table
First and subsequent F5 DSCBs
Extent table
For second or third DD entry
+4
X'40'
Indicator byte*
Current volume sequence number
+8
↑ First TIOT DD entry

*Indicator byte

0	1	2	3	4	5	6	7	
0	0	·	·	·	·	·	·	Absolute-track request
0	1	·	·	·	·	·	·	Blank area-type request
1	1	·	·	·	·	·	·	Contiguous space request
·	·	1	·	·	·	·	·	Second DD statement
·	·	·	·	·	·	·	1	Index requested
·	·	·	·	·	·	1	·	Prime area requested
·	·	·	·	·	1	·	·	Overflow area requested

Extended Description for Diagram 08

IGG032I1

This function is entered from the Allocate initialization routine when an ISAM data set is to be allocated.

1 Determines whether this is a cylinder or absolute-track request. [ALPHA]

Checks for entry from the multivolume-loop module; if multivolume request, transfers control to next load (IGG032I2); checks validity of duplicate-name F1 DSCB if one has been found. [CHECK]

2 Verifies UCBs for this and preceding request are same for duplicate-name requests. [CONTINUE]

Calculates the address of legitimate duplicate-name F1 DSCB. [LEGITIMT]

3 For the first DD statement, determines if a second DD statement exists for this ISAM allocation, checks validity of request type (that is, absolute track, cylinder) and space type (index, prime, overflow). Sets bit in the next RIOT entry to indicate request type and space type of this TIOT entry.

For second or third DD statement, ensures that request type (contiguous cylinder, overflow, etc.) is appropriate. [NONAME]

4 Sets up work area for this pass: TIOT DD entry pointer, primary quantity requested, embedded index quantity, absolute track address. [MOVEFLDS]

IGG032I2

5 Sets up channel program to read F5 DSCBs. [GOREAD]

Sets up the F5 DSCB in virtual storage to search for available extents. [SUCCIO]

6 For absolute-track requests:

- Ensures that quantity requested is on cylinder boundaries. [NONEHERE]
- Finds the extent that contains the specified tracks. [COMPRTA]
- Unless the last F5 extent has been reached, determines if the last F5 extent will fill the request. [CHECKOUT]
- When the extent containing the specified track is found, determines if the space is available. [STOPHERE]
- Converts XXYYZ of F5 extent to RTA1/RTA2 and builds a one-entry extent table. [GOODQTY]
- If there is more than one F5 DSCB on the VTOC, reads in next F5. [UPDATESK]

7 For contiguous requests in cylinders, finds smallest F5 extent on the volume that fills the request. [CYLREQ]

For non-contiguous (no-option-specified) requests in cylinders, finds the smallest extent that fills the request or builds and sorts by RTA a five-entry list of the largest extents encountered that are insufficient to fill the request. [BLANK]

8 Selects the extents (from two to all five) that are needed from the list just built to fill the request. Enters the extents selected in the extent table. [PICKLIST]

A If an ISAM data set is being allocated, R2 points to a list of DSAB addresses at the time SVC 32 is issued. The contents of the list depends on the number of DD statements associated with the data set being allocated:

Word 1—address of DSAB associated with the DD statement being allocated

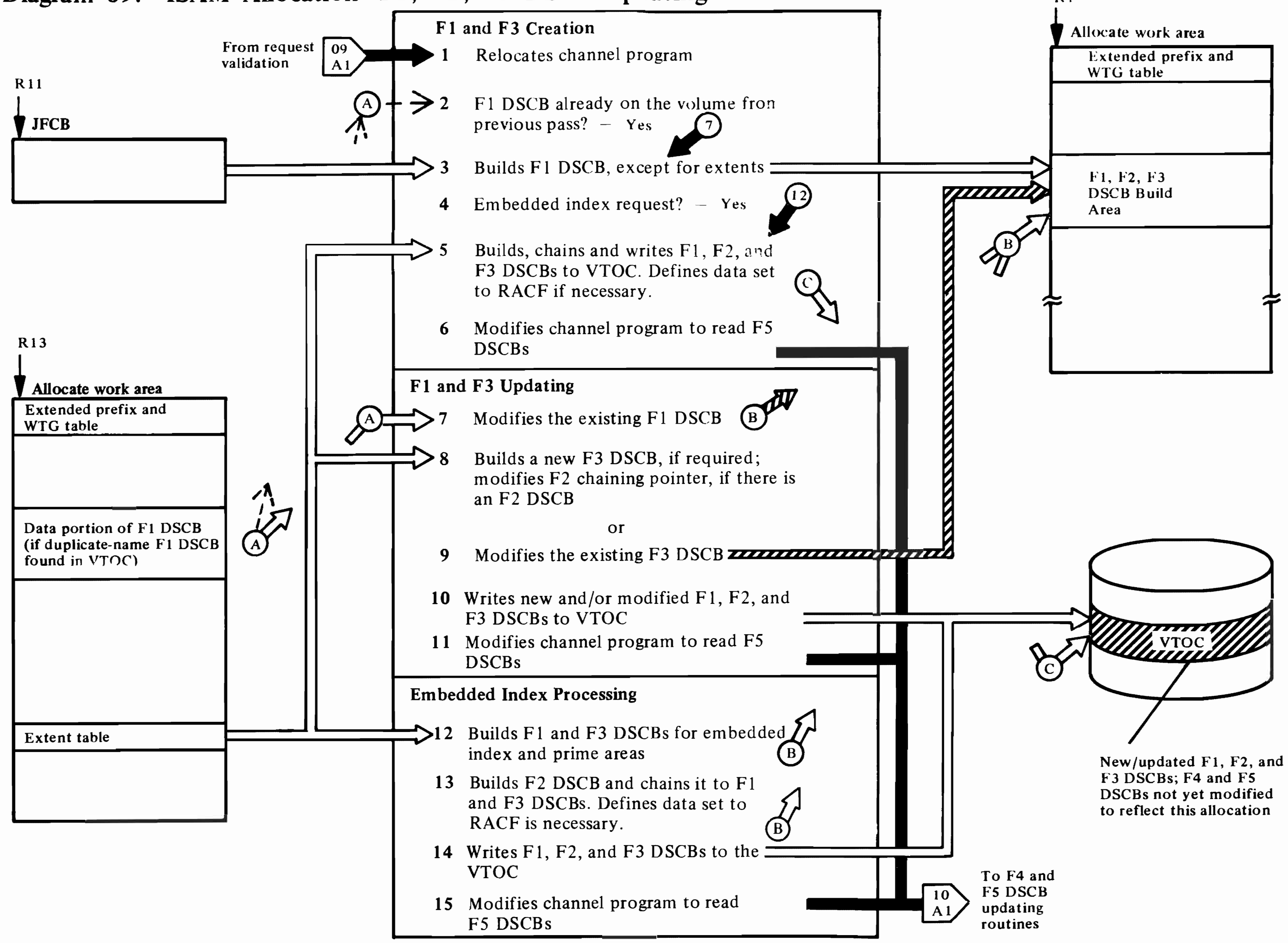
Word 2—address of DSAB that represents the first DD statement associated with the data set (the DD statement containing the DD name)

Word 3—address of the DSAB that represents the second DD statement (zero if there is only one DD statement)

Word 4—address of the DSAB that represents the third DD statement (zero if there are only two DD statements and absent if there is only one DD statement)

Word 5—zero if there are three DD statements, otherwise absent.

Diagram 09. ISAM Allocation–F1, F2, F3 DSCB Updating


From request validation
09 A1
R11
JFCB
R13
Allocate work area
Extended prefix and WTG table
Data portion of F1 DSCB (if duplicate-name F1 DSCB found in VTOC)
Extent table
R1
Allocate work area
Extended prefix and WTG table
F1, F2, F3 DSCB Build Area
VTOC
New/updated F1, F2, and F3 DSCBs; F4 and F5 DSCBs not yet modified to reflect this allocation
10 A1
To F4 and F5 DSCB updating routines
A
B
C
7
12

F1 and F3 Creation

1 Relocates channel program

2 F1 DSCB already on the volume fron previous pass? – Yes

3 Builds F1 DSCB, except for extents

4 Embedded index request? – Yes

5 Builds, chains and writes F1, F2, and F3 DSCBs to VTOC. Defines data set to RACF if necessary.

6 Modifies channel program to read F5 DSCBs

F1 and F3 Updating

7 Modifies the existing F1 DSCB

8 Builds a new F3 DSCB, if required; modifies F2 chaining pointer, if there is an F2 DSCB

or

9 Modifies the existing F3 DSCB

10 Writes new and/or modified F1, F2, and F3 DSCBs to VTOC

11 Modifies channel program to read F5 DSCBs

Embedded Index Processing

12 Builds F1 and F3 DSCBs for embedded index and prime areas

13 Builds F2 DSCB and chains it to F1 and F3 DSCBs. Defines data set to RACF is necessary.

14 Writes F1, F2, and F3 DSCBs to the VTOC

15 Modifies channel program to read F5 DSCBs

Extended Description for Diagram 09

IGG032I3

1 Relocates channel program for reading and writing F1, F2, and F3 DSCBs for this and subsequent loads. [BETA]

2 Determines whether F1 DSCB already exists; if so, transfers control to IGG032I4. Sets RACF indicator in DSCB (DS1DSIND=X'40') if automatic data set protection (JFCFLGS1=JFCBADSP). [FINE]

3 Builds all but extent fields of F1 DSCB. [FROMJFCB]

For a second or third DD statement, updates volume sequence number. Puts volume serial number in F1 DSCB. [NOTFIRST]

4 Tests for an embedded-index request and if one is specified, transfers control to IGG032I5. [FILLEXTS]

5 Fills in extents in F1 DSCB, builds an F3 and fills in its extents if required. [LOOP1]

Builds an F2 DSCB if needed; chains F1 to F2 DSCB. [BUILDF2]

Finds an F0 DSCB (hole) for new F1, F2, or F3 DSCB. Sets up channel program for writing F1, F2, and F3 DSCBs to VTOC. [CHANGE2]

If DSCB has RACF indicator set, then issues RACDEF TYPE=DEFINE macro instruction for first volume of data set and RACDEF TYPE=ADDVOL for subsequent volumes. Sets error code X'A8' or X'AC' for RACDEF return codes 4 (duplicate name) or 8 (not authorized to define). [ONEONLY]

Checks for need to enqueue on VTOC with a "set-must-complete" and if required, does so. [CHKSMC]

Links to resident conversion routine to convert RTA1/RTA2 of extent table to CCHHCCHH for F1 and F3 DSCBs. [LOOP]

6 Relocates channel program for updating F5 DSCBs. Transfers control to IGG032I6. [XCTLDADS]

IGG032I4

7 When this routine receives control, there will be an F1 DSCB in virtual storage from the previous pass. Determines if an F3 needs to be built and if not, adds extents to the F1 until maximum (three extents) is reached or until the extent table entries are all processed. [F1EXISTS]

8 If no F3 exists, builds F3 DSCB extents until extent table entries are exhausted, then writes F3 to VTOC. [BUILDF3]

9 If there is an F2 DSCB on the volume, reads it in, updates the pointer to the F3 DSCB, and writes the updated F2 DSCB back to the VTOC [UPDATF2]

If there is already an F3 DSCB on the VTOC for this data set, reads in F2 to find address of F3. [F3EXISTS]

Finds the first available extent in the F3 DSCB, adds an extent to the F3 DSCB for each extent table entry. [ISAF3]

Writes the F3 DSCB to the VTOC. Updates the number of extents in the F1 DSCB. [THRU]

10 Does a "set-must-complete" enqueue on the VTOC if required; links to IOS to read and write from the VTOC. [EXEC]

11 Sets up channel program for updating the F5s. Transfers control to IGG032I6. [XCTLDADS]

Branches to resident conversion routine (IECPCNVT) to convert RTA1/RTA2 of extent table to CCHHCCHH format for F1 and F3 DSCBs. [LOOP]

IGG032I5

12 Finds the middle of the prime area requested using extent table entries. [NEXTA]

Determines how much space from the middle of the next extent table entry is needed for the index. [ICK]

Splits extent table entry into prime and index areas. [CHEEK]

Puts number of index extents into the F1 DSCB; converts index entries for the F1 and F3 DSCBs from RTA1/RTA2 to CCHHCCHH. [OUTPULL]

Converts prime area entries for F1 and F3 DSCBs from RTA1/RTA2 format to CCHHCCHH. When F1 DSCB extents are exhausted, builds an F3 to contain the rest of the extent table entries. [GETPRIME]

13 Builds the F2 DSCB; chains F2 to F1. If DSCB is RACF-indicated, issues RACDEF TYPE=DEFINE to define data set to RACF. Sets return codes X'A8' or X'AC' for RACDEF error return codes 4 (duplicate name) or 8 (not authorized to define) respectively. [BUILDF2]

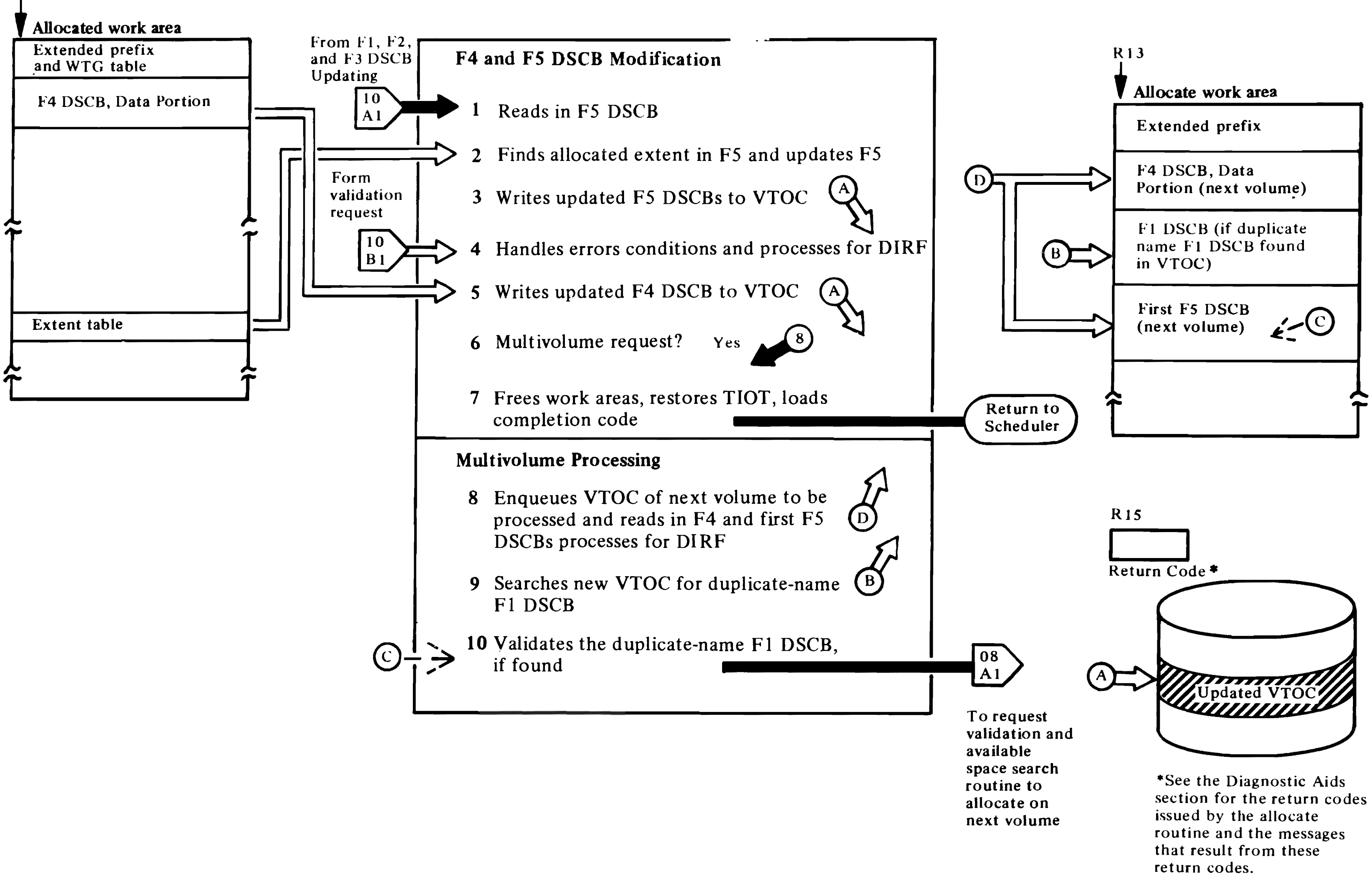
14 Branches to the resident conversion routine (IECPCNVT) to convert RTA to CCHH. [LOOP]

Relocates channel program to write F1, F2, F3s to VTOC. [WRITEF1]

Enqueues on VTOC (with set-must-complete, if required). [SEARCHF0]

15 Relocates channel program for updating the F5 DSCB and branches to IGG032I6. [XCTLDADS]

Diagram 10. ISAM Allocation–F5 and F4 Modification and Multivolume Processing



F4 and F5 DSCB Modification

1 Reads in F5 DSCB

2 Finds allocated extent in F5 and updates F5

3 Writes updated F5 DSCBs to VTOC

4 Handles errors conditions and processes for DIRF

5 Writes updated F4 DSCB to VTOC

6 Multivolume request?

7 Frees work areas, restores TIOT, loads completion code

Multivolume Processing

8 Enqueues VTOC of next volume to be processed and reads in F4 and first F5 DSCBs processes for DIRF

9 Searches new VTOC for duplicate-name F1 DSCB

10 Validates the duplicate-name F1 DSCB, if found

To request validation and available space search routine to allocate on next volume

*See the Diagnostic Aids section for the return codes issued by the allocate routine and the messages that result from these return codes.

Extended Description for Diagram 10

IGG032I6

1 Calculates the TTR of the F1 DSCB and stores it in the JFCB. [CONVERT]

Initializes F5 output area; initializes pointers to input and output areas; reads in first F5 DSCB. [DADSMSTR]

Determines when an extent table entry is to be merged into the F5 DSCBs. [DUELOOP]

Reads first and additional F5 DSCBs. Resets input pointers and counters. Converts extents from CCHHCCHH to RTA1/RTA2 format. [SAVEID]

2 Merges an extent table entry to add available space to the beginning or middle of an existing F5 extent. [ALTEREX]

Merges extent table entries that are returning available space to the end of an existing F5 extent or adding an extent. [TOTALL]

Searches for a hole (F0 DSCB) in the VTOC where a new F5 DSCB can be written. [CHEXT]

3 Converts the RTA1/RTA2 entries to XXYYZ format, inserts F5 chain pointer, completes F5 fields, and writes the F5 DSCB to the VTOC. [WRITEF5]

IGG032I7

4 Tests for error condition and if found, tests DIRF setting; if there was a previous interrupt, takes error exit; if there was no previous interrupt. tests for permanent I/O error; if found, takes error exit; if not a permanent I/O error, forces an F4 to be written to the VTOC and takes an error exit to return to scheduler. [DELTA]

5 Resets DIRF indicator, updates the number of F0 DSCBs left on VTOC, relocates a channel program and writes the F4 DSCB to the VTOC. Tests for a multivolume request. For multivolume requests, updates UCB counter and checks for completion of the multivolume allocation. Increments number of tracks just allocated from total to be allocated and sets up amount to be allocated and absolute track (if requested) on the next volume. [NOERROR]

For an unsuccessful allocation, resets the last 7 bytes of each of the concatenated DD entries to blanks. [MISTAKE]

6 If a multivolume request is being processed, dequeues on this volume and transfers control to IGG032I8 for duplicate-name search. [MIXED]

7 Checks for successful allocation. Fills a TIOT entry for the third DD statement for multiple DD statement requests. [ALLOUT]

If allocation is completed, restores TIOT to the way it was before entering the Allocate routine. [OUT]

Dequeues on VTOC. [FINAL]

Frees Allocate work areas and returns control to calling routine. [CONTINUE]

IGG032I8

8 Relocates the channel program for reading in the F4 DSCB and first F5 DSCB and for searching the VTOC for a duplicate name on the next volume to be processed. Updates the UCB pointer in the DEB. Saves pointer to TIOT DD entry. [RELOCCP]

Links to resident conversion routine to convert VTOC relative track address to absolute address (MBBCCHHR). Issues a RESERVE macro instruction to enqueue on the VTOC of next volume. [ZEROCLR]

9 Reads in data portion of F4 DSCB and the first F5, searches for a duplicate-name F1 on the VTOC, and reads in the duplicate-name F1 DSCB, if found. [EXECUTET]

Sets/resets then tests DIRF indicator. If the DIRF bit (bit 5 of the DS4VTOCI field of the F4 DSCB) is set indicating that one error has been encountered, this routine sets an error code and gives control to the error handling routine. [CHECK]

10 Determines if a duplicate-name DSCB is valid. Calculates the address of the duplicate-name F1 DSCB. [SAMENAME]

Saves F1 and F5 addresses and transfers control to the first load of ISAM Allocate (IGG032I1). [NOTBOS]

Diagram 11. VTOC Conversion–Overview

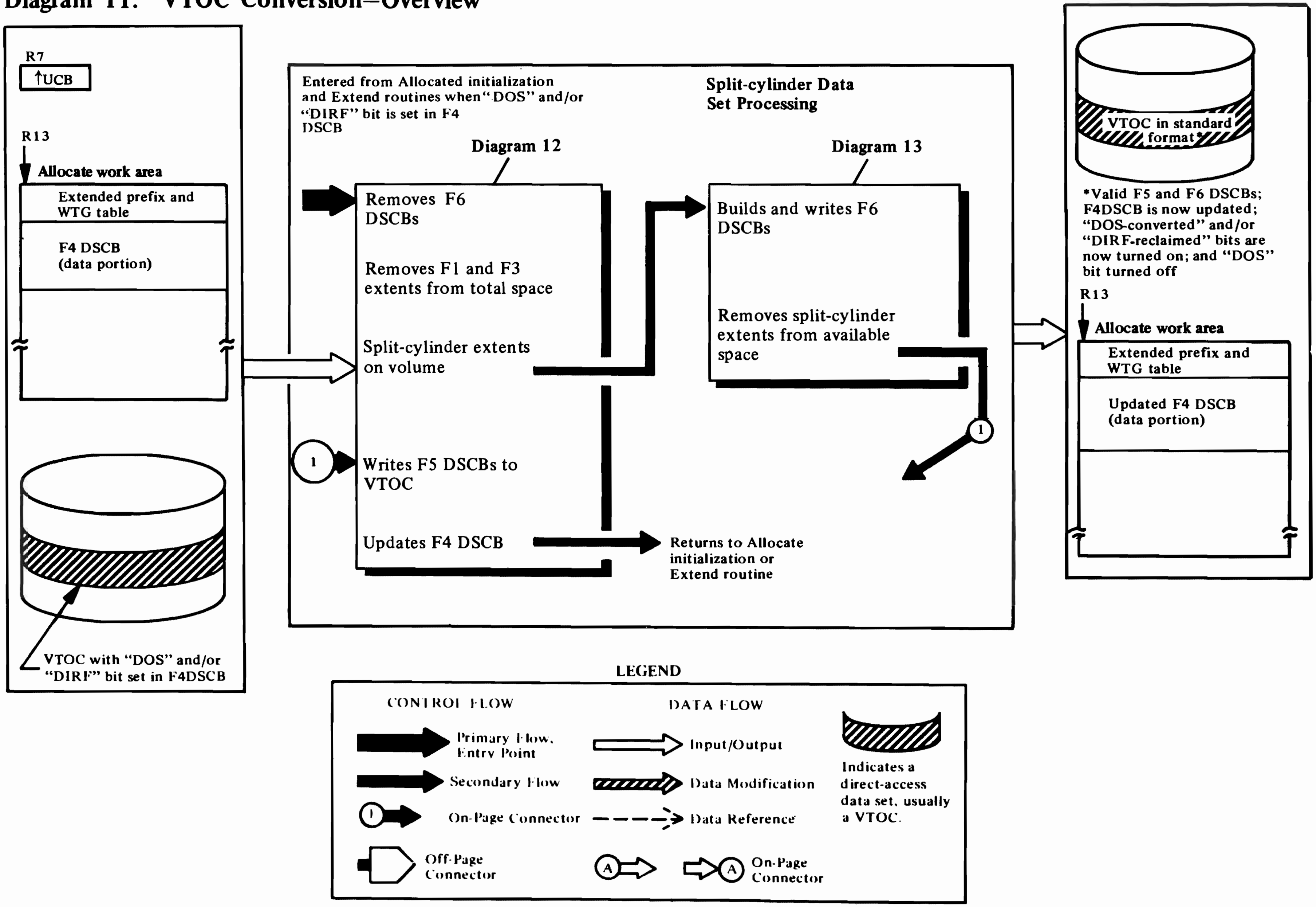
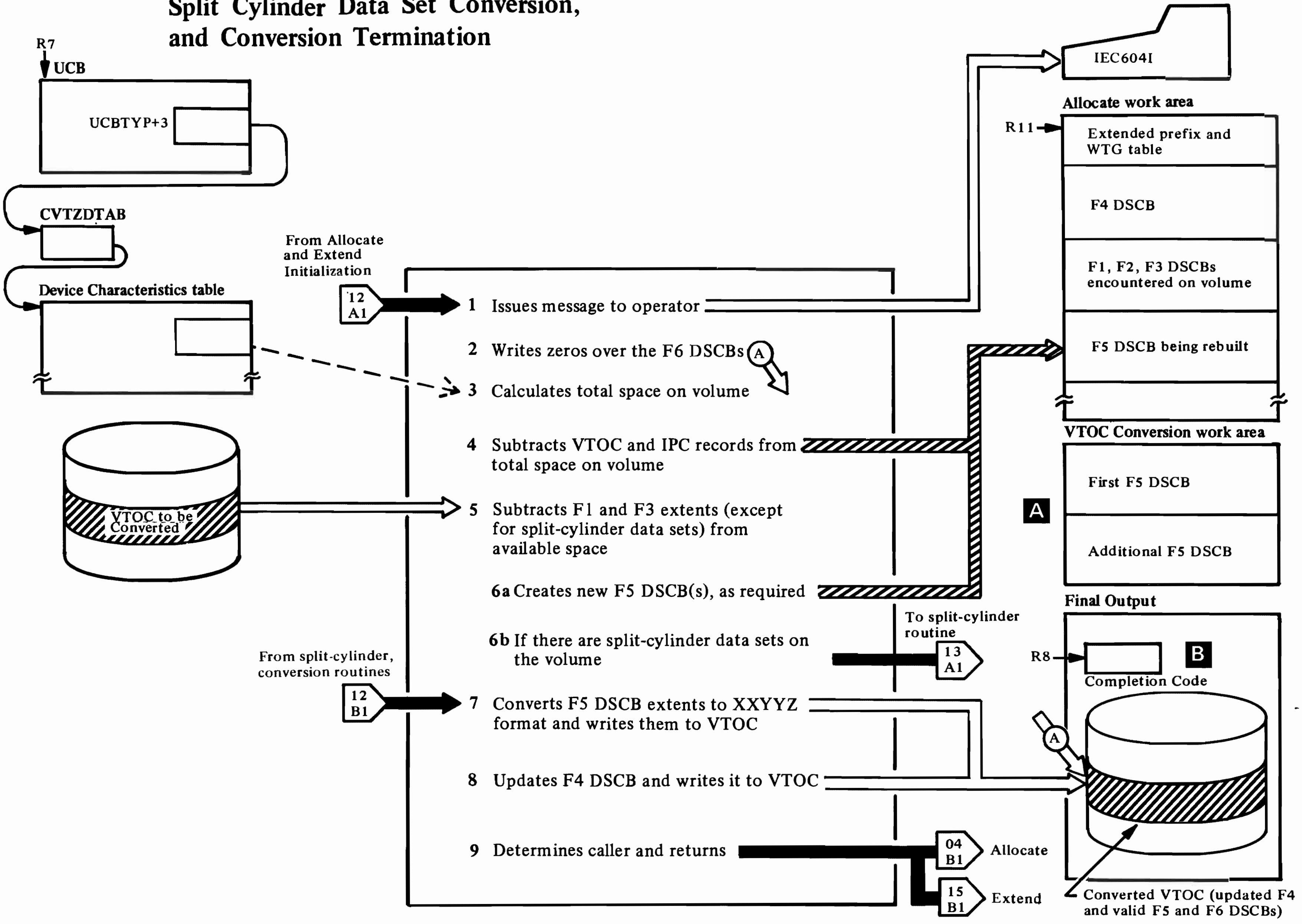

R7
↑UCB
R13
Allocate work area
Extended prefix and WTG table
F4 DSCB (data portion)
VTOC with "DOS" and/or "DIRF" bit set in F4DSCB
Entered from Allocated initialization and Extend routines when "DOS" and/or "DIRF" bit is set in F4 DSCB
Diagram 12
Removes F6 DSCBs
Removes F1 and F3 extents from total space
Split-cylinder extents on volume
1
Writes F5 DSCBs to VTOC
Updates F4 DSCB
Returns to Allocate initialization or Extend routine
Split-cylinder Data Set Processing
Diagram 13
Builds and writes F6 DSCBs
Removes split-cylinder extents from available space
1
VTOC in standard format*
*Valid F5 and F6 DSCBs; F4DSCB is now updated; "DOS-converted" and/or "DIRF-reclaimed" bits are now turned on; and "DOS" bit turned off
R13
Allocate work area
Extended prefix and WTG table
Updated F4 DSCB (data portion)
LEGEND
CONTROL FLOW
Primary Flow, Entry Point
Secondary Flow
1
On-Page Connector
Off-Page Connector
DATA FLOW
Input/Output
Data Modification
Data Reference
A
A
On-Page Connector
Indicates a direct-access data set, usually a VTOC.

Diagram 12. VTOC Conversion–Initialization, Non-Split Cylinder Data Set Conversion, and Conversion Termination



1 Issues message to operator

2 Writes zeros over the F6 DSCBs

3 Calculates total space on volume

4 Subtracts VTOC and IPC records from total space on volume

5 Subtracts F1 and F3 extents (except for split-cylinder data sets) from available space

6a Creates new F5 DSCB(s), as required

6b If there are split-cylinder data sets on the volume

7 Converts F5 DSCB extents to XXYYZ format and writes them to VTOC

8 Updates F4 DSCB and writes it to VTOC

9 Determines caller and returns

R7
UCB
UCBTYP+3
CVTZDTAB
Device Characteristics table
VTOC to be Converted
From Allocate and Extend Initialization
12 A1
From split-cylinder, conversion routines
12 B1
A
IEC604I
Allocate work area
R11
Extended prefix and WTG table
F4 DSCB
F1, F2, F3 DSCBs encountered on volume
F5 DSCB being rebuilt
VTOC Conversion work area
A
First F5 DSCB
Additional F5 DSCB
Final Output
To split-cylinder routine
13 A1
R8
B
Completion Code
A
04 B1
Allocate
15 B1
Extend
Converted VTOC (updated F4 and valid F5 and F6 DSCBs)

Extended Description for Diagram 12

IGG0325Z

1 This function, VTOC Conversion, is entered from the Allocate initialization routine or the Extend initialization routine when either the DOS or DIRF bit is on in the F4 DSCB. These bits are on when (1) an allocation was made on the volume by DOS or (2) an interrupt occurred during an updating of the VTOC, invalidating the contents of the VTOC, or (3) overlapping extents were found in a DSCB. [CONVTEST]

Builds and issues a message (IEC604I) to tell the operator that the VTOC conversion routines have been entered. [BUILDMSG]

Relocates channel program to write F0 DSCBs over F6 DSCBs. Calculates number of DSCBs in the VTOC. [BUILDCHP]

2 Tests for F6 DSCBs on the volume. Writes an F0 DSCB over each F6 DSCB found in the VTOC. [ZEROF6]

3 Calculates the highest track that can be allocated on the volume (as if no data sets existed on the volume). [STARTF5]

4 Determines if the volume is used for initial program loading (IPL) and calculates the lowest track that could be allocated on the volume (as if no data sets existed on the volume). [HIGHTRK]

Subtracts the VTOC extent from the total space on the volume. [CHECK2]

Initializes the channel program to read each F5 DSCB on the volume and gives control to IGG0325P. [INITCHPG]

IGG0325P

5 Tests to determine if every DSCB in the VTOC has been read. If not, it reads the next DSCB. [TESTSW]

Tests to determine if an F1, F2, F3, F5, or F6 DSCB has been read. If an F1, F2, or F3 DSCB has been read, the number of unused DSCBs is decremented. [F1ORF3]

If an F5 DSCB has been read in, determines if there is space in the VTOC Conversion work area (each work area can contain only three F5 DSCBs). Moves the actual device address (CCHHR) of the F5 DSCB to the VTOC Conversion work area. [NEXTADR1]

If an F1 DSCB has been read in, updates the high-water mark. Initializes pointers and counter to convert the F1 extents to RTA1/RTA2 format. [CHECK1]

If an F3 DSCB has been read in, initializes pointers and counters to convert the F3 extents to RTA1/RTA2 format. [CHECK3]

Converts all non-split-cylinder data set extents to RTA1/RTA2 format and enters them in the extent table. [CONVERT]

When the extent table contains five entries, this routine sorts the entries into ascending RTA sequence. [ORDER]

Compares each entry in the extent table with the extents in the F5 DSCBs to find the F5 extent that contains the extent table entry. Subtracts the tracks allocated to the entry from the F5 extent, if this extent comes at the beginning or end of the F5 extent or takes all of the extent. [NEWCOMP]

Subtracts the tracks allocated to the entry from the F5 extent when the extent table entry is in the middle of the F5 extent. This results in the creation of two extents. [TWOEXTS]

When the first F5 is filled (26 extents entered), moves the first rebuilt F5 DSCB to the VTOC Conversion work area, sets up parameters, and passes control to IGG0325R. [PREPPARM]

When all DSCBs have been read, branches to a subroutine that sorts and subtracts any remaining entries in the extent table from the reubuilt F5 DSCBs. If there are split-cylinder data sets on the volume, control is given to IGG0325Q. Otherwise, control is given to IGG0325T. [FINSIHF5]

IGG0325R

6a Compares each entry in the extent table with the extents of the current F5 DSCB. [STARTCOM]

Moves all of the extents in the current F5 forward to wipe out an extent that was totally allocated. Determines if there are chained F5s. [WIPEOUT]

Zeros out the last extent in an F5 or zeros out the last F5 in the chain. [COMPLAST]

Modifies an F5 extent when the allocated extent begins at the beginning of the F5 extent. [REPLACE]

Moves the modified F5 DSCB to the VTOC Conversion work area and finds the next F5 in the work area chain. [WRITENOW]

Finds next F5 in chain and moves an extent from the beginning of this F5 to the end of the previous F5. Moves the previous F5 DSCB to the VTOC Conversion work area. [MERGERTN]

6b Determines if there are more entries in the extent table to be subtracted. If not, moves the first F5 to the Allocate work area and transfers control back to IGG0325W if split-cylinders extents are to be subtracted; otherwise, control is returned to IGG0325P. [UPDATPTR]

Determines if the allocated extent is within the F5 extent. If not, points to the next F5 extent or moves the next F5 in the chain to the Allocate work area. [MORECK]

Modifies an F5 extent when the extent table entry falls in the middle of the F5 extent. Moves the F5 DSCB to the VTOC Conversion work area. Moves the next F5 DSCB in the chain to the Allocate work area and moves the last extent from the previous F5 to the beginning of the next F5. [EXPANDCK]

Modifies an F5 extent when the extent table entry ends at the same place as the F5 extent (that is, their RTA2s are equal). Moves the F5 DSCB to the VTOC conversion work area. [REPLACE2]

Creates a new F5 DSCB and moves the last extent of a previous F5 into it. Moves the new F5 to the VTOC Conversion work area and decrements the number of unused DSCBs. [FINXPAND]

IGG0325T

7 Converts each F5 extent from RTA1/RTA2 to XXYYZ format. [CONVNEXT]

Inserts the F5 indicator bytes and the F5 identification byte. Writes the F5 and decrements the hole count (DS4DSREC field) in the F4 DSCB. [MOVEMORE]

Finds F0 DSCBs over which the rebuilt F5 DSCBs will be written. [FINDF0]

Moves the next F5 in the chain to the Allocate work area. [MOVEF5IN]

Determines whether F0 DSCBs must be written over any F5 DSCBs that have not been overwritten with rebuilt F5 DSCBs. [F0TEST]

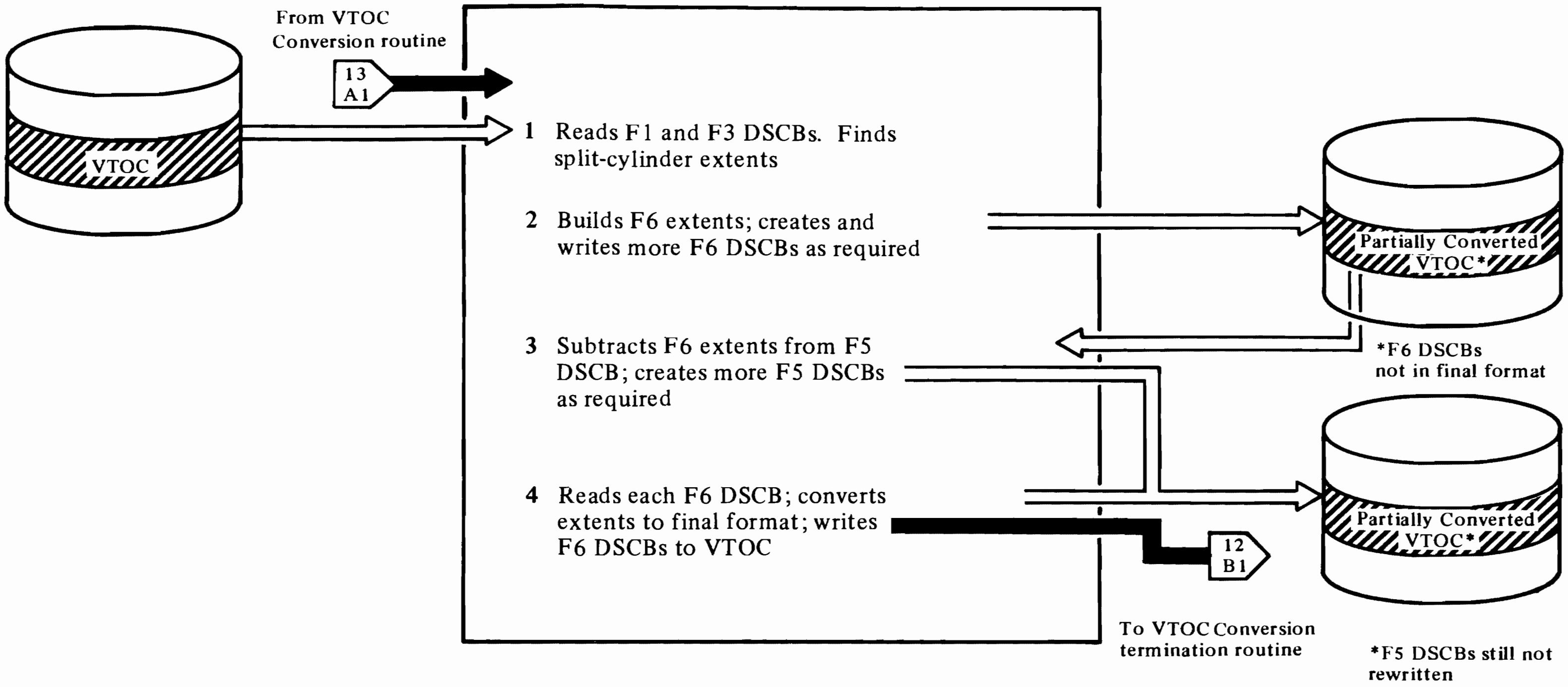
Extended Description for Diagram 12 (Continued)

8 When all the F5 DSCBs have been read, converted, and written back, turns off the DOS bit and/or turns on the *DOS-converted* and/or *DIRF-reclaimed* bits, and writes the F4 DSCB back to the VTOC. [TURNOFF]

A When there is more than one F5 DSCB in the VTOC being converted, when more than one F5 is built during conversion processing, or if split-cylinder data sets are found, a VTOC Conversion work area is obtained. The work area is defined by the IECVTCWA mapping macro, and will contain as many as three F5 DSCBs. All F5 DSCBs are created, modified, and maintained in virtual storage. They are written to the VTOC at the conclusion of conversion processing.

