IBM **Palo Alto Systems Center Bulletin**

R E C O V E R Y / R E S T A R T

IMS/VS VERSION 1.0.1
IMS/VS VERSION 1.1.1
IMS/VS VERSION 1.1.3

Prepared By: Wayne Weikel
Palo Alto - DB/DC Systems Center
Palo Alto, California

G320-6007

CONTENTS

INTRODUCTION

The purpose of this bulletin is to expand upon the documentation currently available in the IMS SRL's. It is intended to be used as a supplement to the IMS manuals and not as a replacement. The manuals should be considered the most current source of information since they are updated by technical news letters.

The material presented in this bulletin is not entirely the original work of the author. It has been gathered from various sources including the IMS 360 Recovery/Restart Bulletin (ZZ20-2801), various World Trade documents, and user experiences.

Program Isolation is an optional feature only in IMS/VS 1.0.1. Since most version 1.0.1 installations use Program Isolation, and because it is a standard feature in all later releases, Program Isolation is an assumed feature of the IMS system through out this document unless specifically stated otherwise.

Those few users who have elected to not use the feature should probably refer to the Recovery/Restart Bulletin ZZ20-2801.

It is extremely important that all users of IMS/VS have well documented procedures relating to the operation of the system. Those installations which enjoy the highest degree of success are those which do have explicit procedures for handling problems arising in the area of Recovery/Restart. IMS/VS provides excellent facilities and utilities to ensure total system integrity but it falls upon the user to determine which of these should be used to accomplish the desired objective. Because of the diversity within IMS systems, no one procedure can be developed for all users. Therefore, each user must establish those procedures which best operate within his environment. Hopefully this bulletin will act as an aid and guide toward such development.

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to any justifications or implementation.

IMS/VS RECOVERY/RESTART PHILOSOPHY

The recovery/restart philosophy is founded upon the concept that when a failure occurs the entire system can be restored to its status at the time of failure and the cause of failure can be alleviated.

This concept covers all the components of the entire system, that is, the Operating System, the IMS/VS Control Region, each dependent region, the data bases, and the Message Queues.

Operating system failures may have to be corrected within that system, however, an IMS/VS restart will ensure system integrity within the IMS/VS system.

Within the IMS/VS system the recovery/restart philosophy is based upon the concept of a synchronization point existing for each component within the system.

The IMS/VS Control Region has synchronization points called system checkpoints. These points occur on a regular basis based upon the number of records written to the log tape. This value is specified during System Generation in the CPLOG parameter of the IMSCTF macro. The data base synchronization points occur on a data base record basis when the dependent region which is updating that record reaches a synchronization point. This point may be called a COMMIT point in that if the system fails later this update will <u>not</u> be removed from the data base. This point in time is synonymous with the dependent region synchronization point.

Each dependent region has a synchronization point defined as that point in time when:

a) The dependent region terminates normally.

b) The dependent region issues a DL/I checkpoint call.

c) The dependent region asks for the next message and single mode operation was specified for that transaction.

Message queue synchronization points occur on an individual message basis. Input message synchronization points occur when the entire message is received and a type '35' log record is written. Output message synchronization points occur when the entire message has been transmitted and some action at the receiving end indicates the message has been received. This action could be an operator asking for the next message, or the receipt of an input message, or some hardware notification. A type '36' log record is written at this time.

USE OF SYNCHRONIZATION POINTS.

When a failure occurs, the Recovery/Restart facilities of IMS/VS will restore the system to its condition at the failure point. To accomplish this:

a) The control blocks are restored to the last system checkpoint. The old log tape is read forward from that point and all commands are reprocessed thus establishing the condition of the lines, terminals, etc.

b) All data base changes since the last synchronization point for each dependent region are removed. This establishes the data bases at the last synchronization point of each processing region.

c) All input messages which were on the queue at the last system checkpoint are restored to the queue.

   All input messages received since the last system checkpoint are added to the queue (except non-recoverable types).

   All input messages which were processed by a dependent region which had reached a synchronization point are removed.

   All output messages produced by a dependent region which has not reached a synchronization point are discarded.

   All output messages acknowledged at the receiving device are removed.

The IMS/VS system is now restored to the status which existed at the last synchronization point of each dependent region. All processing in each dependent region between the last synchronization point and the time of failure is re-processed after the region is started.

RECOVERY COMPONENTS.

The above described recovery is made possible through the use of the recovery components of IMS/VS.

a) The Log Tape.

   The log tape contains a record for each change made to the message queues, a record for each data base update, a record for each dependent region start or termination, and records for each system checkpoint.

b) The Dynamic Log data set.

The dynamic log data set contains a copy of the log record for each data base update since the last synchronization point.

c) The Checkpoint ID Table.

The checkpoint ID table contains an entry for each active dependent region and an entry for the latest system checkpoint.

d) Data Base Image Copy.

The data base image copy is a copy of the data sets at a given point in time. It is used when a recovery of a track or a complete data set is required.

e) Change Accumulation Data Set.

The change accumulation data set contains the data base changes from the log tapes. These changes are in the physical sequence of the data sets. The data set is used when data base data set recovery is required.

These components along with the restart facilities and supplied utilities provide the user with the ability to restart the system after a failure with no loss of system integrity.

IMS/VS LOGGING
LOGGING - GENERAL

The very heart of the IMS/VS System Restart facilities and Data Base Recovery capabilities is the system log. The System log data set is created during each execution of the online control region and is optional during the batch. Because of its importance to system integrity, its use should be considered mandatory for all batch executions. The user should never DD DUMMY the log data set. The log is used by IMS to record all essential information required to perform system restart and data base recovery. A by-product of recording this information is the ability to produce statistics information relative to system performance.

The log is written using BSAM variable blocked spanned records. In IMS 1.1.1 and later releases the user may specify VB only in online control regions. Only VBS is allowable in batch regions.

DUAL LOGGING.

Dual logging is a feature of IMS/VS. Since the system log is of primary importance to system integrity, a second or back up copy is a very desireable feature. In the event of reading or writing errors, the second copy of the log may be used. The dual log feature simply involves the writing of two output log tapes. There is no attempt made to synchronize the EOF on each volume, therefore, one log data set might span multiple volumes while the other is contained on one volume. If a write error occurs during creation, the system continues on the alternate data set. If a read error occurs during Emergency Restart, Backout, etc., the second log may be used as input and the function re-executed.

IMS/VS records the volume serial numbers of only the primary log data set in QBLKS. The volume serial numbers of the secondary log data set are not recorded. When system checkpoint messages are issued they contain the volume serial number of the primary log as recorded in QBLKS.

Should the primary log be lost because of I/O errors operation is continued on the secondary log, however, the checkpoint messages contain a blank volume serial number.

Should the system be restarted after the loss of the primary log the operator must specify the volume serial number of the secondary log in the restart command. Failure to do so will cause IMS/VS to request the mounting of the last known primary log volume because when nothing is specified the volume serial number is obtained from QBLKS.

During a WARM or EMERGENCY restart the old log data set is identified and read using the IMSLOGR DD Card. The data set name in this card is "IMSLOG". In order to restart from a secondary

log the data set name must be changed to "IMSLOG2". It is recommended the user establish a "restart from secondary log" procedure in which the IMSLOGR DD data set name is "IMSLOG2".

When the primary log is lost the system should be immediately terminated with a CHECKPOINT FREEZE followed by a WARM START using the special procedure and specifying the secondary log volume serial number. This will cause IMS/VS to resume with both log data sets. If this is not done the operator must carefully record and correlate the checkpoint messages with the secondary log volume serial numbers.

Loss of the secondary log data set leaves the system in the same state as single log operation. Should the primary log then be lost IMS/VS will abend. It is recommended that after loss of the secondary log IMS/VS be terminated with a checkpoint freeze and restarted but at a more convenient point in time.

LOG BLOCKSIZE AND LOGICAL RECORDS.

Via JCL the user may over-ride the generated LRECL and BLKSIZES for the system log. The chart below shows the generated sizes.

IMS VERSION		RECFM	BLKSIZE **	LRECL
2.4.1	BATCH	VBS	900 (Default)	N/A
2.4.1	ONLINE	VBS	Greater of 1036 or Long Msg LRECL + 12	BLKSIZE-4
1.0.1	BATCH	VBS	1920 (Default)	BLKSIZE-4
1.0.1	ONLINE	VBS	Greater of 1042 or Long Msg LRECL + 18	BLKSIZE-4
1.1.1+	BATCH	VBS	1920 (Default)	BLKSIZE-4
1.1.1+	ONLINE	VBS*	Greater of 1048 or Long Msg LRECL + 24	BLKSIZE-4

* The user may specify RECFM=VB.

** If the user specified BLKSIZE is greater than the minimum it will be rounded up to the next doubleword.

LOG TAPE WRITE AHEAD.

The log tape write ahead optional feature of IMS/VS is designed to protect data base integrity in those instances where main memory is lost, or where execution of DFSFLOTO is unsuccessful.

The feature forces all data base updates to be physically written to the log prior to physically writing the data base buffers to

disk. Then should memory be lost or should the log terminator fail, the user may close the log tape and execute Emergency Restart or Batch Backout, and be assured that all data base updates will be backed out to the previous synchronization point.

Without the feature a data base update could occur and the corresponding log record be placed into the log buffer. If the buffer were lost, and Backout or Emergency Restart were attempted, that update would not be backed out. Without Log Tape Write Ahead any loss of the log buffers necessitates a full data base recovery!

There are some performance implications relative to using Log Tape Write Ahead. The performance degradation would be greatest in the batch environment and least in the online environment. As more and more activity is logged, the greater the probability that the log records representing data base changes have been flushed from the log buffers to tape prior to the data base buffers being flushed from the data base buffer pool. Naturally much more logging occurs in online systems therefore the chance of having to wait for a log write before a data base write is considerably less. Weighed against the data base recovery alternative, Log Tape Write Ahead is the suggested mode of operation.

The Log Tape Write Ahead feature is supported for both ISAM/OSAM and VSAM data bases. The feature is activated with the OPTIONS card in the buffer pool initialization data set. The format of the OPTIONS card parameter is:

LTWA=YES | NO

Notice the default is NO. If the installation should establish a standard requiring operation with Log Tape Write Ahead, consideration should be given to changing the default to YES.

Another requirement when using this feature is the use of BISAM (or VSAM Direct mode) to manage the data bases. Since IMS/VS cannot predict when buffers are written, the log tape write ahead option does not apply to data bases managed with QISAM.

The following rules apply tc forcing the use of BISAM.

1. BISAM is always used to manage an ISAM data set that is accessed from the IMS/VS Control region (a message processing environment).

2. For a batch processing environment BISAM is used to manage the ISAM data set of a HISAM data set group when:

   a) a PCB is sensitive to a logical parent that exists in the data set group or:

   b) multiple PCBs are sensitive to segments within the data set group.

The simplest way to ensure the use of BISAM is to code an extra PCB for the data set group. The PCB will never be referenced and will not require application programming changes.

In the above discussion BISAM is equivalent to VSAM Direct mode and QISAM is equivalent to VSAM Skip Sequential mode.

CLOSING THE LOG

IMS_FAILURE.

IMS/VS has implemented the (E)STAE for MVS systems and the STAE TERM=YES for VS 1.4 and later systems to attempt to ensure correct closure of log tapes should IMS/VS fail. These STAE routines flush the data base buffers, flush the log buffers, close the log tape, proceed with cleanup operations, and reissue the abend to pass control to any User STAE. These STAE routines should gain control on all ABENDS.

ERROR_IN_IM/VS_STAE_ROUTINE.

When message DFS616 is issued during abnormal termination, it indicates that the usual service of the STAE exit could not be provided because a problem exists with the log writing module or the log tape itself. The DFS616 message is issued to the operating system console, by a WTOR. This notifies the operator and holds up the IMS/VS termination which allows the operator to obtain a dump of memory contents before they are destroyed. The basic instructions for this case are:

a) Keep the operating system up.

b) Obtain a memory dump using the DUMP command.

c) Reply "TERM" to the DFS616 message.

d) Execute DFSFLOTO.

e) /ERESTART.