B See "DADSM Allocate Return Codes" for return codes issued and resultant messages.

Diagram 13. VTOC Conversion—Split-Cylinder Data Sets


From VTOC Conversion routine
13 A1
VTOC
Partially Converted VTOC*
*F6 DSCBs not in final format
Partially Converted VTOC*
12 B1
To VTOC Conversion termination routine
*F5 DSCBs still not rewritten

1 Reads F1 and F3 DSCBs. Finds split-cylinder extents

2 Builds F6 extents; creates and writes more F6 DSCBs as required

3 Subtracts F6 extents from F5 DSCB; creates more F5 DSCBs as required

4 Reads each F6 DSCB; converts extents to final format; writes F6 DSCBs to VTOC

Extended Description for Diagram 13

This function is entered from Diagram 12 when a VTOC being converted contains split-cylinder data set allocations.

IGG0325Q

1 Finds the first unused DSCB in the VTOC and initializes the F6 pointer in the F4. [BUILDF6]

Tests to determine if all DSCBs in the VTOC have been read. If not, reads the next DSCB. [READNEXT]

Determines if an F1 or F3 DSCB has been read. If so, it initializes a pointer and a counter to be used in testing for split-cylinder extents. [TESTF1]

Tests to determine if the next extent is a split-cylinder extent. [TESTNEXT]

Converts a split-cylinder extent from CCHH/CCHH to RTA1/RTA2 format. [CONVERT]

If the RTA1/RTA2 was the first split-cylinder extent, builds the first F6. [CONT]

If the RTA1/RTA2 was not the first split-cylinder extent, transfers control to IGG0325U. [XCTL325U]

When all DSCBs have been read and tested for split-cylinder data set extents, writes out the F6 and transfers control to IGG0325W. [XCTL325W]

IGG0325U

2 Compares the RTA1/RTA2 just converted to the F6 extent. If it lies within the F6 extent, transfers control to IGG0325V to check for F6 overlapping. [COMPARE]

If the split-cylinder extent does not lie within the F6 extent, updates pointers and counters to compare it with the next F6 extent or reads in the next F6 in the chain. [NOLAP]

Creates a new F6 with the split-cylinder extent as the first entry. Chains the new F6 to the previous F6 and writes the old and new F6 DSCBs. [NEWF6]

Rereads the first F6 and transfers control back to IGG0325Q. [RD1STF6]

Creates a new F6 extent in the current F6 DSCB if the split-cylinder extent lies between two F6 extents. If this new F6 extent overlaps the F6 following it, control is transferred to IGG0325V. [NEWENTRY]

Moves all F6 extents one extent to the right and writes the F6 to the VTOC. [MOVE]

If the F6 is chained, reads in the next F6 and continues to move each F6 extent to the right. [LASTF6CK]

If the split-cylinder extent lies totally within an existing F6 extent, increments the data set count by one and writes the F6 to the VTOC. [NEWDSCNT]

For a new F6 extent this routine moves the RTA1/RTA2 of the split-cylinder extent into the F6 and initializes the data set count to one. [NEWEXTST]

IGG0325V

Tests to determine if the newly created F6 extents end before the next F6 extent begins. Reads in the next F6 extent if this is the last extent in the current F6. [AGAIN]

If the new F6 extent did not overlap another F6 extent or after all overlapping F6 extents have been combined into one extent, this section restores the original F6 and transfers control back to IGG0325U. [RESTORE]

Combines two F6 extents. [COMBINE]

Moves all extents in the current F6 forward to wipe out the extent that was just combined. [WIPEOUT]

Writes a F0 over the last F6, zeros out the F6 chaining pointer in the previous F6, and writes out this F6. [COMPLAST]

Tests to determine if the split-cylinder extent ends within the updated F6 extent. [LASTCOMP]

Reads in next F6 in chain and moves its first extent into the end of the previous F6. Writes the previous F6 to VTOC. [MERGERTN]

IGG0325W

3 Moves the first F5 DSCB into the Allocate work area. [BUILDTBL]

Adds an F6 extent to the extent table. [ADDENTRY]

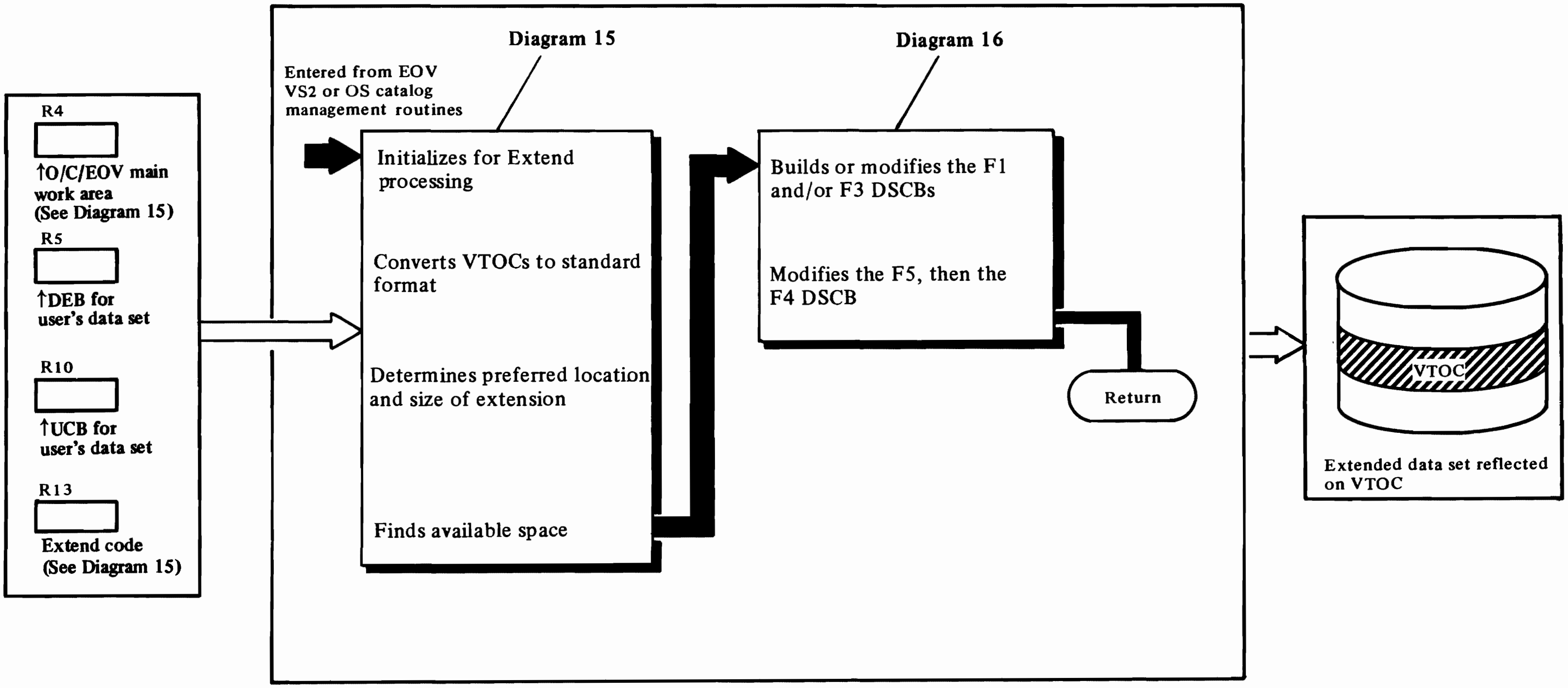
Initializes pointers and counters and transfers control to IGG0325R to subtract the entries in the extent table from the free space in the F5 DSCBs. [SUBTRACT]

Updates pointers to the next F6 extents to be added to the extent table. Reads in the F6 when all the extents from the previous F6 have been added. After all F6 extents have been subtracted from the F5 DSCBs, it rereads the first F6. [RETURNPT]

4 Converts the F6 extents from RTA1/RTA2/number of data sets to RTA/number-of-shared-cylinders/number-of-data-sets format. [FINISHF6]

Inserts the F6 indicator bytes and the F6 identifier byte. Writes the F6 and reads the next F6 in the chain. After all the F6 DSCBs have been converted, transfers control to IGG0325T. [MOVE]

Diagram 14. Extend Overview


R4
↑O/C/EOV main work area (See Diagram 15)
R5
↑DEB for user's data set
R10
↑UCB for user's data set
R13
Extend code (See Diagram 15)
Diagram 15
Entered from EOV VS2 or OS catalog management routines
Initializes for Extend processing
Converts VTOCs to standard format
Determines preferred location and size of extension
Finds available space
Diagram 16
Builds or modifies the F1 and/or F3 DSCBs
Modifies the F5, then the F4 DSCB
Return
VTOC
Extended data set reflected on VTOC

LEGEND

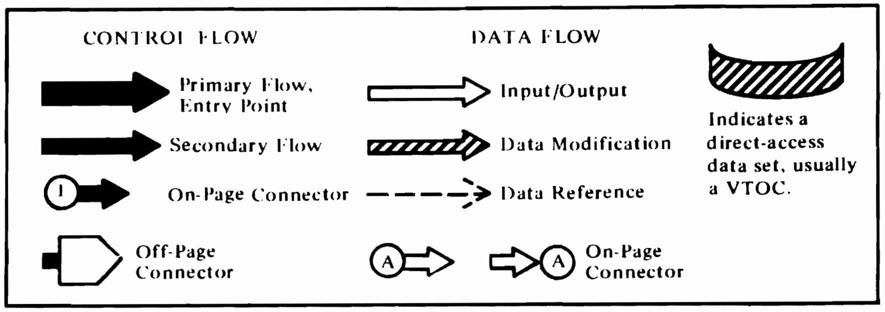
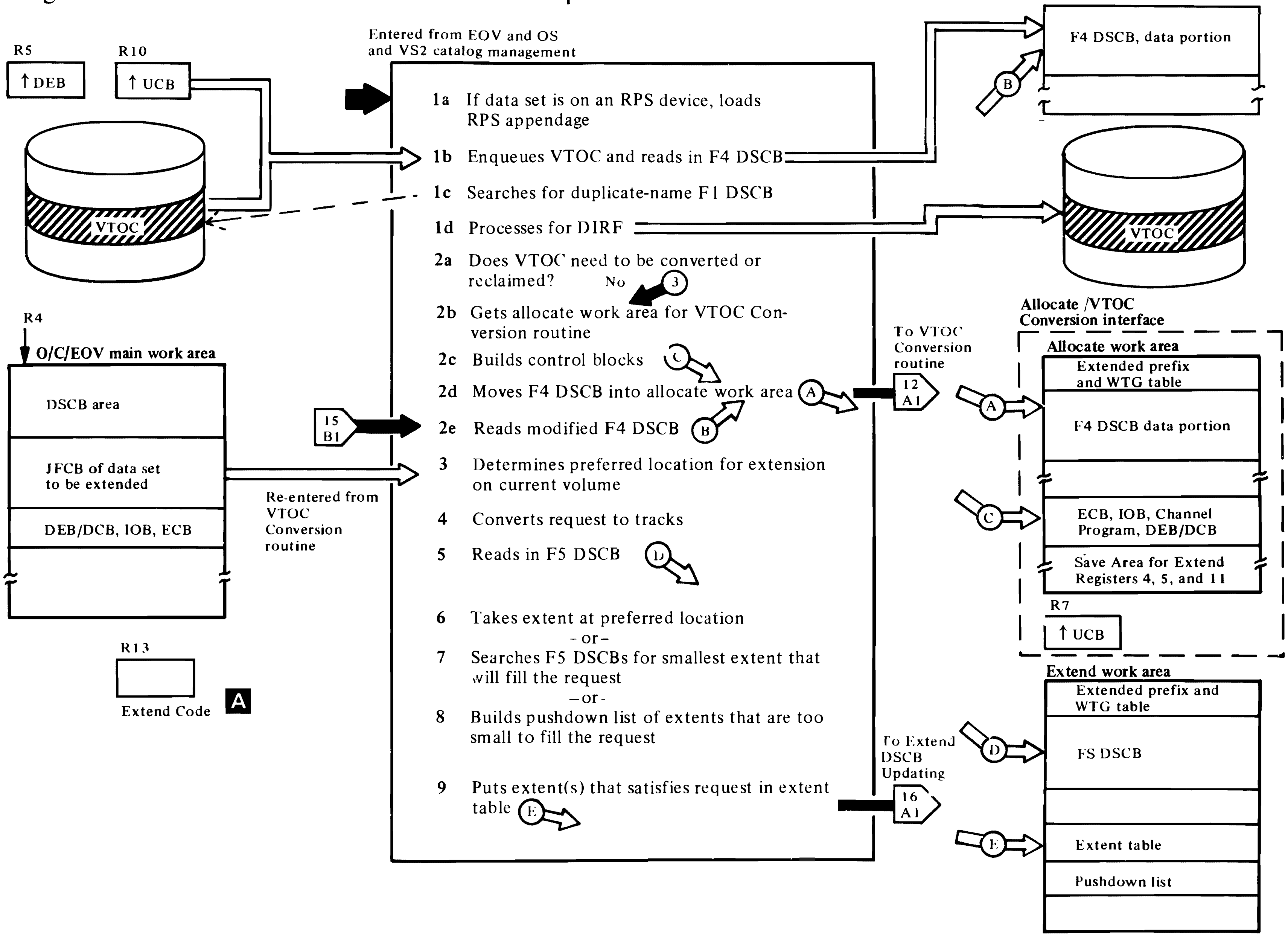


Diagram 15. Extend–Initialization and Free Space Search



Entered from EOV and OS and VS2 catalog management

1a If data set is on an RPS device, loads RPS appendage

1b Enqueues VTOC and reads in F4 DSCB

1c Searches for duplicate-name F1 DSCB

1d Processes for DIRF

2a Does VTOC need to be converted or reclaimed? No 3

2b Gets allocate work area for VTOC Conversion routine

2c Builds control blocks C

2d Moves F4 DSCB into allocate work area A

2e Reads modified F4 DSCB B

3 Determines preferred location for extension on current volume

4 Converts request to tracks

5 Reads in F5 DSCB D

6 Takes extent at preferred location
-or-
7 Searches F5 DSCBs for smallest extent that will fill the request
-or-
8 Builds pushdown list of extents that are too small to fill the request

9 Puts extent(s) that satisfies request in extent table E

R5
↑DEB
R10
↑UCB
VTOC
R4
O/C/EOV main work area
DSCB area
JFCB of data set to be extended
DEB/DCB, IOB, ECB
R13
Extend Code
A
15
B1
Re-entered from VTOC Conversion routine
To VTOC Conversion routine
12
A1
To Extend DSCB Updating
16
A1
O/C/EOV main work area
F4 DSCB, data portion
B
VTOC
Allocate /VTOC Conversion interface
Allocate work area
Extended prefix and WTG table
A
F4 DSCB data portion
C
ECB, IOB, Channel Program, DEB/DCB
Save Area for Extend Registers 4, 5, and 11
R7
↑UCB
Extend work area
Extended prefix and WTG table
D
F5 DSCB
E
Extent table
Pushdown list

Extended Description for Diagram 15

IGG0553A

1a Entered (1) from EOV when a data set is to be extended, (2) from VS2 catalog management when a data space is to be extended, or (3) from OS catalog management when a catalog is to be extended. Gets the Extend work area. [BEGIN]

Tests for the RPS feature. If present, gets 128-byte RPS work area and copies the appendage vector table (AVT) into it. Loads module IGG019EK and enters its address in the AVT. Modifies the DEB in the O/C/EOV work area to point to the RPS AVT. [INITRRPL]

1b Links to resident conversion routine to convert VTOC address from relative track address to absolute (CCHHR). Relocates channel program to read the F4 DSCB and to search for a duplicate-name F1 DSCB. Enqueues the VTOC. If extending the data set on the current volume, the CCWs are altered so that no search of the F1 DSCBs for a duplicate name is made. [NORPS]

1c Tests for invalid, duplicate-name F1 DSCB, for a volume containing a data set allocated by DOS, or for a VTOC containing invalid information, resulting from a previous interrupt (DIRF bit set). When a DOS volume or a volume on which the DIRF bit was set (system failed during VTOC updating, a permanent I/O error occurred during the reading or writing of a DSCB, or overlapping extents were found in a DSCB) is encountered, control is transferred to the VTOC conversion interface routine (IGG0553F). [SKIPWR]

1d Processes for DIRF: sets/resets the DIRF bit, and if no previous interrupt has occurred, writes the F4 back to the VTOC with the DIRF bit set. If a previous interrupt has occurred, the F4 is not written back. The DOS and DIRF bits are bits 0 and 5 of the DS4VTOCI field of the F4 DSCB. [EXECUTET]

IGG0553F

2a This routine is entered (1) when a data set is to be extended on a volume containing a data set allocated by DOS, (2) when the DIRF bit is set in the F4 DSCB indicating an invalid VTOC caused by a previous interrupt during DADSM processing, or (3) when overlapping extents are encountered in a DSCB. The VTOC must be converted to standard format or reclaimed by the VTOC conversion routines before the data set can be extended. This routine is also re-entered when the VTOC conversion is complete.

2b This section of code tests for first or second entry; if first entry, obtains an Allocate work area.

2c Builds the DEB/DCB, IOB, and ECB in the work area.

2d The F4 DSCB for the volume is moved into this work area from the O/C/EOV work area and control is transferred to IGG0325Z to convert the VTOC. [BEGIN]

2e This section of code is entered when this module is entered for the second time (after the VTOC has been converted to standard format or reclaimed). Restores Extend registers. Tests for errors in conversion processing, and, if there were none, reads the modified F4 DSCB into the O/C/EOV work area, repeats the search for a duplicate-name F1 DSCB, and passes control to IGG0553G. If an error has occurred, passes control to IGG0553E. [ERRORTST]

IGG0553G

3 Verifies that there are at least three free VTOC records (F0 DSCBs). If extending on current volume, sets up to calculate the last track written. [BEGIN]

4 For current-volume allocations, converts last track used by the data set to be extended from relative to absolute address (CCHHR). This track is used as the preferred location for the extension. [COMPRTA]

Converts primary quantity request in cylinders to tracks. [CYLTRKRC]

For requests in records, converts records to tracks and calculates overhead. [INRECORD]

If the request specifies rounding to cylinder boundary, this routine increments to next cylinder boundary. [ROUNDUP]

Adjusts space request for user labels, if requested. [CLEANUP]

IGG0553B

5 Relocates a channel program to read in the first F5 DSCB. [BEGIN]

Converts the extent in an F5 DSCB from XXYYZ format to RTA and a number of tracks. [SETUP]

Tests for last (twenty-sixth) F5 extent and, if it is the last extent, determines whether there is another F5. [TESTLOC]

6 Begins the search of the F5 DSCBs for space to fill the request by determining whether the F5 extent examined is the preferred extent and whether enough space is available at that location. [BEGSURCH]

7 Compares extent just found, which is larger than request, with extent being held to determine which is closest in size to the request. If the new extent is closer in size, it becomes the "held" extent. [COMPARE]

8 If no extent has been held, compares extent just found, which is smaller than request, with the smallest of the five entries in the pushdown list and saves the larger extent. It then arranges the modified pushdown list in order of decreasing extent size. [QTNOTAVL]

9 When quantity and preferred track are available, builds a one-entry table. [ONEDADSM]

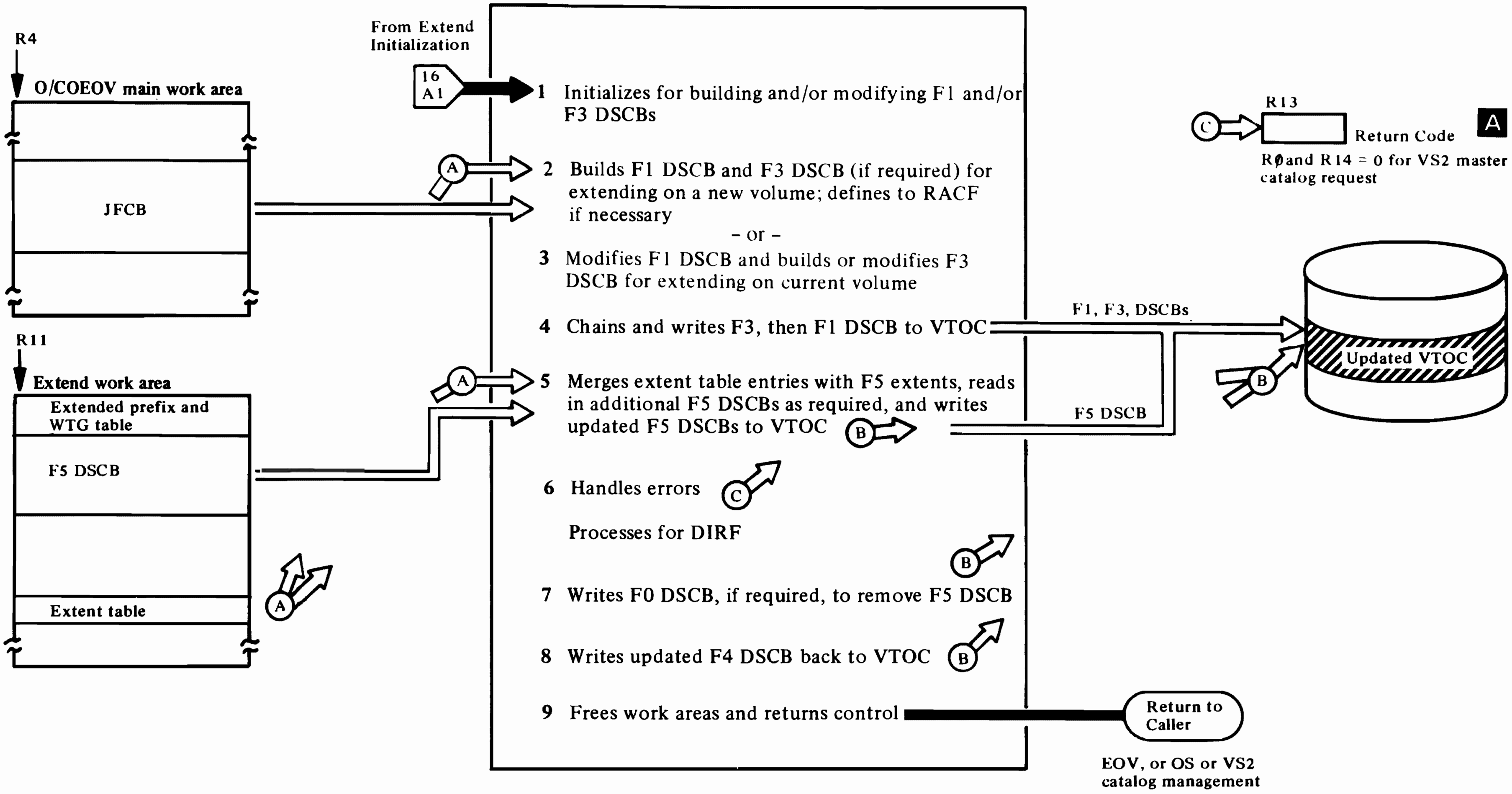
If an extent has been held (that is, an extent has been found equal to or larger than the request), builds a one-entry extent table. [FILLIT]

The extents in the pushdown list are examined to determine whether two or more of the extents can be used to fill the request. If so, the extents that satisfy the request are entered in the extent table. [PICKLIST]

A Extend Code:

Code	Meaning
01	Extend a data set on current volume (EOV request).
02	Extend the OS catalog on the current volume (OS catalog management request).
04	Extend a data set on a new volume (EOV request).
129(X'81')	Extend a data space on the current volume (VSAM catalog request).

Diagram 16. Extend–DSCB Updating



1 Initializes for building and/or modifying F1 and/or F3 DSCBs

2 Builds F1 DSCB and F3 DSCB (if required) for extending on a new volume; defines to RACF if necessary

- or -

3 Modifies F1 DSCB and builds or modifies F3 DSCB for extending on current volume

4 Chains and writes F3, then F1 DSCB to VTOC

5 Merges extent table entries with F5 extents, reads in additional F5 DSCBs as required, and writes updated F5 DSCBs to VTOC

6 Handles errors

Processes for DIRF

7 Writes F0 DSCB, if required, to remove F5 DSCB

8 Writes updated F4 DSCB back to VTOC

9 Frees work areas and returns control

Extended Description for Diagram 16

IGG0553C

1 If the task that requires more direct-access space has not obtained step-must-complete status (by issuing ENQ with SMC=STEP or STATUS macro), this routine branches to the STATUS routine to obtain step-must-complete status. [BEGIN]

2 Reads F1 DSCB to virtual storage if extending on current volume. [READF1]

Determines whether extents to be added will bring total for data set to more than 16. If so, returns to caller via last load. Determines whether extent(s) can be added to the F1 and whether an F3 needs to be built. [NVL]

If an F3 DSCB already exists for this data set, this routine reads it into extend work area. [NOTF1]

For extensions on new volumes, builds a new F1 from the JFCB. [BUILDF1]

If the data set is RACF-indicated (DEBXDSSI is on), issues RACDEF TYPE=ADDVOL to add new volume to data set definition. If R15=0, successful; if R15 is not zero, sets RACF error indicator (DXEXTSW=RACERROR) and exits to IGG0553E. [NOSECUR]

3 Converts RTA1/RTA2 from extent table to CCHHCCHH and moves converted extents to the F1 or F3 DSCB being built or modified. Accounts for user label track. [STLOOP]

4 If no F3 exists for this data set, this routine searches for an F0 DSCB. If there is already an F3 on the volume or after an F0 DSCB is found, this routine writes the F3 to the VTOC. [WROUTF3]

Chains F1 DSCB to F3 DSCB if an F3 exists, sets number of extents in the F1. If extending on a new volume, this routine searches the VTOC for an F0 DSCB, over which the new F1 can be written. Writes the F1 DSCB to the VTOC. [WRITEF1]

Modifies channel program to read and write the F5 and F4 DSCBs. [MODIFYCP]

IGG0553D

5 Gets F5 input work area address, initializes pointers and counters, and modifies channel program to read first F5 DSCB. Reads first F5 DSCB. [DADSMSTR]

Determines when and where extent table should be merged into the output for the F5 DSCB. [DUELOOP]

Merges the extent table entries with the F5 DSCB extents if allocation is made from the beginning or middle of an F5 extent. [ALTEREX]

Merges the extent table entries with the F5 extent when the allocation is made from the end of the F5 extent or if all of the extent is allocated. [TOTALL]

Converts F5 extents from RTA1/RTA2 to XXYYZ format and writes the F5 to the VTOC. [WRITEF5]

Determines whether there is more than one F5 DSCB on the VTOC. [CHECHAIN]

Writes the last F5 to VTOC. [WRITEFNL]

Finds an F0 DSCB over which the new F5 DSCB can be written. [CHEXT]

If more F5 DSCBs were read from the VTOC than were written back, sets an indicator to cause the next load (F4 DSCB updating routine) to write an F0. [TSTZERO]

When the second and subsequent F5 DSCBs are to be read into virtual storage, resets input pointers and extent counter, then initiates the read operation. [READF5]

IGG0553E

6 Resets DIRF switch and tests DIRF bit in the F4 DSCB. Checks for permanent I/O error passed from previous routines. [CHKIOERR]

7 Determines whether an F0 DSCB needs to be written to the VTOC. If so, modifies the channel program and writes the F0 DSCB. [START]

8 Updates the F4 DSCB count and writes the F4 DSCB to the VTOC. [WRITEF4]

9 Dequeues the VTOC. If a permanent I/O error occurred while reading from or writing to the VTOC during Extend processing, issues message IEC603I. Frees the Extend work area(s) (if RPS feature is present on the device, restores the work area DEB) and determines whether to exit to EOV, to VS2 catalog management, or OS catalog management. Loads appropriate return code in register 13. [EXITHERE]

For normal return to EOV or VSAM, loads the address of the SVRB into register 15, restores the calling routine's registers (2 through 12) and transfers control to the catalog management routine using the XCTL macro or to EOV using the IECRES macro. [TESTRTRN]

A Return Code

- **To EOV:**

  1(1) successful secondary allocation on current volume

  4(4) successful secondary allocation on new volume

  -1(-1) error in secondary allocation on current volume
  - fewer than three free VTOC records (format-0 DSCBs)
  - quantity not available
  - too many extents (more than 16) for the data set on this volume

  -4(-4) error in secondary allocation on the new volume
  - duplicate name on the new volume
  - fewer than three free VTOC records (format-0 DSCBs)
  - quantity not available

  -8(-8) unable to convert VTOC to standard format or a volume with the alternate tracks on cylinder 0 was encountered (DOS allocation)

  -12(-C) Unable to define new volume to RACF

  -16(-10) permanent I/O error

- **To VS2 Catalog Management:**

  129(81) successful secondary allocation on current volume

  -129(-81) error in secondary allocation on new volume
  - fewer than three free VTOC records (format-0 DSCBs)
  - quantity not available
  - too many extents (more than 16) for the data space on this volume

- **To OS Catalog Management:**

2(2) successful secondary allocation on current volume

-2(-2) error in secondary allocation on current volume

- fewer than three free VTOC records (format-0 DSCBs)
- quantity not available
- too many extents (more than 16) for the data set on this volume

-8(-8) unable to convert VTOC to standard format or a volume with the alternate tracks on cylinder 0 was encountered (DOS allocation)

-16(-10) permanent I/O error

Diagram 17. Scratch Overview

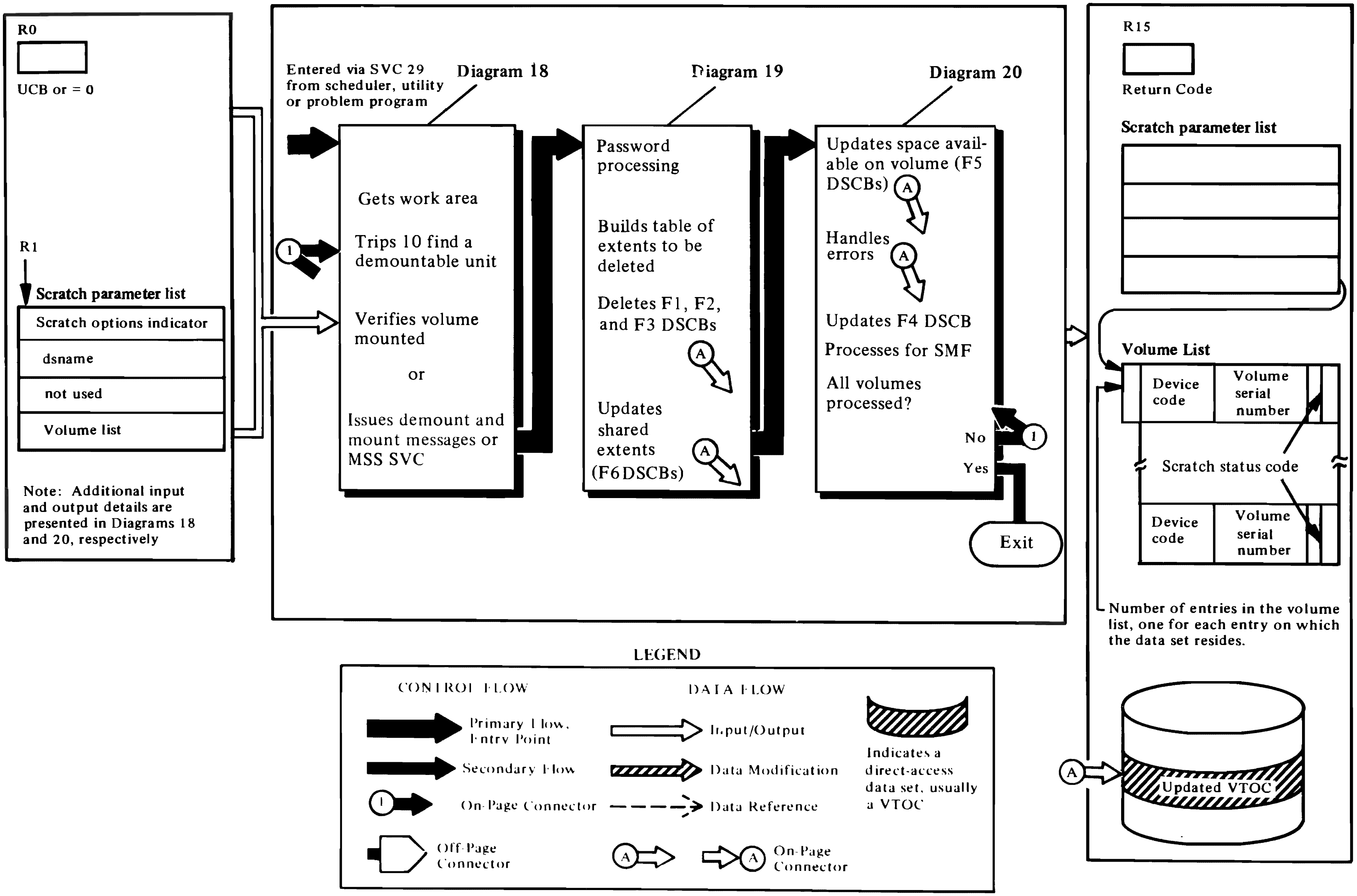
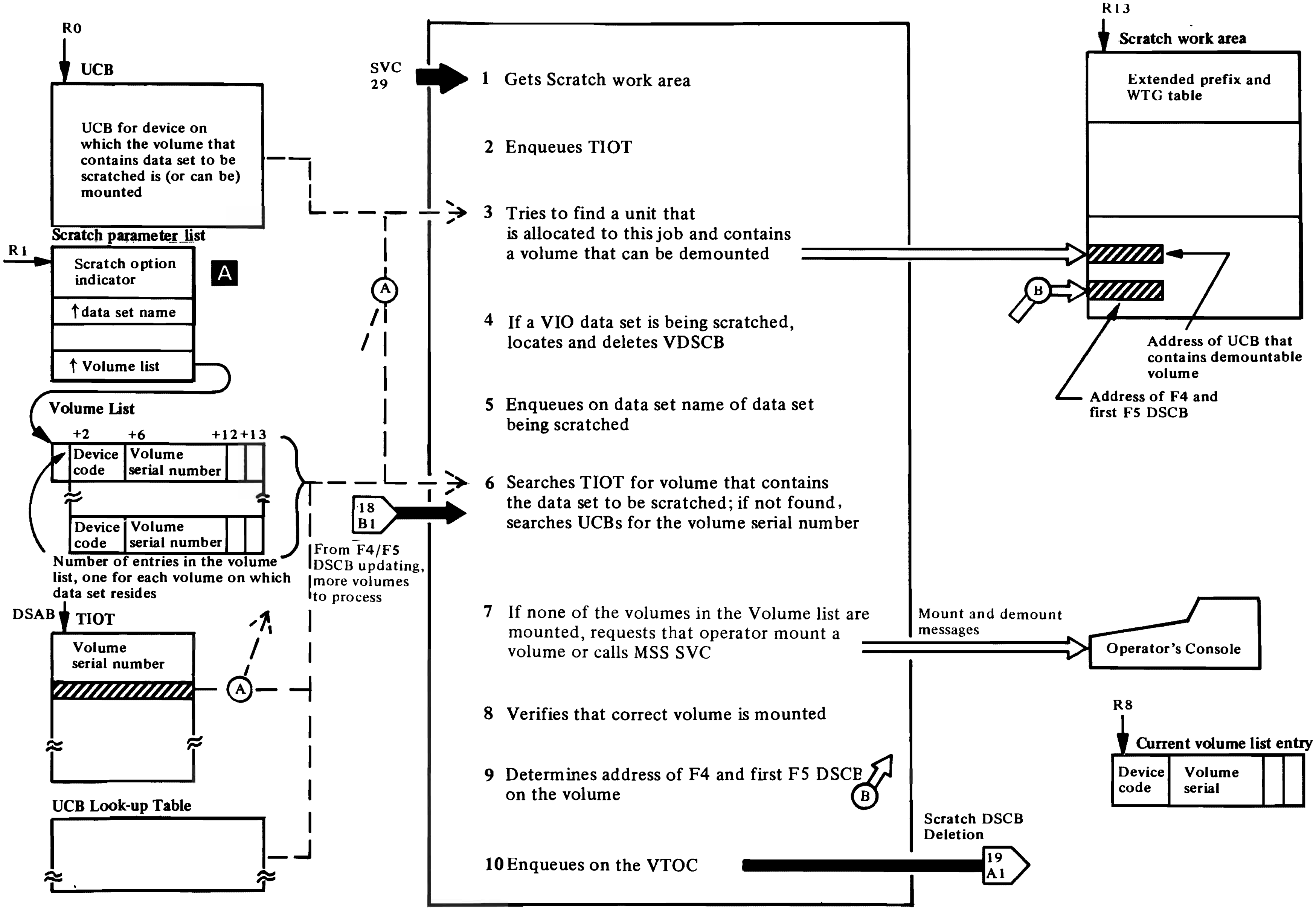

R0
UCB or = 0
R1
Scratch parameter list
Scratch options indicator
dsname
not used
Volume list
Note: Additional input and output details are presented in Diagrams 18 and 20, respectively
Entered via SVC 29 from scheduler, utility or problem program
Diagram 18
Gets work area
Trips 10 find a demountable unit
Verifies volume mounted
or
Issues demount and mount messages or MSS SVC
Diagram 19
Password processing
Builds table of extents to be deleted
Deletes F1, F2, and F3 DSCBs
Updates shared extents (F6DSCBs)
Diagram 20
Updates space available on volume (F5 DSCBs)
Handles errors
Updates F4 DSCB
Processes for SMF
All volumes processed?
No
Yes
Exit
1
A
R15
Return Code
Scratch parameter list
Volume List
Device code
Volume serial number
Scratch status code
Device code
Volume serial number
Number of entries in the volume list, one for each entry on which the data set resides.
Updated VTOC
LEGEND
CONTROL FLOW
Primary Flow, Entry Point
Secondary Flow
On-Page Connector
Off-Page Connector
DATA FLOW
Input/Output
Data Modification
Data Reference
On-Page Connector
Indicates a direct-access data set, usually a VTOC

Diagram 18. Scratch–Initialization, Volume Mounting and Volume Verification



1 Gets Scratch work area

2 Enqueues TIOT

3 Tries to find a unit that is allocated to this job and contains a volume that can be demounted

4 If a VIO data set is being scratched, locates and deletes VDSCB

5 Enqueues on data set name of data set being scratched

6 Searches TIOT for volume that contains the data set to be scratched; if not found, searches UCBs for the volume serial number

7 If none of the volumes in the Volume list are mounted, requests that operator mount a volume or calls MSS SVC

8 Verifies that correct volume is mounted

9 Determines address of F4 and first F5 DSCB on the volume

10 Enqueues on the VTOC

R0
UCB
UCB for device on which the volume that contains data set to be scratched is (or can be) mounted
Scratch parameter list
R1
Scratch option indicator
A
↑data set name
↑ Volume list
Volume List
+2
+6
+12
+13
Device code
Volume serial number
Device code
Volume serial number
Number of entries in the volume list, one for each volume on which data set resides
DSAB
TIOT
Volume serial number
A
A
UCB Look-up Table
SVC 29
18 B1
From F4/F5 DSCB updating, more volumes to process
B
B
R13
Scratch work area
Extended prefix and WTG table
Address of UCB that contains demountable volume
Address of F4 and first F5 DSCB
Mount and demount messages
Operator's Console
R8
Current volume list entry
Device code
Volume serial
Scratch DSCB Deletion
19 A1

Extended Description for Diagram 18

This function is entered from the SVC SLIH as a result of the issuance of SVC 29, which is part of the expansion of the SCRATCH macro. The purpose of this function is to delete data sets and VSAM data spaces.

IGC0002I

1 Issues the IECRES GET macro to get the Scratch work area (see *OS/VS2 Open/Close/EOV Logic* for a description of the IECRES macro). [BEGIN]

Checks for a caller executing in supervisor state or in a system storage protection key (key 0 through 7). If neither, this routine branches to the resident address validation routine (IEA0VL00) to verify that the volume list passed by the caller has the same protection key as indicated in the TCB. [VALCK]

2 Before enqueuing on the TIOT, checks the Scratch option indicator in the Scratch parameter list to determine if the scratch request has been issued by the dynamic allocation component, which will have already enqueued on the TIOT. If the request is not from dynamic allocation, this routine attempts to enqueue on the TIOT. If this resource is not available, the data set is in the process of being opened or closed, and therefore cannot be scratched. [BYPASS]

3 When the caller passes the address of a UCB in register 0, searches the DSAB chain for a TIOT DD entry that contains this UCB address. This search will verify that the unit associated with the UCB is allocated to the job. If the unit is allocated to this job, this routine verifies that the volume on the device is eligible for demounting. [USERSUCB]

This routine is entered (1) when no UCB address is passed, (2) when the UCB address passed represents a device that is not allocated to the job step, and (3) when the UCB address represents a volume that is not eligible for demounting. The function of this routine is to search the DSAB chain for a TIOT entry whose UCB contains the volume serial number of a volume in the volume list (to determine if any volume in the volume list is mounted on a unit allocated to this job). If so, this routine verifies that the UCB represents a volume that is on-line and available for demounting. [FINDUNIT]

4 This routine is entered when a VIO data set is being scratched. A VIO data set allocation request is identified by (1) the address of a virtual UCB is passed in register 0 or (2) the current volume list entry is a serial number of a virtual volume and the data set name is temporary. The function of this routine is to search the DSABs for a TIOT entry containing (1) a virtual UCB whose virtual DSCB contains the same name as the data set being scratched and (2) the volume serial number in the virtual UCB that matches the volume serial number in the current entry of the volume list. If a match is found and if the data set has ever been open, this routine issues the VSCRATCH macro and zeros the data portion of the virtual DSCB. [VIODSTST]

5 Attempts to enqueue on the resource named SYSDSN. If the resource is not available for exclusive control, the data set is currently allocated to this or another address space and cannot be scratched. [DSNAMENQ]

If there are no entries in the volume list, if the validity check in step 1 fails, if the data set to be scratched is allocated, or if the VDSCB cannot be found, Scratch processing is terminated by branching to IGG0290D. [ERREXIT]

IGG0290E

6 Builds a DEB/DCB and an IOB with the information required by IOS to read from and write to the VTOC of the volume from which the data set is to be deleted. [BLDBLKS]

Searches the DSAB chain for a TIOT entry whose UCB contains the volume serial number of the current volume-list entry. [FINDUCB]

If the current entry in the volume list is not mounted on a device allocated to the job, searches the UCB look-up table for a UCB with a volume serial number that matches the volume serial number of the current entry in the volume list. [TABLSRCH]

If the current volume list entry was found on an unallocated unit, builds the request blocks required to dynamically allocate the unit and issues the DYNALLOC macro to request that the unit be allocated to this job. [ALLOCVOL]

If the volume was not mounted on any unit or if the dynamic allocation request was unsuccessful, sets up to request that the volume be mounted on an available, demountable unit. [MOUNTVOL]

IGG0290F

7 Determines if the unit associated with the UCB is eligible for demounting. [MNT00100]

If another volume is to be mounted after the current volume is demounted, enqueues on the volume serial number of the next volume. [MNT00200]

Calls a subroutine that builds the demount message IEC502E, then completes the message and calls another subroutine that issues the message. If the volume is an MSS virtual volume, this module calls a subroutine which builds a parameter list and issues the macro ICBMNTDE to demount the volume. [MNT00400]

Builds the demount message. If the "MONITOR SPACE" command has been issued or if demounting activity is being monitored for SMF, this routine issues SVC 78 (LSPACE, see Diagram 26) to gather information about the space available on the volume for the demount message or for recording in the type-19 SMF data set record. This routine also issues SVC 98 (VOLSTAT) to write a volume statistics record on the SYS1.LOGREC data set. [MNT11000]

Calls a subroutine that sets the mount bit in the UCB. Builds and issues a mount message IEC601D or, if the UCB indicates both an MSS virtual volume and a JES3-managed unit, builds an SSOB and passes it to JES3 via the IEFSSREQ macro for mounting approval. Upon approval, issues SVC 126 by use of the ICBMNTDE macro to mount the virtual volume. Upon disapproval, loads the return code of 8 into register 15, dequeues from the volume serial number (resource name is SYSZVOLS), and returns to the calling routine. [MNT03000]

If the UCB indicates an MSS virtual volume that is not JES3-managed, issues SVC 126 by use of the ICBMNTDE macro to mount the virtual volume. [MNT03005]

Also issues a TPUT macro to write message IEC108I to the TSO terminal, unless it is an MSS virtual volume in which case the message is ignored. [MNT03000]

8 Builds and executes a channel program to read the volume label (record 3 of track 0, cylinder 0), then calls a subroutine that resets the mount bit in the UCB. [MNT04000]

If the operator mounted the wrong volume, branches to a subroutine that builds and issues a message telling the operator that the volume is not accepted. In the case of MSS virtual volumes, the message indicates that a hardware error occurred. [MNT04200]

Calls a subroutine that converts the actual device address (CCHHR) of the VTOC to a relative track

Extended Description for Diagram 18 (Continued)

address (TTR), then moves the TTR of the VTOC and the volume serial number to the UCB. [MNT05000]

Updates the volume disposition in the UCB for the newly mounted volume. [MNT05100]

Loads the return code in register 15, dequeues from the volume serial number (resource name is SYSZVOLS), and returns to the calling routine. [MNT01500]

IGG0290E

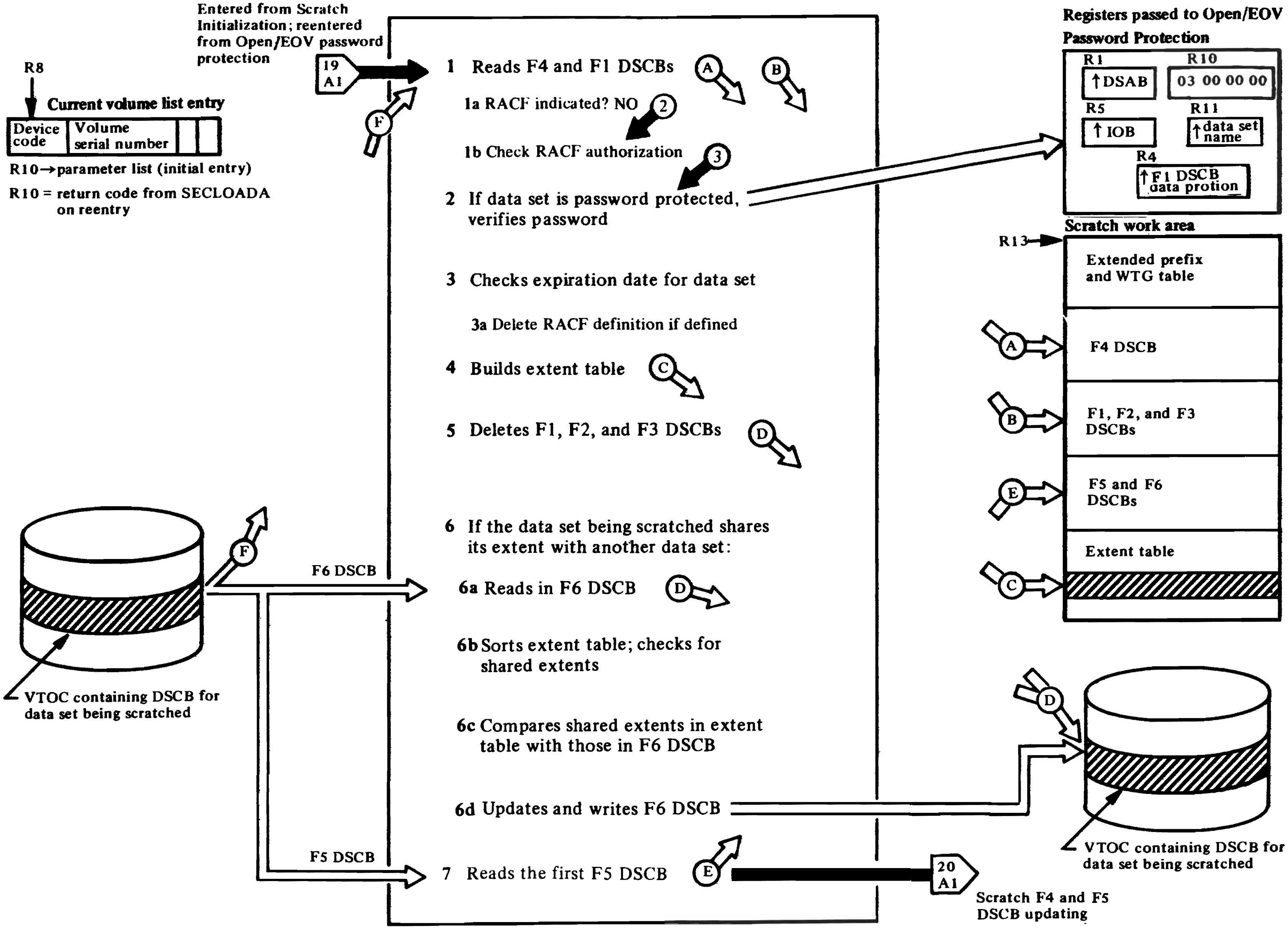
9 When the volume is mounted, tests to see if the device has the RPS feature. If the RPS feature is present, gets a work area to be used by the RPS appendage, loads the appendage (IGG019EK), and puts its address in the appendage vector table (AVT). [MOUNTED]

10 This routine enqueues on the VTOC using the RESERVE macro with the "set-must-complete" option. [ISSUENQ]

A Scratch Option Indicator:

Indicator	Meaning
41 00 50 00 —	override expiration date
41 00 40 00 —	no override of expiration authorized
41 80 D0 00 —	override expiration date, TIOT enqueued by dynamic UNALLOCATION
41 80 C0 00 —	no override authorized, TIOT enqueued by dynamic allocation
41 80 50 00 —	override expiration date, request from job or step termination component
41 80 40 00 —	no override of expiration data authorized, request from job or step termination component

Diagram 19. DSCB Deletion and Split-cylinder DSCB Updating



1 Reads F4 and F1 DSCBs

1a RACF indicated? NO

1b Check RACF authorization

2 If data set is password protected, verifies password

3 Checks expiration date for data set

3a Delete RACF definition if defined

4 Builds extent table

5 Deletes F1, F2, and F3 DSCBs

6 If the data set being scratched shares its extent with another data set:

6a Reads in F6 DSCB

6b Sorts extent table; checks for shared extents

6c Compares shared extents in extent table with those in F6 DSCB

6d Updates and writes F6 DSCB

7 Reads the first F5 DSCB

Entered from Scratch Initialization; reentered from Open/EOV password protection
19 A1
R8
Current volume list entry
Device code
Volume serial number
R10→parameter list (initial entry)
R10 = return code from SECLOADA on reentry
A
B
F
2
3
C
D
E
VTOC containing DSCB for data set being scratched
F6 DSCB
F5 DSCB
Registers passed to Open/EOV Password Protection
R1
↑DSAB
R10
03 00 00 00
R5
↑ IOB
R11
↑data set name
R4
↑F1 DSCB data protion
Scratch work area
R13
Extended prefix and WTG table
F4 DSCB
F1, F2, and F3 DSCBs
F5 and F6 DSCBs
Extent table
VTOC containing DSCB for data set being scratched
20 A1
Scratch F4 and F5 DSCB updating

Extended Description for Diagram 19

IGG0290A

1 This routine is entered from Scratch initialization and reentered from SECLOADA; the entry condition is determined from the contents of register 10.

(1) If this is the first entry to this module, register 10 will contain a pointer to the Scratch parameter list

(2) If reentered from SECLOADA, register 10 can contain one of the following:

00 00 00 00	—	Password verified
10 00 00 00	—	Password error
20 00 00 00	—	No PASSWORD data set on SYSRES

Tests for an entry from SECLOADA. If this is not an entry from the SECLOADA module, relocates a channel program to read the F4 DSCB and F1 DSCB for the data set to be scratched into the Scratch work area. [BEGIN]

Branches to a subroutine that links to IOS to read in the F4 DSCB and the F1 DSCB to be deleted. Tests to verify that an F1 DSCB has been read in. [EXECUTE]

This routine issues the ENQ macro to enqueue on the data set name of the data set to be scratched. If the return code from the ENQ macro request indicates that some task could be open to the data set being scratched, this routine examines the DEBs assigned to this job, comparing the starting CCHH in the first extent in the DEB to the CCHH of the first data extent in the DSCB to ensure that no task is currently open to the data set. [NEWDSCB]

1a If it is RACF-indicated (DS1DSIND=X'40'), bypasses password checking.

1b Sets error code 8 if a VSAM data space. Issues a RACHECK macro or ALTER authorization to the data set/volser entity. If the data set RACHECK fails, checks ALTER authorization to DASDVOL volume serial entity, and permits SCRATCH if so authorized. If neither is authorized, sets error code 8 and exits; otherwise, continues. [PSWDCHK]

2 Determines whether the data set has password protection; if the password has not already been verified, passes control to the SECLOADA module of Open/EOV password protection (see *OS/VS2 Open/Close/EOV Logic*). [PSWDCHK]

When entered from SECLOADA and register 10 contains a zero, which indicates that the password has been successfully verified, this routine restores registers and addresses used in password processing. [PASSOKAY]

3 If the data set is not password-protected, receives control after the F1 DSCB has been read in. If the expiration date for the data set has not been overridden (indicator set in Scratch parameter list), verifies that the expiration date is passed. [NAMEOKAY]

IGG0299A

3a Checks DSCB (DS1DSIND=X'40') to see if it is RACF defined. If it is defined, issues a RACDEF DELETE macro instruction to remove the volume from the data set definition.

Return code=0—Successful; continue
Return code=4—Volume not defined;
set error code=8
unless authorized to the DASDVOL
Return code=8—Not authorized to delete;
set error code=8
unless authorized to the DASDVOL
[BEGIN]

Checks the number of extents to be deleted and if there are more than 16, sets the DOS bit in the DS4VTOCI field and writes the modified F4 DSCB back to the VTOC. This is done so that the next time an allocation is made on the volume, the VTOC Conversion routine will be executed in order to correct the F5 and F6 DSCBs. [BLDTAB]

Sets/resets the DIRF bit and tests for previous premature termination of a DADSM function. If no such interruption occurred, writes the F4 back to the VTOC. [MOVECCW]

If no premature termination occurred or the F4 has been written back to the VTOC, sets/resets the DIRF bit for writing out in IGG0290D. Then the channel program is modified to write an F0 DSCB over the F1 and to read in an F2 and/or an F3 if they exist on the VTOC. [SKPWR]

Determines if there is a DSCB in virtual storage with extents to be converted. [TESTPTR]

4 When F3 DSCBs are read in, sets up work area and branches to a subroutine that converts the F3 extents to RTA1/RTA2 format. If the data set resides on an MSS virtual volume, a subroutine is called to build a parameter list describing the extents which will be passed to SVC 126 to free the staging volume space. [MOREXTS]

Links to the resident conversion routine to convert the F1 and F3 extents from CCHHR to RTA1/RTA2 format and stores each converted extent in the extent table. [CVTEXTS]

5 Links to a subroutine that writes an F0 over an F1 or F3 DSCB or an F2 DSCB (whose extents have previously been converted and entered in the extent table). When the F0 is written, the channel program then reads in the next DSCB in the chain if there is one. [ZEROUT]

6a When there are no more F1, F2, or F3 DSCBs in the chain, inserts in the CCW the address of the F6 DSCB for split-cylinder data sets or the F5 DSCB for non-split-cylinder data sets, so that one or the other is read in by the channel program that writes the last F0 DSCB. [LASTDSCB]

IGG0290B

Modifies a channel program that searches for F0 DSCBs and reads or writes F6 DSCBs. If there are no entries in the extent table or if the DOS or DIRF bit is set, Scratch processing is terminated and control is passed to IGG0290D. [BEGIN]

6b Sorts the extent table into ascending RTA sequence. [NEWPASS]

Tests for shared cylinders and exits to IGG0290D if not present. Sets ups pointers and counters to build or modify an F6 DSCB. [TESTCYLS]

6c Checks each extent entry in the extent table to determine if it represents a shared extent. Searches for the extent in the F6 DSCB. When found, the number of data sets sharing the extent is decremented by one. If the number of data sets reach zero, the XXYYZ for this extent is converted to RTA1/RTA2 and moved to the extent table. [COMPARE]

When an entry in the extent table is encountered that does not share cylinders, this routine increments the extent table pointers and counter. [NOTSPLIT]

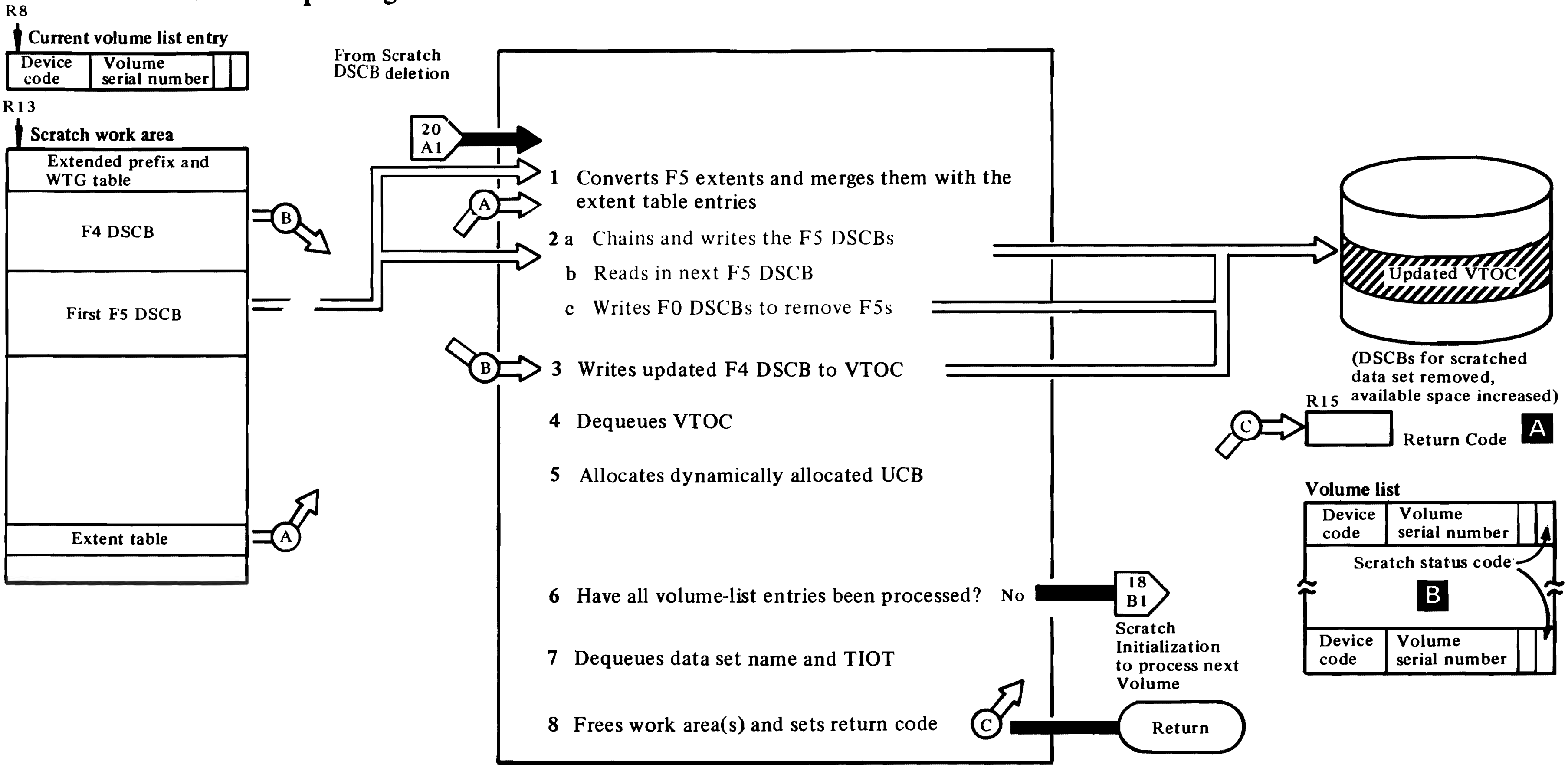
6d When data sets other than the one being scratched are still using the extent currently being examined, moves the extent (with a decremented number of data sets) to the output area to be written back to the VTOC. [LOOKAHED]

When the F6 DSCB extents have been compared with the split-cylinder extent in the extent table and when this F6 is chained to another F6, reads in the next F6 in the chain. [CHECKIN]

When extent entries have been made in an F6 output DSCB, determines whether a chaining pointer needs to be put in the F6 before it is written out. A pointer is not needed if (1) only one entry is left in the extent table and it is for a split-cylinder extent, and (2) only one extent is left in the input F6 DSCB that has a data set count of one and is contained in the single extent in the extent table. When neither of the above conditions is met, a pointer is added to the F6 and the F6 is written back to the VTOC. [DADSOUT1]

7 When there are no unprocessed entries in the extent table, reads into virtual storage the first F5 DSCB on the VTOC and passes control to IGG0290C. [ENDUPD2]

Diagram 20. Scratch–Format-5 and Format-4 DSCB Updating



1 Converts F5 extents and merges them with the extent table entries

2 a Chains and writes the F5 DSCBs

b Reads in next F5 DSCB

c Writes F0 DSCBs to remove F5s

3 Writes updated F4 DSCB to VTOC

4 Dequeues VTOC

5 Allocates dynamically allocated UCB

6 Have all volume-list entries been processed? No

7 Dequeues data set name and TIOT

8 Frees work area(s) and sets return code

Extended Description for Diagram 20

IGG0290C

This function is entered after the extent table has been built and the F1, F2, and F3 DSCBs have been removed from the VTOC.

1 Sets up counters and pointers in input and output F5 work areas. Calls a subroutine that links to the resident conversion routine to convert the XXYYZ of the first F5 to RTA1/RTA2. [BEGUN]

Compares the RTA1/RTA2 of the extent table entries with the RTA1/RTA2 of the F5 DSCB to determine where to merge the extent table entries. [MERGE1]

Merges extent table entries with the F5 extents when overlapping extents are detected. [OVERLAP]

Determines when an F5 DSCB must be written to the VTOC and when a new F5 must be read in. [OUTPUT1]

Modifies a channel program and links to IOS to read in the second and subsequent F5 DSCBs; calls a subroutine to convert the F5 extents from XXYYZ to RTA1/RTA2. [DADSMIN]

Converts XXYYZ of F5 extents to RTA1/RTA2. Sets up extent counters and pointers. [CVTORTAO]

2 Determines if two F5 output extents are contiguous and if so, combines them into a single extent. [COLLAPSE]

Converts the F5 output extents from RTA1/RTA2 to XXYYZ, modifies the channel program to write the F5 to the VTOC, and branches to a subroutine that links to IOS to write the F5 to the VTOC. [DADSMOUT]

When a new F5 DSCB has been built and an F0 must be found to write it over, branches to a subroutine that searches for an F0 on the VTOC. [FINDHOLE]

IGG0290D

If a VTOC is enqueued, tests to determine if errors were encountered while processing this volume. If no errors were encountered, updates the number of F0 DSCBs on the VTOC and writes the F4 DSCB back to the VTOC. [BEGIN]

4 Dequeues VTOC. [TESTEXIT]

Builds and issues message IEC603I if an I/O error occurred or if overlapping extents were encountered in the F5 DSCB. [TSTIOERR]

5 If the unit (UCB) was dynamically allocated, builds the request block required to deallocate the unit and issues the DYNALLOC macro (SVC 99) to deallocate the unit. [TSTALLOC]

6 Determines if all entries in the volume list have been processed. If there are unprocessed entries, passes control to the Scratch initialization function (IGG0290E) to process the next volume in the volume list. [TESTLAST]

If a volume was mounted during Scratch initialization, sets up to request that the volume be demounted. A test is made to see if an MSS parameter list had been built. If yes, the routine issues ICBACREL to call SVC 126 to relinquish the data space. [DEQTEST1]

IGG0290F

Calls a subroutine that builds the demount message IEC502E. Completes the message and calls another subroutine that issues the message. For MSS virtual volumes, an SVC 126 is issued. [MNT00400]

Builds the demount message. If the "MONITOR SPACE" command has been issued or if demounting activity is being monitored for SMF, issues SVC 78 (LSPACE, see Diagram 26) to gather information about the space available on the volume for the demount message or for the SMF data set (record type-19). Also issues SVC 98 (VOLSTAT) to write a volume statistics record on the SYS1.LOGREC data set. [MNT11000]

IGG0290D

7 If the data set name was enqueued upon exclusively, dequeues from the data set name (resource name is SYSDSN). [DEQTEST2]

If the TIOT was enqueued upon, dequeues (resource name is SYSZTIOT). [DEQTEST3]

When the Scratch processing is complete and was successful or partly successful, determines whether data set activity is being monitored and, if so, builds and writes a type-17 record to the SMF data set. [TESTSMF]

8 When all of the entries in the volume list have been processed, issues the IECRES macro to free the Scratch work area(s). See *OS/VS2 Open/Close/EOV Logic* for a description of the function of the IECRES macro. If the scratch was successful or if an invalid volume list was provided, control is returned to the caller with a return code of 0 (if successful) or 12 (if invalid) in register 15. In all other cases, control is passed to IGG029DM. [EXIT]

IGG029DM

This dummy CSECT immediately passes control using the return address in register 14.

A Return Code

Code	Meaning
0	Successful scratch.
4	No volumes containing any part of the data set were mounted, nor did register 0 contain the address of a unit that was available for mounting a volume of the data set to be scratched. (This return code is accompanied by the setting of status code 5 in all entries of the volume list.)
8	An unusual condition was encountered on one or more volumes.
12	Invalid volume. (This return code is not accompanied by the setting of a status code.)

B Scratch Status Code:

Code	Meaning
0	The DSCB for the data set has been deleted from the VTOC on the volume pointed to.
1	The VTOC of this volume does not contain the DSCB to be deleted.
2	The macro instruction failed when the correct password was not supplied by the console operator or TSO terminal operator in the two attempts allowed, or the user tried to scratch a VSAM data space.
3	The DSCB was not deleted because either the OVRD option was not specified or the retention cycle had not expired.
4	A permanent I/O error on an invalid F1 DSCB was encountered when processing this volume.
5	It could not be verified that this volume was mounted nor was there a unit available for mounting the volume. If one or more of the volumes was an MSS mass-storage volume,

Extended Description for Diagram 20 (Continued)

see the explanation of message IEC666I in *OS/VS Message Library: VS2 System Messages.*

6 The operator was unable to mount this volume. If the volume is an MSS virtual volume and is running in a JES3 environment, JES3 would not allow the volume to be mounted.

7 The DSCB was not deleted because the data set was open.

8 The DSCB indicates the data set is defined to RACF but either the accessor is not authorized to the data set or to the volume, or the data set is a VSAM data space, or the data set is not defined to RACF.

Diagram 21. Partial Release—Overview

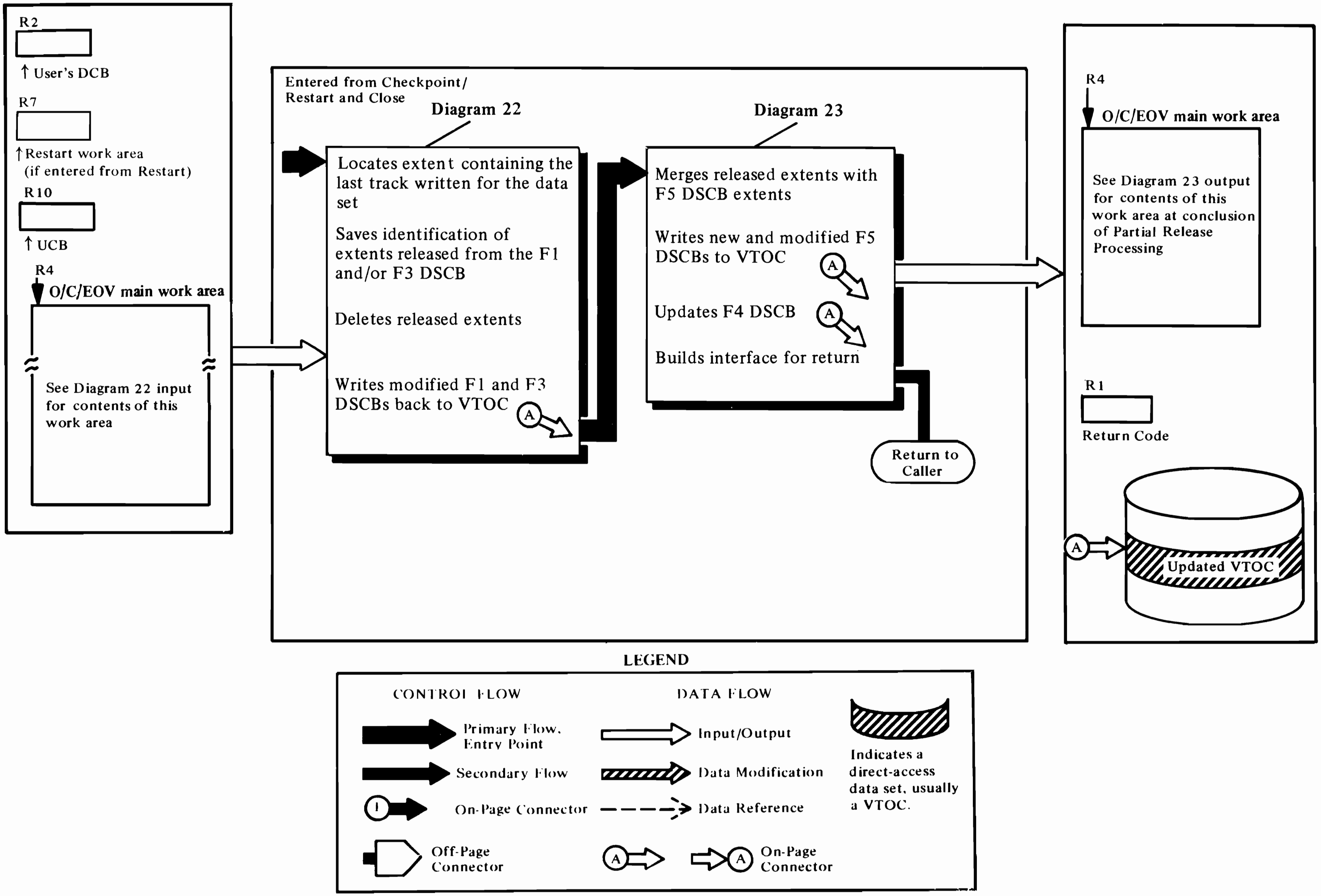
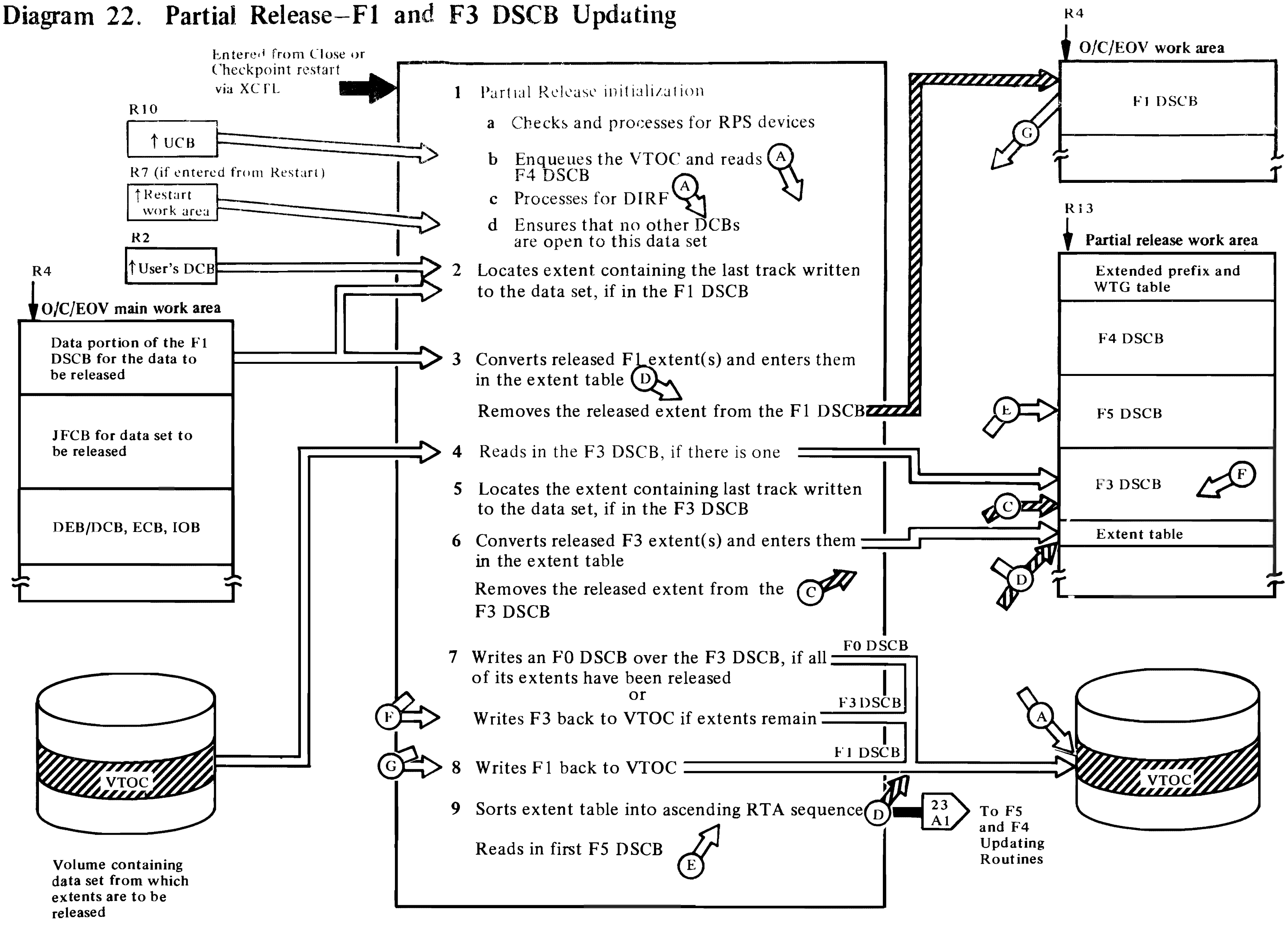

R2
User's DCB
R7
Restart work area
(if entered from Restart)
R10
UCB
R4
O/C/EOV main work area
See Diagram 22 input for contents of this work area
Entered from Checkpoint/ Restart and Close
Diagram 22
Locates extent containing the last track written for the data set
Saves identification of extents released from the F1 and/or F3 DSCB
Deletes released extents
Writes modified F1 and F3 DSCBs back to VTOC
A
Diagram 23
Merges released extents with F5 DSCB extents
Writes new and modified F5 DSCBs to VTOC
A
Updates F4 DSCB
A
Builds interface for return
Return to Caller
R4
O/C/EOV main work area
See Diagram 23 output for contents of this work area at conclusion of Partial Release Processing
R1
Return Code
A
Updated VTOC
LEGEND
CONTROL FLOW
Primary Flow, Entry Point
Secondary Flow
On-Page Connector
Off-Page Connector
DATA FLOW
Input/Output
Data Modification
Data Reference
A
A
On-Page Connector
Indicates a direct-access data set, usually a VTOC.