The installation may consider modifying the text of the DFS616 message issued by DFSFLOSO so that it instructs the operator to obtain a dump before replying.

OPERATING_SYSTEM_FAILURE.

If the failure occurs within the operating system a storage dump should be taken. Execution of the SYSTEM LOG TERMINATOR (DFSFLOTO) will close the log tape. DFSFLOTO is a utility which operates under Operating System control using either a SYS1.DUMP or the Stand Alone dump data set and closes the log tape by finding the log buffers in the dump data set, flushing the buffers to the log, and closing the data set. The program determines the control information necessary to locate log control blocks and buffers in the memory dump from the hex 42 record which IMS/VS writes to the log data set when the system is started, and at the beginning of each subsequent log volume. Having located the log records that were still in memory at the time of failure, the log data set is read until a block is found which contains a record with a sequence number just one less than the first record still in the memory buffers. This comparison is

done on the log sequence number only and not on date/time or any other field. Then all blocks in the active buffers are written. A hex 06 log record type 8 is written and the log data set is closed. This technique preserves the integrity of the log tape and prepares it for usage by Emergency Restart or Backout.

DFSFLOTO FAILURE.

A log tape which is not terminated by IMS/VS or by the log terminator cannot be safely used for Emergency Restart. The following discussion assumes the presence of the log tape write ahead feature in the system.

The user may utilize the System Log Recovery Utility (DFSULTRO) to close an unterminated log. Using this utility requires both a DUP and REP phase. During the REP phase the user must specify CLOSE TRUNC beginning in colunm 16 of the control card. CONTROL TRUNC indicates the output tape will be closed immediately before this error block; this error block and the last record of the previous block will be deleted if that record is spanned (incomplete).

If he wishes, the user may terminate the log tape by copying the tape to the point of errcr and place a trailer label on the new tape. The copy program should sequence check the log records because log tapes are frequently reused and data blocks could possibly be copied from a previous usage of the tape. Once the last block is located, the copy program must ensure that a partially spanned record does not exist. If it does, that record must be removed pricr to writing the EOF.

Once the tape is terminated, the user has two options to accomplish data base backout and restart of the system.

a) The first is to execute Emergency Restart with the BLDQ parameter from the latest dump queue type checkpoint. Emergency Restart will back out the active program data base changes. Using the BLDQ option will rebuild the message queues to reflect those log records which were on the tape. Input messages which did not get logged will be lost (they were in the lost buffers). and output messages will be duplicated if they were enqueued to final destination and dequeued with only the enqueue having been written to the log tape. If BLDQ is not used the message queues may not be secure.

   The user must then assess the impact of the lost messages and re-input those desired. Where possible, inquiry logic should be designed into every application so the terminal users can determine the effect of the entered transactions.

b) The alternative is to use the Data Base Backout utility to backout all active PSBs and cold start IMS. A cold start means the loss of any messages enqueued at the time of failure. If Emergency Restart fails in the first case a cold start may be required.

LOG CONTROL AND USE

An IMS system with high volume transaction and data base activity or many batch data base update programs will generate a large number of system log tapes in a given day. Strict control procedures should be established for the selection, use, retention, and distribution of all log tapes. Failure to include a volume, or use of the wrong volume, or use in the wrong sequence during data base recovery or system restart will result in the loss of integrity. A log should be maintained keeping track of log tape volume serial numbers by date and time of use. This log should at a minimum contain:

1) Volume serial number, date and time of use,

2) Starting serial number for recovery of each data base data set,

3) Record of change accumulation runs,

4) Backup volumes for change accumulation,

5) DUMPQ checkpoint IDs for ERESTART.

If statistics are required the log may also contain a record of volumes per statistics run to avoid duplicate or missing data.

Some installations erase tapes prior to using them as logs to prevent reusing prior run blocks during log copy after an I/O error.

The OS parameter DISP=MOD is not supported by IMS for system log tapes. The system log terminator will not properly position a MOD tape for termination because multiple '42' records would exist and only the first would be selected to determine the buffer addresses.

The minimum retention cycle for log tapes is successful completion of change accumulation for data base recovery and the most recent tape containing a DUMPQ or PURGE checkpoint for system restart. User requirements may dictate longer retention for backup, statistics, etc. In summary, the integrity and control of the system log tapes is a minimum requirement for the integrity of the IMS system.

In addition to the IEFRDER log data set, IMS/VS maintains a Dynamic Log Data Set. This data set is a sequential, variable blocked, spanned, OSAM data set which is reused in a circular fashion. The data set consists of incore buffers within the control region and the DBLLOG disk data set. The device type, the buffer size, and the number of buffers are all specified at SYSGEN in the IMSCTF macro. The number of buffers can be overridden at execution time in the DYBN parameter of the IMS procedure.

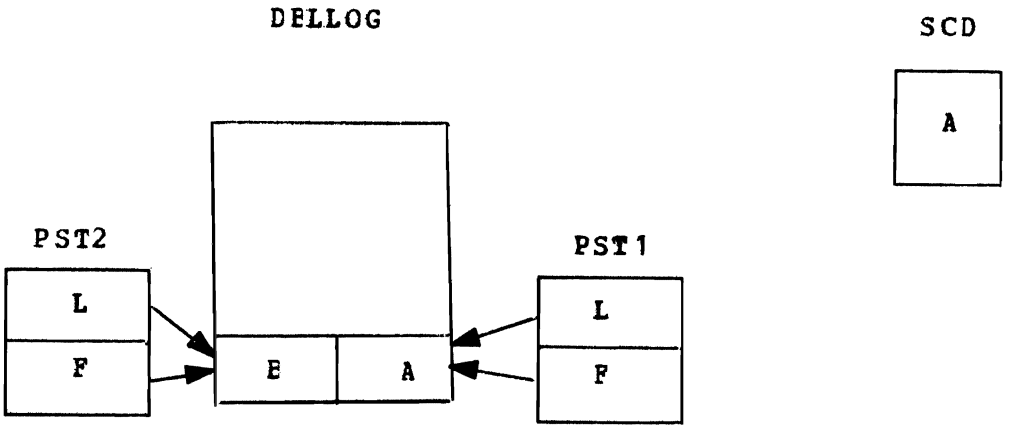
As the dynamic log buffers become full, they are written to the DBLLOG data set. A buffer is written each time there is only one more available buffer remaining. Therefore, to maximize effeciency, the more log buffers specified, the less I/O is required to the DBLLOG data set. If enough buffers were specified, and the programs reached synchronization points often enough, no I/O would ever occur.

The DBLLOG data set allows the logging of all data base changes by each region in a temporary area until a region synchronization is reached. If the region should abend or a ROLL call should be issued, this temporary area will be used to automatically back out all data base changes made by the program. Each active IMS region has an individual log chain built by pointers in the PST. PSTDBLF contains a pointer to the first log record and PSTDBLBK contains a pointer to the last log record. Each log record contains a pointer to the previous log record written by the PST. This chain overlays the date and time fields in the type '51' log records. The SCD contains the RBN of the oldest log record in field SCDDBWRP. The RBN is used to indicate when wrapping occurs. The RBN of the log records is assigned beginning at the highest possible and decreasing to zero. The reuse is from zero to the highest RBN.

In the following examples assume the DBLLOG data set contains only four blocks. Each block will hold only two type '51' records.

EXAMPLE 1.

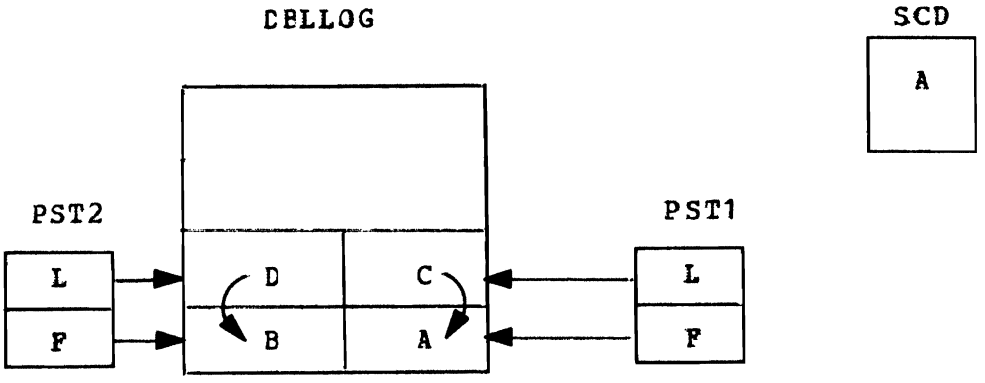
Region 1 changes segment A
Region 2 changes segment B



After logging the changes to segments A and B the first and last PST log chains are established and the SCD contains the RBA of log record A because this is the oldest record in the data set.

EXAMPLE 2.

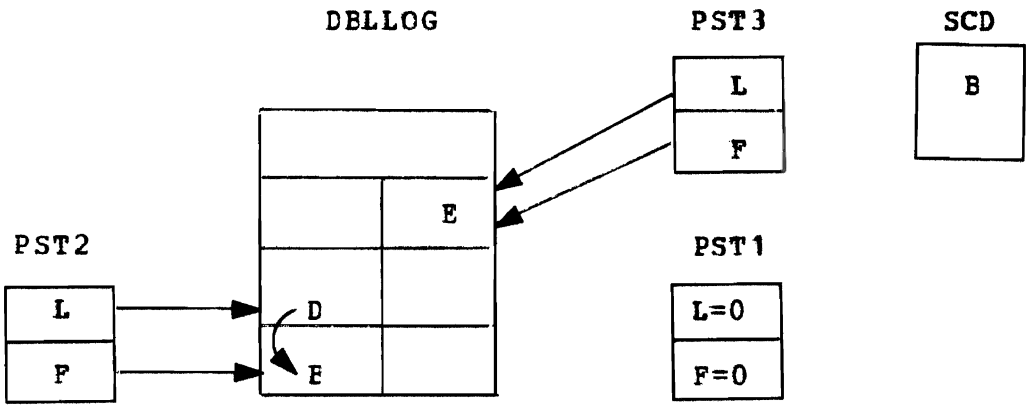
Region 1 changes segment C
Region 2 changes segment D



The changes to segments C and D are logged. The PST last log chains are updated to point to these log records and each log record is chained to the previous per PST. Should backout be necessary at this point for Region 1, the last chain is followed and segment C would be backed out, then Segment A.

EXAMPLE 3.

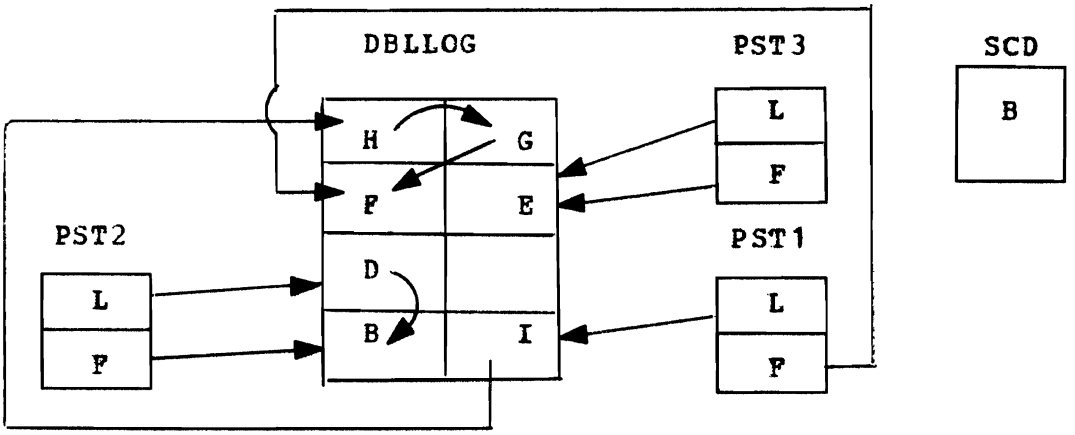
Region 3 changes segment E
Region 1 reaches sync point



When a region reaches a synchronization point the log records are freed, and the PST chains are set tc zero. In this example PST1 is set to zero. An interogation of active PSTs determines the RBN of the next oldest active log record, and the SCD is updated accordingly. In this example the SCD is set to the RBN of log record B.