Diagram 22. Partial Release–F1 and F3 DSCB Updating



Entered from Close or Checkpoint restart via XCTL

1 Partial Release initialization

a Checks and processes for RPS devices

b Enqueues the VTOC and reads F4 DSCB

c Processes for DIRF

d Ensures that no other DCBs are open to this data set

2 Locates extent containing the last track written to the data set, if in the F1 DSCB

3 Converts released F1 extent(s) and enters them in the extent table

Removes the released extent from the F1 DSCB

4 Reads in the F3 DSCB, if there is one

5 Locates the extent containing last track written to the data set, if in the F3 DSCB

6 Converts released F3 extent(s) and enters them in the extent table

Removes the released extent from the F3 DSCB

7 Writes an F0 DSCB over the F3 DSCB, if all of its extents have been released

or

Writes F3 back to VTOC if extents remain

8 Writes F1 back to VTOC

9 Sorts extent table into ascending RTA sequence

Reads in first F5 DSCB

To F5 and F4 Updating Routines

Volume containing data set from which extents are to be released

Extended Description for Diagram 22

IGG020P1

1a Entered from the Reposition I/O module of Checkpoint/Restart or from Close function of O/C/EOV, this routine gets a work area for the Partial Release routines using the IECRES macro, described in *OS/VS2 Open/Close/EOV Logic*. Initializes a channel program and picks up the F4 (VTOC) DSCB address from the UCB. Tests for the RPS feature. If present, gets 128-byte RPS work area and copies the appendage vector table (AVT) into it. Loads module IGG019EK and enters its address in the AVT. Modifies the DEB in the O/C/EOV main work area to point to the RPS AVT. [BEGINA]

b Enqueues the VTOC using a RESERVE macro instruction with the "set-must-complete" option. Branches to a subroutine to read in the F4 DSCB. [SKIPRPS]

c Sets/resets and tests the DIRF indicator, and if no previous interrupt has occurred, writes the F4 to the VTOC with the DIRF bit set. [SETVTOC]

When entered from Checkpoint/Restart, attempts to enqueue exclusively on the data set name (resource name is SYSDSN). [DSENQ1]

When entered from Close, requests a change of status to exclusive control of the TIOT (resource name is SYSZTIOT). Reads the key of the F1 DSCB. (TIOTENQ]

d When entered from Close, searches the DEB table to ensure that this job has no other DCBs open to the data set. Then searches the TCB DEB chain (if entered from Close) or the Restart DEB chain (if entered from Restart) to ensure that this job has no other DCBs open to the data set. [DEBSRCH]

2 For requests made in a number of cylinders, this routine determines whether the last track written to the data set to be released was on a cylinder boundary, and if not, rounds the extent to the next cylinder boundary. [ROUNDCYL]

Determines whether the extent to be released is in the F1 or F3 DSCB and branches accordingly. [SAVEFMT3]

Locates the extent containing the last track written when it is in an F1 DSCB. When found, the request type (cylinder or rounded record) is determined. [F1RUTINE]

3 Determines if last track written is the last track of the extent. If not the last track, inserts the new end of the extent (last track actually used), and links to a subroutine that converts the CCHH to RTA1/RTA2 and enters it in the extent table. [CYLREQ]

When the last track written is the last track of an extent in an F1 DSCB, determines if there is another extent. If the extent containing the last track written is not the last extent of the data set, branches to a subroutine that calls the resident conversion routine to convert the CCHHCCHH of any subsequent extents to RTA1/RTA2 format. These extents are entered in the extent table. [CHECKLST]

IGG020P2

4 Checks to see if there is an F3 for this data set, and, if so, modifies the channel program and reads in the F3 DSCB. [CHECKF3]

5 Locates the extent containing the last track written if this extent is in the F3 DSCB. [BEGINA]

6 When the F3 extent containing the address of the last track written is located, links to a subroutine that converts the portion of the extent that is to be released and any subsequent extents in the F3 DSCB from CCHHCCHH to RTA1/RTA2 format and enters them in the extent table. [LSTRACK]

Calls the resident conversion routine to convert the F3 extents from CCHH to RTA1/RTA2. Enters the converted extents in the extent table. [CVTORTA]

7 When all extents are released from an F3 DSCB, modifies the channel program and links to a subroutine that writes an F0 DSCB over the released F3 DSCB. [WRITEZRO]

Updates the F1 DSCB with the number of extents remaining in the volume for the data set. Tests for SMF and if present, enters the amount of space released in the TCTIOT. [UPDATEF1]

IGG020D0

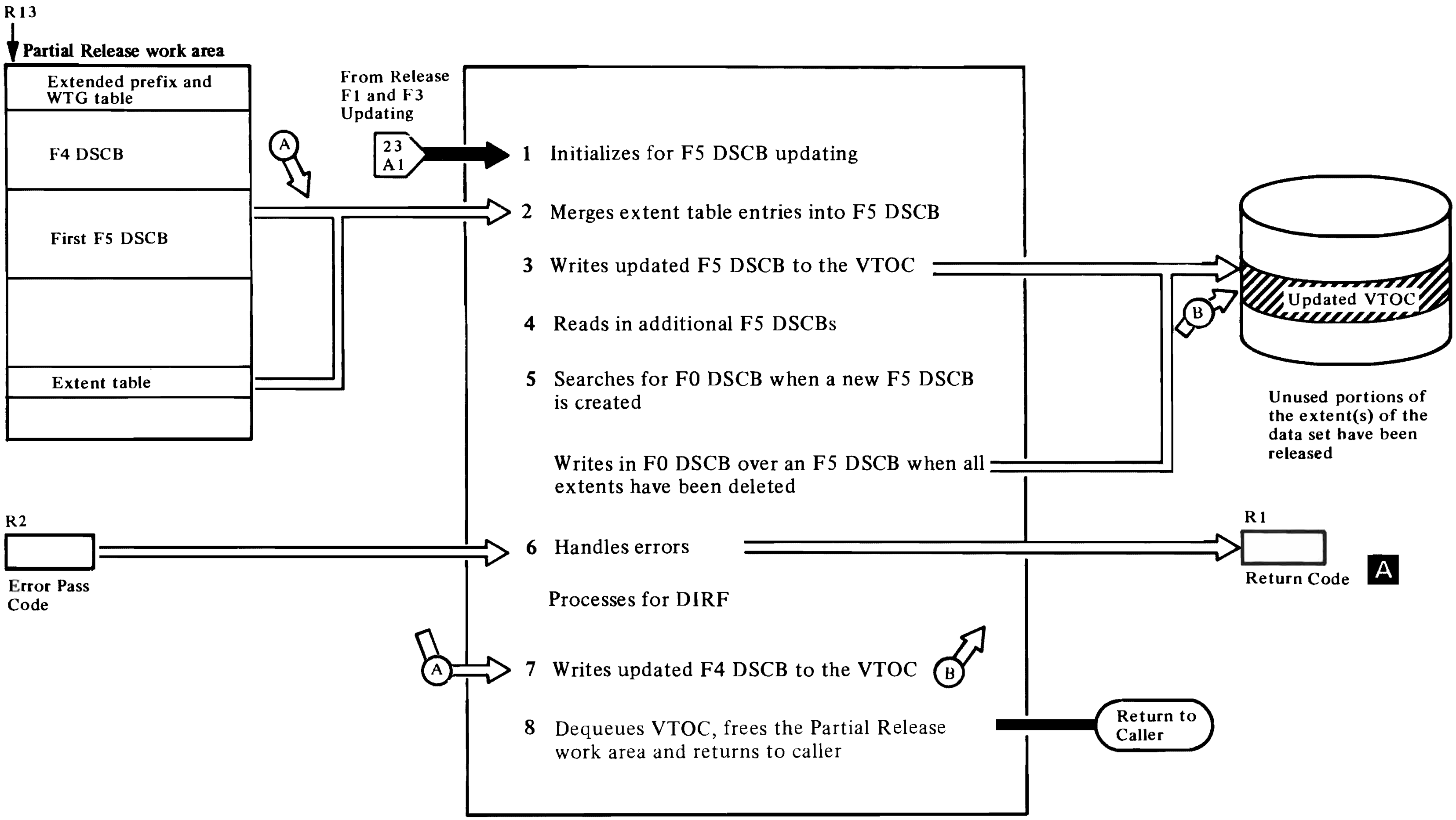
8 Writes back the F1 DSCB to the VTOC. If no space is being released, transfers control to IGG020P3. [BEGINA]

If the DOS or DIRF bit is set, this routine transfers control to IGG020P3, bypassing updating of the F5 DSCB(s). [DOSCHECK]

9 Sorts extent table into ascending RTA sequence. [NEWPASS]

10 Relocates channel program for reading and writing F5 DSCBS. Links to a subroutine to read in first F5 DSCB. [RELOCATE]

Diagram 23. Partial Release–F5 and F4 DSCB Updating


R13
Partial Release work area
Extended prefix and WTG table
F4 DSCB
First F5 DSCB
Extent table
A
From Release F1 and F3 Updating
23 A1
R2
Error Pass Code
A
B
Updated VTOC
B
Unused portions of the extent(s) of the data set have been released
R1
Return Code
A
Return to Caller

1 Initializes for F5 DSCB updating

2 Merges extent table entries into F5 DSCB

3 Writes updated F5 DSCB to the VTOC

4 Reads in additional F5 DSCBs

5 Searches for F0 DSCB when a new F5 DSCB is created

Writes in F0 DSCB over an F5 DSCB when all extents have been deleted

6 Handles errors

Processes for DIRF

7 Writes updated F4 DSCB to the VTOC

8 Dequeues VTOC, frees the Partial Release work area and returns to caller

Extended Description for Diagram 23

This function is entered after the F1 and F3 DSCBs have been updated. In the routines that follow, the F5 DSCBs and the F4 DSCB are updated to reflect the new status of the volume.

IGG020D1

1 Sets up F5 chain pointers, extent counters, and clears output area. [SETICCHH]

2 Merges extent table entries with the F5 DSCBs read in. [MERGE1]

Creates a single extent for each set of contiguous extents. [COLLAPSE]

When an extent (or extents) in the extent table overlaps all or part of an extent in an F5 DSCB, an error code is set, then the extents are merged. The error code causes the DIRF bit to be left on in the F4 DSCB. [OVERLAP]

Determines when to write an F0 or an F5 DSCB. [OUTPUT1]

3 Converts the RTA1/RTA2 extents to XXYYZ, then links to a subroutine that writes the F5 DSCBs to the VTOC. [DADSMOUT]

4 Modifies the channel program and links to a subroutine that reads in an F5 DSCB, then links to a second subroutine that converts the F5 extents from XXYYZ to RTA1/RTA2. [DADSMIN]

Converts the F5 extents from XXYYZ to RTA1/RTA2 format. [CVTORTA0]

5 Determines when it is necessary to find the address of an F0 DSCB, when a new F5 DSCB is created. [FINDHOLE]

IGG020P3

6 Tests for error entries and if a permanent I/O error has occurred, the F4 DSCB is not rewritten. [BEGIN]

The DIRF bit is set/reset and tested, if a previous interrupt has occurred, the F4 DSCB is not written back to the VTOC. [TESTDIRF]

Determines whether overlapping extents were encountered (extents to be released already existed in the F5 DSCBs). [XTENTEST]

7 Sets up channel program, then writes the F4 DSCB. [WRTF4]

When entered from Checkpoint/Restart, this routine dequeues the data set name (SYSDSN) if it had been enqueued exclusively. [NOWRT]

Dequeues from VTOC. [DEQVTOC]

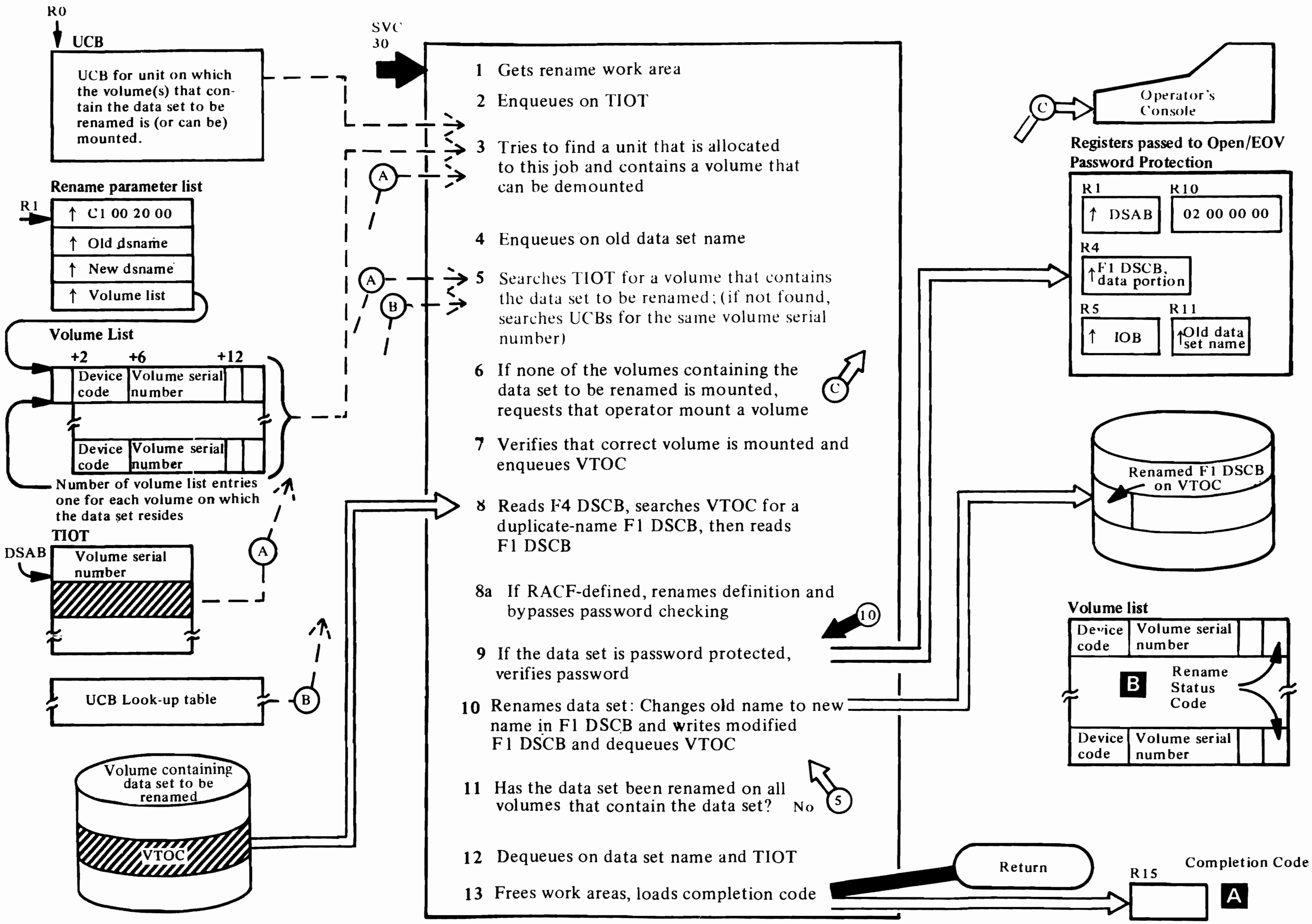
If I/O errors or overlapping extents were encountered, issues message IEC603I. [BUILDMSG]

8 Frees work areas obtained during previous processing and restores the work area DEB if RPS feature is present on device. Determines caller and returns. [RPSTEST]

A Return Code:

Code	Meaning
0	No errors detected.
2	Unable to find extent in F1 DSCB.
4	Unable to find extent in F3 DSCB.
8	Data set is open.
16	Permanent I/O error.
32	No space in VTOC.

Diagram 24. Rename Data Set



1 Gets rename work area

2 Enqueues on TIOT

3 Tries to find a unit that is allocated to this job and contains a volume that can be demounted

4 Enqueues on old data set name

5 Searches TIOT for a volume that contains the data set to be renamed; (if not found, searches UCBs for the same volume serial number)

6 If none of the volumes containing the data set to be renamed is mounted, requests that operator mount a volume

7 Verifies that correct volume is mounted and enqueues VTOC

8 Reads F4 DSCB, searches VTOC for a duplicate-name F1 DSCB, then reads F1 DSCB

8a If RACF-defined, renames definition and bypasses password checking

9 If the data set is password protected, verifies password

10 Renames data set: Changes old name to new name in F1 DSCB and writes modified F1 DSCB and dequeues VTOC

11 Has the data set been renamed on all volumes that contain the data set? No

12 Dequeues on data set name and TIOT

13 Frees work areas, loads completion code

Extended Description for Diagram 24

The Rename function is entered from the SVC SLIH as a result of the issuance of SVC 30, which is part of the expansion of the RENAME macro. Its purpose is to rename data sets, which is done by changing the data set name field of the F1 DSCB.

IGC00030

1 Issues the IECRES GET macro to get the Rename work area (see *OS/VS2 Open/Close/EOV Logic* for a description of the IECRES macro). [BEGIN]

Checks for a caller executing in supervisor state or in a system storage protection key (key 0 through 7). If neither, branches to the resident address validation routine (IEA0VL00) to verify that the volume list passed by the caller has the same protection key as indicated in the TCB. [VALCHECK]

2 Attempts to enqueue on the TIOT using the resource name SYSZTIOT. If the resource in not available, the data set is in the process of being opened or closed and therefore cannot be renamed. [BYPASS]

3 When the caller passes the address of a UCB in register 0, searches the DSAB chain for a TIOT DD entry that contains this UCB address. This search will verify that the unit associated with the UCB is allocated to the job. If the unit is allocated to this job, verifies that the volume on the device is eligible for demounting. [USERSUCB]

When no UCB address is passed, or when the UCB address passed represents a device that is not allocated to the job step or contains a volume that is not eligible for demounting, this routine searches the DSAB chain for a TIOT entry whose UCB contains the volume serial number of a volume in the volume list (to determine if any volume in the volume list is mounted on a unit allocated to this job). If so, this routine verifies that the UCB represents a volume that is on-line and available for demounting. [FINDUNIT]

Checks for requests to rename a VIO data set, which is not allowed. If such an attempt is made, this routine terminates Rename processing. [VIODSTST]

4 Attempts to enqueue on the data set name using the resource name of SYSDSN. If the resource is not available for exclusive control, the data set is allocated to this or another address space and cannot be renamed. [DSNAMENQ]

When there are no entries in the volume list, when the validity check of the volume list address fails, or when the data set to be renamed is either open or a VIO data set, this routine frees the Rename work area and returns to the caller with an error indication in register 15. [EXIT]

IGG03002

5 Before searching the TIOT, builds a DEB/DCB and IOB with the information required by IOS to read from the VTOC. [BEGIN]

Searches the DSAB chain for a TIOT entry whose UCB contains the volume serial number of the current volume-list entry. [FINDUCB]

If the current entry in the volume list is not mounted on a device allocated to the job, searches the UCB look-up table for a UCB with a volume serial number that matches the volume serial number of the current entry in the volume list. [TABLSRCH]

If the current volume list entry was found on an unallocated unit, builds the request blocks required to dynamically allocate the unit. Issues the DYNALLOC macro to request that the unit be allocated to this job. [ALLOCVOL]

If the volume was not mounted on any unit or if the dynamic allocation request was unsuccessful, requests that the volume be mounted on a demountable, available unit. [MOUNTVOL]

IGG0300F (alias for IGG0290F)

6 Determines if the unit associated with the UCB is eligible for demounting. [MNT00100]

If another volume is to be mounted after the current volume is demounted, enqueues on the volume serial number of the next volume. [MNT00200]

Calls a subroutine that builds the demount message IEC502E, then completes the message and calls another subroutine that issues the message. [MNT00400]

Builds the demount message. If the "MONITOR SPACE" command was issued or if demounting activity is being monitored for SMF, issues SVC 78 (LSPACE, see Diagram 26) to gather information about the space available on the volume for the demount message or for recording in the SMF data set (record type-19). Also issues SVC 98 (VOLSTAT) to write a volume statistics record on the SYS1.LOGREC data set. [MNT11000]

Calls a subroutine that sets the mount bit in the UCB. Builds and issues a mount message IEC601D. If the UCB indicates both an MSS virtual volume and a JES3-managed unit, builds an SSOB and passes it to JES3 via the IEFSSREQ macro for mounting approval. Upon approval, issues SVC 126 by use of th ICBMNTDE macro to mount the virtual volume. Upon disapproval, loads the return code of 8 into register 15, dequeues from the volume serial number (resource name is SYSZVOLS), and returns to the calling routine. [MNT03000]

If the UCB indicates an MSS virtual volume that is not JES3-managed, issues SVC 126 by use of the ICBMNTDE macro to mount the virtual volume. [MNT03005]

Also issues a TPUT macro to write message IEC108I to the TSO terminal. [MNT03010]

7 Builds and executes a channel program to read the volume label (record 3 of track 0, cylinder 0), then calls a subroutine that resets the mount bit in the UCB. [MNT04000]

If the operator mounted the wrong volume, branches to a subroutine that builds and issues a message telling the operator that the volume is not accepted. [MNT04200]

Calls a subroutine that converts the actual device address (CCHHR) of the VTOC to a relative track address (TTR), then moves the TTR of the VTOC and the volume serial number to the UCB. [MNT05000]

Updates the volume disposition in the UCB for the newly mounted volume. [MNT05100]

Loads the return code in register 15, dequeues from the volume serial number (resource name is SYSZVOLS), and returns to the calling routine. [MNT01500]

IGG03001

8 Tests for a return from the Open/EOV password protection function (SECLOADA); if not entered from SECLOADA, enqueues on the VTOC by issuing a RESERVE macro. Next, searches the VTOC for a F1 DSCB that has the new data set name (a *duplicate-name* DSCB). [BEGIN]

Calls a subroutine that reads the F1 DSCB of the data set to be renamed (the old data set name is pointed to by the second word of the Rename parameter list). [NAMEOKAY]

Extended Description for Diagram 24 (Continued)

If the enqueuing of the data set name indicated that some task may be open to the data set being renamed, this routine examines the extents in each of the DEBs associated with this job to verify that no task is currently open to the data set to be renamed. [F1FOUND]

If data set is RACF-indicated, checks user authorization to rename data set by issuing a RACFDEF for ALTER authorization. If not authorized, sets error code=8; otherwise issues RACDEF TYPE=DEFINE, NEWNAME= to change name of RACF definition. If not successful, sets error code of 8. If successful, bypasses password checking. [PSWDCHK]

The RACDEF TYPE=DEFINE, NEWNAME macro instruction is bypassed for RACF-defined data sets if both the JSCBPASS bit and the second bit of the second byte of the RENAME parameter list are set to 1.

8a If data set is RACF-indicated, checks user authorization to rename data set by issuing a RACDEF for ALTER authorization. If not authorized, sets error code=8; otherwise, issues RACDEF TYPE=DEFINE, NEWNAME= to change name of RACF definition. If not successful, sets error code of 8. If successful, bypasses password checking. [PSWDCHK]

9 This routine determines if the data set is password protected. If so, passes control to the SECLOADA module of the Open/EOV password protection function. (See *OS/VS2 Open/Close/EOV Logic* for a description of this function.) [CHECKPSW]

10 When the password has been verified or when the data set to be renamed is not password protected, the new data set name is written over the old data set name, and the F1 DSCB is written back to the VTOC in its original location. [PSWDOKAY]

Dequeues the VTOC. [FINISH]

If the unit on which the renamed data set resides was dynamically allocated, builds the request block required to deallocate the unit and issues the DYNALLOC macro (SVC 99). [TSTALLOC]

11 Determines if all entries in the volume list have been processed. If there are unprocessed entries, gives control to IGG03002 to process the next volume list entry. [TESTLAST]

IGG0300F (alias for IGG0290F)

Calls a subroutine that builds the demount message IEC502E, completes the message and calls another subroutine that issues the message. [MNT00400]

Builds the demount message. If the "MONITOR SPACE" command was issued or if demounting activity is being monitored for SMF, issues SVC 78 (LSPACE, see Diagram 26) to gather information about the space available on the volume for the demount message or for the SMF data set (record type-19). [MNT11000]

IGG03001

12 Dequeues the data set name if it was enqueued for exclusive control (resource name is SYSDSN). [DEQTEST2]

Dequeues from TIOT. [DEQTEST3]

When Rename processing is successful on one or more volumes, determines if SMF information is being gathered. If so, builds and writes an SMF type-18 record, which describes renamed data sets. [TESTSMF]

13 When all the volume list entries have been processed, frees the Rename work area, puts a return code in register 15, and returns to the caller. [EXIT]

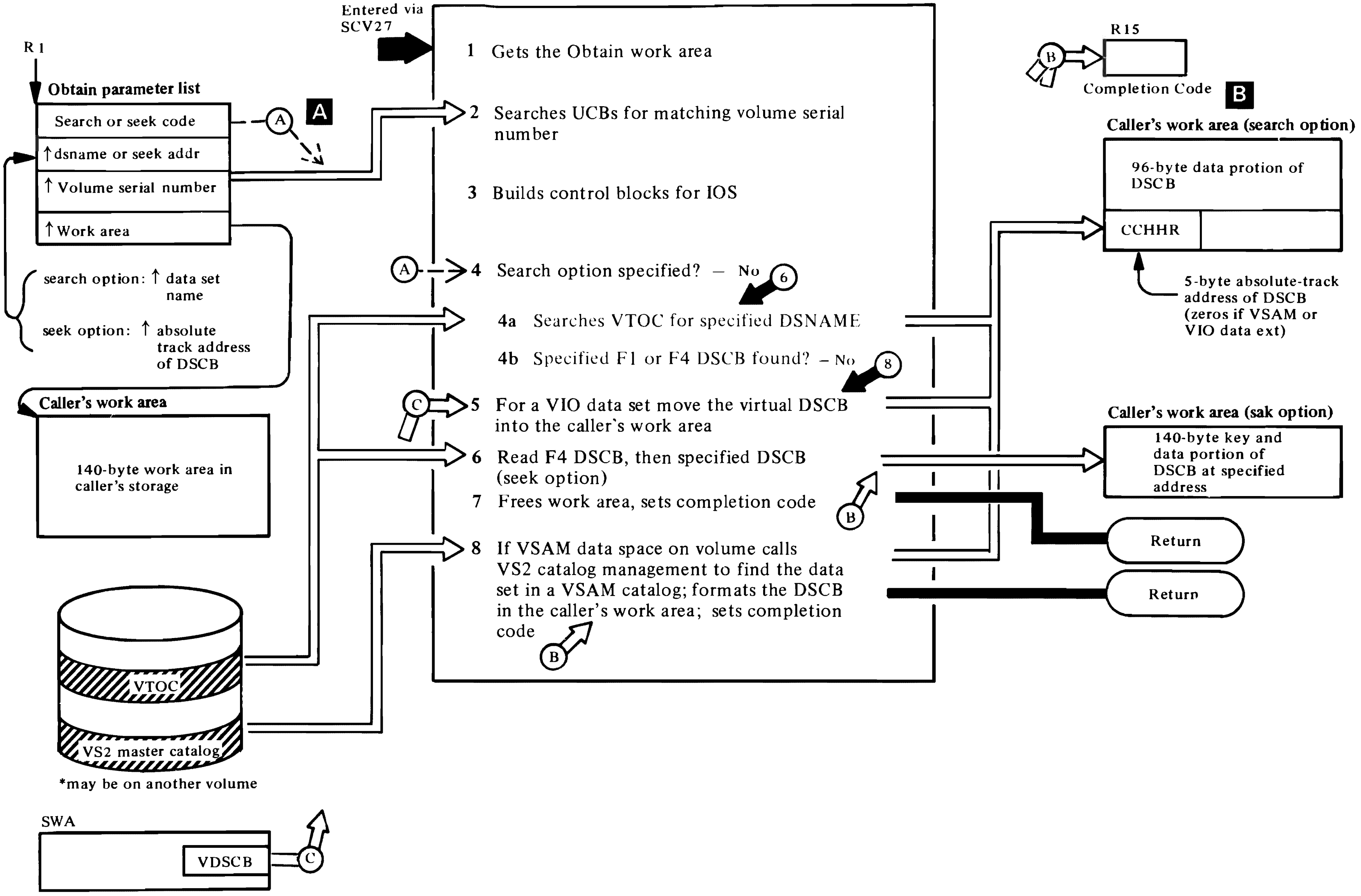
A *Completion Code:

Code	Meaning
0	Successful completion.
4	No volumes containing any part of the data set were mounted, nor did register 0 contain the address of a unit that was available for mounting a volume of the data set to be renamed (this completion code is accompanied by a setting of 5 in the Rename status code of each entry of the volume list). The data set may be a VIO data set, which cannot be renamed.
8	An unusual condition was encountered on one or more of the volumes.
12	The volume list passed to Rename was not valid (this return code is not accompanied by the setting of a Rename status code in the volume list).

B Rename Status Code:

Code	Meaning
0	The F1 DSCB for the data set has been renamed in the VTOC on this volume.
1	The VTOC of this volume does not contain the F1 DSCB of the data set to be renamed.
2	The data set could not be renamed because the data set was password protected and the password was not supplied in the two attempts allowed, or an attempt was made to rename a VSAM data space.
3	An F1 DSCB containing the new data set name already exists in the VTOC of this volume.
4	A permanent I/O error or an invalid F1 DSCB was encountered while trying to rename the data set on this volume.
5	It could not be verified that this volume was mounted nor was there a unit available for mounting the volume.
6	The operator was unable to mount this volume. If the volume is an MSS virtual volume and is running in a JES3 environment, JES3 would not allow the volume to be mounted.
7	The data set was not renamed because it was currently open for processing.
8	The data set is defined to RACF but either the accessor is not alter authorized to the data set or the data set is defined to RACF on multiple volumes.

Diagram 25. Obtain



1 Gets the Obtain work area

2 Searches UCBs for matching volume serial number

3 Builds control blocks for IOS

4 Search option specified? – No 6

4a Searches VTOC for specified DSNAME

4b Specified F1 or F4 DSCB found? – No 8

5 For a VIO data set move the virtual DSCB into the caller's work area

6 Read F4 DSCB, then specified DSCB (seek option)

7 Frees work area, sets completion code

8 If VSAM data space on volume calls VS2 catalog management to find the data set in a VSAM catalog; formats the DSCB in the caller's work area; sets completion code

Entered via SCV27
R1
Obtain parameter list
Search or seek code
↑dsname or seek addr
↑Volume serial number
↑Work area
A
search option: ↑ data set name
seek option: ↑ absolute track address of DSCB
Caller's work area
140-byte work area in caller's storage
VTOC
VS2 master catalog
*may be on another volume
SWA
VDSCB
C
B
R15
Completion Code
B
Caller's work area (search option)
96-byte data protion of DSCB
CCHHR
5-byte absolute-track address of DSCB (zeros if VSAM or VIO data ext)
Caller's work area (sak option)
140-byte key and data portion of DSCB at specified address
Return
Return

Extended Description for Diagram 25

The Obtain function is entered from the SVC SLIH as a result of the issuance of SVC 27, which is part of the expansion of the OBTAIN macro instruction. Its purpose is to read a DSCB from the VTOC of a specified volume into virtual storage. For VIO data sets, its purpose is to retrieve a virtual DSCB from the SWA; for VSAM data sets its purpose is to retrieve equivalent information from the catalog.

IGC0002G

1 Issues the IECRES GET macro to get the Obtain work area (see *OS/VS2 Open/Close/EOV Logic* for a description of the IECRES macro). [BEGIN]

Checks for a caller executing in supervisor state or in a system storage protection key (key 0 through 7). If neither, branches to the resident address validation routine (IEA0VL00) to verify that the volume list passed by the caller has the same protection key as indicated in the TCB. [VALCK]

2 If the search option is specified, determines whether the volume serial number passed in the volume list represents a VIO data set. If so, only a virtual UCB will be present. If (1) this is not a virtual DSCB for a VIO data set or (2) the seek option is specified, this routine searches the UCBs (using the UCB look-up table) for the UCB with a volume serial number that matches the volume serial number passed in the volume list. [ADDROK]

3 When the appropriate UCB has been located, builds a DEB/DCB, IOB, and ECB. The F4 DSCB address is picked up from the UCB, converted from TTR to MBBCCHHR, and stored as the seek address. [BUILDEB]

4 If the search option is specified, searches the VTOC for the F1 or F4 DSCB with the specified data set name, and when found, moves the data portion of the DSCB and its actual device address into the caller's work area. [SRCHDSCB]

5 For VIO data sets, enqueues exclusively on the TIOT (resource name is SYSZTIOT). Searches the DSAB for a TIOT DD entry containing a virtual UCB whose virtual DSCB contains the data set name being obtained and whose volume serial number matches the volume serial number in the parameter list. If a matching data set name and volume serial number is found, moves the data portion of the virtual DSCB from the scheduler work area (SWA) into the caller's work area and dequeues from the TIOT. [VIODSTST]

6 When the seek option is specified, relocates a channel program to read the F4 DSCB, moves the extent containing the VTOC (DS4VTOCE field) to the DEB, and verifies that the seek address passed is within the VTOC. Reads the DSCB and moves it into the caller's work area. [SEEKDSCB]

7 Issues the IECRES FREE macro to free the Obtain work area and returns to the caller with a return code in register 15. [LEAVE]

8 When the search option is specified and no F1 DSCB can be found, passes control to IGC0102G if the DS4AMCAT field of the F4 DSCB indicates that there is a VSAM data space on the volume. [NODSCB]

IGC0102G

Constructs a catalog parameter list and calls catalog management to find the named data set in the master catalog or in a VSAM user catalog. If found, formats the data portions of an F1 DSCB in the caller's work area, then returns to the caller with a return code in register 15. [IGC0102G]

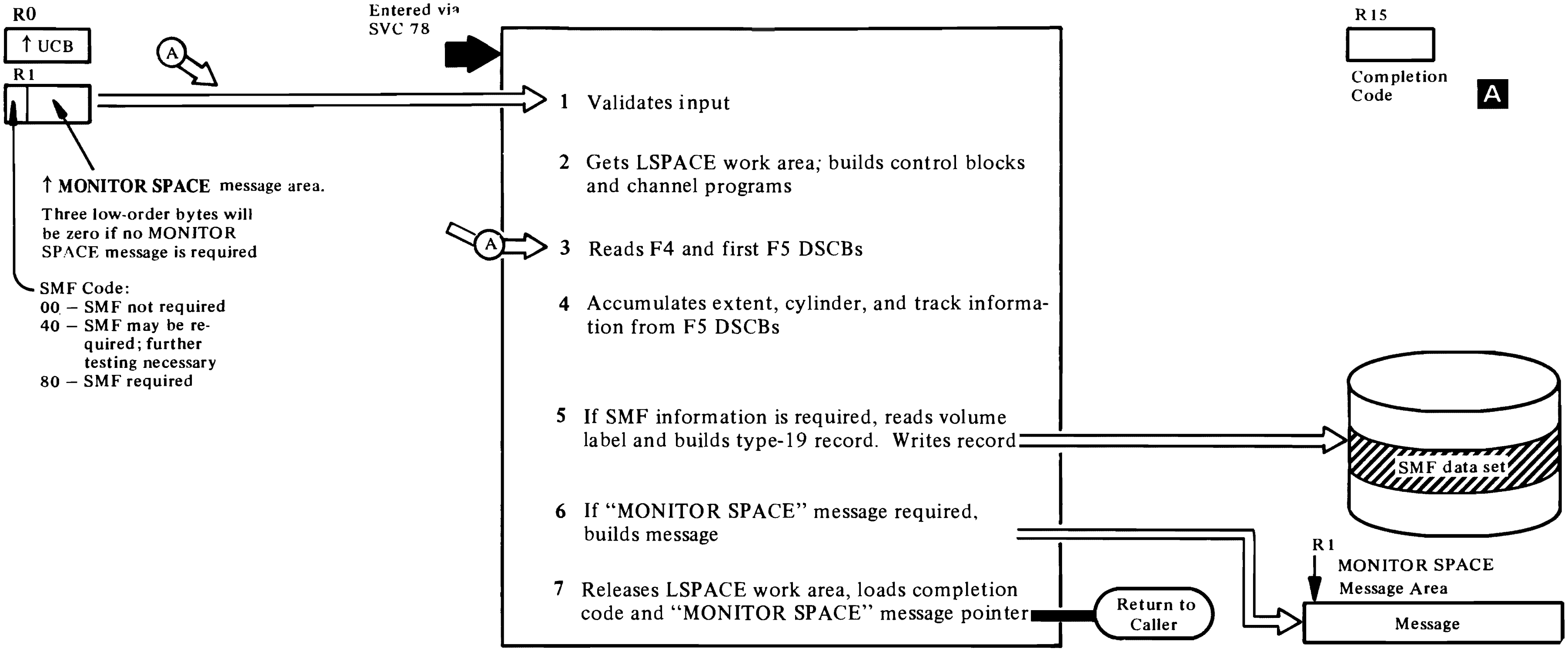
A Search or Seek Code:

Code	Meaning
X'C1 00 00 00'	Search
X'C0 80 00 00'	Seek

B Completion Code:

Code	Meaning
0	Successful completion of Obtain routine.
4	The required volume was not mounted.
8	The F1 DSCB was not found in the VTOC of the specified volume.
12	A permanent I/O error or an invalid F1 or F4 DSCB was encountered when processing the specified volume.
16	The pointer to the work area passed by the caller was invalid.
20	The absolute address passed (seek option) was not within the boundaries of the VTOC.

Diagram 26. LSPACE



↑ **MONITOR SPACE** message area.
Three low-order bytes will be zero if no MONITOR SPACE message is required

SMF Code:
00 – SMF not required
40 – SMF may be required; further testing necessary
80 – SMF required

1 Validates input

2 Gets LSPACE work area; builds control blocks and channel programs

3 Reads F4 and first F5 DSCBs

4 Accumulates extent, cylinder, and track information from F5 DSCBs

5 If SMF information is required, reads volume label and builds type-19 record. Writes record

6 If "MONITOR SPACE" message required, builds message

7 Releases LSPACE work area, loads completion code and "MONITOR SPACE" message pointer

Extended Description for Diagram 26

The LSPACE function is entered from the SVC SLIH as a result of the issuance of SVC 78. Its purpose is to calculate the amount of available space on a given volume. SVC 78 is issued by routines that provide available-space information when a volume is to be demounted. SVC 78 is also issued to gather information for an SMF type-19 record.