EXAMPLE 4.

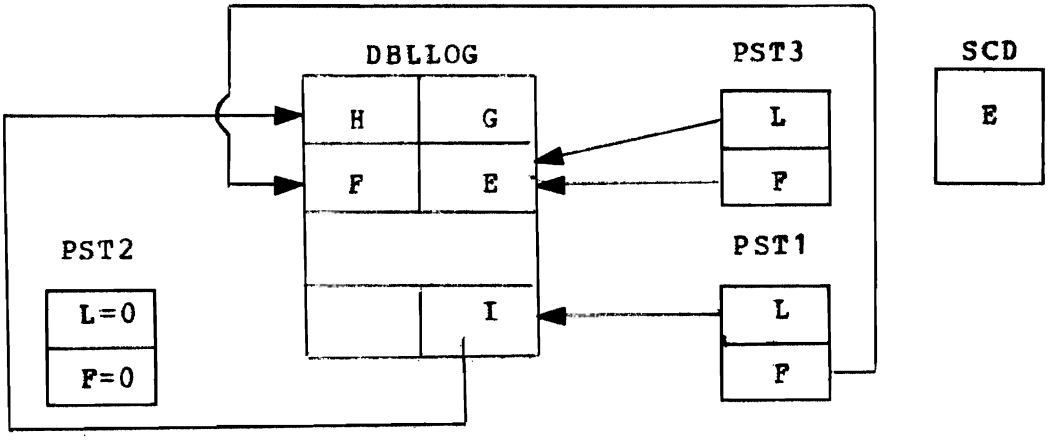
Region 1 changes segments F,G,H, and I.



This example demonstrates the circular reuse of the data set. When the end of the data set was reached the next record was obtained from the highest RBN since that record was available for use.

EXAMPLE 5.

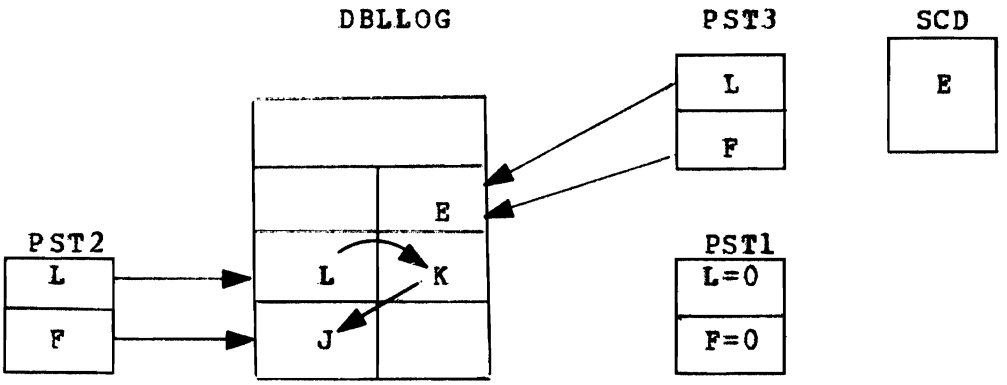
Region 2 reaches sync point.



The log records on the PST2 chain are freed. PST2 pointers are set to zero and the SCD is updated to the RBN of log record E since it now is the oldest in the data set.

EXAMPLE 6.

Region 1 reaches sync point
Region 2 changes segments J, K, and L



All log records for region 1 are freed. Note that the log records for region 2 begin at the old location of segment B since it was the next to assign (desending). Even though the location occupied by the released segment I was available it was not used.

EXAMPLE 7.

Region 1 changes segment M.

At this point the data set is considered full since the next to assign is the RBN occupied by log record E and that RBN is contained in the SCD. Freed space exists both before and after log record E yet the data set is considered full and wrap control is begun. This should explain how a long running job (especially a BMP) can cause the Dynamic Log to fill even though it performs relatively few updates.

WRAP CONTROL.

Once the DBLLOG data set reaches a point when it can no longer assign space sequentially, IMS/VS takes the following action. Message DFS228 "IMSVS.DBLLOG DATASET TOO SMALL" is issued to the Master Terminal Operator and dynamic logging ceases. The system then begins to quiesce the dependent message regions and waits for BMPs to issue a checkpoint call or terminate. Meanwhile data base changes continue to be logged on the system log. If during this time any dependent region abends or issues a ROLL call, the IMS/VS control region will abend with a U823 abend since dynamic backout can not be performed. The user must then perform an Emergency Restart from the last simple checkpoint. If the U823 does not occur before all dependent regions quiesce, IMS resumes normal operation. In almost every case the DBLLOG allocation should be increased. The user could issue a /CHECKPOINT FREEZE, reallocate, and Warm start or reallocate prior to the Emergency Restart.

DYNAMIC BACKOUT.

Dynamic backout is invoked upon any abnormal termination of a MPP or BMP program. Both the transaction code and application program are stopped unless the abend is 777 (deadlock) or 778 (roll). The Master Terminal operator can restart either or both of the stopped resources. If it is desired to allow queueing of the transactions to continue, the transaction must be restarted. Restarting the program prior to determining the exact cause of the abend could well cause the program to again abend.

When the abend occurs message DFS554 is issued to the Master Terminal identifying the application program, the transaction code, the input logical terminal name, the system and user completion codes, and the first 120 bytes of the input transaction in process, if available. The message will no longer be available if the program had issued a GU call to the message queue and received a 'QC' status code prior to abending. If the message was available, message DFS555 is issued to the Master Terminal stating the source LTERM of the transaction and indicating whether the message has been retained or

deleted. With IMS/VS 1.1.1 PTF 3 and later releases the DFS555 is also sent to the inputting logical terminal.

The message will be retained on the queue if the abend occurs before the program issues a GU call to the message queue. It will be assumed to have contributed to the abend if it is received by the program therefore it will be deleted. No other transactions are dequeued even if mode is multiple.

Using the DBLLOG all data base changes are backed out to the program's last synchronization point. If mode equals single the output messages and database updates for the last transaction are backed out. If mode equals multiple all messages and updates are backed out to the last synchronization point or program schedule, whichever is most recent.

I/O_ERROR_ON_DBLLOG.

If a write I/O error occurs on the DBLLOG data set, message DFS226 is issued to the Master Terminal.

If the message is issued during normal operations dynamic logging ceases and an internally generated /CHECKPOINT FREEZE is issued to shut down IMS. If an application program should abend before IMS can be shut down the Control Region will abend with a U818.

If the message is issued during Emergency Restart the Control Region is abended with a U818. (Emergency Restart writes system log records to the DBLLOG in preparation for performing backout).

A read error on DBLLOG during Emergency Restart results in message DFS982 followed by a U982 abend of the Control Region. During normal operation message DFS981 is issued and the data base is stopped.

If the I/O error results in a Control Region abend the DBLLOG data set should be reallocated and Emergency Restart executed with "FORMAT DL" specified.

PROGRAM ISOLATION

Program Isolation is a feature of IMS/VS which allows multiple IMS dependent regions (MPP and BMP) concurrent access to the same data base and segment types. To accomplish this DL/I uses the ENQUEUE/DEQUEUE FACILITY.

ENQUEUE/DEQUEUE

In order to allow multiple users access to the same data base and segment types, some rules must be enforced to assure data base integrity and user protection. It is critically important that one user not delete segments while ancther user is interrogating them, or insert segments thereby destroying anothers position within the record, or update data while the second user is also updating the same data. The technique used to prevent such happenings is the ENQUEUE/DEQUEUE logic. When one user is operating on a database record, that record is unavailable to any other user in the system. The following rules and examples are intended to show how this is accomplished.

ENQUEUE LEVELS

The ENQUEUE/DEQUEUE logic provides for various levels of enqueue and options for each level. IMS/VS does not utilize all the levels and options possible. There is no correlation between the level names and the type of program which issues that particular enqueue. For example, Read Only does <u>not</u> imply usage by inquiry only programs. In fact inquiry only programs will issue Single Update and possibly Read Only enqueues. Whenever an enqueue is issued, a return code is passed to the requestor indicating the status of his request. Any combination of enqueue requests (see levels) which totals 4 or greater causes the return code to be set greater than 0. The requestor may proceed or wait for the desired segment depending upon this return code. The following are the enqueue levels.

LEVEL 1. READ ONLY

A return code of 0 indicates that no one else has requested the segment, or another user has requested it for read only (level 1) or single update (level 2). In any case the requestor may look at the segment.

A return code of 4 indicates another user has, in some fashion, updated this segment and the requestor may not have the segment until the first user releases it.

LEVEL 2. SINGLE UPDATE.

A return code of 0 indicates that no other user has requested the segment, or another user has requested it for read only (level 1).

A return code of 4 indicates another user has requested the segment at a single update level (he intends to update it) or an exclusive level (he has updated it). The requestor may not look at the updated segment until the first user has finished.

LEVEL 3. EXCLUSIVE.

A return code of 0 indicates that no other user has requested the segment.

A return code of 4 indicates that another user has requested the segment at any one of the three levels.

ENQUEUE OPTIONS.

For each level of enqueue, the requestor is allowed one of three options.

FEEDBACK

With this option the enqueue is not actually performed but the appropriate feedback is returned to reflect the status had it been performed. The Return Code is always zero, the feedback word is zero if this user may enqueue, the feedback word is non-zero if the resource is held by another user.

TEST

With this option if the return code is 0 no enqueue is done. If the return code is 4, the segment is enqueued at the requested level and the requestor must wait for a previous enqueue.

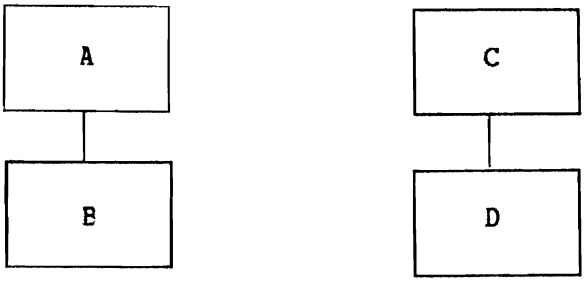
NORMAL.

With this option the segment is enqueued at the requested level. If the return code is 0 the requestor may continue. If the return code is 4 the requestor must wait for a prior request.

DEADLOCK

A return code of 8 from any enqueue request indicates a deadlock situation. A deadlock develops when one requestor is enqueued on a segment and asks for a second. The second segment is enqueued upon by a second user who has asked for the first segment. Obviously, making each wait is not the answer since neither requestor can satisfy his second request, so one requestor is selected to be abended. When this occurs all data base updates performed by the abended user are backed out and the user is then re-scheduled for processing.

An example of a deadlock:



1. User one enqueues on segment A.
2. User two enqueues on segment C.
3. User one enqueues on segment C. (must wait).
4. User two enqueues on segment A.

Since user one is in a wait and won't free segment A, user two can never get segment A.

The following table explains which program is selected for ABEND.

		And if the waiting program that completes the deadlock circuit				
If calling program whose request will cause a deadlock		Is a MSG program with		Is a BMP program with		Is multiple programs
		Single Mode Tran	Mult Mode Tran	Single Mode Tran	Mult Mode or no Tran	
Is BMP program with	Mult Mode or no Tran	DO B	DO B	DO B		
	Sngl Mode	DO B	DO B			
Is MSG program with	Mult Mode Tran	DO B		DO A		
	Sngl Mode Tran					

A = ABEND the calling program
B = ABEND the program with which the calling program would deadlock.

When a deadlock situation develops, one of the programs is Abended and backed out to the previous synchronization point. If the program is an MPP it is then rescheduled at a priority of 15. This is an attempt to continue execution with the least impact upon the end user.

ENQUEUE RULES

1. Only one user may be positioned within a data base record at any one time.

   A. To ensure this, each root is enqueued at the Single Update level (level 2) when retrieved.

   B. If the request is for a concatenated segment, the root of the logically related segment is enqueued at the Single Update level.

2. Any segment which is changed, whether data content or prefix content, is enqueued at the Exclusive level (level 3).

3. All Exclusive enqueues are held until the requestor reaches a synchronization point at which time they are dequeued. A synchronization point is defined as one of the following.

   A. Program termination.

   B. Checkpoint call.

   C. A Get Unique to the message queue if mode = single.

4. In an HD organization, whenever a dependent segment is updated, an enqueue is issued on the root RBN + 1. This is used as a switch to indicate one of the dependents is enqueued at an exclusive level. When another user is accessing dependents, a test of the root RBN + 1 indicates whether or or not it is necessary to test each dependent accessed. The test of the root RBN + 1 is performed by issuing an Exclusive Feedback enqueue. If the return code is 0 no further tests are required. If the return code is 4 each segment must be individually tested. (See examples).

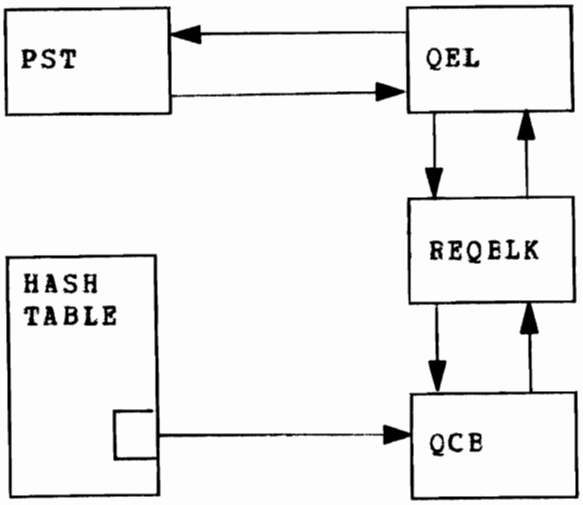
5. In HISAM all enqueues are done using the relative record number of the logical record. The prime logical record is controlled by the Single Update level enqueue on the logical record containing the root. If the dependent exists in an overflow logical record, a Read Only Test enqueue is done on that logical record. (see figure 16.)

ENQUEUE CONTROL BLOCKS

All enqueues are accomplished by building control blocks which identify the user and the data base resource. These control blocks are built in the storage specified by the IMSCTF macro and the CORE= parameter. Each control block is 16 bytes in length. The PST, in field PSTQELA, contains a pointer to the first of these blocks, the QEL. The QEL is assigned once per PST when the first enqueue request is made. Internally there exists a HASH TABLE which is used to provide an anchor point address. Each enqueue request for a RBN or RRN is passed through a hashing technique to provide an anchor point in the table. The anchor point is a four byte pointer to the QCB which is the second of the control blocks. The QCB contains the RBN of the segment, the DMB and the DCB among other things, and identifies the data base resource. The QCB in turn is chained to the REQBLK which is built for each request for the resource.

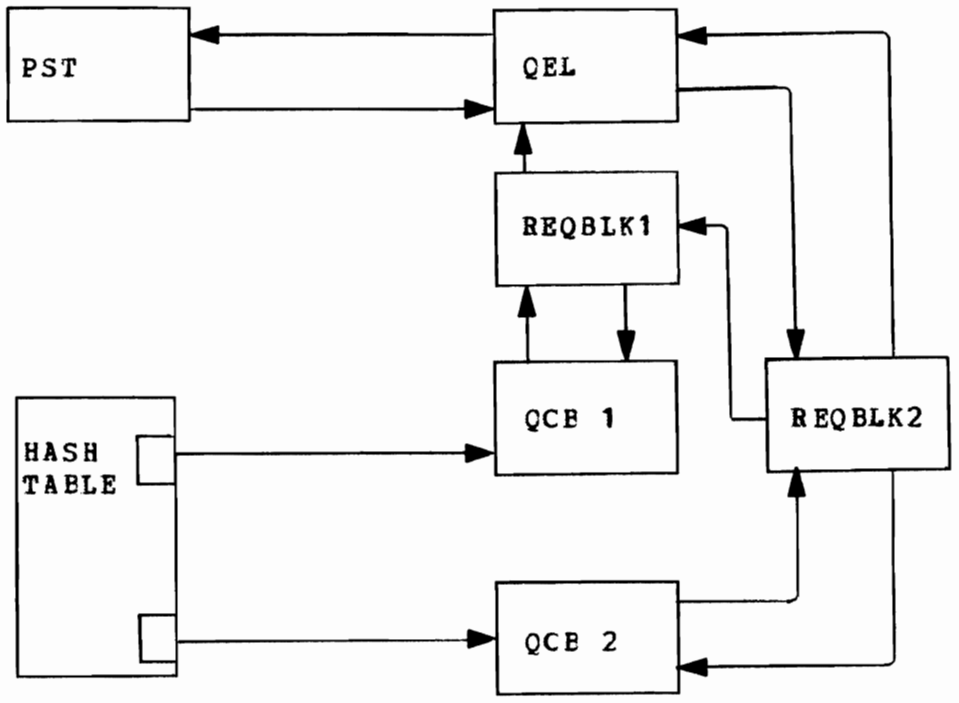
The following examples show how the various blocks are chained together for different situations.

EXAMPLE 1. First request for a data base resource.



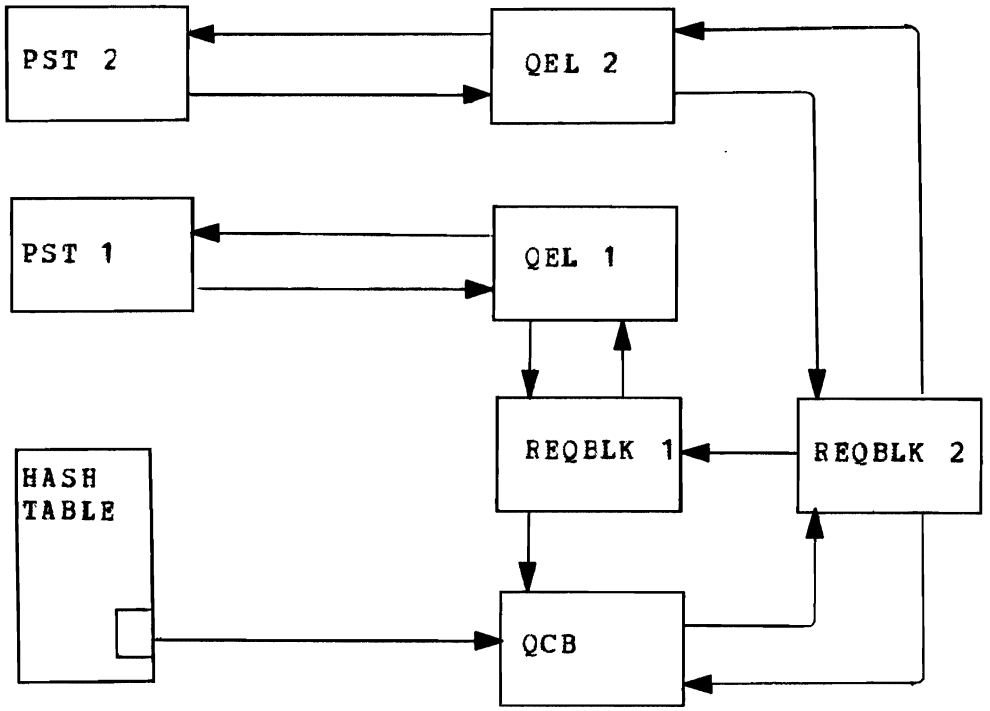
The first enqueue request issued by a program uses 48 bytes of storage.

EXAMPLE 2. A request for another resource made by the same program.



The request for a second resource results in 32 bytes of storage being used.

EXAMPLE 3. Two requests for the same resource made by two programs.



The second request for the same resource from a different PST results in 32 bytes of storage being used.

ENQUEUE EXAMPLES - HD ORGANIZATION

The following examples are intended to explain the logical path through the enqueue code and the level of enqueue performed on each segment.

1. RETRIEVE A ROOT.

DL/I will follow one of two paths when a GU or GN call is made to the root. A GU causes DL/I to access the index segment and follow the pointer to the root. A GN Root qualified on a key value operates just like a GU. An unqualified GN will follow the PTF pointer when PTB has also been specified for the root segment. If PTF only has been specified, DL/I will retrieve the next index segment and follow the pointer to the root. The following shows the logic of retrieval via the two paths.

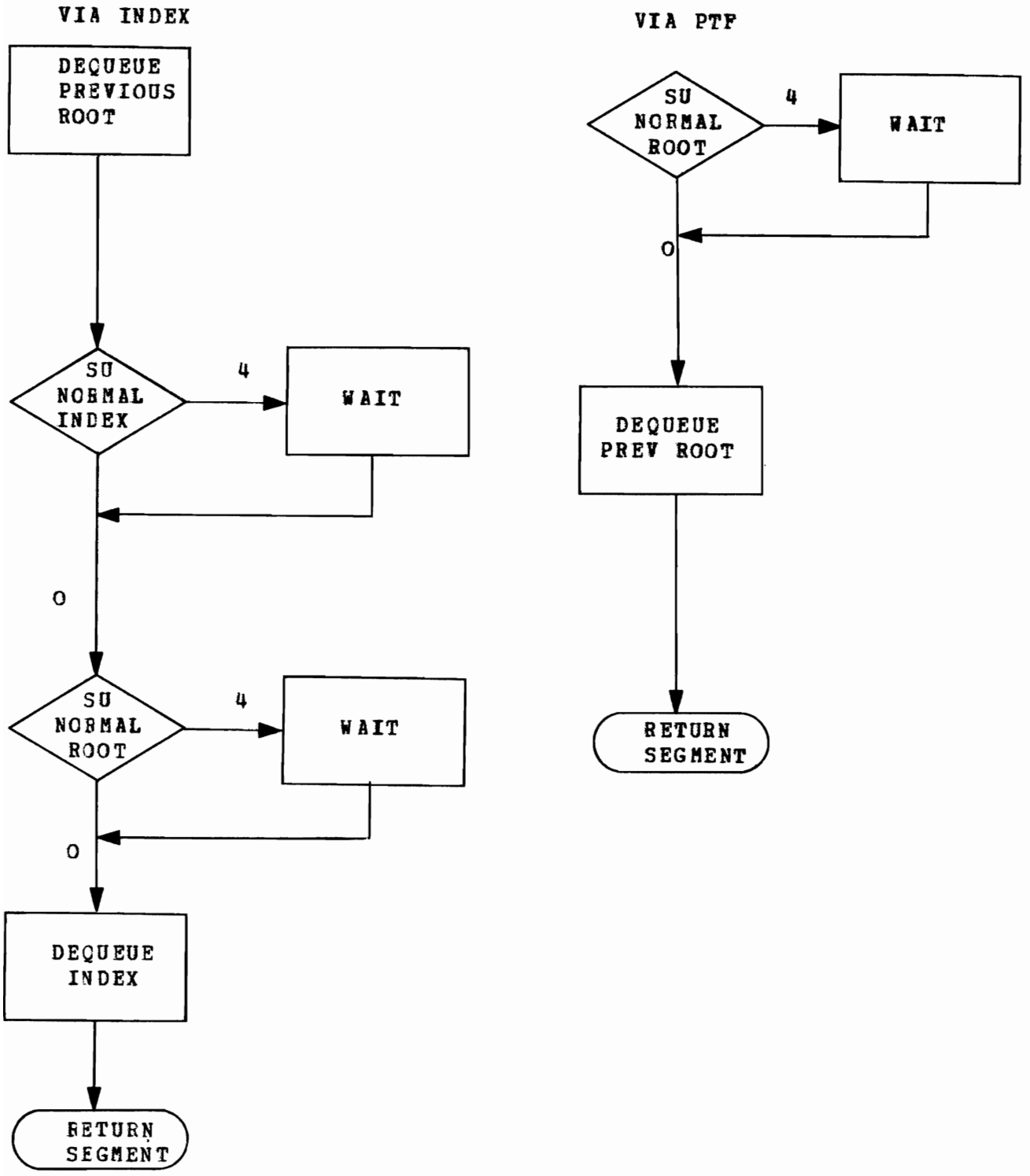
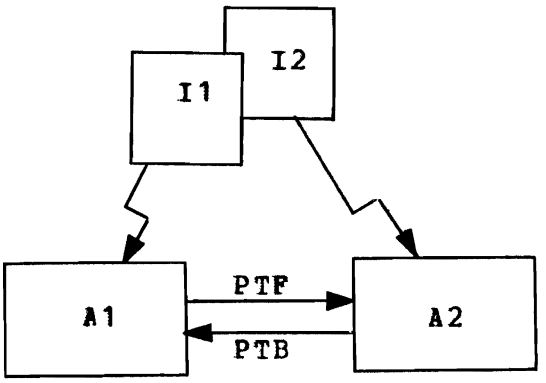


FIGURE 1.