IGC0007H

1 Validates SMF indicator; checks for a caller running in a system storage protection key (keys 0 through 7) or supervisor state and, if neither, branches to the validity checking routine to verify that the beginning and ending addresses of the message area have the same protection key as the TCB for this task. [BEGIN]

Validates UCB address. [UCBADDR]

2 Gets a work area for LSPACE processing and builds a DEB/DCB, IOB, and ECB in the work area. [BEGINC]

Converts TTR of the F4 (VTOC) DSCB to MBBCCHHR. Relocates the channel program to the LSPACE work area and initializes it to read the F4 DSCB and the first F5 DSCB. Issues a RESERVE macro instruction to enqueue the VTOC. [GOCVT]

When (1) the UCB address passed in register 0 is not for a direct-access device, (2) an invalid parameter is passed in the LSPACE parameter list, or (3) the UCB represents a device that is not ready, sets up the error message, frees the work area, loads a completion code in register 15, and returns to the caller. [EXIT]

IGC0107H

3 Reads the F4 and first F5 DSCB. [BEGIN]

4 Relocates the channel program to read subsequent F5 DSCBs. [SETNXT]

Totals the cylinders and tracks in the F5 extents; saves the largest extent encountered; counts the number of extents. [ADDIN]

Checks to determine if all F5s on VTOC have been read; if not, branches to read the next one. [CHECKOUT]

5 Tests for SMF requirements, and, if required, modifies channel program to read the volume label and branches to the routine that reads the label. [TESTSMF]

Builds type-19 SMF record (see *OS/VS System Management Facilities (SMF)* for a description of the type-19 record). Writes the record. Tests to determine if a "MONITOR SPACE" message is required. [BLDSMF]

6 When "MONITOR SPACE" message is required, links to a routine to unpack and convert to decimal the information required, then moves the message to the 30-byte message area. [INITCVT]

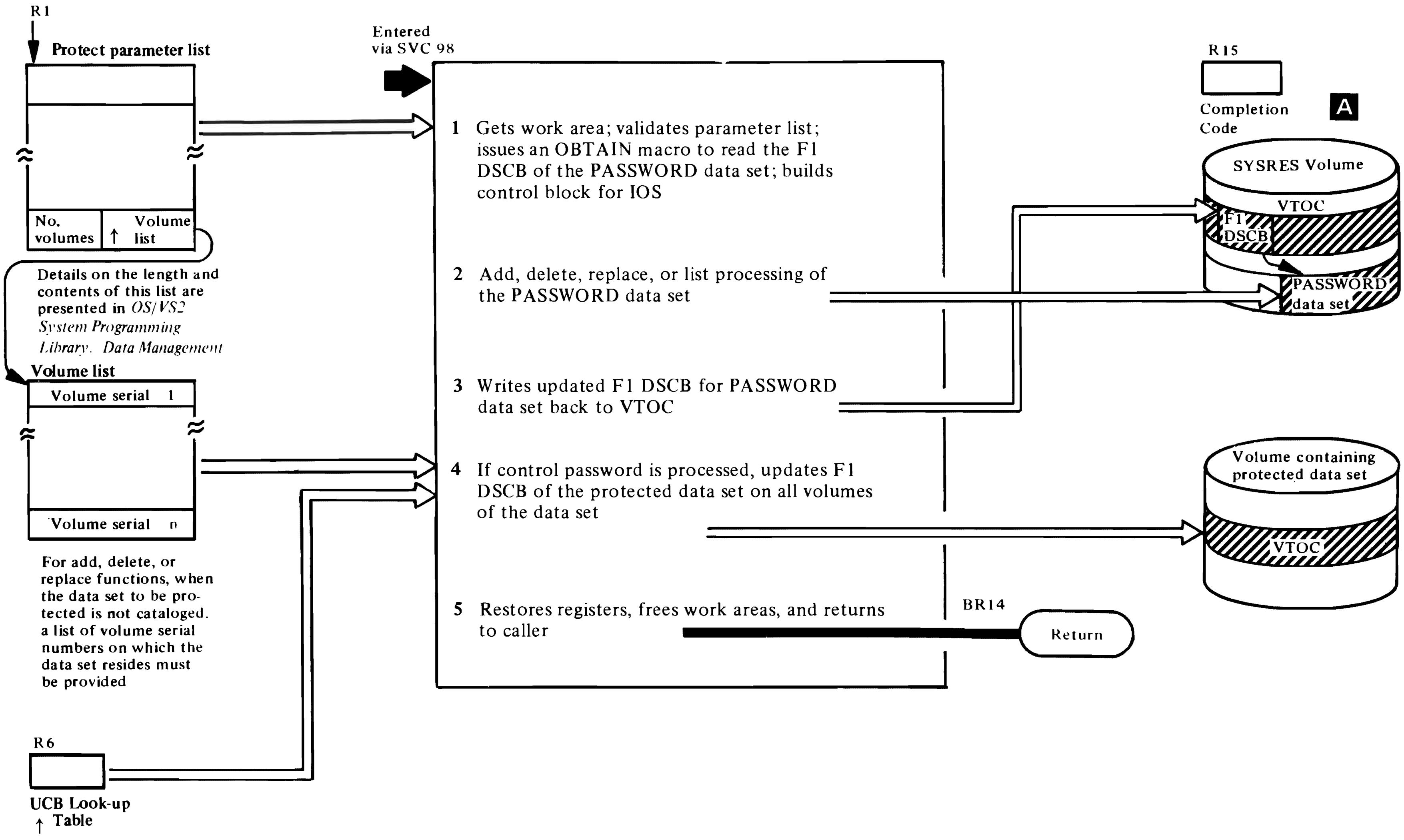
Dequeues the VTOC, tests for error exits and sets up error messages, if required. [SETDEQ]

7 Frees the LSPACE work area, loads the completion code into register 15, loads the message location into register 1, and returns to the caller. [FREE]

A *Completion Code and Resultant Message:

Code	Message	Condition
0	None	Successful LSPACE processing
4	LSPACE—PERMANENT I/O ERROR	I/O error while reading DSCBs
8	LSPACE—NON-STANDARD OS VOLUME	Last allocation on this volume made under DOS
12	LSPACE—INVALID PARAMETER (or) LSPACE—NOT A DIRECT-ACCESS VOL (or) LSPACE-UCB NOT READY	UCB address invalid UCB not for direct-access device UCB-not-ready bit is on indicating device not ready
16	None	Invalid message area address or SMF indicator

Diagram 27. Protect



R1
Protect parameter list
No. volumes
↑ Volume list
Details on the length and contents of this list are presented in *OS/VS2 System Programming Library: Data Management*
Volume list
Volume serial 1
Volume serial n
For add, delete, or replace functions, when the data set to be protected is not cataloged, a list of volume serial numbers on which the data set resides must be provided
R6
UCB Look-up
↑ Table
Entered via SVC 98
R15
Completion Code
A
SYSRES Volume
VTOC
F1 DSCB
PASSWORD data set
Volume containing protected data set
VTOC
BR14
Return

1 Gets work area; validates parameter list; issues an OBTAIN macro to read the F1 DSCB of the PASSWORD data set; builds control block for IOS

2 Add, delete, replace, or list processing of the PASSWORD data set

3 Writes updated F1 DSCB for PASSWORD data set back to VTOC

4 If control password is processed, updates F1 DSCB of the protected data set on all volumes of the data set

5 Restores registers, frees work areas, and returns to caller

Extended Description for Diagram 27

IGC0009H

1 Gets a work area for building I/O supervisor control blocks and reading in the F1 DSCB for the PASSWORD data set. Validates the parameter list pointed to by register 1. Validates parameter list entries. [INITCOR]

Determines if a PASSWORD data set entry is to be added, replaced, deleted, or listed. Verifies that the buffer address (list request) passed has the same storage protection key as the current TCB. [CHK001]

Sets up a parameter list for enqueuing the PASSWORD data set and issues an ENQ macro instruction. If an entry is to be added to the PASSWORD data set, a second ENQ is issued. [CHKEND]

Determines the record capacity constant of the SYSRES device. Gets a work area (subpool 251) for use by Obtain routine. Issues the OBTAIN macro instruction to read in the F1 DSCB for the PASSWORD data set. Builds control blocks and a channel program for subsequent processing. [TABREC]

IGC0109H

2 A branch table routes requests based on the contents of register 10, which were established in the previous routine. [ENT010]

Links to a routine that builds the protection mode indicator byte, then links to a second subroutine that builds the balance of the 80-byte data area. [COM050]

Modifies channel program and links to a subroutine that searches the PASSWORD data set for the 44-byte name specified in the parameter list. When the data set name is found, links to a routine that issues an EXCP to search for the 44-byte data set name plus the 8-byte password. [ADD010]

When a data set name is being replaced, links to a subroutine that searches the PASSWORD data set to verify that the new name is not already in the PASSWORD data set. [COM005]

When a password record is being deleted, links to a subroutine that searches for the record to be deleted. [COM010]

Entered when a record is to be listed, this routine links to a subroutine that searches for the specified record. The 80 bytes including the counter, protection mode indicator, and optional, 77-byte data field are read into the buffer provided by the requestor. [LST010]

When a new record is to be added to the end of the PASSWORD data set, sets up the IOB seek field of the last block written to the track. If there is no room to write another block to the track, obtains the next track address. [ENBLK010]

Using the input parameter list, creates the protection mode byte for the password being added or replaced. For control passwords, an indicator is set to indicate F1 DSCB updating. [BLDMD010]

Creates the 80-byte data field for a PASSWORD data set record. [BLDEN010]

Searches PASSWORD data set and writes new records or replaces records in the PASSWORD data set. [SRCH020]

Error handling routine; loads register 11 with error return code. [ERET040]

Exit routine; dequeues the enqueued PASSWORD data set, sets up XCTL ID, and issues SVC 7(XCTL) to transfer control to IGC0209H. [RET010]

IGC0209H

Tests for error entries and sets up enqueue and dequeue lists. [IGC0209H]

3 If the F1 DSCB for the PASSWORD data set is to be updated, enqueues the SYSRES VTOC, updates the DS1LSTAR field (address of last block written), and links to a subroutine that writes the DSCB back to the VTOC. The SYSRES VTOC is then dequeued. [UPDATE]

4 When a control password is being changed, checks for a volume list provided by the user. If no volume list was provided, a LOCATE macro instruction is issued to create a volume list. [SETFLAG]

Searches the UCBs for the UCB of a mounted volume with a volume serial number that matches the volume serial number in the volume list. [LOOKUP]

Issues DYNALLOC (SVC 99) to request allocation of a volume and device for the data set. [MATCHID]

Issues an OBTAIN macro to read in the data portion (last 96 bytes) of the F1 DSCB of the data set being protected. [GETDSCB]

Determines if the protection status bits of the F1 DSCB are to be modified. If no change of protection status is required, an exit is taken via BR 14. [FLAGDS]

When the protection status bits must be modified, enqueues the VTOC, reissues the OBTAIN macro instruction, modifies the status bits, writes the F1 DSCB back to the VTOC, and dequeues the VTOC. [NQ]

5 As each F1 DSCB is written back to its VTOC, checks for additional entries in the volume list. Makes another pass through this module for each volume list entry. When all volume list entries have been processed, frees work areas, restores registers 0, 1, and 15, lands the return address of the calling routine in register 14, and returns control to the caller via a branch on register 14. [STATUSOK]

A Completion Code:

Register 15	Explanation
0	The PASSWORD data set was successfully updated.
4	The password of the data set name was already in the PASSWORD data set.
8	The password of the data set name was not in the PASSWORD data set.
12	A control password is required or the one supplied is incorrect.
16	The supplied parameter list was incomplete or incorrect.
20	There was an I/O error in the password data set.
24**	The PASSWORD data set was full.
28	The validity check of the buffer address failed.
32*	The LOCATE macro failed. LOCATE's return code is in register 1 and the number of indexes searched is in register 0.
36*	The OBTAIN macro failed. OBTAIN's return code is in register 1.
40*	The DSCB could not be updated.
44	The PASSWORD data set does not exist.
48*	Tape data set cannot be protected.

Extended Description for Diagram 27 (Continued)

Register 15	Explanation
52*	Data set in use.
56*	The data set uses the virtual storage access method (VSAM).

*For these return codes, the PASSWORD data set has been updated, but the DSCB has not been flagged to indicate the protected status of the data set.

**For this return code, a message is written to the console indicating that the PASSWORD data set is full.

PROGRAM ORGANIZATION

The DADSM routines are supervisor call (SVC) routines, except for Extend and Partial Release. At system initialization, the DADSM routines are made resident in virtual storage in 4096-byte pages of the pageable link pack area (PLPA).

The DADSM routines transfer control among themselves using the IECRES macro instruction. The IECRES macro is documented in *OS/VS2 Open/Close/EOV Logic*. Except for the Partial Release and Extend, the DADSM routines return control to their callers using a direct branch. The Partial Release routine uses the XCTL macro when returning to Checkpoint/Restart. The Extend routine uses the XCTL macro instruction when returning to the catalog management routines.

Charts containing brief module descriptions and the logic that determines flow of control among the DADSM routines follow. These charts can be used to trace the flow of control, both backward and forward, through the DADSM object modules. The figures indicate conditions at exit from each object module that passes control, under non-error conditions, to more than one other module.

At the right of each box that represents a DADSM CSECT are register settings and work area contents used by each CSECT as input. Immediately following the input description is the description of the output from each CSECT: register settings, work area contents, and control blocks written to direct access storage devices.

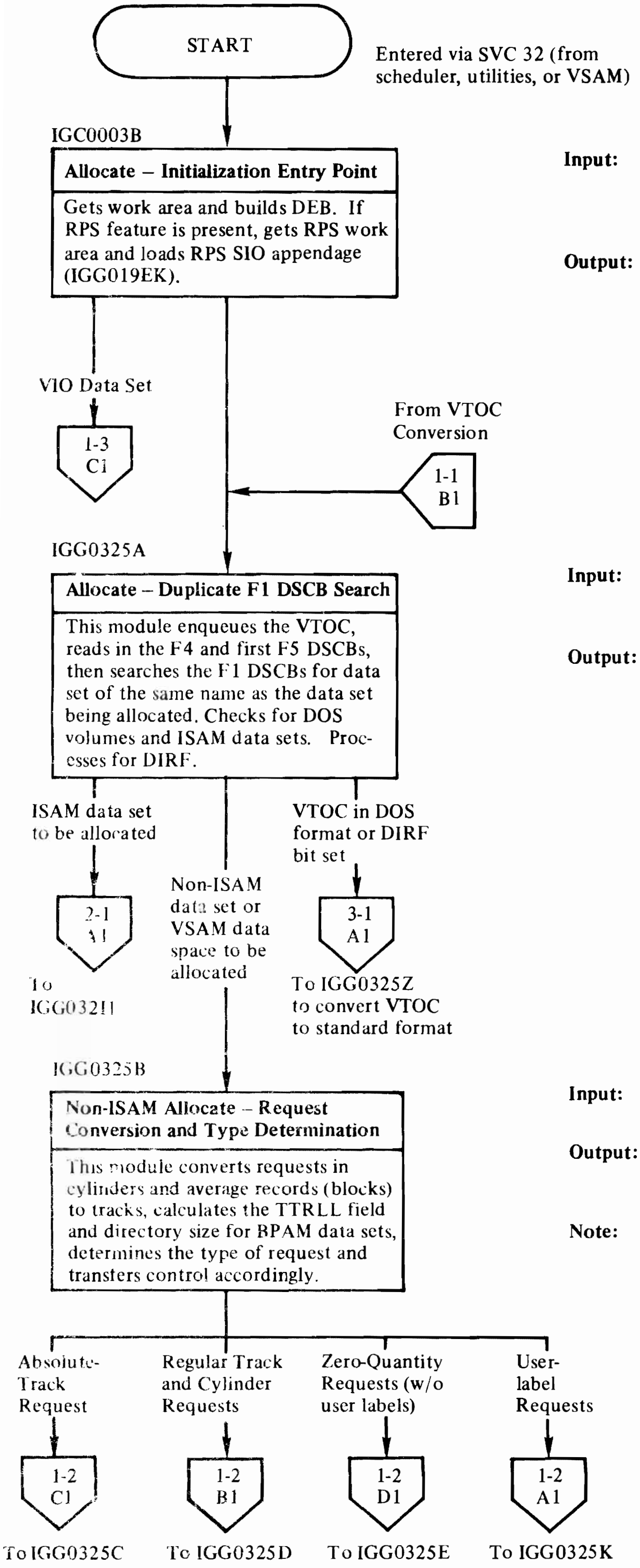

START
Entered via SVC 32 (from scheduler, utilities, or VSAM)
IGC0003B
Allocate – Initialization Entry Point
Gets work area and builds DEB. If RPS feature is present, gets RPS work area and loads RPS SIO appendage (IGG019EK).
VIO Data Set
1-3 C1
From VTOC Conversion
1-1 B1
IGG0325A
Allocate – Duplicate F1 DSCB Search
This module enqueues the VTOC, reads in the F4 and first F5 DSCBs, then searches the F1 DSCBs for data set of the same name as the data set being allocated. Checks for DOS volumes and ISAM data sets. Processes for DIRF.
ISAM data set to be allocated
2-1 A1
To IGG032II
Non-ISAM data set or VSAM data space to be allocated
VTOC in DOS format or DIRF bit set
3-1 A1
To IGG0325Z to convert VTOC to standard format
IGG0325B
Non-ISAM Allocate – Request Conversion and Type Determination
This module converts requests in cylinders and average records (blocks) to tracks, calculates the TTRLL field and directory size for BPAM data sets, determines the type of request and transfers control accordingly.
Absolute-Track Request
1-2 C1
To IGG0325C
Regular Track and Cylinder Requests
1-2 B1
To IGG0325D
Zero-Quantity Requests (w/o user labels)
1-2 D1
To IGG0325E
User-label Requests
1-2 A1
To IGG0325K

Chart 1. (Part 1 of 3) Allocate Initialization and Non-ISAM Allocation

Input: R0 → JFCB or partial DSCB (if bit 0 of R0 = 0, entry was from scheduler; if bit 0 of R0 = 1, entry was from a utility). R1 → fullword location containing a pointer to a UCB or if a VIO data set is being allocated, R1 → device code. R2 →

Output: DSAB address list if ISAM data set is being allocated. R4 ≠ 0, R7 → UCB or device code R11 → JFCB or partial DSCB, R13 → Allocate work area (DEB/DCB, ECB, IOB).

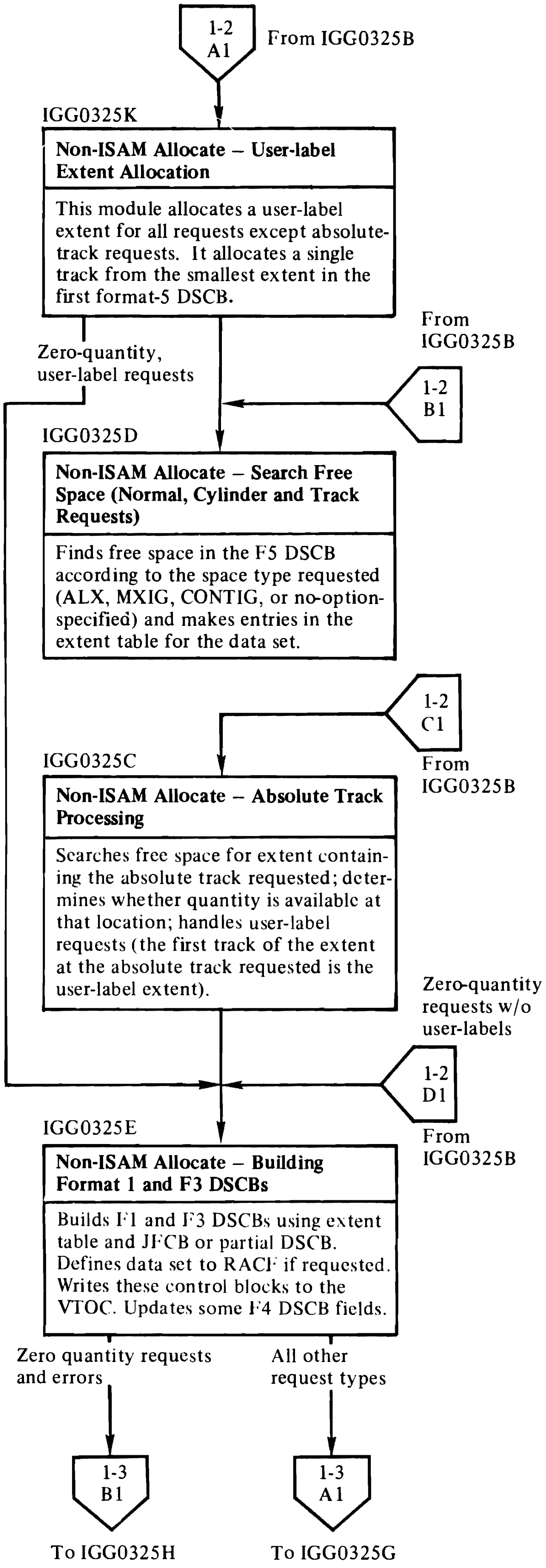
Input: R11 → JFCB or partial DSCB, R7 → UCB, R4 = 0 if entered from IGG0325Z (VTOC Conversion), R13 → Allocate work area (DEB/DCB, ECB, IOB).

Output: R13 → Allocate work area (F4 and first F5 DSCBs and their absolute track addresses, duplicate-name F1 DSCB, if found, DEB/DCB, IOB, ECB), R11 → JFCB or partial DSCB.

Input: R13 → Allocate work area (F4, first F5, DEB/DCB, IOB, ECB, channel program), R11 → JFCB or DSCB.

Output: Registers same as at input, except R10 = the space requested in tracks. A modified channel program will reside in the work area.

Note: All modules, except IGC0003B, IGG0325I and IGGOCLF2, passs control to IGG0325H when error conditions are encountered: see "DADSM Allocate Return Codes" for error codes and resultant messages.



Non-ISAM Allocate – User-label Extent Allocation

This module allocates a user-label extent for all requests except absolute-track requests. It allocates a single track from the smallest extent in the first format-5 DSCB.

Input: R13 → Allocate work area (data portion of F4 DSCB and first F5 DSCB and their addresses, DEB/DCB, IOB, ECB, channel program to read F5 DSCBs), R11 → JFCB or partial F1 DSCB.

Output: R13 →Allocate work area (contents of work area unchanged, except user-label extent will be added to the extent table and removed from the F5 DSCB).

Non-ISAM Allocate – Search Free Space (Normal, Cylinder and Track Requests)

Finds free space in the F5 DSCB according to the space type requested (ALX, MXIG, CONTIG, or no-option-specified) and makes entries in the extent table for the data set.

Input: R13 → Allocate work area (contents same as IGG0325K input), R10 = space requested in tracks, and R11 → JFCB or partial F1 DSCB.

Output: Work area now contains extent table entries for allocated extents; first F5 DSCB may not be in virtual storage.

Non-ISAM Allocate – Absolute Track Processing

Searches free space for extent containing the absolute track requested; determines whether quantity is available at that location; handles user-label requests (the first track of the extent at the absolute track requested is the user-label extent).

Input: Same as input to IGG0325K; in addition, R10 = space requested in tracks.

Output: R13 → Allocate work area (first F5 may have been overwritten with subsequent F5s, extent table contains a single entry; otherwise contents of work area unchanged).

Non-ISAM Allocate – Building Format 1 and F3 DSCBs

Builds F1 and F3 DSCBs using extent table and JFCB or partial DSCB. Defines data set to RACF if requested. Writes these control blocks to the VTOC. Updates some F4 DSCB fields.

Input: Same as input to IGG0325K; in addition, the work area will contain the extents allocated to this request in the extent table, if it is not a zero-quantity request; the channel program will have been modified to write F1 and F3 DSCBs.

Output: F1 and F3 DSCBs to the VTOC Registers and work areas are as at input. If directories were requested, the absolute track address (CCHH) of the first F1 extent will be saved in the work area. To IGG0325H for zero-quantity requests – R8 = 0; for error conditions, R8 = error code (non-zero).

Note: All modules except IGC0003B, IGG03251, and IGG0CLF2, transfer control to IGG0325H when error conditions are encountered; see "DADSM Allocate Return Codes" in the Diagnostic Aids section for error codes and resultant message.

Chart 1. (Part 2 of 3) Allocate Initialization and Non-ISAM Allocation

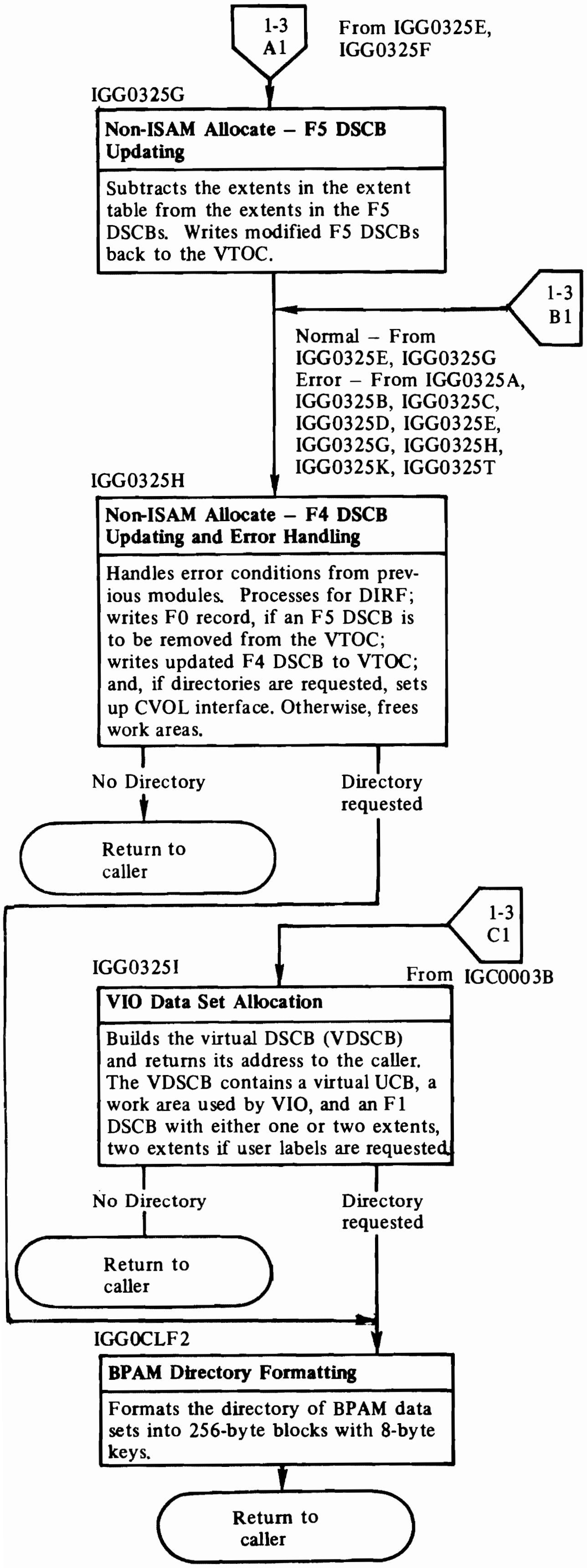

1-3
A1
From IGG0325E,
IGG0325F
IGG0325G
Non-ISAM Allocate – F5 DSCB Updating
Subtracts the extents in the extent table from the extents in the F5 DSCBs. Writes modified F5 DSCBs back to the VTOC.
1-3
B1
Normal – From
IGG0325E, IGG0325G
Error – From IGG0325A,
IGG0325B, IGG0325C,
IGG0325D, IGG0325E,
IGG0325G, IGG0325H,
IGG0325K, IGG0325T
IGG0325H
Non-ISAM Allocate – F4 DSCB Updating and Error Handling
Handles error conditions from previous modules. Processes for DIRF; writes F0 record, if an F5 DSCB is to be removed from the VTOC; writes updated F4 DSCB to VTOC; and, if directories are requested, sets up CVOL interface. Otherwise, frees work areas.
No Directory
Directory
requested
Return to
caller
1-3
C1
IGG0325I
From IGC0003B
VIO Data Set Allocation
Builds the virtual DSCB (VDSCB) and returns its address to the caller. The VDSCB contains a virtual UCB, a work area used by VIO, and an F1 DSCB with either one or two extents, two extents if user labels are requested.
No Directory
Directory
requested
Return to
caller
IGG0CLF2
BPAM Directory Formatting
Formats the directory of BPAM data sets into 256-byte blocks with 8-byte keys.
Return to
caller

Chart 1. (Part 3 of 3) Allocate Initialization and Non-ISAM Allocation

Input: R13 → Allocate work area (data portion of F4 DSCB, absolute address of F4 and first F5 DSCB, extent table, DEB/DCB, IOB, ECB, channel program to search for F0 DSCBs and read and write F5 DSAB(s)), R11 → JFCB.

Output: New and/or updated F5 DSCBs to VTOC. Registers and work areas are as at input.

Input: R13 → Allocate work area (updated F4 DSCB, control blocks and channel program to write to the VTOC).

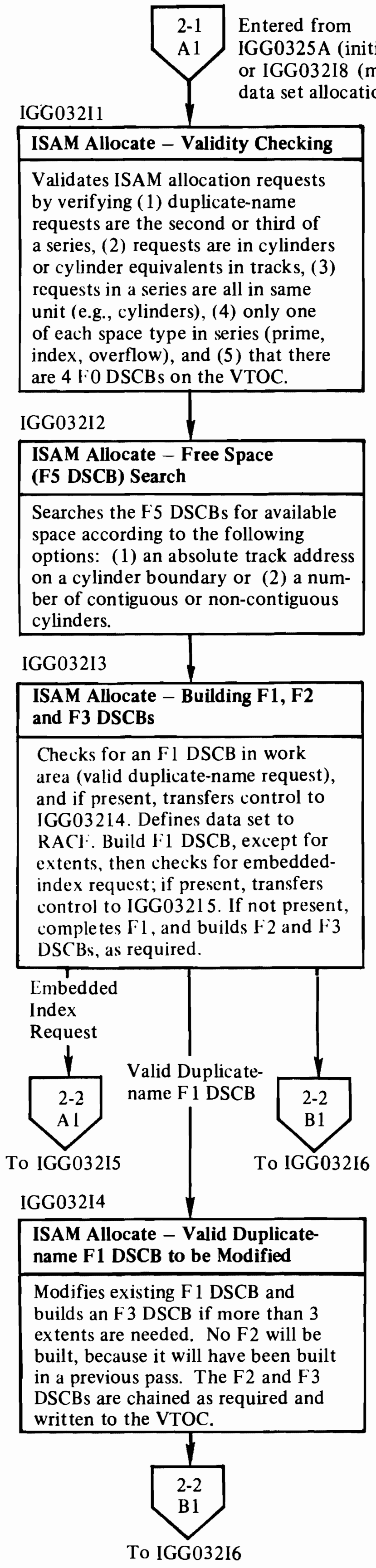
Output: F4 and F0 DSCBs written to VTOC. R15 = return code. Device code field now →VDSCB. R1 = number of directory blocks to be written and R2 = blocks/track for the particular DA device.

Input: R7 → device type, R11 → JFCB, and R13 → Allocate work area.

Output: A VDSCB will be built in the SWA and R15 = return code. Device code field now → VDSCB.

Input: R1 = number of directory blocks requested, R2 = number of directory blocks per track, R13 → Allocate work area.

Output: Formatted directory and return code in R15.



Entered from IGG0325A (initial entry) or IGG03218 (multivolume data set allocation)

IGG03211

ISAM Allocate – Validity Checking

Validates ISAM allocation requests by verifying (1) duplicate-name requests are the second or third of a series, (2) requests are in cylinders or cylinder equivalents in tracks, (3) requests in a series are all in same unit (e.g., cylinders), (4) only one of each space type in series (prime, index, overflow), and (5) that there are 4 F0 DSCBs on the VTOC.

Input: R11 → JFCB, R13 → Allocate work area (ADSABLST → list of DSABs, address of F4 DSCB, first F5 DSCB, F1 DSCB if a duplicate-name F1 DSCB was found, DEB/DCB, IOB, ECB, channel program).

Output: Registers and work area as at input; in addition, the work area now contains the amount of space requested on this track request is being processed. ATIOTPTR current entry in TIOT.

IGG03212

ISAM Allocate – Free Space (F5 DSCB) Search

Searches the F5 DSCBs for available space according to the following options: (1) an absolute track address on a cylinder boundary or (2) a number of contiguous or non-contiguous cylinders.

Input: Same as output from IGG03211

Output: Registers as at input. One or more extent table entries will now be contained in the extent table in the work area (DADSMTBL) and the first F5 DSCB may or may not be in the work area.

IGG03213

ISAM Allocate – Building F1, F2 and F3 DSCBs

Checks for an F1 DSCB in work area (valid duplicate-name request), and if present, transfers control to IGG03214. Defines data set to RACF. Build F1 DSCB, except for extents, then checks for embedded-index request; if present, transfers control to IGG03215. If not present, completes F1, and builds F2 and F3 DSCBs, as required.

Input: Same as output from IGG03212

Output: To IGG03216 – work area is as at input. An F1 DSCB (and F2 and F3 DSCBs, as required) will have been written to the VTOC.
To IGG03214 – the valid duplicate-name F1 DSCB will be contained in the work area, which is as at input.
To IGG03215 – A new F1 DSCB, except for extents, will have been built in the work area.

IGG03214

ISAM Allocate – Valid Duplicate-name F1 DSCB to be Modified

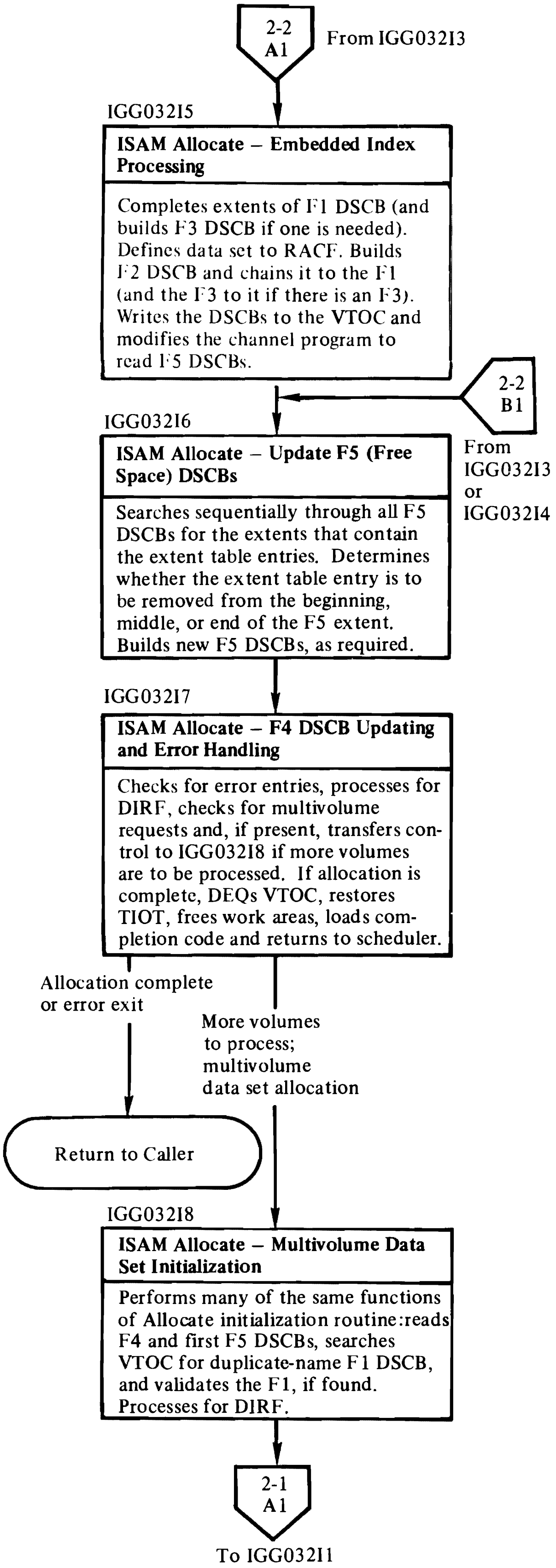
Modifies existing F1 DSCB and builds an F3 DSCB if more than 3 extents are needed. No F2 will be built, because it will have been built in a previous pass. The F2 and F3 DSCBs are chained as required and written to the VTOC.

Input: R13 → Allocate work area (valid duplicate-name F1 DSCB, extent table with one or more entries, F4 DSCB, DEB/DCB, IOB, ECB).

Output: Updated F1 DSCB and an F3 DSCB if one is needed. Work area contents as at entry.

Note: All modules transfer control to IGG03217 on error conditions; see "DADSM Allocated Return Codes" for the error codes issued by each module and the resultant messages.

Chart 2. (Part 1 of 2) ISAM Allocation



IGG03215

ISAM Allocate – Embedded Index Processing

Completes extents of F1 DSCB (and builds F3 DSCB if one is needed). Defines data set to RACF. Builds F2 DSCB and chains it to the F1 (and the F3 to it if there is an F3). Writes the DSCBs to the VTOC and modifies the channel program to read F5 DSCBs.

Input: R13 → Allocate work area (new F1 DSCB, except for extents, extent table with one or more entries, F4 DSCB, DEB/DCB, IOB, ECB, channel program).

Output: F1 and F2 (and F3 if needed) DSCBs chained and written to VTOC. Work area contents unchanged.

IGG03216

ISAM Allocate – Update F5 (Free Space) DSCBs

Searches sequentially through all F5 DSCBs for the extents that contain the extent table entries. Determines whether the extent table entry is to be removed from the beginning, middle, or end of the F5 extent. Builds new F5 DSCBs, as required.

Input: R13 → Allocate work area (extent table with one or more entries, F4 DSCB, address of first F5 DSCB).

Output: Updated F5 DSCBs written to VTOC.

IGG03217

ISAM Allocate – F4 DSCB Updating and Error Handling

Checks for error entries, processes for DIRF, checks for multivolume requests and, if present, transfers control to IGG03218 if more volumes are to be processed. If allocation is complete, DEQs VTOC, restores TIOT, frees work areas, loads completion code and returns to scheduler.