Note that a Single Update (level 2) enqueue is done on the desired root segment for any kind of a get call. Note also that a dequeue is issued for the previously enqueued segment. The

question answered by this logic is "Has any other user established position on this Data Base Record?" The first user would enqueue at a level 2, the second user would also, the resulting total of 4 means the second user must wait. Applying that logic to the following data base would result in the following enqueues.



GU A (KEY = 1)

1. S.U. NORMAL (2) ON I1.
2. S.U. NORMAL (2) ON A1.
3. DEQUEUE I1.

GN A (FOLLOWS THE PTF POINTER)

1. S.U. NORMAL (2) ON A2.
2. DEQUEUE A1.

GU A (KEY = 2) (ASSUME PREVIOUS GU TO A1)

1. DEQUEUE A1.
2. S.U. NORMAL (2) ON I2.
3. S.U. NORMAL (2) ON A2.
4. DEQUEUE I2.

FIGURE 2.

2. RETRIEVE A DEPENDENT.

The logic of retrieval of a dependent is:

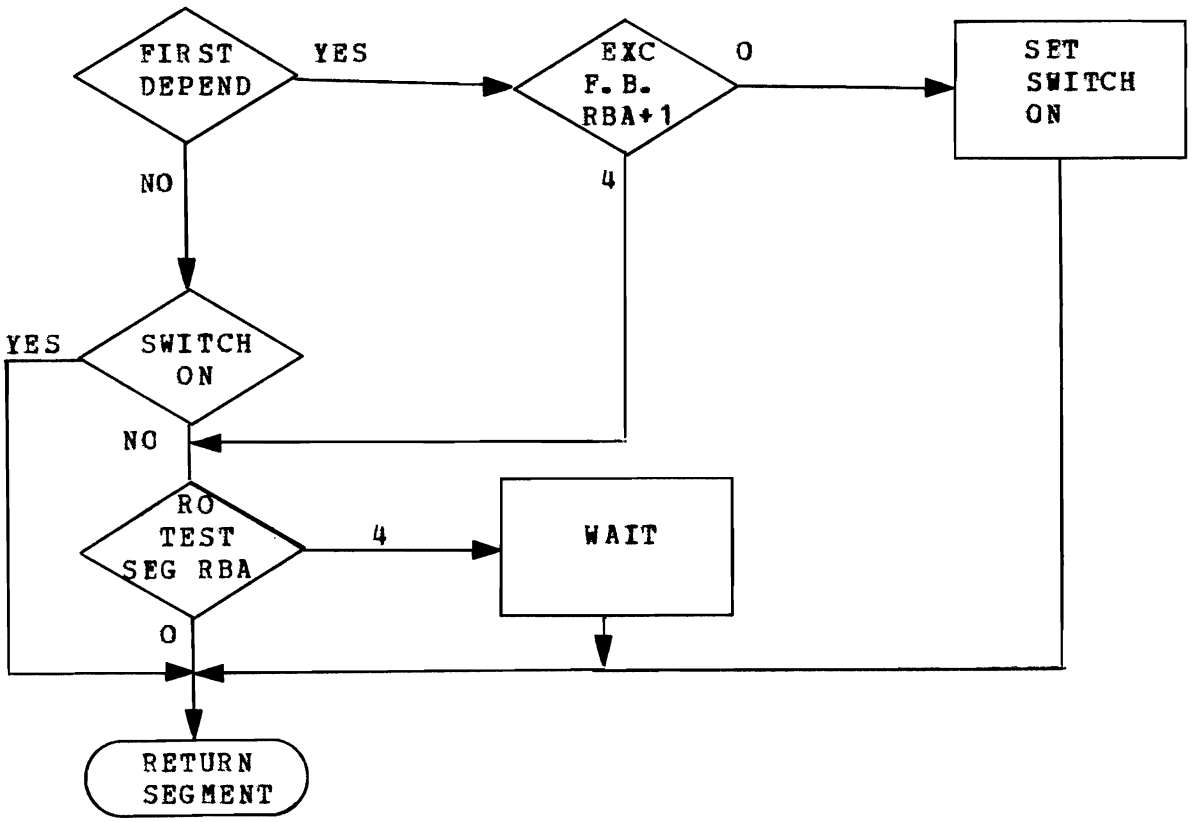
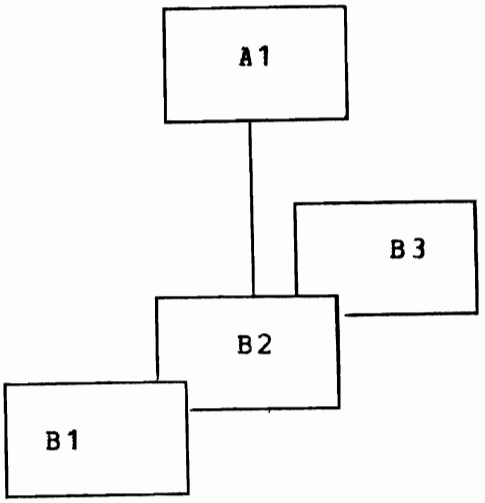


FIGURE 3.

If this is the first dependent accessed, an Exclusive Feedback enqueue is done on the Root RBA +1 (See rule number 4). If any updates had been done on any dependent, the return code would be 4, and each dependent must be tested to determine if the desired segment is indeed the one which was updated. If no dependents were updated, it is unnecessary to test each segment therefore a switch is set. It is cheaper to simply test a switch on succeeding calls than to perform an enqueue. The questions answered in this logic diagram are "Has any dependent segment been updated". If yes, "Is this the dependent?"

Applying that logic to the following data base would result in the following enqueues.



Assume positioning is established on A1.

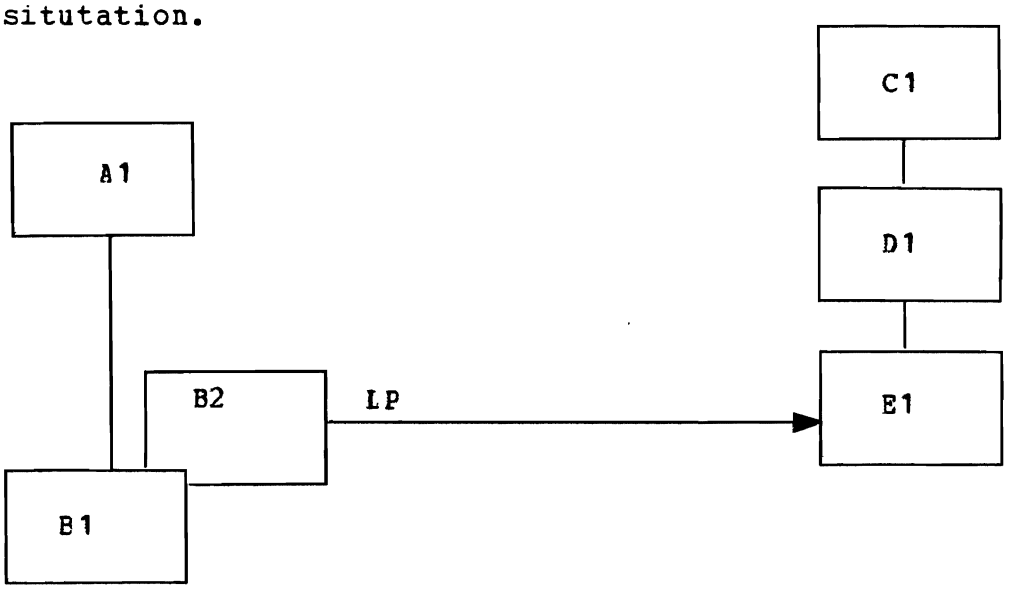
```
GN      B    (KEY=B3)

1. EXC. FEEDBACK (3) ON A1+1.      IF ZERO
2. R.O. TEST     (1) ON B1.
3. R.O. TEST     (1) ON B2.
4. R.O. TEST     (1) ON B3.
```

FIGURE 4.

Note that the test (point 1) can save many enqueues.

3. RETRIEVE A DEPENDENT IN A LOGICAI RELATIONSHIP.

When a concatenated segment is retrieved, the root of the record containing the logically related segment is enqueued at the Single Update level (rule 1B). Since the logically related segment may occur at a lower level, DL/I will test each segment on the path to the root. The following example shows such a situtation.




```
GU    A    (KEY = A1)

1. S.U. NORMAL     (2) ON A1.

GN     B/D (KEY = B2)

1. EXC. FEEDBACK (3) ON A1+1.    IF ZERO

2. R.O. TEST     (1) ON B1.

3. R.O. TEST     (1) ON B2.

4. R.O. TEST     (1) ON E1.

5. R.O. TEST     (1) ON D1.

6. R.O. TEST     (1) ON C1.

7. S.U. NORMAL   (2) ON C1.

8. EXC. FEEDBACK (3) ON C1+1.  IF ZERO

9. R.O. TEST     (1) ON E1.
```

FIGURE 5.

Note that if the test at number 8 fails, DL/I does not test each segment on the path down to the logical parent. DL/I has saved

the RBN of that segment and "jumps" intervening segments. It is worth noting that if the pointer to segment E were symbolic, the path to segment E would be identical to a GU path call through segments C, D, and E and the enqueues would be the same as explained in the Retrieve Root section (i.e. enqueue on the Index, enqueue on C1 instead of points 4, 5, and 6).

4. ISSUE A Q COMMAND CODE CALL.

The logic for the Q command code calls is the same as the Retrieve Dependent calls with the addition of two steps. First, any segment retrieved with a Q command is enqueued at the Single Update level. Secondly, a switch is turned on indicating a Q command was issued for that DMB. If another user had enqueued the segment at a level 2 or higher, the requesting user must wait.

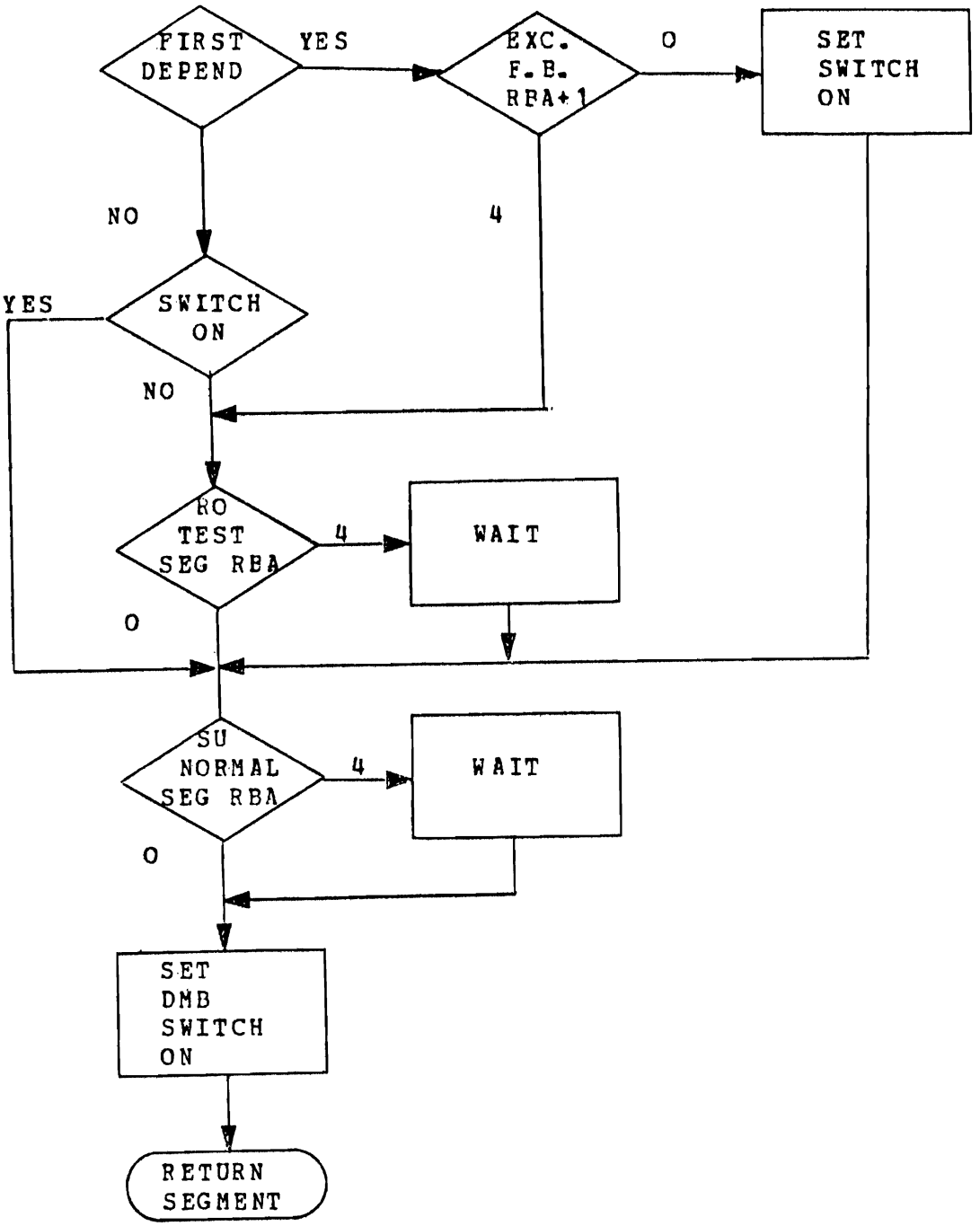
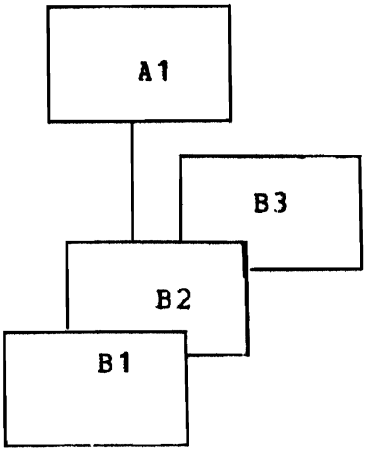


FIGURE 6.

The questions answered by the above logic are "Has any dependent been updated?", and if yes, "Is this the dependent?" "Has another user issued a Q command for this segment?" Applying that logic to the following data base would result in the following enqueues.



Assume position is established on A1.

```
GN   B     *Q      (KEY = B3)

1. EXC. FEEDBACK (3) ON A1+1.   IF ZERO
2. R.O. TEST     (1) ON B1.
3. R.O. TEST     (1) ON B2.
4. R.O. TEST     (1) ON B3.
5. S.U. NORMAL   (2) ON B3.
6. TURN ON SWITCH
```

FIGURE 7.

The Single Update enqueue on B3 will not prevent another user from looking at the segment but will prevent anyone from retrieving it with the intention of updating it.

5. GET HOLD A DEPENDENT.

The get hold logic is identical to the get logic with one addition. A test is made to insure that no other user is holding the segment with a Q command. The logic is:

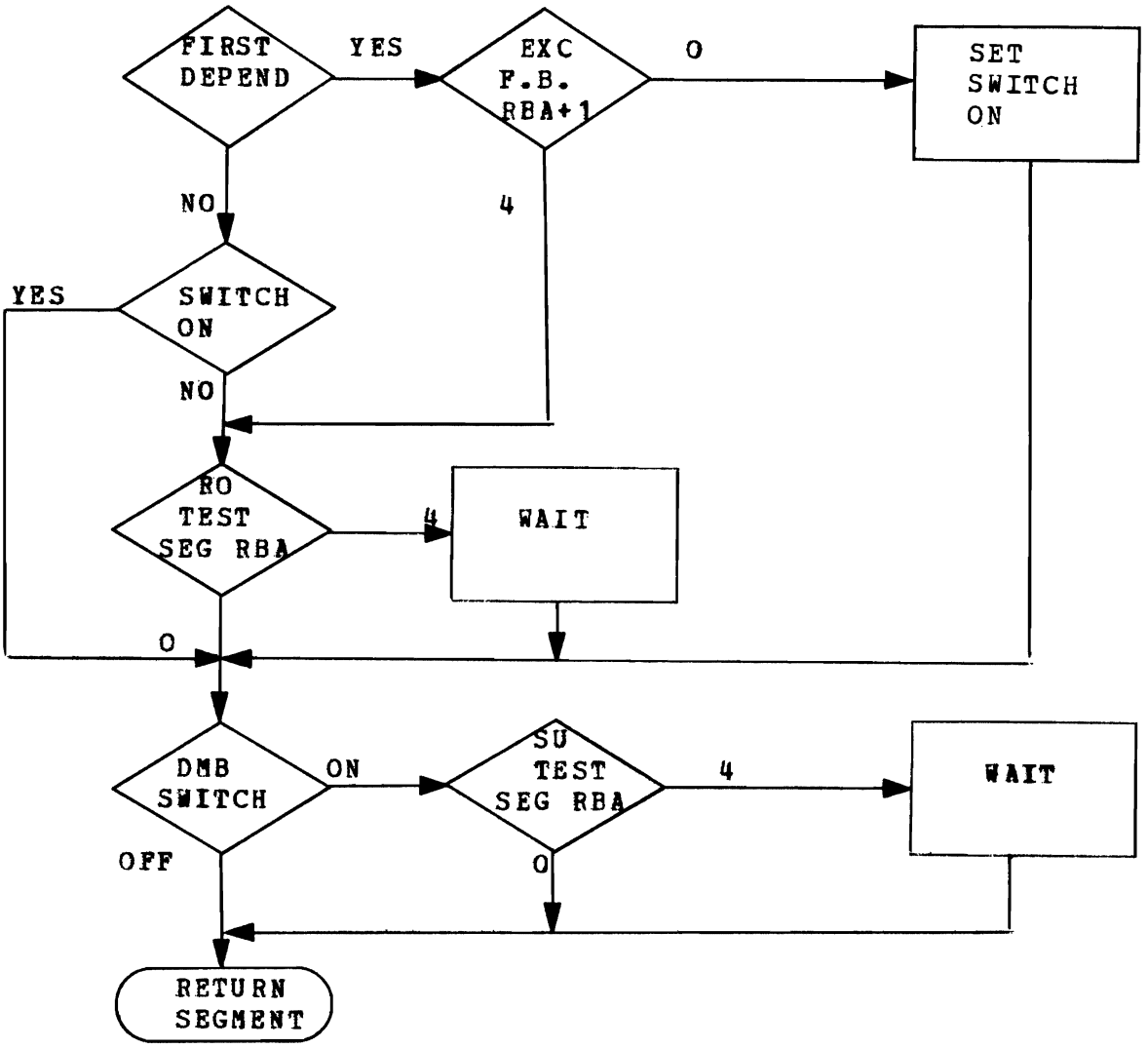
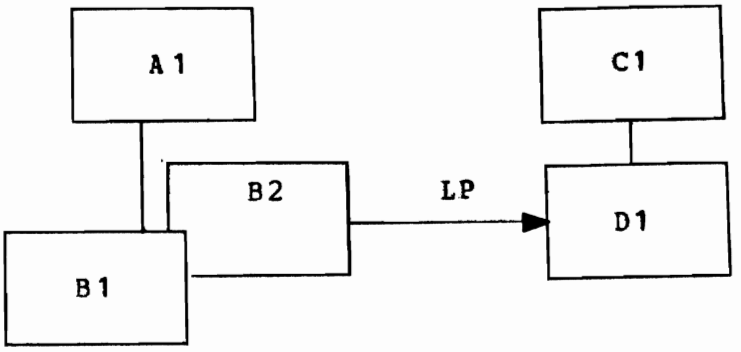


FIGURE 8.

The questions answered here are "Has any dependent been updated?" and if yes, "Is this the segment?" "Has any one issued a Q call for this DMB?" and it yes, "was it for this segment?"

Applying that logic to the following data base would result in the following enqueues.




```
GU      A      (KEY = 1)

1. S.U. NORMAL    (2) ON A1.

GHN     B/D     (KEY = B2)

1. EXC. FEEDBACK (3) ON A1+1.     IF ZERO
2. R.O. TEST     (1) ON B1.
3. R.O. TEST     (1) ON B2.
4. R.O. TEST     (1) ON D1.
5. R.O. TEST     (1) ON C1.
6. S.U. NORMAL   (2) ON C1.
7. EXC. FEEDBACK (3) ON C1+1.     IF ZERO
8. R.O. TEST     (1) ON D1.
   IF DMB SWITCH IS OFF
9. S.U. TEST     (2) ON B2.
   IF DMB SWITCH IS OFF
10. S.U. TEST     (2) ON D1.
```

FIGURE 9.

Note that the test for previous Q calls is performed after all other tests. The test is performed immediately prior to returning the segment to the requestor.

6. REPLACE ANY DEPENDENT.

Any replaced segment is enqueued at the exclusive level. When the segment is a dependent, another enqueue is issued on the root RBA+1. This acts as a switch to inform any other user that he must do a test enqueue when retrieving dependents of this root. The replace logic is:

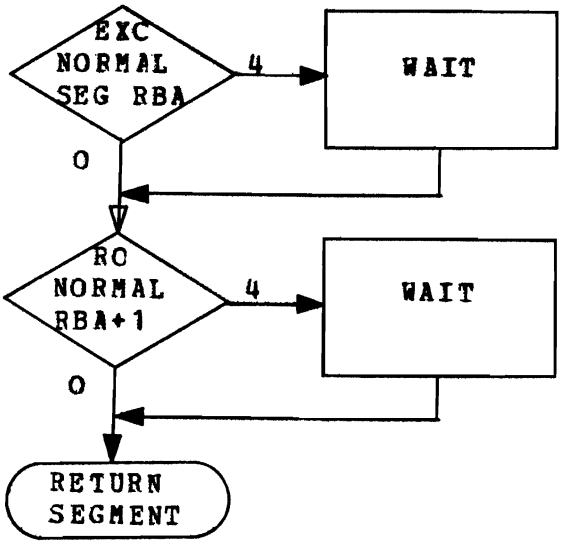


FIGURE 10.

Note that no waits could ever occur. The Get Hold call would have waited if the enqueue were level 2 or greater and the RBA+1 will never be enqueued above a level 1.

The enqueues that would result from a replace call after the Get Hold in Figure 9 are:

REPL B/D (B DATA ONLY CHANGED)

1. EXC. NORMAL (3) ON B2.
2. R.O. NORMAL (1) ON A1+1.

REPL B/D (D DATA ONLY CHANGED)

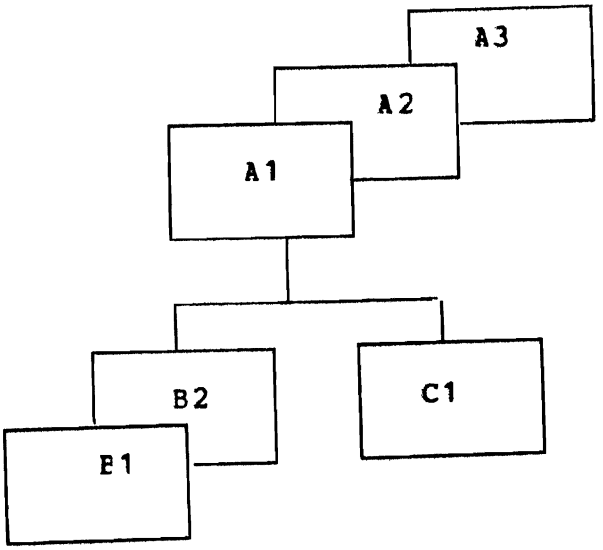
1. EXC. NORMAL (3) ON D1.
2. R.O. NORMAL (1) ON C1+1.