Input: Error entries (from IGG03211, IGG03212, IGG03213, IGG03214, IGG03215, IGG03216, IGG03218) – R5 = error code. Normal entry (from IGG03216) – R5 = 0, R13 → Allocate work area (F4 DSCB, ECB, DEB/DCB, IOB, channel program).

Output: Updated F4 DSCB written to VTOC.
To IGG03218 – R13 → Allocate work area.
R10 → DD entry in TIOT.
To Caller – R15 = completion code.

IGG03218

ISAM Allocate – Multivolume Data Set Initialization

Performs many of the same functions of Allocate initialization routine:reads F4 and first F5 DSCBs, searches VTOC for duplicate-name F1 DSCB, and validates the F1, if found. Processes for DIRF.

Input: R10 → DD statement entry in TIOT. R13 → Allocate work area (CTR field used to calculate address of UCB for this pass, DEB/DCB, IOB, ECB, channel program).

Output: R13 → Allocate work area (F4 DSCB, first F5 DSCB, F1 DSCB if valid duplicate-name DSCB found, DEB/DCB with new UCB pointer in DEB, IOB, ECB).

Chart 2. (Part 2 of 2) ISAM Allocation

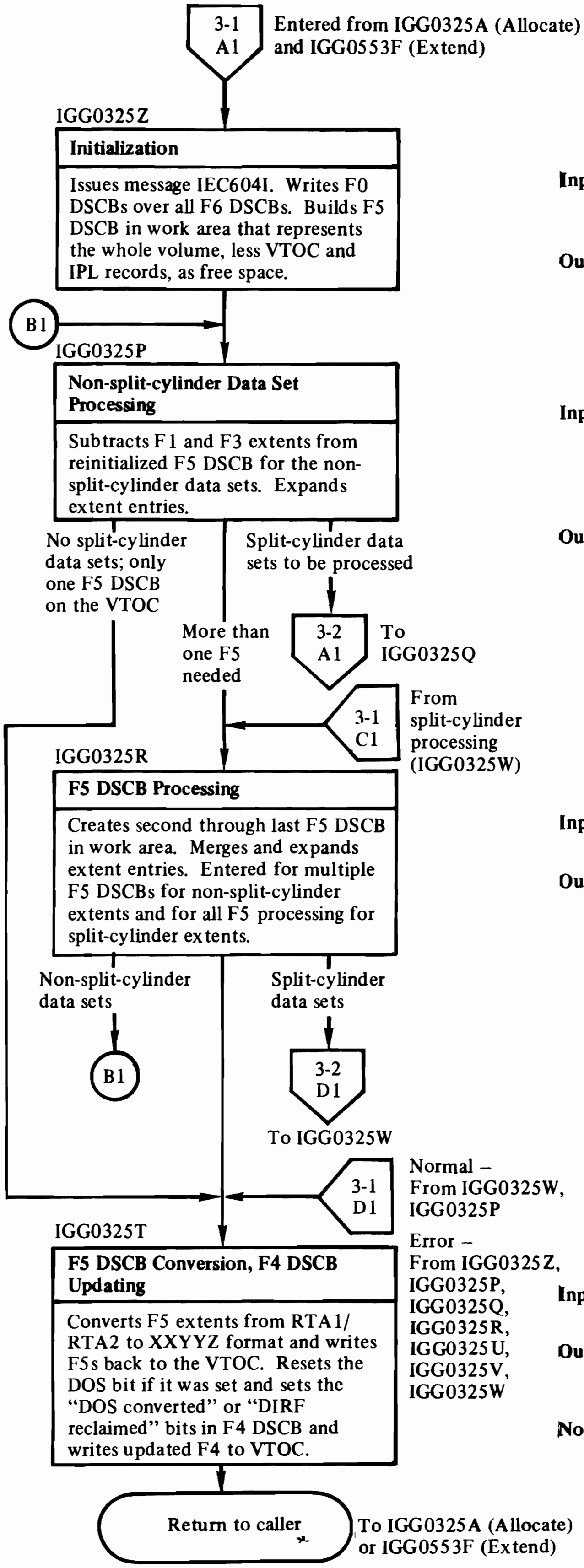

3-1 A1
Entered from IGG0325A (Allocate) and IGG0553F (Extend)
IGG0325Z
Initialization
Issues message IEC604I. Writes F0 DSCBs over all F6 DSCBs. Builds F5 DSCB in work area that represents the whole volume, less VTOC and IPL records, as free space.
B1
IGG0325P
Non-split-cylinder Data Set Processing
Subtracts F1 and F3 extents from reinitialized F5 DSCB for the non-split-cylinder data sets. Expands extent entries.
No split-cylinder data sets; only one F5 DSCB on the VTOC
Split-cylinder data sets to be processed
More than one F5 needed
3-2 A1
To IGG0325Q
3-1 C1
From split-cylinder processing (IGG0325W)
IGG0325R
F5 DSCB Processing
Creates second through last F5 DSCB in work area. Merges and expands extent entries. Entered for multiple F5 DSCBs for non-split-cylinder extents and for all F5 processing for split-cylinder extents.
Non-split-cylinder data sets
Split-cylinder data sets
B1
3-2 D1
To IGG0325W
3-1 D1
Normal – From IGG0325W, IGG0325P
Error – From IGG0325Z, IGG0325P, IGG0325Q, IGG0325R, IGG0325U, IGG0325V, IGG0325W
IGG0325T
F5 DSCB Conversion, F4 DSCB Updating
Converts F5 extents from RTA1/RTA2 to XXYYZ format and writes F5s back to the VTOC. Resets the DOS bit if it was set and sets the "DOS converted" or "DIRF reclaimed" bits in F4 DSCB and writes updated F4 to VTOC.
Return to caller
To IGG0325A (Allocate) or IGG0553F (Extend)

Chart 3. (Part 1 of 2) VTOC Conversion Routine

Input: R7→UCB, R13→Allocate work area (F4 DSCB, DEB/DCB, ECB, IOB, channel program).

Output: Same as input, except F5 in work area has been reinitialized. Channel program is modified.

Input: R13→Allocate work area. When entered form IGG0325Z work area contains reinitialized F5 DSCB. When entered form IGG0325R, R1 = 0 and Allocate work area contains last F1 or F3 DSCB read and first F5 DSCB.

Output: To IGG0325R – R13 → Allocate work area (F4 DSCB, last F1 or F3 DSCB read, and first F5 DSCB, and CCW12 field contains number of extents in extent table to be subtracted and the address of the first extent to be subtracted). The Allocate work area is also chained, via the work area prefix to the VTOC Conversion work area(s). To IGG0325Q – high-water mark (DS4HPCHR field) in the F4 DSCB is now valid. First F5 DSCB is now in VTOC Conversion work area extension. To IGG0325T – If there is more than one F5 DSCB or if more than one F5 DSCB was found on the VTOC the second and subsequent F5s (or their addresses) will be in the VTOC Conversion work area extension.

Input: R13→Allocate work area (F4 DSCB, last F1 or F3 read in, first F5, address of VTOC Conversion work area).

Output: Registers and work areas unchanged, except R1 = 0.

Input: R13→Allocate work area (F4 DSCB)→VTOC Conversion work area.

Output: Valid F5 and updated F4 written to VTOC. R4 = 0 to indicate reentry to IGG0325A. If the VTOC cannot be converted, issues message IEC602I.

Note: All VTOC Conversion modules transfer control to IGG0325T when error conditions are encountered.

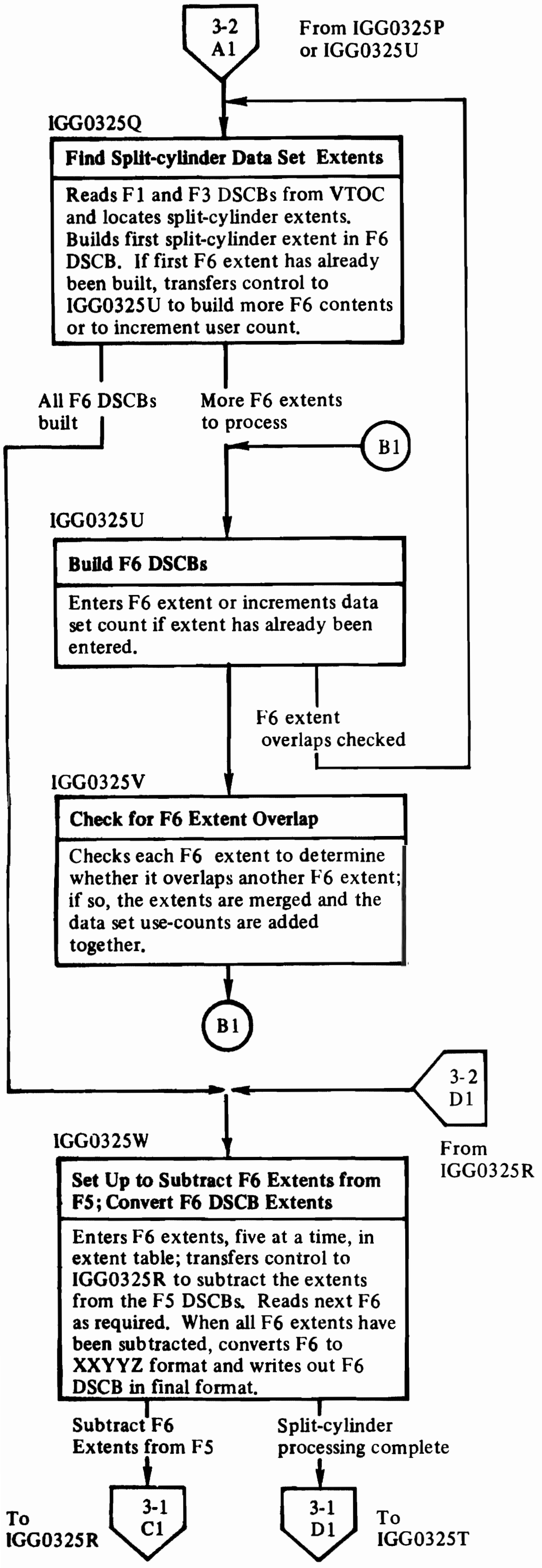

3-2 A1
From IGG0325P or IGG0325U
IGG0325Q
Find Split-cylinder Data Set Extents
Reads F1 and F3 DSCBs from VTOC and locates split-cylinder extents. Builds first split-cylinder extent in F6 DSCB. If first F6 extent has already been built, transfers control to IGG0325U to build more F6 contents or to increment user count.
All F6 DSCBs built
More F6 extents to process
B1
IGG0325U
Build F6 DSCBs
Enters F6 extent or increments data set count if extent has already been entered.
F6 extent overlaps checked
IGG0325V
Check for F6 Extent Overlap
Checks each F6 extent to determine whether it overlaps another F6 extent; if so, the extents are merged and the data set use-counts are added together.
B1
3-2 D1
From IGG0325R
IGG0325W
Set Up to Subtract F6 Extents from F5; Convert F6 DSCB Extents
Enters F6 extents, five at a time, in extent table; transfers control to IGG0325R to subtract the extents from the F5 DSCBs. Reads next F6 as required. When all F6 extents have been subtracted, converts F6 to XXYYZ format and writes out F6 DSCB in final format.
Subtract F6 Extents from F5
Split-cylinder processing complete
To IGG0325R
3-1 C1
3-1 D1
To IGG0325T

Chart 3. (Part 2 of 2) VTOC Conversion Routine

Input: R13 → Allocate work area → VTOC Conversion work area. From IGG0325P, Allocate work area contains F4 DSCB with valid DS4HPCHR field. From IGG0325U, work area contains F4 and first F6 DSCBs; R1 = 0 to indicate reentry from IGG0325U.

Output: To IGG0325U – R13 → Allocate work area (F4 DSCB, last F1 or F3 DSCB read, first F6 DSCB). R6 → current shared-cylinder entry in extent table. R8 → first F6 extent.
To IGG0325W – R13 → Allocate work area (F4 and first F6 DSCB).

Input: From IGG0325Q – R11 → JFCB or partial DSCB, R13 → Allocate work area (F4 DSCB, last F1 or F3 DSCB read in, first F6 DSCB), R6 → RTA1/RTA2 entry in extent table, R8 → first F6 extent.
From IGG0325V – R13 → work area (F4 DSCB, current F6 DSCB), R6 → RTA1/RTA2 of data set, R7 = extent number being compared, R8 → current F6 extent, R10 = number of extents in last F6, R1 = 0 to indicate reentry from IGG0325V.

Output: To IGG0325Q – R13 → Allocate work area (F4 DSCB, F1 or F3 DSCB being processed, and first F6 DSCBs), R1 = 0 to indicate reentry from IGG0325U.
To IGG0325V – R13 → Allocate work area (F4 DSCB, F1 or F3 DSCB being processed, and first F6 DSCBs), R6 → RTA1/RTA2 of data set, R7 = extent number being compared, R8 → current F6 extent, R10 = number of extents in last F6.

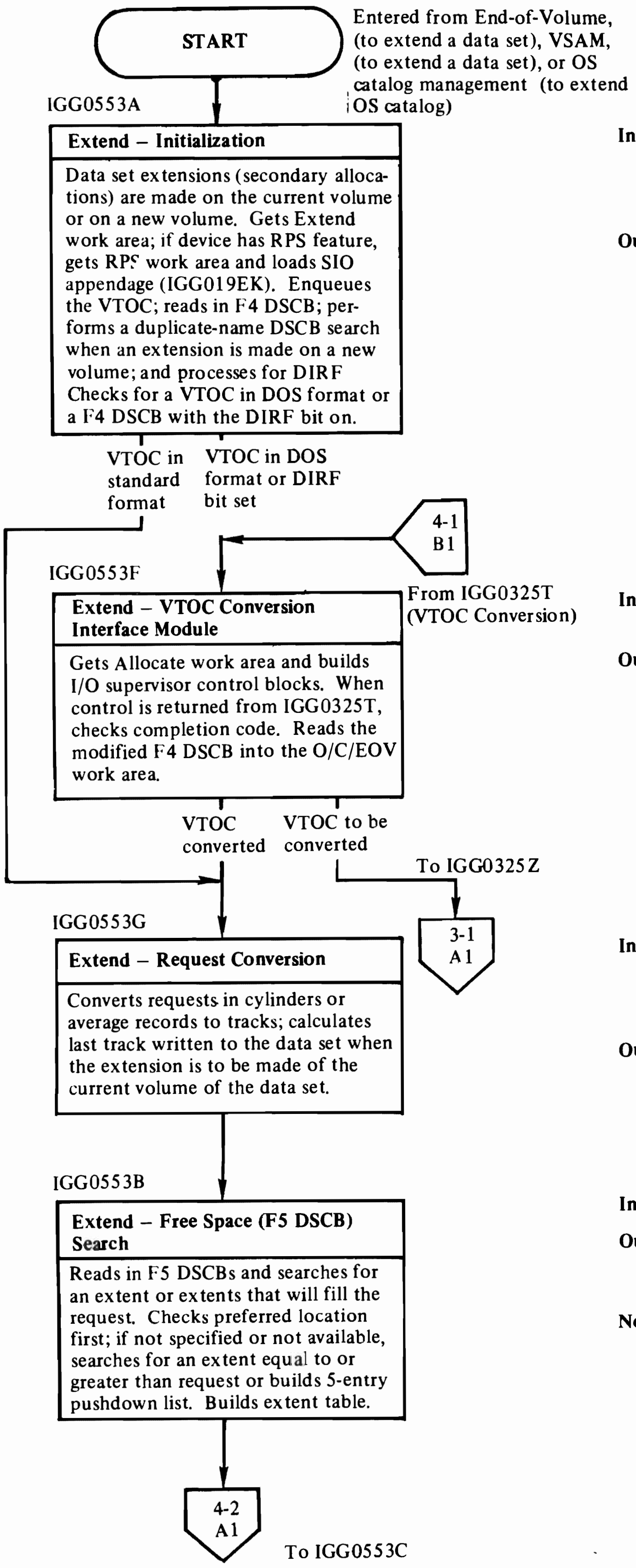
Input: R13 → Allocate work area (F4 DSCB and F6 DSCB being compared), R6 → RTA1/RTA2 of data set, R7 = number of the extent being compared, R8 → current extent, R10 = number of extents in last F6 DSCB.

Output: Same as input except R1 = 0 to indicate reentry from IGG0325V.

Input: R13 → Allocate work area.
From IGG0325Q -- Allocate work area contains F4 and first F6 DSCBs.
From IGG0325R – first F5 DSCB will also be in the work area and R1 = 0.

Output: To IGG0325R – R13 → Allocate work area (extent table entries, F4 DSCB, F6 DSCB, first F5 DSCB), R1 → parameter list (number of extents to be subtracted and → first extent).
To IGG0325T – R13 → Allocate work area (F4 DSCB and first F5 DSCB). Format 6 DSCBs are in final format, F5 DSCBs in temporary format.

Note: All VTOC Conversion nodules transfer control to IGG0325T when error conditions are encountered.



Entered from End-of-Volume, (to extend a data set), VSAM, (to extend a data set), or OS catalog management (to extend OS catalog)

Extend – Initialization

Data set extensions (secondary allocations) are made on the current volume or on a new volume. Gets Extend work area; if device has RPS feature, gets RPS work area and loads SIO appendage (IGG019EK). Enqueues the VTOC; reads in F4 DSCB; performs a duplicate-name DSCB search when an extension is made on a new volume; and processes for DIRF Checks for a VTOC in DOS format or a F4 DSCB with the DIRF bit on.

Input: R4 → O/C/EOV work area (JFCB of data set to be extended, I/O supervisor control blocks, channel program), R5→ DEB of data set to be extended, R10 → UCB, R13 = entry conditions (Diagram 15).

Output: R4 → O/C/EOV work area (contents as in input, plus F4 DSCB). R5 → DEB, R11 → Extend work area.

Extend – VTOC Conversion Interface Module

From IGG0325T (VTOC Conversion)

Gets Allocate work area and builds I/O supervisor control blocks. When control is returned from IGG0325T, checks completion code. Reads the modified F4 DSCB into the O/C/EOV work area.

Input: From IGG0553A – same as IGG0553A output. From IGG0325T – R8 = completion code.

Output: To IGG0325T – R7 → UCB, R13 → Allocate work area (F4 DSCB, address of first F5 DSCB, DEB/DCB, IOB, ECB, channel program, contents of R4, R5, and R11).
To IGG0553G – same as output from IGG0553A.

Extend – Request Conversion

Converts requests in cylinders or average records to tracks; calculates last track written to the data set when the extension is to be made of the current volume of the data set.

Input: R4 →O/C/EOV work area (F4 DSCB, address of first F5 DSCB, JFCB, DEB/DCB, IOB, ECB, channel program for reading from and writing to the VTOC), R5 → DEB, R11 → Extend work area.

Output: R12 = space requested in tracks, R13 = relative track address of preferred location, R11 → Extend work area, R4 →O/CEOV work area which is unchanged, R5→DEB.

Extend – Free Space (F5 DSCB) Search

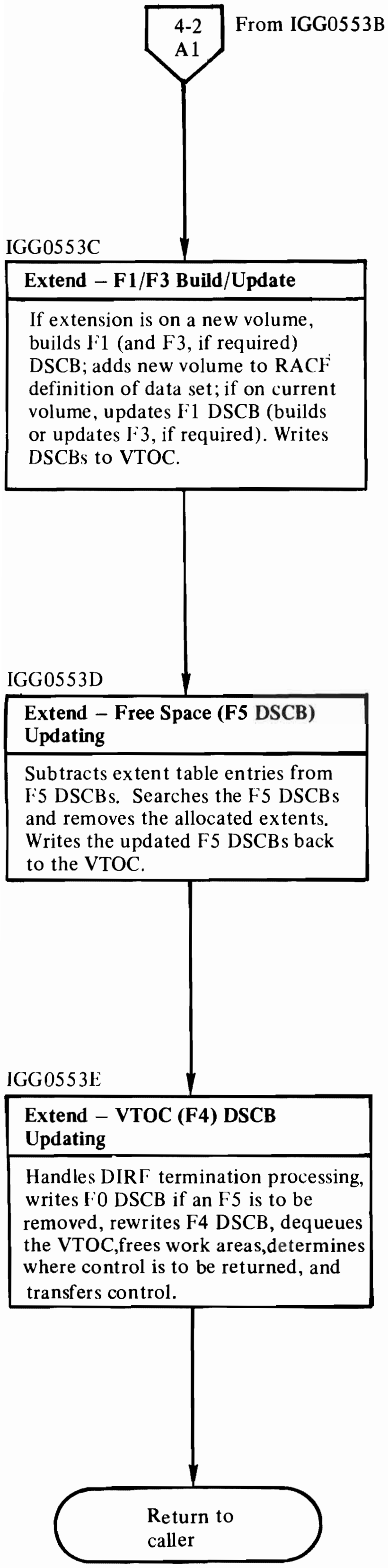
Reads in F5 DSCBs and searches for an extent or extents that will fill the request. Checks preferred location first; if not specified or not available, searches for an extent equal to or greater than request or builds 5-entry pushdown list. Builds extent table.

Input: See IGG0553G Output.

Output: R4 → O/C/EOV work area (same as at input, except that the extent table will now contain one or more entries), R11 → Extend work area, R5 → DEB.

Note: When error conditions are encountered, all extend modules transfer control to IGG0553E with entry code (or modified entry code) in R13.

Chart 4. (Part 1 of 2) Extend



IGG0553C

Extend – F1/F3 Build/Update

If extension is on a new volume, builds F1 (and F3, if required) DSCB; adds new volume to RACF definition of data set; if on current volume, updates F1 DSCB (builds or updates F3, if required). Writes DSCBs to VTOC.

Input: R4 → O/C/EOV work area (F4 DSCB, JFCB, DEB/DCB, IOB, channel program, extent table), R11 → Extend work area, R5 → DEB.

Output: New/updated F1 DSCB (and F3, as required). Contents of work areas as at input. R2 → extent table, R4 → O/C/EOV work area, R11 → Extend work area.

IGG0553D

Extend – Free Space (F5 DSCB) Updating

Subtracts extent table entries from F5 DSCBs. Searches the F5 DSCBs and removes the allocated extents. Writes the updated F5 DSCBs back to the VTOC.

Input: Register and work area same as IGG0553C output.

Output: Updated F5 DSCB(s). R4 → O/C/EOV work area (F4 DSCB; a pointer may be set to indicate where an F0 DSCB is to be written over an F5 DSCB to remove it from the VTOC). R11 → Extend work area.

IGG0553E

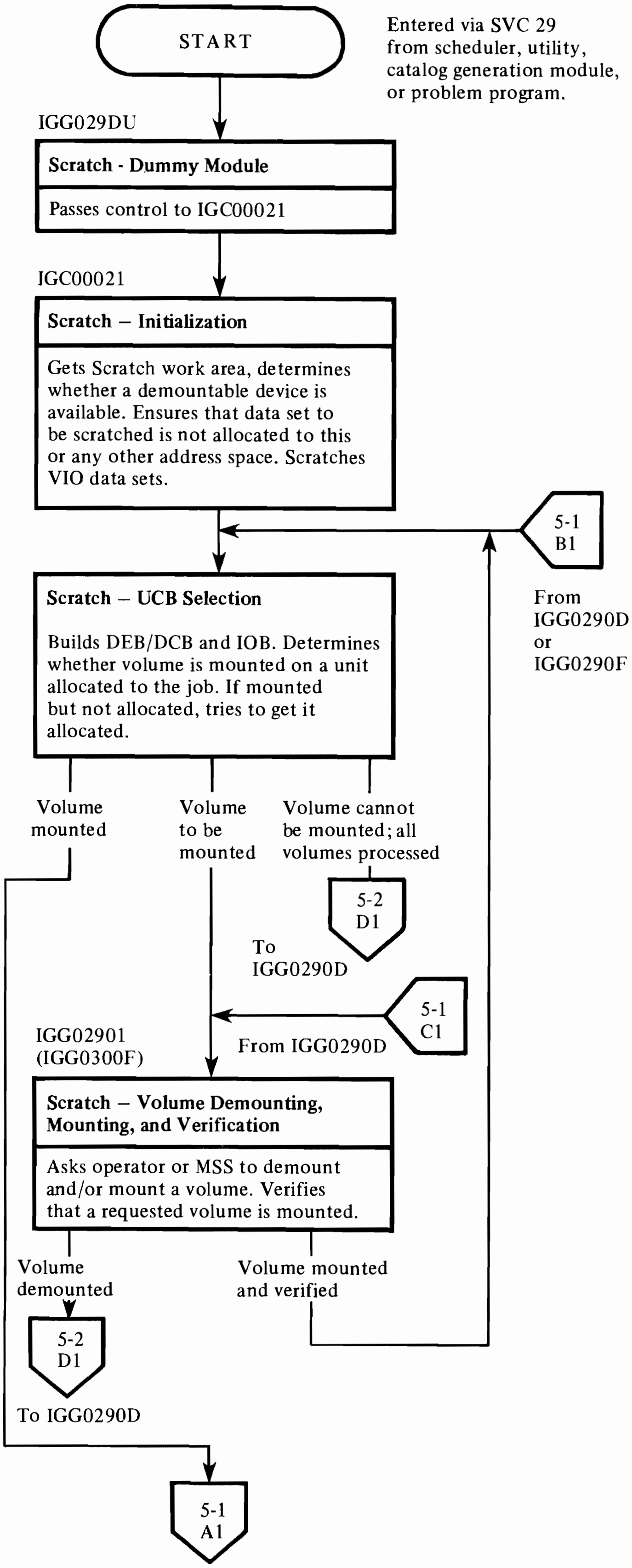
Extend – VTOC (F4) DSCB Updating

Handles DIRF termination processing, writes F0 DSCB if an F5 is to be removed, rewrites F4 DSCB, dequeues the VTOC,frees work areas,determines where control is to be returned, and transfers control.

Input: R4 → O/C/EOV work area (F4 DSCB and write-F0-indicator), R11 → Extend work area, R13 = 0 if no error has occurred.

Output: Updated F4 DSCB written to the VTOC.
To EOV, VSAM or OS catalog management - restores caller's registers from SVRB estended save area, R13 = return code. To VSAM – R0 and R14 = 0.
To VSAM – R0 and R14 = 0.

Chart 4. (Part 2 of 2) Extend


START
Entered via SVC 29 from scheduler, utility, catalog generation module, or problem program.
IGG029DU
Scratch - Dummy Module
Passes control to IGC00021
IGC00021
Scratch – Initialization
Gets Scratch work area, determines whether a demountable device is available. Ensures that data set to be scratched is not allocated to this or any other address space. Scratches VIO data sets.
5-1 B1
From IGG0290D or IGG0290F
Scratch – UCB Selection
Builds DEB/DCB and IOB. Determines whether volume is mounted on a unit allocated to the job. If mounted but not allocated, tries to get it allocated.
Volume mounted
Volume to be mounted
Volume cannot be mounted; all volumes processed
5-2 D1
To IGG0290D
5-1 C1
From IGG0290D
IGG02901 (IGG0300F)
Scratch – Volume Demounting, Mounting, and Verification
Asks operator or MSS to demount and/or mount a volume. Verifies that a requested volume is mounted.
Volume demounted
Volume mounted and verified
5-2 D1
To IGG0290D
5-1 A1

Input: R0 → UCB or = 0, R1 → Scratch parameter list (see Diagram 18 for contents).

Output: Registers unchanged.

Input: R0 → UCB or = 0, R1 → Scratch parameter list (see Diagram 18 for contents).

Output: R13 → Scratch work area (PRUCBPTR field → demonstrable UCB).

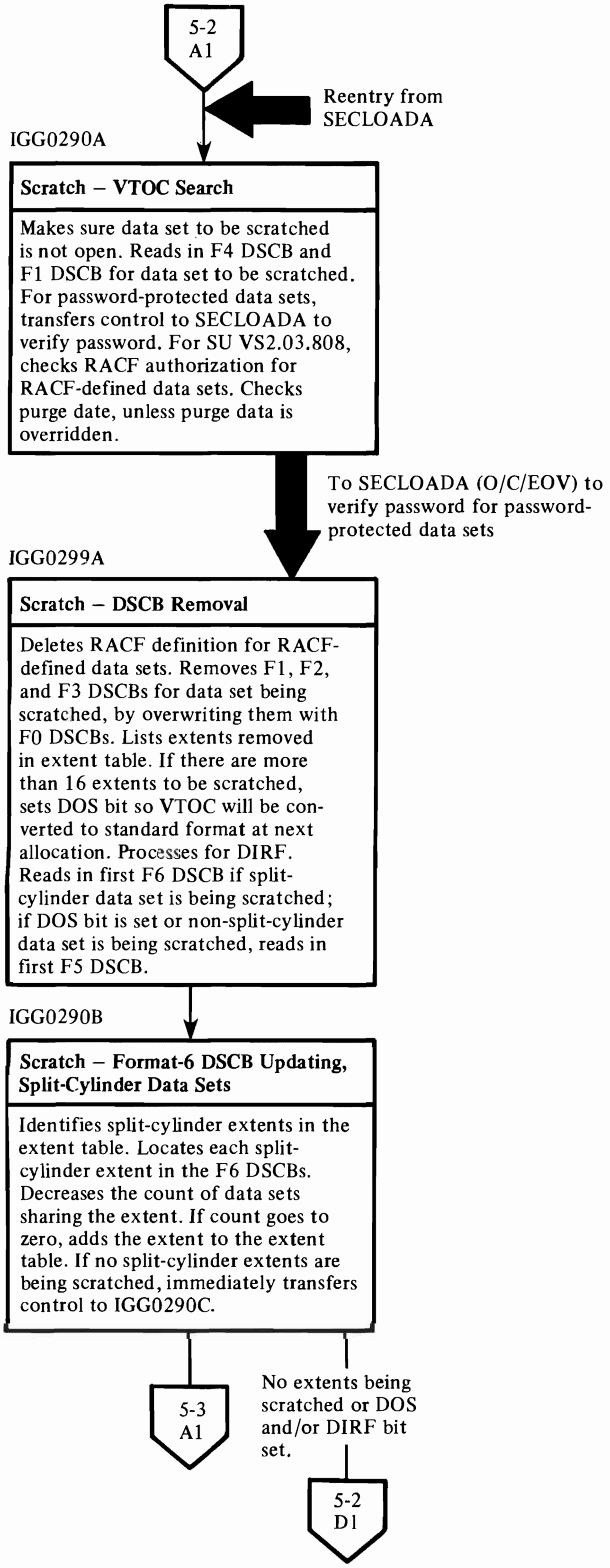
Input: R9 → CVT, R10 → Scratch parameter list, R13 → Scratch work area (VOLNUM = number of volumes remaining to be processed and VOLPTR →current entry in volume list). When reentered from IGG0290F, R14 = code indicating success or failure of volume mounting/verification.

Output: R8 → current entry in volume list. R13 → Scratch work area (DEB contains address of UCB for volume containing data set to be scratched).
To IGG02901 – R2 → DEB, R3 → volume serial number, R4 = 290E, R11 → UCB.

Input: R2 → DEB, R3 → volume serial number of volume to be demounted or mounted, R4 = last four characters of 1D of module to receive control next, R5 = number of volumes to be scratched, R8 → current entry in volume list, R10 → Scratch parameter list, R13 → Scratch work area, R11 → UCB.

Output: Register contents are the same as at entry, except R14 = a return code. Messages 1F0502E and/or IEC601D are written to operator. Message IEC1081 is written to TSO terminal. Volumes demounted, mounted, and/or verified.

Chart 5. (Part 1 of 3) Scratch



5-2
A1

Reentry from SECLOADA

IGG0290A

Scratch – VTOC Search

Makes sure data set to be scratched is not open. Reads in F4 DSCB and F1 DSCB for data set to be scratched. For password-protected data sets, transfers control to SECLOADA to verify password. For SU VS2.03.808, checks RACF authorization for RACF-defined data sets. Checks purge date, unless purge data is overridden.

Input: From IGG0290E – see IG0290F output. From SECLOADA – R10 = password status code. R13 → Scratch work area (DEB/DCB, IOB, ECB, channel program). R8 → current Scratch parameter list entry.

Output: To SECLOADA – R1 → DSAB, R4 → data portion of F1 DSCB, R5 → IOB, R10 = X'03 00 00 00' to identify a request from Scratch, R11 → data set name.
To IGG0299A – R13 → Scratch work area (DEB/DCB, IOB, ECB, channel program, F1 DSCB, F4 DSCB, address of first F5 DSCB).

Note: When error conditions are encountered, all Scratch modules except IGG0290F set the appropriate error code (if any) in the status byte of the current volume list entry and branch to IGG0290D. IGG0290F returns to either IGG0290E or IGG0290D.

To SECLOADA (O/C/EOV) to verify password for password-protected data sets

IGG0299A

Scratch – DSCB Removal

Deletes RACF definition for RACF-defined data sets. Removes F1, F2, and F3 DSCBs for data set being scratched, by overwriting them with F0 DSCBs. Lists extents removed in extent table. If there are more than 16 extents to be scratched, sets DOS bit so VTOC will be converted to standard format at next allocation. Processes for DIRF. Reads in first F6 DSCB if split-cylinder data set is being scratched; if DOS bit is set or non-split-cylinder data set is being scratched, reads in first F5 DSCB.

Input: R8 → current volume list entry, R13 →Scratch work area (F4 DSCB, F1 DSCB, address of first F5 DSCB, DEB/DCB, IOB, LCB, channel program).

Output: F1, F2, and F3 DSCBs for data set being scratched will have been overwritten with F0 DSCBs. R13 → Scratch work area (contents as at input, except it now contains extent table entries and the first F6 DSCB if a split-cylinder data set is being scratched, or the first F5 DSCB if a non-split-cylinder data set is being scratched). F4 DSCB will have been written a VTOC with DIRF bit set. R8 → volume list entry being processed.

IGG0290B

Scratch – Format-6 DSCB Updating, Split-Cylinder Data Sets

Identifies split-cylinder extents in the extent table. Locates each split-cylinder extent in the F6 DSCBs. Decreases the count of data sets sharing the extent. If count goes to zero, adds the extent to the extent table. If no split-cylinder extents are being scratched, immediately transfers control to IGG0290C.

Input: R8 → current volume list entry, R13 → Scratch work area (F4 DSCB, first F6 DSCB if split-cylinder data sets to be scratched, F5 DSCB if non-split-cylinder data set to be scratched, DEB/DCB, IOB, ICB, channel program, extent table with split-cylinder extents marked with X'FF'.)

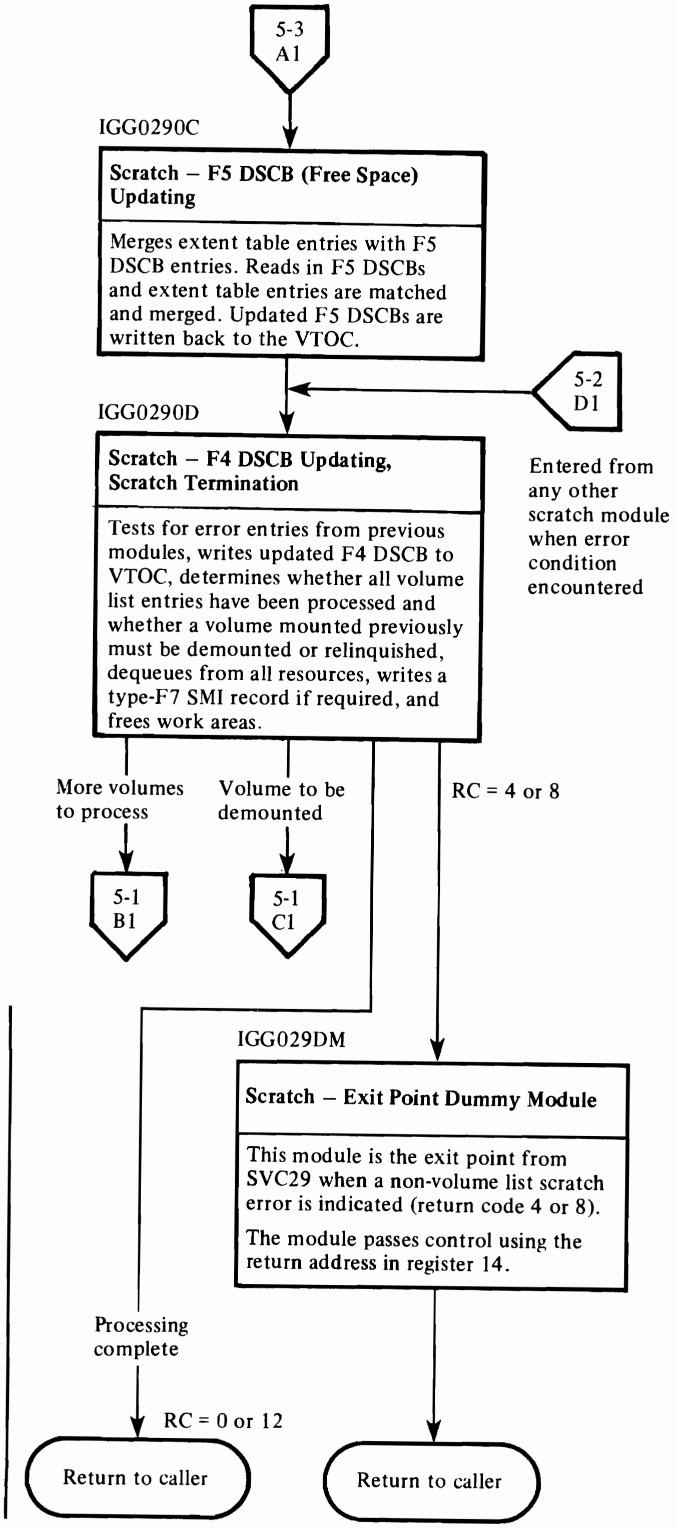
Output: Updated F6 DSCBs written to VTOC. Extent table now contains only free extents (split-cylinder extents with the other data sets continuing to reside in the extent have been removed from extent table). R13 → Scratch work area (F4 DSCB, extent table, first F5 DSCB, DEB/DCB, IOB, ECB, channel program). R8 → volume list entry.

5-3
A1

No extents being scratched or DOS and/or DIRF bit set.

5-2
D1

Chart 5. (Part 2 of 3) Scratch



IGG0290C

Scratch – F5 DSCB (Free Space) Updating

Merges extent table entries with F5 DSCB entries. Reads in F5 DSCBs and extent table entries are matched and merged. Updated F5 DSCBs are written back to the VTOC.