REPL B/D (BOTH B AND D CHANGED)

1. EXC. NORMAL (3) ON B2.
2. R.O. NORMAL (1) ON A1+1.
3. EXC. NORMAL (3) ON D1.
4. R.O. NORMAL (1) ON C1+1.

FIGURE 11

It is worth noting that a segment replace call enqueues only the replaced segment. Once the replacement is done the user can move to another data base record and a second user can interrogate the original data base record. The second user has access to all <u>but</u> the updated segments. For example:




```
USER 1.     GU      A       (KEY = A1)
                    B       (KEY = B2)
            REPL
            GN      A

ENQUEUES:

1. S.U. NORMAL    (2)    ON A1.

2. EXC. FEEDBACK  (3)    ON A1+1.    IF ZERO

3. R.O. TEST      (1)    ON B1.

4. R.O. TEST      (1)    ON B2.

5. EXC. NORMAL    (3)    ON B2.       (ENQ replaced)

6. R.O. NORMAL    (1)    ON A1+1.     (indicate update)

7. S.U. NORMAL    (2)    ON A2.       (ENQ next root)

8. DEQUEUE                  A1.

USER 2.   GU    A    (KEY = A1)
                B    (KEY = B1)
```

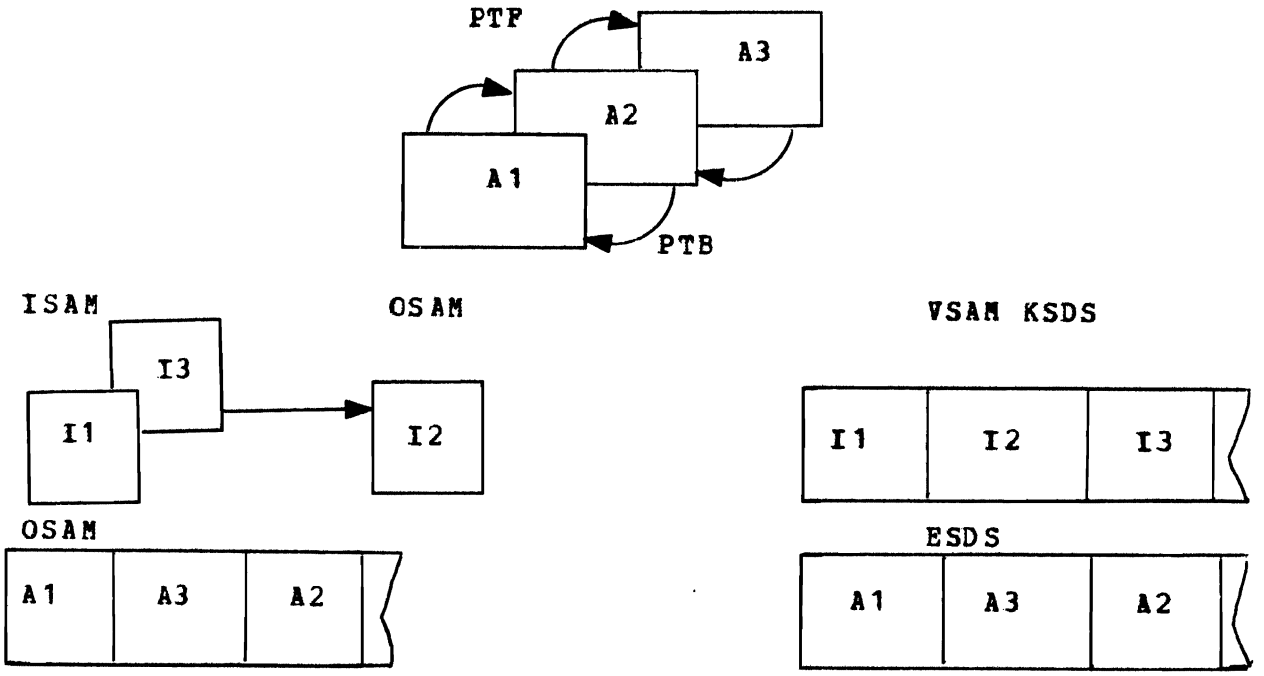
```
ENQUEUES:

1. S.U. NORMAL    (2)    ON A1.

2. EXC. FEEDBACK  (3)    ON A1+1.      (won't be zero)

3. R.O. TEST      (1)    ON B1.

          GN   C

1. R.O. TEST      (1)    ON C1.
```

FIGURE 12

If user two asks for segment B2, he will have to wait but any other segment in that data base record may be accessed.

7. ROOT INSERTION.

The following example shows the insertion of ROOT A2. The diagrams show the data bases after the insertion.



ISRT A2

1. EXC. NORMAL (3) ON I3.
2. EXC. NORMAL (3) ON I2
3. EXC. NORMAL (3) ON A1.
4. EXC. NORMAL (3) ON A3.

1. EXC. NORMAL (3) ON I2.
2. EXC. NORMAL (3) ON A1.
3. EXC. NORMAL (3) ON A3.

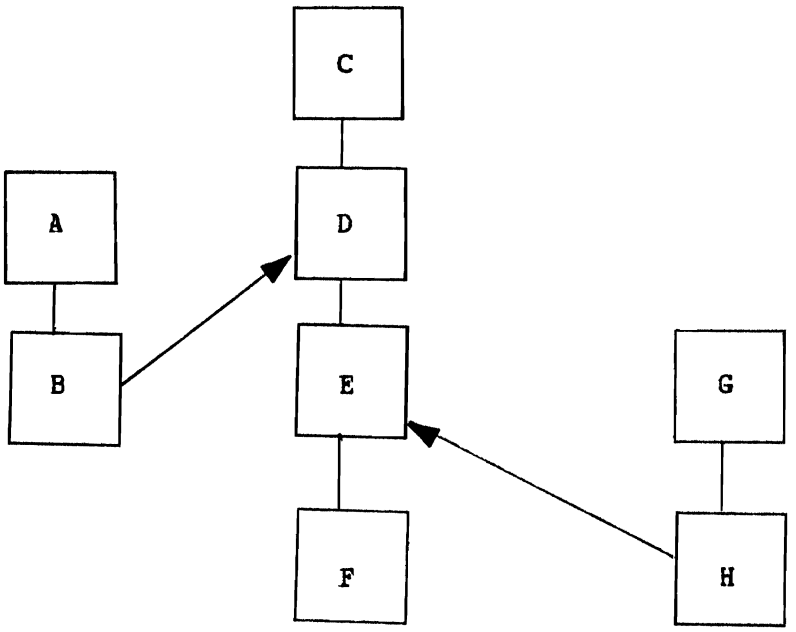
FIGURE 13

In the ISAM case the enqueue on I3 is the result of the pointer to I2 being added. Since the index is a HISAM file, I2 must be enqueued because on retrieval HISAM tests only the logical record containing the desired segment and does not test any logical records along the path. A later retrieval of I2 would look at I3 to obtain the pointer, but only I2 would be enqueued at a test level.

In the VSAM case only I2 is enqueued. In both cases the root segments A1 and A3 are enqueued because of changes to the PTF and PTB pointers. A2 is never enqueued because all paths to the segment are enqueued.

8. SEGMENT DELETION.

When a delete call is issued for a parent segment, delete follows its own logic to find the segments at lower levels. It does not use the logic as described earlier in dependent retrieval. The following example is a simplified version and does not show all the delete rule options possible. Figure 14 shows the logic involved.



A delete of segment C:

1. Follows its own logic to retrieve segment F.

2. Delete of segment F will free the space.

3. Delete of segment E:

   A. Frees space if E is logically deleted (no H exists).

   B. Updates E (PD bit) if not.

4. Delete of segment D:

   A. Frees space if all dependents were freed and if segment D is logically deleted (no B exists).

   B. Updates segment D (PD bit) if not.

5. Delete of segment C:

   A. Frees space if all dependents were freed.

   B. Updates segment C (PD bit) if not.

FIGURE 14.

Figure 15 shows the enqueues which would result as a result of deleting segment C.

1. S.U. NORMAL	(2)	ON C.	(VIA RETRIEVE)
2. EXC NORMAL	(3)	ON D.	
3. EXC NORMAL	(3)	ON E.	
4. EXC NORMAL	(3)	ON F.	
5. EXC. NORMAL	(3)	ON E.	(CHANGE POINTER TO F TO ZERO).
6. DEQUEUE		F.	
7. EXC. NORMAL	(3)	ON D.	(IF E TO BE FREED - POINTER CHANGE)
8. DEQUEUE		E.	(IF FREED)
9. EXC. NORMAL	(3)	ON C.	(IF D TO BE FREED - POINTER CHANGE)
10. DEQUEUE		D.	(IF FREED)
11. DEQUEUE		C.	(IF FREED)

FIGURE 15

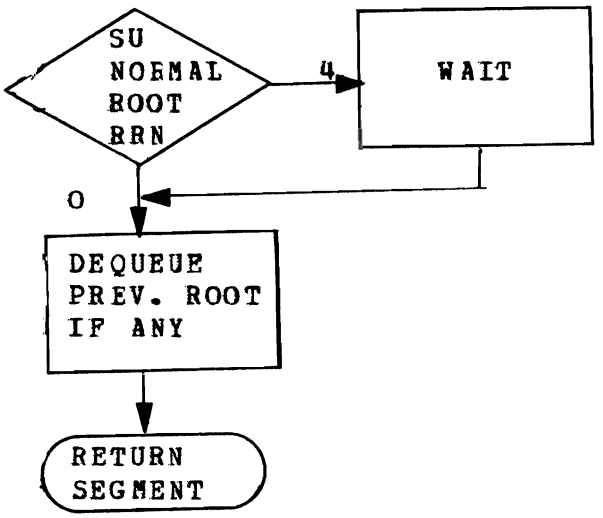
Note that a Exclusive Normal is done for each segment in the path to the lowest level. This insures that there are no outstanding enqueues. If any of these are found, the requestor must wait. Note also that any segment which is updated as a result of a delete is also enqueued. For example, if the delete of segment E had been requested, a twin of E or segment D would have been updated (PTF/PTB or PCF) and consequently enqueued when E was freed.

ENQUEUE EXAMPLES - HISAM

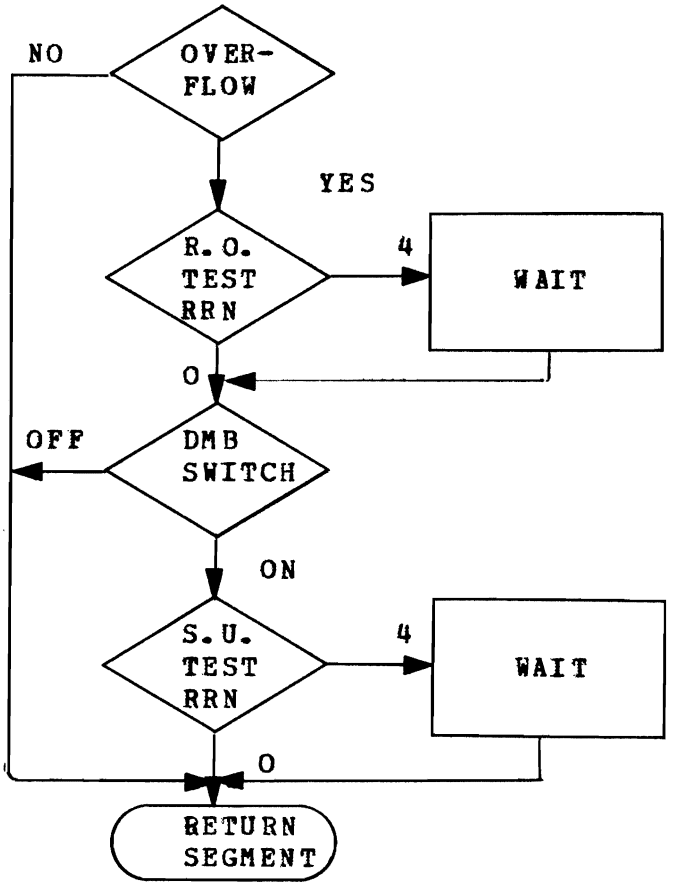
The following example shows the logic of enqueueing in a HISAM data base. Note that all enqueues are on the Relative Record Number. An enqueue for a root segment effectively enqueues all segments within that logical record, therefore, for dependents the only test necessary is to determine whether or not the segment resides in an overflow logical record since the root enqueue handles all the segments in the Prime logical record.