Input: R8 → current volume list entry, R13 →Scratch work area (F4 DSCB, first F5 DSCB, extent table, DEB/DCB, IOB, ECB, channel program).

Output: Updated F5 DSCBs written back to VTOC. Work area now contains only F4 DSCB and I/O supervisor control blocks. R8 → current volume list entry.

IGG0290D

Scratch – F4 DSCB Updating, Scratch Termination

Tests for error entries from previous modules, writes updated F4 DSCB to VTOC, determines whether all volume list entries have been processed and whether a volume mounted previously must be demounted or relinquished, dequeues from all resources, writes a type-F7 SMI record if required, and frees work areas.

Input: R8 → current volume list entry, R13 → Scratch work area (F4 DSCB, EROCDE field = error code if an error has been encountered in previous processing).

Output: Updated F4 DSCB written to the VTOC.
To caller R15 = return code 0 or 12.
To IGG0290E R10 → Scratch parameter list. R13 → Scratch work area (VOLNUM = number of volumes remaining to process, VOLPER → next entry in the volume list).
To IGG0290F R2 → DEB, R3 → volume serial number, R4 = 290D, R8 → current volume list entry, R11 → UCB, R13 → Scratch work area.
To IGG029DM – R1 → Scratch parameter list. R14 = return address, R15 = return code 4 or 8.

IGG029DM

Scratch – Exit Point Dummy Module

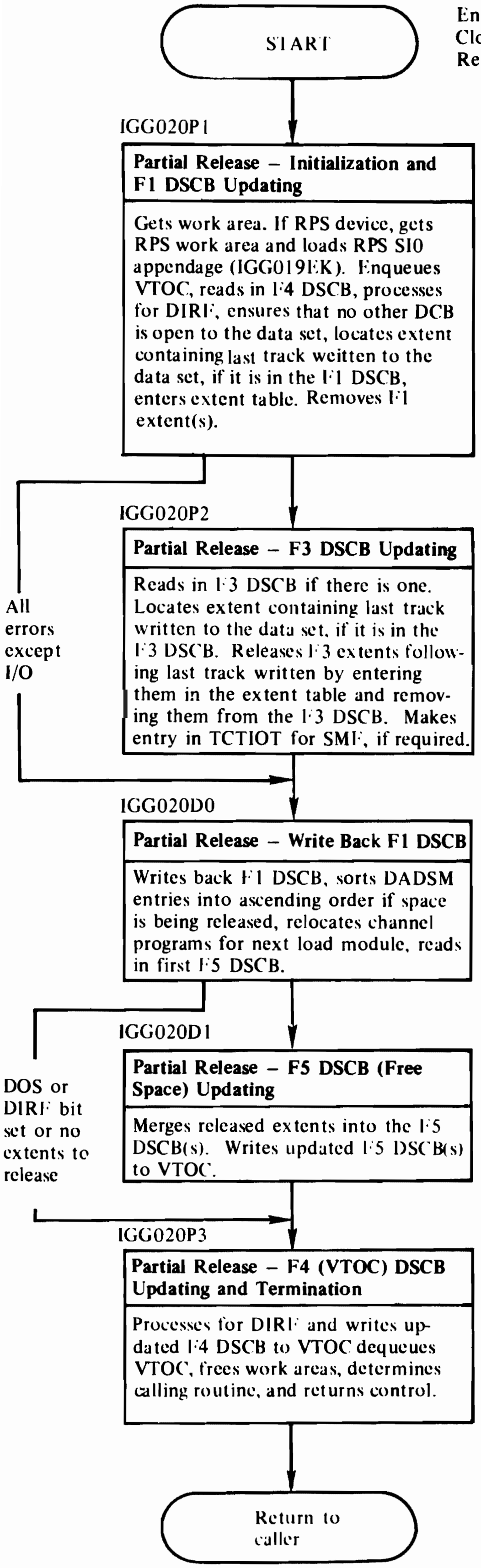
This module is the exit point from SVC29 when a non-volume list scratch error is indicated (return code 4 or 8).

The module passes control using the return address in register 14.

Input: R1 – scratch parameter list.
R14 = return address.
R15 = return code 4 or 8.

Output: To caller –
R15 = return code 4 or 8.

Chart 5. (Part 3 of 3) Scratch


START
Entered from Close and Checkpoint/ Restart
IGG020P1
Partial Release – Initialization and F1 DSCB Updating
Gets work area. If RPS device, gets RPS work area and loads RPS SIO appendage (IGG019EK). Enqueues VTOC, reads in F4 DSCB, processes for DIRF, ensures that no other DCB is open to the data set, locates extent containing last track weitten to the data set, if it is in the F1 DSCB, enters extent table. Removes F1 extent(s).
All errors except I/O
IGG020P2
Partial Release – F3 DSCB Updating
Reads in F3 DSCB if there is one. Locates extent containing last track written to the data set, if it is in the F3 DSCB. Releases F3 extents following last track written by entering them in the extent table and removing them from the F3 DSCB. Makes entry in TCTIOT for SMF, if required.
IGG020D0
Partial Release – Write Back F1 DSCB
Writes back F1 DSCB, sorts DADSM entries into ascending order if space is being released, relocates channel programs for next load module, reads in first F5 DSCB.
DOS or DIRF bit set or no extents to release
IGG020D1
Partial Release – F5 DSCB (Free Space) Updating
Merges released extents into the F5 DSCB(s). Writes updated F5 DSCB(s) to VTOC.
IGG020P3
Partial Release – F4 (VTOC) DSCB Updating and Termination
Processes for DIRF and writes updated F4 DSCB to VTOC dequeues VTOC, frees work areas, determines calling routine, and returns control.
Return to caller

Input: R2 → DCB of data set to be released, R4 → O/C/EOV work area (data portion of F1 DSCB, JFCB of data set to be released, DEB/DCB, IOB, ECB, channel program). R7 → Restart work area (if entered from Restart). R10 → UCB of data set to be released.

Output: R13 → O/C/EOV work area (F1 DSCB, DEB/DCB, IOB, ECB, channel program). R11 → Partial Release work area (F4 DSCB, extent table if F1 DSCB extent(s) were released). F4 DSCB will have been written to VTOC with DIRF bit set. R5 = number of extents in extent table.

Input: See IGG020P1 output.

Output: Modified F3 written to VTOC, or if all extents were released in the F3 DSCB, an F0 DSCB will be written over it. TCTIOT will be updated if SMF information has been requested. Number of extents remaining on the volume for this data set will be updated in the F1 DSCB in the O/C/EOV work area, R11 Partial Release work area (F3 DSCB is there is one, F4 DSCB, extent table).

Input: See IGG020P2 output; also if R2 contains anything other than a zero or negative value, an error (other than I/O error) was detected in either IGG020P1 or IGG020P2.

Output: F1 DSCB written back to VTOC, DADSM table sorted in ascending order, first F5 DSCB read in. R9 = number of extents to be released, R11→Partial release work area (first F5 DSCB, F4 DSCB, sorted extent table), R13 → I/O support work area (updated F1 DSCB).

Input: R11→Partial release work area (F4 DSCB, first F5 extent table).

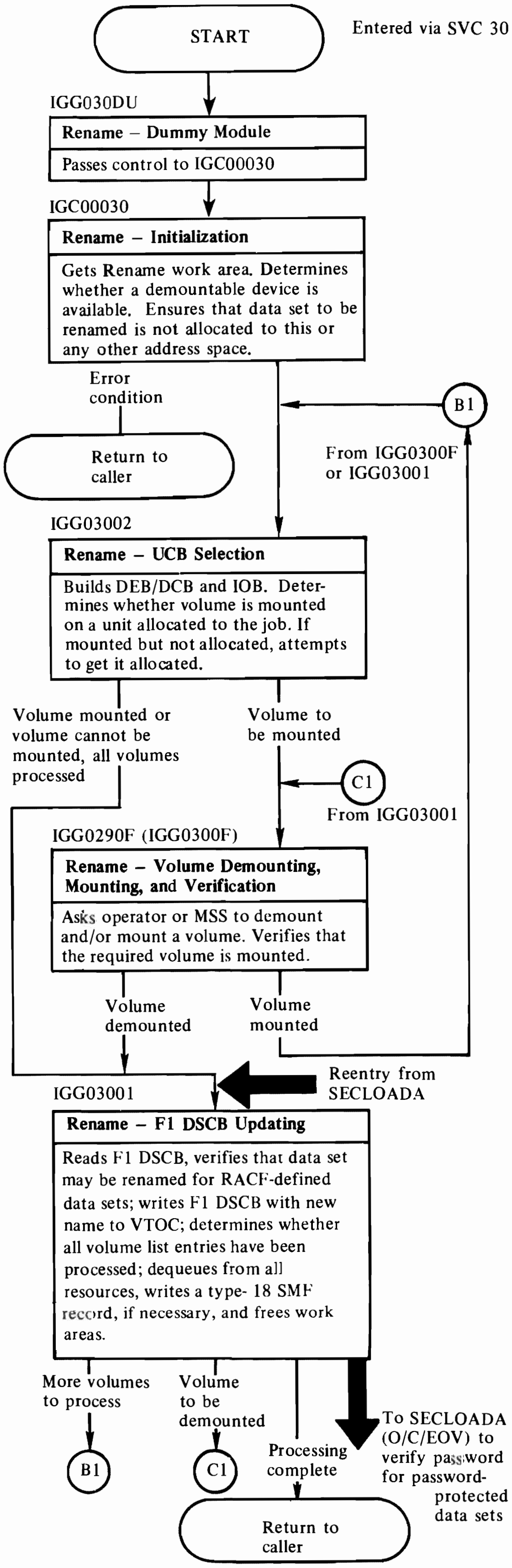
Output: Updated F5 DSCBs to VTOC.

Input: Error R2 - error code (permanent I/O error, no space in VTOC, extent not found, another DCB is open to the data set). Normal - R11 Partial Release work area (updated F4 DSCB), R13 O/C/EOV work area (F1 DSCB, DEB/DCB, IOB, ECB, channel program).

Output: F4 DSCB written to VTOC. Updated F1 DSCB in O/C/EOV work area. The Partial Release work area will have been freed, as well as the RPS work area, if one existed.

Note: All Partial Release modules transfer control to IGG020P3 when I/O errors are encountered. IGG020P1 or IGG020P2 transfers control to IGG020D0 for all errors except I/O. IGG020P2 also transfers control to IGG020D0 for normal processing. IGG020P2 then writes back the F1 DSCB to the VTOC and, if an error was detected in IGG020P1 or IGG020P2, or if an error occured in IGG020D0, control is transferred to IGG020P3.

Chart 6. Partial Release


START
Entered via SVC 30
IGG030DU
Rename – Dummy Module
Passes control to IGC00030
IGC00030
Rename – Initialization
Gets Rename work area. Determines whether a demountable device is available. Ensures that data set to be renamed is not allocated to this or any other address space.
Error condition
Return to caller
B1
From IGG0300F or IGG03001
IGG03002
Rename – UCB Selection
Builds DEB/DCB and IOB. Determines whether volume is mounted on a unit allocated to the job. If mounted but not allocated, attempts to get it allocated.
Volume mounted or volume cannot be mounted, all volumes processed
Volume to be mounted
C1
From IGG03001
IGG0290F (IGG0300F)
Rename – Volume Demounting, Mounting, and Verification
Asks operator or MSS to demount and/or mount a volume. Verifies that the required volume is mounted.
Volume demounted
Volume mounted
Reentry from SECLOADA
IGG03001
Rename – F1 DSCB Updating
Reads F1 DSCB, verifies that data set may be renamed for RACF-defined data sets; writes F1 DSCB with new name to VTOC; determines whether all volume list entries have been processed; dequeues from all resources, writes a type- 18 SMF record, if necessary, and frees work areas.
More volumes to process
B1
Volume to be demounted
C1
Processing complete
To SECLOADA (O/C/EOV) to verify password for password-protected data sets
Return to caller

Input: R1 → Rename parameter list (see Diagram 24 for contents) R0 → WCB or = 0.

Output: Registers unchanged.

Input: R1 → Rename parameter list (see Diagram 24 for contents). R0 → UCB or = 0.

Output: R9 → CVT, R10 → Rename parameter list, R13 → Rename work area (PRUCBPTR field → UCB representing a demountable volume), R15 = error code, if an error was encountered.

Input: R9 → CVT, R10 → Rename parameter list, R13 → Rename work area (VOLNUM = number of volumes to be processed, VOLPTR → current entry in volume list). When reentered from IGG0290F, R14 = code indicating success or failure of volume mounting/verification.

Output: R8 → current volume list entry, R9 → CVT, R10 → Rename parameter list.
To IGG0300F – R2 → DEB, R3 → volume serial number, R4 = 3002, R11 → UCB, R13 → Rename work area.

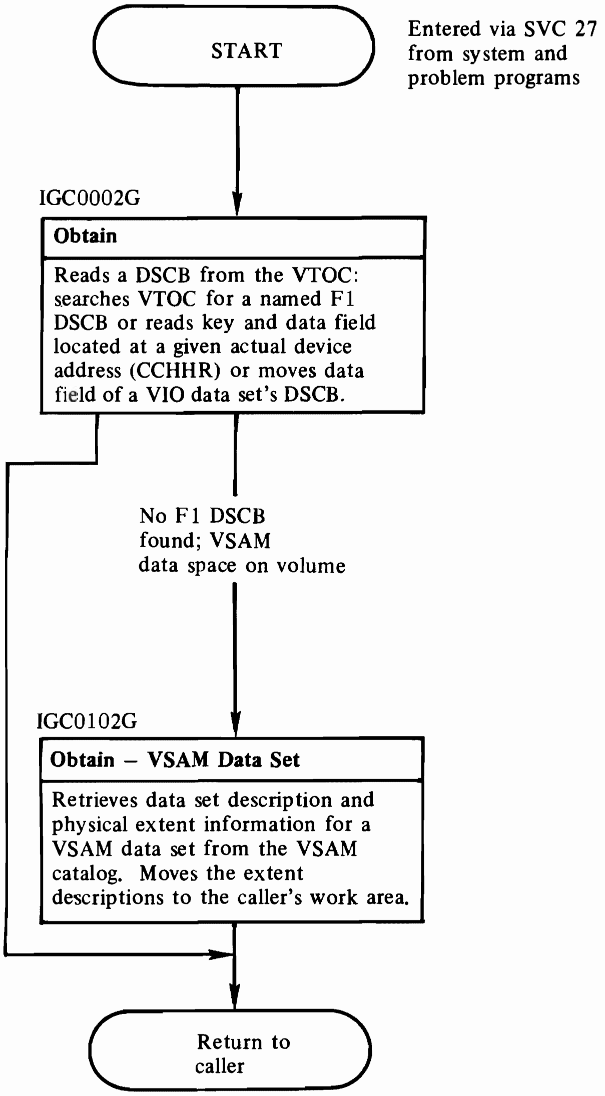
Input: R2 → DEB, R3 → volume serial number of volume to be demounted or mounted, R4 = last four characters of ID of module to receive control next, R5 = number of volumes to be renamed, R8 → current entry in volume list, R10 → Rename parameter list, R13 → Rename work area.

Output: Register contents are the same as at entry, except R14 = a return code. Messages IEC502E and/or IEC601D are written to operator, and message IEC108I is written to the TSO terminal. Volumes demounted, mounted, and/or verified.

Input: R8 → current entry in volume list, R10 → Rename parameter list, R13 → Rename work area (DEB/DCB, IOB, ECB, channel program). SMF record type-18 to SMF data set.
From SECLOADA – R10 = password status code, R13 → Rename work area (OLDPLPTR → Rename parameter list).

Output: To SECLOADA– R1 → DSAB, R4 → data portion of F1 DSCB, R5 → IOB, R10 = X'02 00 00 00' to identify Rename request, R11 → old data set name, R13 → Rename work area.
To IGG0300F (volume to be demounted) – R2 → DEB, R3 → volume serial number, R4 = 3001, R11 → UCB, R13 → Rename work area.
To IGG03002 (more volumes to process) – rename F1 DSCB has been written to VTOC, R10 → volume list.
Return to caller – R15 = completion code.

Chart 7. Rename


START
Entered via SVC 27 from system and problem programs
IGC0002G
Obtain
Reads a DSCB from the VTOC: searches VTOC for a named F1 DSCB or reads key and data field located at a given actual device address (CCHHR) or moves data field of a VIO data set's DSCB.
No F1 DSCB found; VSAM data space on volume
IGC0102G
Obtain – VSAM Data Set
Retrieves data set description and physical extent information for a VSAM data set from the VSAM catalog. Moves the extent descriptions to the caller's work area.
Return to caller

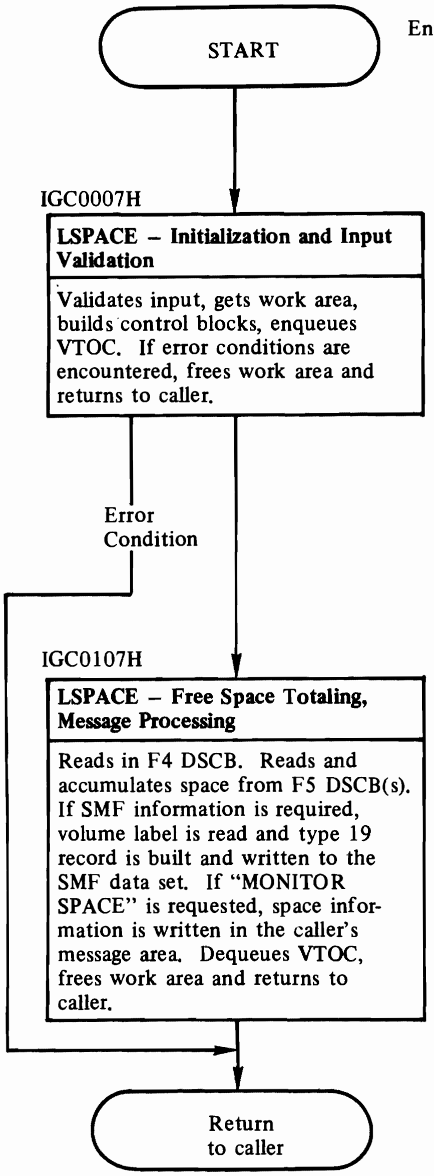
Input: R1 → Obtain parameter list (option code,→to data set name or to absolute track address, volume serial number, 140-byte work area provided by caller).

Output: Search option – data portion of named DSCB and address (CCHHR) of DSCB. Seek option – key and data protion of DSCB. R15 = return code.

Input: R10→obtain parameter list (option code→data set name,→volume serial number,→140-byte work area provided by caller); R13 → Obtain work area.

Output: Data portion of named DSCB. R15 = return code.

Chart 8. Obtain


START
IGC0007H
LSPACE – Initialization and Input Validation
Validates input, gets work area, builds control blocks, enqueues VTOC. If error conditions are encountered, frees work area and returns to caller.
Error Condition
IGC0107H
LSPACE – Free Space Totaling, Message Processing
Reads in F4 DSCB. Reads and accumulates space from F5 DSCB(s). If SMF information is required, volume label is read and type 19 record is built and written to the SMF data set. If "MONITOR SPACE" is requested, space information is written in the caller's message area. Dequeues VTOC, frees work area and returns to caller.
Return to caller

Entered via SVC 78

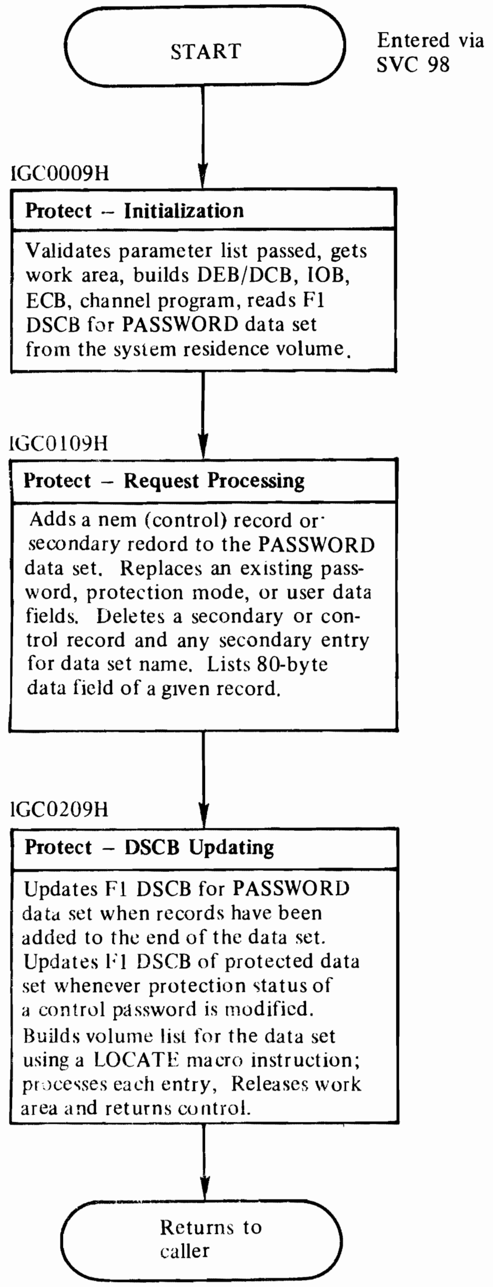
Input: R0 → UCB, R1 = SMF code in high order byte and → message area in low order bytes.

Output: R7 → UCB, R9 → "MONITOR SPACE" message area, R13 → LSPACE work area (DEB/DCB, IOB, ECB, channel program to read F4 and first F5 DSCB, absolute address of F4 DSCB). If the device has the RPS feature, the channel program will have been modified to contain a set-sector command. If an error is encountered, an error message will be moved to the caller's message area. R15 = completion code.

Input: See output from IGC0007H.

Output: If SMF information is requested, a type 19 record is written to the SMF data set. If "MONITOR SPACE" command has been issued, accumulated extent, cylinder, and track information will be moved to the caller's message area, pointed to by R1. R15 = return code.

Chart 9. LSPACE



START

Entered via SVC 98

IGC0009H

Protect – Initialization

Validates parameter list passed, gets work area, builds DEB/DCB, IOB, ECB, channel program, reads F1 DSCB for PASSWORD data set from the system residence volume.

IGC0109H

Protect – Request Processing

Adds a nem (control) record or· secondary redord to the PASSWORD data set. Replaces an existing password, protection mode, or user data fields. Deletes a secondary or control record and any secondary entry for data set name. Lists 80-byte data field of a given record.

IGC0209H

Protect – DSCB Updating

Updates F1 DSCB for PASSWORD data set when records have been added to the end of the data set. Updates F1 DSCB of protected data set whenever protection status of a control password is modified. Builds volume list for the data set using a LOCATE macro instruction; processes each entry, Releases work area and returns control.

Returns to caller

Input: R1 → protect parameter list (see *OS/VS 2 System Programming Library: Data Management* for contents of parameter list), R3 → CVT, R4 → current TCB.

Output: R6→O/C/EOV work area (DEB/DCB, 10B, ECB, channel program, F1 DSCB for the PASSWORD data set). R4→Protect parameter list, R10 = entry code for IGC0109H, R11 = completion code.

Input: R4 → protect parameter list, R10 = code to indicate which routine is to be involved, add, replace, delete, or list. R11 = completion code. R5 Protect work area (DEB/DCB, 10B, ECB, channel program, F1 DSCB for PASSWORD data set).

Output: When add, delete, or replace routines are executed, the PASSWORD data set will be updated. When list routine is executed, the caller's 80 byte buffer will contain the record he requested. R5→work area; R4→protect parameter list, R11 = completion code for IGC0209H.

Input: R5 → O/C/EOV work area (F1 DSCB for password data set, DEB/DCB, IOB, ECB, channel program). R4 → protect parameter list.

Output: F1DSCB for PASSWORD data set may be modified; modified F1 DSCBs for the protected data set will be written to the VTOC of each volume on which the data set resides, if the protection status has been changed.

Chart 10. Protect

MODULE DIRECTORY

This section contains a table of cross-references to the DADSM object modules with the function of each, the method of operation diagram and the module flowchart in which reference to the object module can be found. This information is followed by a table that tells which DADSM object modules are link-edited together at system generation. Note that the object module names, CSECT names, and the names that appear on microfiche of the DADSM object modules are the same.

Object Module Name	Module Function	Method of Operation Diagram	Module Flowchart
IGC0002G	Obtain	25	8
IGC0002I	Scratch—initialization	18	5 Part 1 of 2
IGC0003B	Allocate—initialization entry point	4	1 Part 1 of 3
IGC00030	Rename—initialization	24	7
IGC0007H	LSPACE—available space totaling, message processing	26	9
IGC0009H	Protect—initialization	27	10
IGC0102G	Obtain—VSAM data set	25	8
IGC0107H	LSPACE—initialization and input validation	26	9
IGC0109H	Protect—request processing	27	10
IGC0209H	Protect—DSCB updating	27	10
IGG0CLF2	Non-ISAM allocate—BPAM Directory Formatting	6	1 Part 3 of 3
IGG019EK	RPS channel program appendage module	4,6,15,18, 20,22,24, and 25	1, Part 1 of 4, and 12
IGG020D0	Partial release—write back F1 DSCB	22	6
IGG020D1	Partial release—F5 DSCB (free space) updating	23	6
IGG020P1	Partial release—initialization and F1 DSCB updating	22	6
IGG020P2	Partial release—F3 DSCB updating	22	6
IGG020P3	Partial release—F4 (VTOC) DSCB updating and termination	23	6
IGG029DM	Scratch—Exit point dummy module	20	5 Part 2 of 2
IGG029DU	Initial load of scratch. Dummy module—passes control to IGC0002I.		
IGG0290A	Scratch—password protection interface, VTOC search	19	5 Part 1 of 2
IGG0290B	Scratch—F6 DSCB updating, split-cylinder data sets	19	5 Part 2 of 2
IGG0290C	Scratch—F5 DSCB (free space) updating	20	5 Part 2 of 2
IGG0290D	Scratch—F4 (VTOC) DSCB updating	20	5 Part 2 of 2

Object Module Name	Module Function	Method of Operation Diagram	Module Flowchart
IGG0290E	Scratch—UCB Selection	18	5 Part 1 of 2
IGG0290F (alias= IGG0300F)	Scratch and Rename—volume demounting, mounting, and verification	18	5 Part 1 of 2
IGG0299A	Scratch—DSCB removal	19	11 Part 2 of 2
IGG030DU	Initial load of rename dummy module. Passes control to IGC00030.		
IGG03001	Rename—F1 DSCB updating	24	7
IGG03002	Rename—UCB Selection	24	7
IGG032I1	ISAM allocate—verify checking	8	2 Part 1 of 2
IGG032I2	ISAM allocate—free space (F5 DSCB) search	8	2 Part 1 of 2
IGG032I3	ISAM allocate—building F1, F2, and F3 DSCBs	9	2 Part 1 of 2
IGG032I4	ISAM allocate—valid duplicate name F1 DSCB to be modified	9	2 Part 1 of 2
IGG032I5	ISAM allocate—embedded index processing	9	2 Part 2 of 2
IGG032I6	ISAM allocate—update F5 (free space) DSCBs	10	2 Part 2 of 2
IGG032I7	ISAM allocate—F4 DSCB updating and error handling	10	2 Part 2 of 2
IGG032I8	ISAM allocate—multivolume data set initialization	10	2 Part 2 of 2
IGG0325A	Allocate—duplicate F1 DSCB search	4	1 Part 1 of 3
IGG0325B	Non-ISAM allocate—request conversion and type determination	4	1 Part 1 of 3
IGG0325C	Non-ISAM allocate—absolute track processing	5	1 Part 2 of 3
IGG0325D	Non-ISAM allocate-search free space (normal cylinder and track requests)	5	1 Part 2 of 3
IGG0325E	Non-ISAM allocate-building F1 and F3 DSCBs	5	1 Part 2 of 3
IGG0325G	Non-ISAM allocate—F5 DSCB updating	6	1 Part 3 of 3
IGG0325H	Non-ISAM allocate—F4 DSCB updating and error handling	6	1 Part 3 of 3
IGG0325I	VIO Data Set Allocation	6	7 Part 3 of 3
IGG0325K	Non-ISAM allocate—user-label extent allocaiton	5	1 Part 2 of 3
IGG0325P	VTOC conversion—non-split-cylinder data set processing	12	3 Part 1 of 2
IGG0325Q	VTOC conversion—find split-cylinder data set extents	13	3 Part 2 of 2

Object Module Name	Module Function	Method of Operation Diagram	Module Flowchart
IGG0325R	VTOC conversion—F5 DSCB processing	12	3 Part 1 of 2
IGG0325T	VTOC conversion—F5 conversion, F4 DSCB updating	12	3 Part 1 of 2
IGG0325U	VTOC conversion—build F6 DSCBs	13	3 Part 2 of 2
IGG0325V	VTOC conversion—check for F6 extent overlap	13	3 Part 2 of 2
IGG0325W	VTOC conversion—set up to subtract F6 extents from F5; convert F6 DSCB extents	13	3 Part 2 of 2
IGG0325Z	VTOC Conversion—initialization	12	9 Part 1 of 2
IGG0553A	Extend—initialization	15	4 Part 1 of 2
IGG0553B	Extend—free space (F5 DSCB) search	15	4 Part 1 of 2
IGG0553C	Extend—F1/F3 build/update	16	4 Part 2 of 2
IGG0553D	Extend—Free space (F5 DSCB) updating	16	4 Part 2 of 2
IGG0553E	Extend—VTOC (F4) DSCB updating	16	4 Part 2 of 2
IGG0553F	Extend—VTOC conversion interface	15	4 Part 1 of 2
IGG0553G	Extend—request conversion	15	4 Part 1 of 2

Load Module-to-Object Module Cross-Reference

The following table indicates which object modules are included in each of the DADSM load modules.

Load Module	Object Modules Included in This Load Module
IGC0003B	IGC0003B, IGG0325A, IGG0325B, IGG0325C, IGG0325D, IGG0325E, IGG0325G, IGG0325H, IGG0325I, IGG0325K, IGG032I1, IGG032I2, IGG032I3, IGG032I4, IGG032I5, IGG032I6, IGG032I7, IGG032I8, IGG0CLF2
IGG0325Z	IGG0325P, IGG0325Q, IGG0325R, IGG0325T, IGG0325U, IGG0325V, IGG0325W, IGG0325Z
IGG0553A	IGG0553A, IGG0553B, IGG0553C, IGG0553D, IGG0553E, IGG0553F IGG0553G
IGC0002I	IGC0002I, IGG0290A, IGG0299A, IGG0290B, IGG0290C, IGG0290D, IGG0290E, IGG0290F, IGG029DU, IGG029DM
IGG020P1	IGG020P1, IGG020P2, IGG020P3, IGG020D0, IGG020D1
IGC00030	IGC00030, IGG03001, IGG03002, IGG030DU
IGC0002G	IGC0002G, IGC0102G
IGC0007H	IGC0007H, IGC0107H
IGG019EK	IGG019EK
IGC0009H	IGC0009H
IGC0109H	IGC0109H
IGC0209H	IGC0209H

DATA AREAS

This section lists the control blocks and work areas used extensively by the DADSM routines and tells where the contents of these data areas can be found. This section also (1) presents and describes control blocks and work areas used only by the DADSM routines and (2) shows how related work areas are chained.

System Data Areas Used by the DADSM Routines

The following data areas and control blocks used by the DADSM routines can be found in *OS/VS2 Data Areas*, available only in microfiche:

Allocate work area (IECALLWA mapping macro)

Extend work area (IECEXTWA mapping macro)

Data set control blocks (format-1 through format-6). The DSCBs can also be found in *OS/VS2 System Programming Library: Debugging Handbook.*

O/C/EOV main work area (IECDSECT mapping macro)

Partial Release work area (IECPRLWA mapping macro)

Scratch work area (IECSCRWA mapping macro)

Data Areas Used Only by the DADSM Routines

The following work areas and control blocks are used only by the DADSM routines and are described in this book:

Extended Prefix to DADSM Work Areas

Extent (DADSM) table

LSPACE work area (IECLSPWA mapping macro)

Obtain work area (IECOBTWA mapping macro)

Password data set record

Prefix to DADSM work areas

Protect work area

Recovery routine audit trail

Rename work area (IECRENWA mapping macro)

Transfer control table

VTOC Conversion work area (IECVTCWA mapping macro)

Extended Prefix

The DADSM work areas are obtained using the IECRES GET macro instruction. An extended prefix is obtained only with the initial work area obtained to process a DADSM request. The extended prefix provides a register save area and the addresses of other data areas. The contents of the extended prefix follows:

Offset				
Decimal	**Hex**	**Length**	**Name**	**Description**
0	(0)	0	IECEXTPR	Label to identify the beginning of the extended prefix
0	(0)	72	IECREGSV	18-word register save area
72	(48)	4	IECWTGTB	Pointer to WTG table
76	(4C)	4	IECRRPRM	Pointer to recovery routine parameter list
80	(50)	0	IECDAEND	Label to identify the end of the extended prefix

Extent Table

The extent (or DADSM) table is used to pass descriptions of allocated or released space extents modules of the Allocate, Extend, Scratch, and Partial Release routines. The extent table used in the Allocate and Extend routines is 24 bytes long and can contain as many as 5 extent entries. The table used in the Scratch and Partial Release routines is 68 bytes long to accommodate as many as 16 extent entries.

Flags	Number of entries	Used hole count	RTA1	RTA2	Remaining portion contains four (Allocate and Extend routines) or fifteen (Scratch and Partial Release routines) one-word entries
+0	+1	+2	+4	+6	+8

|←One Extent Entry→|

The information contained in the *flags* field differs for each routine and can be determined by examining the program listing.

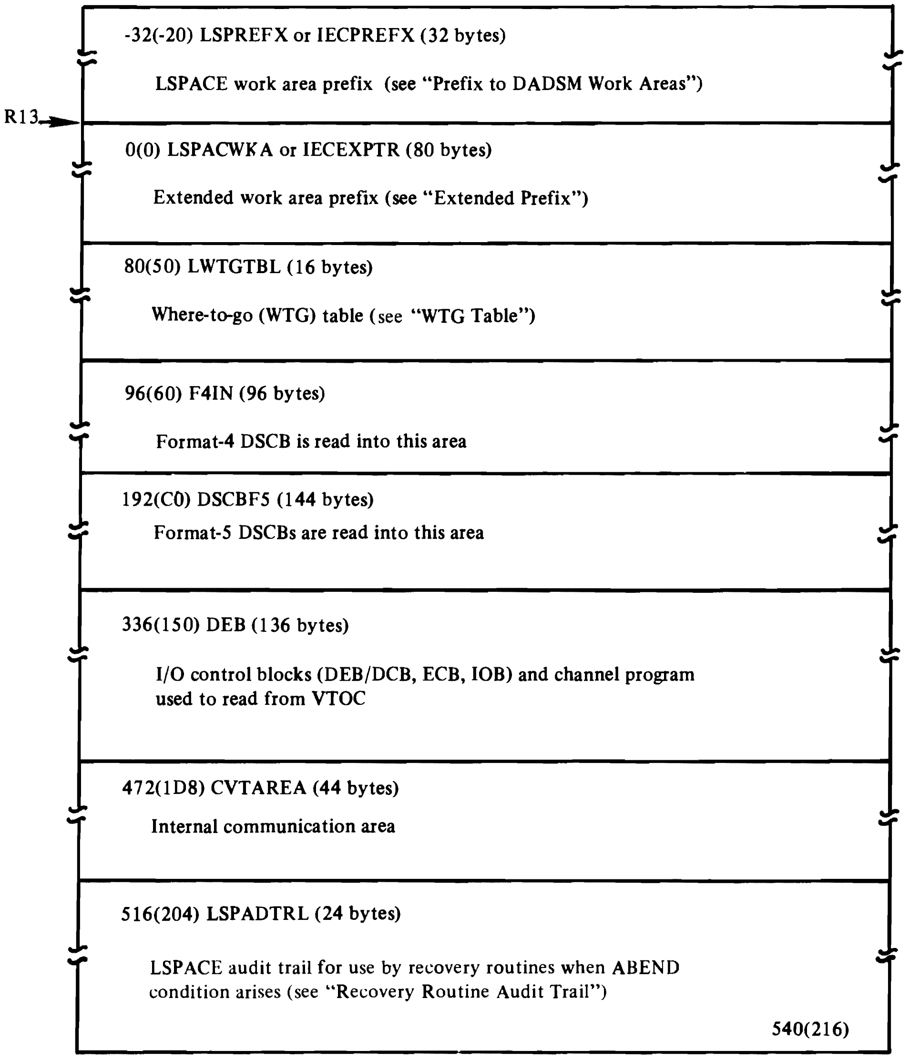
The *number of entries* field contains the count of the number of extent entries in the extent table.

The *used-hole field* contains the count of the number of format-1, format-2, format-3, format-5, and format-6 DSCBs added to and removed from the VTOC during allocation or releasing space.

Each extent entry contains two binary numbers: RTA1 and RTA2. RTA1 is the relative track address of the first track of an extent area; RTA2 is the relative track address of the last track of the area, plus one. A relative track address is a binary number representing the displacement of the track from the first track of the volume. For example, the relative track address of track seven, cylinder zero is seven. The relative track address of track 3 on cylinder 2 is 23, if the volume has ten tracks per cylinder.

LSPACE Work Area

The LSPACE work area, defined by IECLSPWA, is used to read DSCBs from the VTOC of volumes for which available space information is being gathered. Addressability for the work area is established at the beginning of the extended prefix (label LSPACWA). The extended prefix contains a register save area and pointers, and is followed by a WTG table (see "Extended Prefix" and "WTG Table" in this section). The last 24 bytes of the work area is an audit trail used by the DADSM recovery routines to release resources and record diagnostic information when an abnormal termination condition arises during LSPACE processing (see "Recovery Routine Audit Trail" in this section).



-32(-20) LSPREFX or IECPREFX (32 bytes)

LSPACE work area prefix (see "Prefix to DADSM Work Areas")

R13

0(0) LSPACWKA or IECEXPTR (80 bytes)

Extended work area prefix (see "Extended Prefix")

80(50) LWTGTBL (16 bytes)

Where-to-go (WTG) table (see "WTG Table")

96(60) F4IN (96 bytes)

Format-4 DSCB is read into this area

192(C0) DSCBF5 (144 bytes)

Format-5 DSCBs are read into this area

336(150) DEB (136 bytes)

I/O control blocks (DEB/DCB, ECB, IOB) and channel program used to read from VTOC

472(1D8) CVTAREA (44 bytes)

Internal communication area

516(204) LSPADTRL (24 bytes)

LSPACE audit trail for use by recovery routines when ABEND condition arises (see "Recovery Routine Audit Trail")

540(216)

Obtain Work Area

The Obtain work area, defined by the IECOBTWA mapping macro, is used to read DSCBs from the VTOC. Addressability is established at the beginning of the extended prefix, label OBTNWKA. The work area prefix is used to chain this work area to other work areas and storage obtained to process the Obtain request (see "Prefix to DADSM Work Areas"). The extended prefix contains a register save area and pointers and is followed by a where-to-go (WTG) table used to transfer control. (See "Extended Prefix" and "WTG Table" in this section for details). The last 24 bytes of the work area is an audit trail used by the DADSM recovery routines to release resources and record diagnostic information when an abnormal termination condition arises during Obtain processing (see "Recovery Routine Audit Trail" in this section).

-32(-20) OBTPREFX or IECPREFX (32 bytes)

Obtain work area prefix (see "Prefix to DADSM Work Areas")

R1B →

0(0) OBTNWKA or IECEXTPR (80 bytes)

Extended prefix and WTG table (see "Extended Prefix")

80(50) OWTGTBL (16 bytes)

Where-to-go Table (see "WTG Table" for details)

96(60) F4DSCB (96 bytes)

Format-4 DSCB is read into this area

192(60) OBTDSCB (144 bytes)

DSCB is read into this area, then moved to Caller's area

336(150) DEB (200 bytes)

I/O control blocks (DEB/DCB, ECB, IOB) and channel program area

536(218) ERCODE (20 bytes)

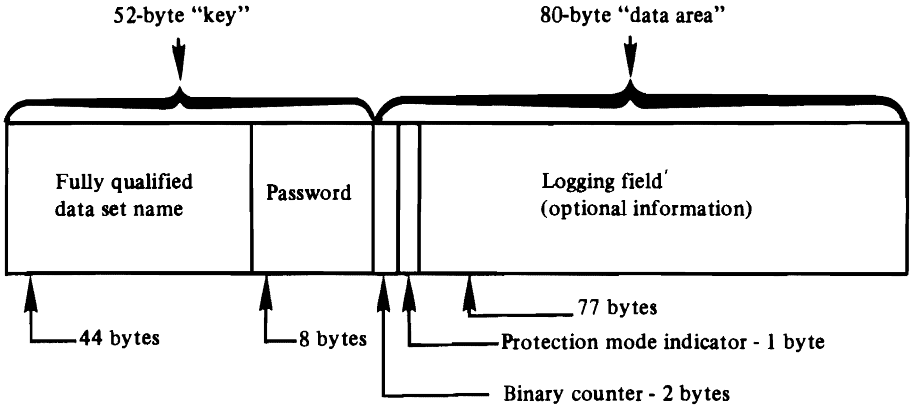
Internal communication area

556 (22C) OBTADTRL (24 bytes)

Obtain audit trail for use by recovery routines when ABEND conditions arise during Obtain processing (see "Recovery Routine Audit Trail")

580(244)

Password Data Set Record Format



When a password-protected data set is opened, the password entered by the operator is matched against the 52-byte "key" area. The data set name and password must be left-justified in their areas and all unused bytes must be blanks (X'40'). The password may be one to eight alphameric characters. When the password record is generated, the counter field is set to zero (X'0000') or any initial value the user desires. Each time the data set is successfully opened (except when the data set is deleted or renamed), the binary counter is incremented by one.

The protection mode indicator field can be set to any of four values:

- X'00' to indicate that the password is a secondary password and the protected data set is to be read only.
- X'80' to indicate that the password is the control password and the protected data set is to be read only.
- X'01' to indicate that the password is a secondary password and the protected data set is to be read and written.
- X'81' to indicate the password is the control password and the protected data set is to be read and written.

Since the DSCB of the protected data set is updated only when the control password is changed, it is possible to request protection attributes for secondary passwords that conflict with the protection attributes of the control password. The 77-byte logging field can contain any type of logging information that the installation chooses (for example, the date on which the counter was reset or the previous password used).

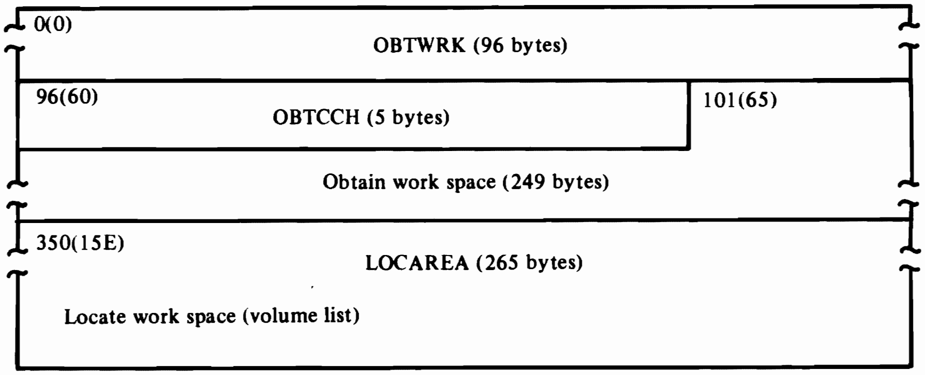
Prefix to DADSM Work Areas

The DADSM work areas are obtained using the IECRES GET macro instruction. A prefix is obtained with each of these work areas. The prefix is used to chain the work areas together. See "DADSM Work Area Chaining" for an example of the chaining scheme. The contents of the prefix fields follow:

Offset				
Decimal	**Hex**	**Length**	**Name**	**Description**
0	(0)	0	IECPREFX	Beginning label for the prefix to all O/C/EOV and DADSM work areas
0	(0)	4	IECCORID	ID of this area. DADSM IDs: ALWA, Allocate work area; EXWA, Extend work area; LSWA, LSPACE work area; OBWA, Obtain work area, PRWA, Partial Release work area; REWA, Rename work area; SCWA, Scratch work area; RPSA, RPS work area; VCWA, VTOC conversion work area.
4	(4)	4	Unused	
8	(8)	4	IECCORLN	Length of this area
12	(C)	4	IECCORAD	Address of the beginning of the extended prefix for this work area
16	(10)	1	IECCORKY	Storage protection key of this work area
17	(11)	2		
19	(13)	1	IECCORSP	Subpool of this work area
20	(14)	4	IECCORPS	Pointer to initial (base) prefix
24	(18)	4	IECCORP1	Address of prefix of previous work area
28	(1C)	4	IECCORP2	Address of prefix of next work area

Protect Work Area

This work area is used by the protect routine to read from and write to the VTOC, using the OBTAIN macro instruction. It is not chained from or related to the other DADSM work areas.



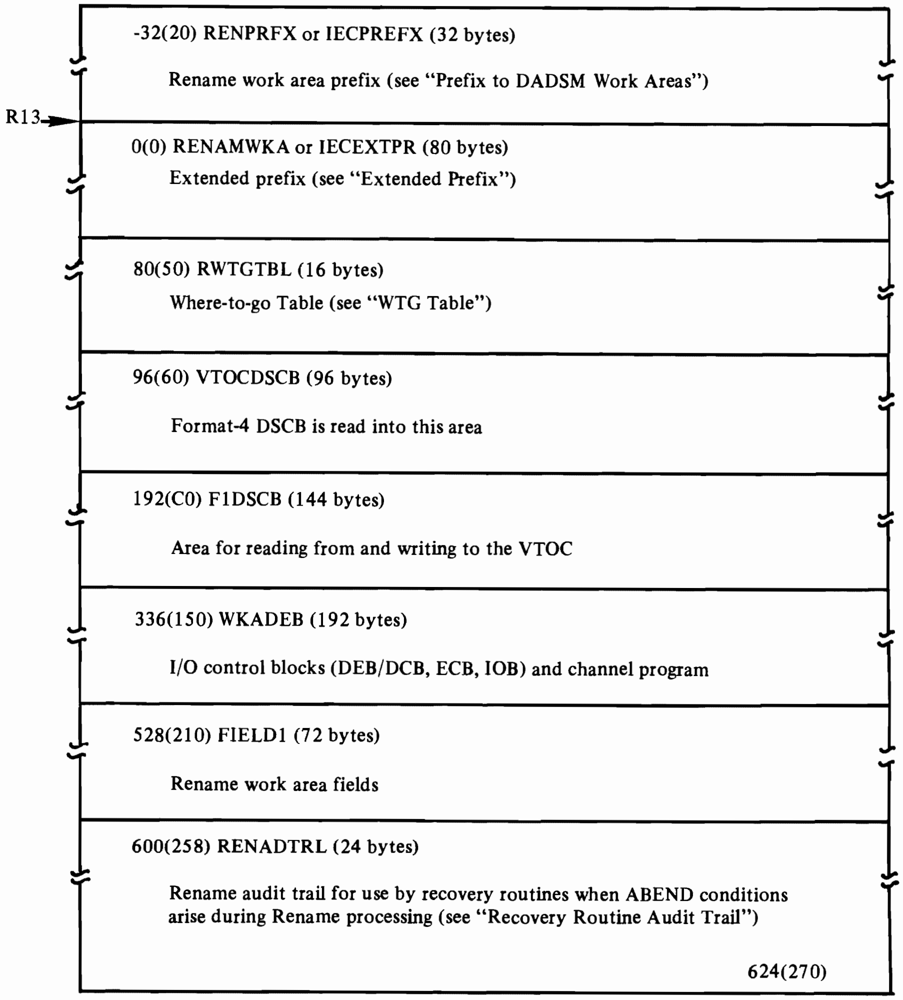
Recovery Routine Audit Trail

This data area is used by the O/C/EOV-DADSM recovery routines (described in *OS/VS2 Open/Close/EOV Logic*) in freeing resources allocated to a DADSM routine that has encountered an abnormal termination condition. This audit trail tells which resources to free. The contents of the DADSM audit trail follow:

Offset				
Decimal	**Hex**	**Length**	**Name**	**Description**
0	(0)	24	ALLADTRL	DADSM Audit Trail Words
0	(0)	4	DSMADTW1	DADSM switches
0	(0)	1	DSMADTB1	General status switches
	1...		DSMISAM	ISAM Allocate
	.1..		DSMVIO	VIO Allocate
	.... .1..		DSMRPSAP	RPS appendage loaded
	.... ..1.		DSMUCBAL	UCB dynamically allocated
	1		DSMASWAP	SYSEVENT swap indicator
1	(1)	1	DSMADTB2	ENQ status switches
	1...		DSMVTOCR	Enqueued on SYSVTOC
	.1..		DSMSMCE	Enqueued SMC on SYSVTOC
	..1.		DSMTIOTE	Enqueued on SYSZTIOT
	...1		DSMDSNE	Enqueued on SYSDSN
2	(2)	1	DSMADTB3	Reserved
3	(3)	1	DSMADTB4	Reserved
4	(4)	4	DSMADTW2	Pointer to SYSZTIOT, the ENQ minor name for Obtain, Rename, and Scratch
8	(8)	4	DSMADTW3	Pointer DSNAME for Obtain (search option), Rename, Partial Release, and Scratch. Pointer to the VDSCB for VIO allocate
12	(C)	4	DSMADTW4	DSAB address for Rename and Scratch; DSAB list address for ISAM Allocate
16	(10)	4	DSMADTW5	Reserved
20	(14)	4	DSMADTW6	Reserved

Rename Work Area

The Rename work area, defined by the IECRENWA mapping macro, is used to read and write DSCBs. Addressability is established at the beginning of the extended prefix at label RENAMWKA. The work area prefix is used to chain this work area to other work areas and storage obtained to process the rename request (see "Prefix to DADSM Work Areas"). The extended prefix contains a register save area and pointers and is followed by a WTG table (see "Extended Prefix" and "WTG Table" for contents of these areas). The last 24 bytes of the work area is an audit trail used by the DADSM recovery routines to release resources and record diagnostic information when an abnormal termination condition arises during Rename processing (see "Recovery Routine Audit Trail" for details).



-32(20) RENPRFX or IECPREFX (32 bytes)

Rename work area prefix (see "Prefix to DADSM Work Areas")

R13

0(0) RENAMWKA or IECEXTPR (80 bytes)

Extended prefix (see "Extended Prefix")

80(50) RWTGTBL (16 bytes)

Where-to-go Table (see "WTG Table")

96(60) VTOCDSCB (96 bytes)

Format-4 DSCB is read into this area

192(C0) F1DSCB (144 bytes)

Area for reading from and writing to the VTOC

336(150) WKADEB (192 bytes)

I/O control blocks (DEB/DCB, ECB, IOB) and channel program

528(210) FIELD1 (72 bytes)

Rename work area fields

600(258) RENADTRL (24 bytes)

Rename audit trail for use by recovery routines when ABEND conditions arise during Rename processing (see "Recovery Routine Audit Trail")

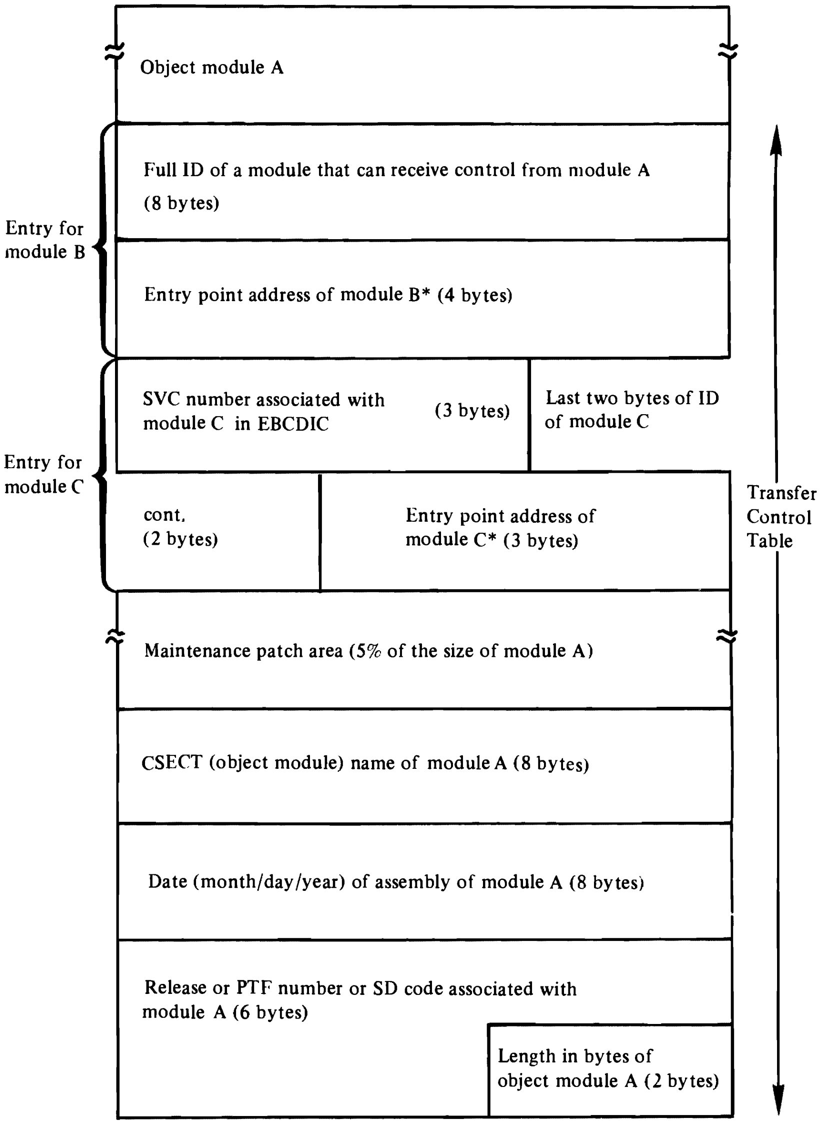
624(270)

Transfer Control Table

The Transfer Control Table (a list of module names and their entry point addresses) is used to transfer control among the DADSM modules and the Close and EOV modules that give control to and receive control from the DADSM modules. The XCTL table appears at the end of each DADSM module as a result of the expansion of the XCTLTABL macro instruction. There are two options that can be used in coding the XCTLTABL macro, depending on how the module ID is coded in the `ID=` operand of the macro. If the full, 8-character module ID is coded, the result is as shown for "Entry for module B" in the illustration below. If only the last two characters of the ID are coded in the `ID=` operand of the macro, the result is as shown in "Entry for module C" below. The XCTL table in a given module will include an entry for every other module that can receive control from the given module. In the illustration, module A's XCTL table includes two entries, one for module B and one for module C. These entries are followed by the maintenance patch area which is 5% of the size of the module it is associated with. The patch area is followed by the 8-byte module name, then the date the source module was assembled. Next appears the 6-byte release or PTF number, and the length in bytes of the load module.

The XCTLTABL macro generates V-type address constants (VCONs). These VCONs contain the entry point address of the object module with which the VCON is associated, if the object modules are in the same load module. If the object modules are in different load modules, the VCON will not be resolved and will contain zeros. When the IECRES LOAD macro is issued to transfer control, the service routine, IFG019RA, checks to see if the VCON contains an address; if so, control is passed to the address in the VCON. If the VCON is unresolved, the service routine initiates a search of the link pack area for the entry point address of the object module to receive control.

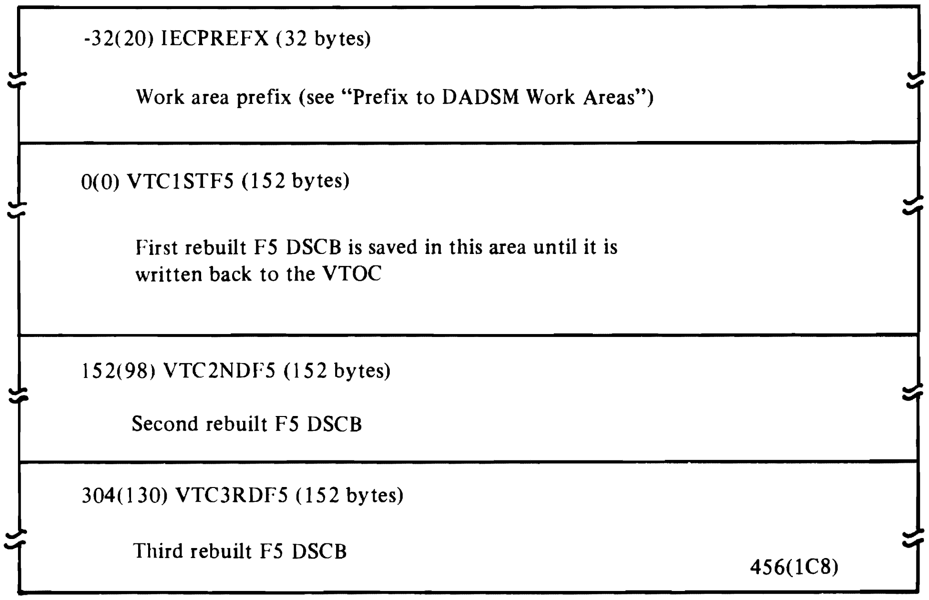
If you are modifying the DADSM routines, you should be aware that every unresolved VCON must have a defined entry point name or alias in the link pack area.


Object module A
Entry for module B
Full ID of a module that can receive control from module A (8 bytes)
Entry point address of module B* (4 bytes)
Entry for module C
SVC number associated with module C in EBCDIC
(3 bytes)
Last two bytes of ID of module C
cont. (2 bytes)
Entry point address of module C* (3 bytes)
Transfer Control Table
Maintenance patch area (5% of the size of module A)
CSECT (object module) name of module A (8 bytes)
Date (month/day/year) of assembly of module A (8 bytes)
Release or PTF number or SD code associated with module A (6 bytes)
Length in bytes of object module A (2 bytes)

* If the object module is not in the same load module as module A, this field will contain zeros an unresolved VCON

VTOC Conversion Work Area

This work area, defined by the IECVTCWA mapping macro, is used during the conversion or reclamation of a VTOC to save the rebuilt format-5 DSCBs until they can be rewritten to the converted VTOC. Each work area can hold as many as three format-5 DSCBs. Each DSCB is chained to the previous format-5 DSCB using the VTCBPTR1, VTCBPTR2, or VTCBPTR3 field. This work area and all other related work areas are chained to the Allocate work area from the work area prefix (see "Prefix to DADSM Work Areas"). The extended prefix contains a register save area and pointers (see "Extended Prefix" for details).



-32(20) IECPREFX (32 bytes)

Work area prefix (see "Prefix to DADSM Work Areas")

0(0) VTC1STF5 (152 bytes)

First rebuilt F5 DSCB is saved in this area until it is written back to the VTOC

152(98) VTC2NDF5 (152 bytes)

Second rebuilt F5 DSCB

304(130) VTC3RDF5 (152 bytes)

Third rebuilt F5 DSCB

456(1C8)

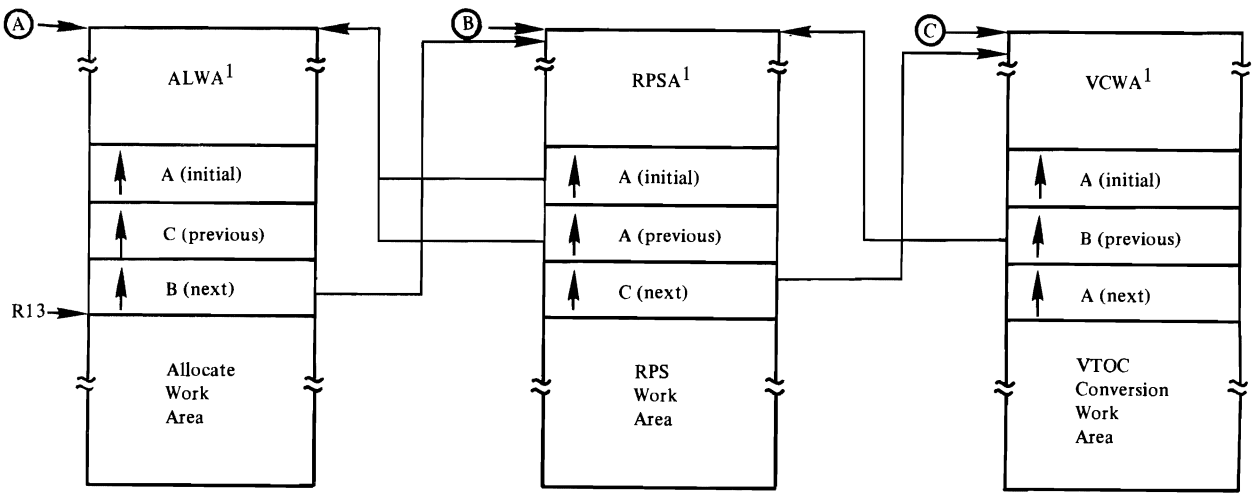
WTG Table

The WTG table is used in conjunction with the IECRES LOAD macro instruction to transfer control. The IECRES macro is described in *OS/VS2 Open/Close/EOV Logic*.

Offset				
Decimal	**Hex**	**Length**	**Name**	**Description**
0	(0)	4	WTGWTOPR	Prefix for WTO
4	(4)	8	WTGMODNM	Name of object module that is to receive control.
12	(C)	4	WTGMODEP	Entry point address of object module that is to receive control.

DADSM Data Area Chaining

To facilitate getting and releasing work areas, the DADSM work areas are chained together. To locate the work areas, it is therefore only necessary in most cases to know the contents of register 13 at the time a DADSM routine is executing. Register 13 contains the address of the extended prefix of a DADSM work area during most DADSM processing. This address is preceded by the prefix which contains the address of subsequently obtained work areas associated with the DADSM request. For example, Figure 6 shows the work area chaining that results from the following conditions: (1) an allocation request (SVC 32) is issued (2) for a device that has the RPS feature and (3) the previous allocation on the volume was made under DOS, which necessitates VTOC conversion.



[1] Four-character identifiers that appear in the first four bytes of each prefix. The other DADSM work area identifiers are: EXWA, Extend work area; LSWA, LSPACE work area; OBWA, Obtain work area; PRWA, Partial Release work area; REWA, Rename work area; and SCWA, Scratch work area.

Figure 6. Example of chaining of DADSM work area—allocation request on a device with the RPS feature, previous allocation mode under DOS

DIAGNOSTIC AIDS

Reclamation of VTOCs by the VTOC Conversion Routine

If a system fails during a function in which the VTOC is being updated (Allocate, Extend, Scratch, or Partial Release processing) or if a permanent I/O error occurs when a DSCB is being read from or written to the VTOC, the VTOC may be in error. To ensure that the error caused by the interruption of DADSM processing is detected and corrected, an indicator called the *DIRF bit* (bit 5 of the DS4VTOCI field of the format-4 DSCB) is set when entering the routines that update the VTOC. The indicator is reset at exit, if no permanent I/O error has been encountered. Scratch and Partial Release routines set the DIRF bit when they encounter DSCBs with overlapping extents. The Allocate and Extend routines check the DIRF bit and request that the VTOC be reclaimed and restored by calling the VTOC Conversion routine if they find the indicator on during initialization. The VTOC Conversion routine is able to reclaim space that might have been lost during a system failure or I/O error, but when the same space has been allocated to two data sets, the VTOC Conversion routine cannot resolve the problem; the system programmer must decide how to correct the error.

Deactivating Automatic VTOC Reclamation

There are situations in which it may not be desirable to automatically convert or reclaim a VTOC that has the DIRF bit set in the format-4-DSCB. By applying a superzap or in-storage alteration to IGG0325Z, you can prevent the VTOC from being automatically reclaimed. The job control language and control statement for altering IGG0325Z follow:

```
//          EXEC      PGM=IMASPZAP
//SYSLIB    DD        DSN=SYS1.LPALIB,DISP=OLD
//SYSPRINT  DD        SYSOUT=A
//SYSIN     DD        *
  NAME      IGG0325Z  IGG0325Z
  VERIFY    0000      0700
  REP       0000      0701
/*
```

DOS VTOC Conversion

The same routines that reclaim an invalid VTOC convert a VTOC on which an allocation has been made by DOS. This conversion is requested during initialization for Allocate and Extend processing when the *DOS bit* is on. The DOS bit is bit 0 of the DS4VTOCI field of the format-4 DSCB.

Listing a VTOC Using IEHLIST

When VTOC problems are encountered, the VTOC of the volume should be printed out and examined. The IEHLIST utility (see "Listing a Volume Table of Contents" in *OS/VS Utilities*) can be used to print the VTOC; an example of the job control language and control statement follows:

```
//            EXEC    PGM=IEHLIST
//DD1         DD      UNIT=3330,DISP=OLD,                          X
//                    VOL=(PRIMATE,RETAIN,SER=111111)
//SYSPRINT DD         SYSOUT=A
//SYSIN       DD      *
  LISTVTOC          DUMP,VOL=3330,111111
/*
```

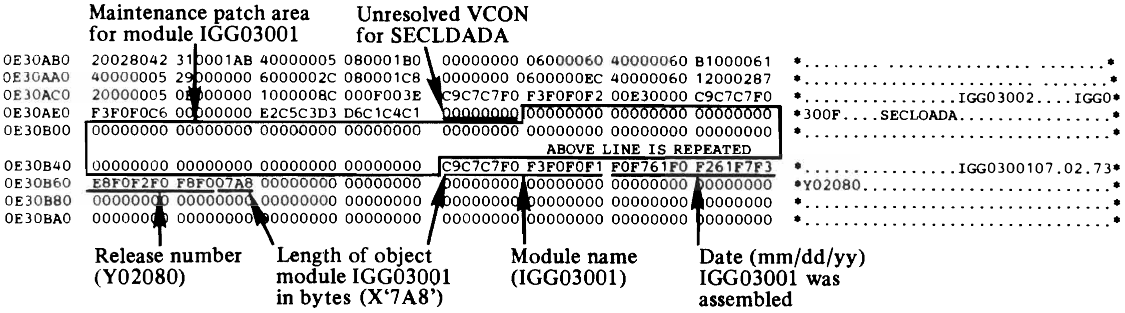
Note: The PRIVATE subparameter is need on the DD1 statement to ensure that the VTOC being processed is not updated by another task during the listing process.

Using the Transfer Control Table

The transfer control table can be used to add a temporary change to a DADSM module and to locate a given object module in a storage dump. One example of a transfer control table appears in the "Data Areas" section (see "Transfer Control Table").

Locating an Object Module in a Storage Dump

Unlike other system control program components, the module name of DADSM modules appears at the **end** of the module in the storage dump. The 8-byte module name follows the maintenance patch area and precedes the date of assembly, the release or PTF number, and length of the moulde. Figure 7 shows the module name for module IFG03001. The object code **precedes** this module name.



```
0E30AB0  20028042 310001AB 40000005 080001B0  00000000 06000060 40000060 B1000061  *........ ............... .......*
0E30AA0  40000005 29000000 6000002C 080001C8  00000000 0600000EC 40000060 12000287  *................................*
0E30AC0  20000005 0E000000 1000008C 000F003E  C9C7C7F0 F3F0F0F2 00E30000 C9C7C7F0  *................IGG03002....IGG0*
0E30AE0  F3F0F0C6 00000000 E2C5C3D3 D6C1C4C1  00000000 00000000 00000000 00000000  *300F....SECLOADA................*
0E30B00  00000000 00000000 00000000 00000000  00000000 00000000 00000000 00000000  *................................*
                                                       ABOVE LINE IS REPEATED
0E30B40  00000000 00000000 00000000 00000000  C9C7C7F0 F3F0F0F1 F0F761F0 F261F7F3  *................IGG0300107.02.73*
0E30B60  E8F0F2F0 F8F007A8 00000000 00000000  00000000 00000000 00000000 00000000  *Y02080..........................*
0E30B80  00000000 00000000 00000000 00000000  00000000 00000000 00000000 00000000  *................................*
0E30BA0  00000000 00000000 00000000 00000000  00000000 00000000 00000000 00000000  *................................*
```

Maintenance patch area for module IGG03001
Unresolved VCON for SECLDADA
Release number (Y02080)
Length of object module IGG03001 in bytes (X'7A8')
Module name (IGG03001)
Date (mm/dd/yy) IGG03001 was assembled

Figure 7. Locating the transfer control table in a storage dump

Adding a Temporary Change in the Maintenance Patch Area of a DADSM Object Module

A maintenance patch area is included in each DADSM module as the result of the expansion of the XCTLTABL macro. The maintenance patch area is 5% of the size of the object module and is located at the end of each module, immediately preceding the 8-byte module name. You can use the IMASPZAP utility to add a modification to an object module:

1. Locate the Transfer Control Table in the source code.
2. Locate the maintenance patch area in the Transfer Control Table. The beginning of the patch area is at the third label from the end of the transfer control table (and the end of the module).
3. Using the displacement (location-counter value) of the maintenance patch area, verify that the patch area is all zeros, and apply the alteration to the object module.

Note:

- Remember to perform a cold start of the system after the alteration. A warm start will not reformat the PLPA to include your modification.
- Be careful not to zap the unresolved V-type address constants in the transfer control table, which are zeros and immediately precede the maintenance patch area.

DADSM Allocate Return Codes

The table that follows identifies each of the abnormal termination messages initiated by the DADSM Allocate modules and identifies the object module (CSECT) that detects the condition that results in the message.

Return Code	Message Identification	Text	Detected by
X'04'	IEF253I	Direct Access—Duplicate Name on Volume	IGG0325A, IGG032I1, IGG032I8
X'08'	IEF254I	Direct Access—No Space in VTOC	IGG0325B, IGG0325C, IGG0325D, IGG032I2
X'0C'	IEF255I	Direct Access—Permanent I/O Error or Invalid F1 DSCB found	IGG0CLF2, IGG0325A, IGG0325C, IGG0325D, IGG0325E, IGG0325G, IGG0325H, IGG0325P, IGG0325Q IGG0325R, IGG0325T IGG0325U, IGG0325V, IGG0325W, IGG0325Z, IGG032I2, IGG032I3, IGG032I4, IGG032I5, IGG032I6, IGG032I7, IGG032I8
X'10'	IEF256I	Direct Access—Absolute Track Not Available	IGG0325C, IGG0325I, IGG032I2
X'14'	IEF257I	Direct Access—Space Requested Not Available	IGG0325C, IGG0325D, IGG0325K, IGG0325I, IGG032I2, IGG032I7
X'18'	IEF258I	Direct Access—Invalid Record Length	IGG0325A
X'1C'	IEF260I	Wrong DSORG or DISP	IGG032I1, IGG032I3
X'20'	IEF261I	No Prime Area Request for ISAM Data Set	IGG032I1
X'24'	IEF262I	Prime Area Must be Requested Before OVFLOW	IGG032I1
X'28'	IEF263I	Space Request Must Begin on Cylinder Boundary	IGG032I2
X'2C'	IEF264I	Duplication of DSNAME Elements Not Allowed—Same Area Requested Twice	IGG032I1
X'34'	IEF266I	Invalid JFCB Pointer	IGG0325A
X'38'	IEF140I	Directory Space Requested is Larger Than Amount Available on This Volume	IGG0325B, IGG0325E
X'40'	IEF273I	Invalid User-Label Request	IGG0325A, IGG0325Z
X'4C'	IEF127I	No Space Parameter Given for New Data Set	IGG0325I, IGG0325C, IGG032I1
X'50'	IEF128I	Invalid Request for ISAM Index	IGG032I1
X'54'	IEF129I	Multivolume Index Not Allowed	IGG032I1
X'58'	IEF130I	DSNAME Element Wrong—Must be PRIME, INDEX, or OVFLOW	IGG032I1
X'5C'	IEF131I	Multivolume OVFLOW Request Not Allowed	IGG032I1
X'60'	IEF132I	SPACE Parameter Wrong—CYL and ABSTR Conflict	IGG032I1
X'64'	IEF133I	SPACE Parameter Wrong—CYL and CONTIG Conflict	IGG032I1
X'68'	IEF134I	Subparameter Wrong in Space Parameter—Must be CYL or ABSTR	IGG032I1
X'6C'	IEF135I	Primary Space Request May Not be Zero	IGG032I2, IGG0325C, IGG0325I
X'70'	IEF136I	Duplication in Allocation—Index Area Requested Twice	IGG032I1
X'80'	IEF267I	Directory Space Request is Larger Than Primary Request	IGG0325B, IGG0325I
X'84'	IEF145I	Space Request Must be ABSTR for DOS Volume	IGG032I1
X'8C'	IEF141I	Index Request Must Precede Prime for ISAM Data Set	IGG032I1
X'90'	IEF143I	Last Concatenated DD Card Unnecessary or Invalid for this Data Set	IGG032I1
X'94'	IEF476I	Overlapping Data Sets in VTOC	IGG0325P, IGG0325R
X'98'	IEF477I	Overlapping DOS Split Cylinder Data Sets in VTOC	IGG0325R
X'9C'	IEF478I	VTOC Error May Exist—Analyze VTOC Listing	IGG0325Z
X'A0'	IEF479I	Possible VTOC error on Second or Later Volume of ISAM Prime Data Set	IGG032I8
X'A4'	IEF703I	New Data Sets Not Allowed on Stacked Pack format DOS Volume	IGG0325A
X'A8'	IEF719I	Duplicate name defined to RACF	IGG032I3, IGG032I5, IGG0325E
X'AC'	IEF720I	Not authorized to define data sets	IGG032I3, IGG032I5, IGG0325E

APPENDIX: EXTENT FORMAT CONVERSION

All allocation and releasing of space are made in a number of tracks; therefore, each of the formats below must be converted to the relative-track-address format used by DADSM, called the RTA1/RTA2 format. RTA1 is the relative track address of the first track of the extent. RTA2 is the relative track address of the last track plus one. The last track address is increased by one during the conversion process to simplify the calculations performed by the DADSM allocation and space releasing routines.

The three extent formats used are:

DSCB Format	Extent Format	Meaning
1, 3, and 4	CCHH/CCHH	The absolute cylinder (CC) and head address (HH) of the first and last track of the data set
5 (free space)	XXYYZ	The first track of the available area (XX), The number of full cylinders (YY) in the available area, and the number of tracks (Z) in addition to the full cylinders
6 (shared cylinders)	XXYYZ	The first track of the available area (XX), and the number of full cylinders (YY) in the available area; the Z is a count of the number of data sets sharing the cylinders

When an allocation or space release has been completed by the DADSM routines, the extents in RTA1/RTA2 format must again be converted to the appropriate form (XXYYZ for format-5 and format-6 DSCBs or CCHH/CCHH for format-1, and format-3 DSCBs). Two resident routines are used to make the conversion, IECPCNVT to convert from CCHH/CCHH to RTA1/RTA2 and IECPRLTV to reverse the conversion. The conversions from XXYYZ to RTA1/RTA2 (and from RTA1/RTA2 to XXYYZ) for both the format-5 and format-6 are done by the DADSM allocate and space releasing routines.

INDEX

For additional information about any subject listed in the index, refer to the publications that are listed under the same subject in *OS/VS2 Master Index of Logic,* GY28-0694.

OS/VS2 DADSM Logic (File No. S370-30) Printed in U.S.A. SY26-3828-2

IBM®

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

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

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

OS/VS2 DADSM Logic

SY26-3828-2

Reader's Comment Form

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

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

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

List TNLs here:

If you have applied any technical newsletters (TNLs) to this book, please list them here:

Last TNL ____________________

Previous TNL ________________

Previous TNL ________________

Fold on two lines, tape, and mail. No postage necessary if mailed in the U.S.A. (Elsewhere, any IBM representative will be happy to forward your comments.) Thank you for your cooperation.

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

SY26-3828-2

Reader's Comment Form

Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

IBM Corporation
P.O. Box 50020
Programming Publishing
San Jose, California 95150

Fold and Tape

IBM®

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

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

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

IBM / Technical Newsletter

This Newsletter No. SN26-0938
Date March 30, 1979

Base Publication No. SY26-3828-2
File No. S370-30

Prerequisite Newsletters None

OS/VS2 MVS DADSM Logic

This technical newsletter, a part of Release 3.8 of OS/VS2, provides replacement pages for the subject publication. These replacement pages remain in effect for any subsequent releases unless specifically altered. Pages to be inserted and/or removed are:

cover, edition notice
93-95
125, 126
131-133

Each technical change is marked by a vertical bar to the left of the change.

Summary of Amendments

Changes included in this newsletter are summarized under "Summary of Amendments" following the list of diagrams.

Note: Please file this cover letter at the back of the publication to provide a record of changes.

IBM Corporation, P.O. Box 50020, Programming Publishing, San Jose, California 95150

Printed in U.S.A.