HISAM

GET ROOT



GET HOLD DEPENDENT



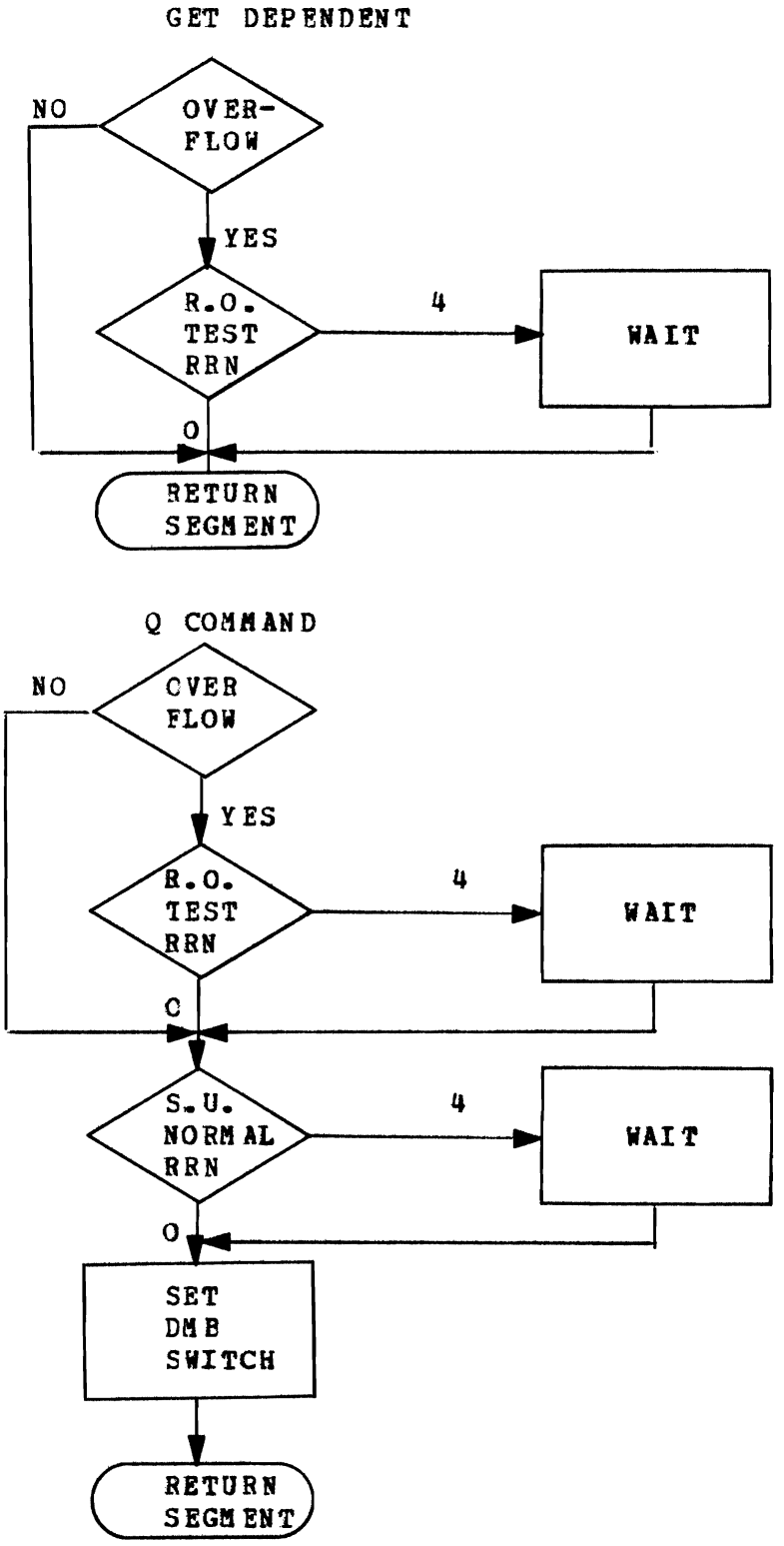

GET DEPENDENT
NO
OVER-
FLOW
YES
R.O.
TEST
RRN
4
WAIT
0
RETURN
SEGMENT
Q COMMAND
NO
OVER
FLOW
YES
R.O.
TEST
RRN
4
WAIT
0
S.U.
NORMAL
RRN
4
WAIT
0
SET
DMB
SWITCH
RETURN
SEGMENT

FIGURE 16.

A user wait is more likely to occur in a HISAM data base because an enqueue of a logical record effectively enqueues a large number of segments. If a dependent segment is updated or retrieved using a Q command code another user is prevented from accessing any segment in the logical record until the first reaches a synchronization point. This should be considered when selecting the data base access method.

APPLICATION PROGRAM - CHECKPOINT/RESTART

The Checkpoint/Restart facility of IMS/VS enables the user to checkpoint any application program except initial data base load programs, and should the need arise, restart the program from that point. User written initial load programs are checkpointed and restartable under the facilities of the Utility Control Facility. Those programs will not issue checkpoint calls.

After Apar 40236 for 1.0.1 and Apar 40366 and Post Req 48421 for 1.1.1 and for later releases of IMS/VS there exists a new symbolic call interface for Checkpoint and Restart DL/I calls. This interface allows Cobol and PL/I programs to specify OS/VS checkpoints. Prior to the availability of this interface these programs had to invoke an assembly language subroutine to issue these calls.

The 1.0.1 user may issue only the BASIC CHECKPOINT CALL requesting IMS/VS to take an OS/VS checkpoint. IMS/VS 1.0.1 has no facilities for extended restart therefore only an OS/VS restart is supported.

With the new interface there exists two categories of checkpoint calls. A discussion of each follows.

BASIC CHECKPOINT CALL.

The format of the Basic Checkpoint call is:

ASSEMBLER:
```
CALL ASMTDLI      ([COUNT,]CHKP,IOPCB,IOAREA[,CKFUNC])[,VL]
                                                  [,DCB]
```

PL/I:
```
CALL PLITDLI      (COUNT,CHKP,IOPCB,IOAREA[,CKFUNC])
```

ANS COBOL:
```
CALL 'CBLTDLI' USING [COUNT,]CHKP,IOPCB,IOAREA[,CKFUNC].
```

where:

COUNT — is the address of a binary fullword containing the number of parameters following (required for PL/I).

CHKP — is the address of the call function "CHKP".

IOPCB — is the address of the I/O PCB or the "dummy" I/O PCB supplied by the CMPAT option in PSBGEN.

IOAREA is the address of the user's I/O area. In applications which access the IMS/VS message queues, the CHKP call implies a message GU and a message may be returned in this area. In batch and batch message programs the I/O area must contain the 8 byte checkpoint ID. This is used for operator or programmer communication, and since it must be entered in the event of restart, should consist of EBCDIC (printable) characters.

CKFUNC is the address of an 8 byte area containing the value "OSVSCHKP". If specified and OS checkpoint dataset DD cards are provided, IMS/VS will issue an OS/VS checkpoint for the user's region, using IMS supplied DCBs, before proceeding with the DL/I CHKP call. The DD names associated with the IMS supplied DCBs are CHKDD and CHKDD2. If both data sets are specified, they are used alternately. In on-line regions, the OS checkpoint data set(s) are reset to record one before issuing the OS checkpoint. In batch regions all checkpoint records are kept sequentially and are thus available for restart.

DCB is the address of a user supplied DCB. If specified, IMS/VS will issue an OS/VS checkpoint for the user's region, using the user supplied DCB, before proceeding with the DL/I CHKP call.

NOTE: The optional parameters "CKFUNC" and "DCB" are mutually exclusive. The "DCB" is valid only for batch or batch-message programs written in ASSEMBLER Language. If the basic CHKP call is used, the following rules apply:

1. The user can request that IMS/VS issue an OS checkpoint for the user's region.

2. The user cannot issue an OS checkpoint. If a restart were attempted OSAM and other obscure system problems would occur.

EXTENDED CHECKPOINT CALL.

The format of the Extended Checkpoint call is:

ASSEMBLER:

```
CALL ASMTDLI     ([COUNT,]CHKP,IOPCB,IOLEN,IOAREA
                 [,1STLEN,1STAREA,.....,NTHLEN,NTHAREA])[,VL]
```

```
PL/I:
CALL PLITDLI     (COUNT,CHKP,IOPCB,IOLEN,IOAREA
                  [,1STLEN,1STAREA,.....,NTHLEN,NTHAREA]);

ANS COBOL:
CALL 'CBLTDLI'  USING [COUNT,]CHKP,IOPCB,IOLEN,IOAREA
                [,1STLEN,1STAREA,.....,NTHLEN,NTHAREA].
```

where:

COUNT — is the address of a binary fullword containing the number of parameters following (required for PL/I).

CHKP — is the address of the call function "CHKP".

IOPCB — is the address of the I/O PCB or the "dummy" I/O PCB supplied by the CMPAT option in PSBGEN.

IOLEN — is the address of a binary fullword containing the length of the largest I/O area used by the user program. This parameter is not used by the checkpoint call but is used by the Restart call. It is required so that the same source language statement could serve for both the CHKP and XRST calls.

IOAREA — is the address of the user's I/O area. In applications which access the IMS/VS message queues the CHKP call implies a message GU and a message may be returned in this area. In batch and batch message programs the I/O area must contain the 8 byte checkpoint ID. This is used for operator or programmer communication and, since it must be entered in the event of restart, should consist of EBCDIC (printable) characters.

1STLEN — is the address of a binary fullword containing the length of the first area to be saved.

1STAREA — is the address of the first area to be saved.

NTHLEN — is the address of a binary fullword containing the length of the last area to be saved.

NTHAREA — is the address of the last area to be saved.

NOTE: The length and area parameters are optional and a maximum of seven (7) areas may be specified. The area should never contain addresses such as ADCON's or VCONS. At restart time the program will probably reside in a different location thus invalidating saved addresses.

SYSTEM ACTION AT CHECKPOINT.

When DL/I receives a CHKP call, it writes to the data base all buffers which were modified by the PST. In Batch when an OS/VS checkpoint is requested the data bases are closed. Apar P51969 for IMS/VS 1.1.1 and 1.1.3 will cause the data bases not to be closed for all other checkpoint calls.

A log record (type 18) is created which contains the checkpoint ID passed with the call.

For a message or batch message region all Program Isolation enqueues are released. All Dynamic Log records for that PST are released.

The positioning of each data base PCB (except GSAM) is set to the beginning of the data base by clearing the level table. There are three reasons why this is necessary.

A. Although DL/I repositions the data bases at restart, the exact position cannot be guaranteed when non-unique or non-keyed segments are involved (see Restart).

B. It is necessary to force the user to re-establish data base positioning thereby re-enqueueing any necessary parent segments. Consider, for example, the following sequence of calls. GU ROOT, GN DEPENDENT 1, CHKP, GN DEPENDENT 2. The CHKP frees all enqueues. If the GN DEPENDENT 2 returned dependent 2, the user would be positioned on the dependent without an enqueue on the root. This would be a severe integrity exposure.

C. When a user program issues a CHKP call which releases all enqueues, that program has relinquished all rights to any segments which were enqueued, and has reached a synchronization point. A higher priority program could then access a data base record which this program had previously enqueued before this program is again dispatched. For this reason the checkpointed program must re-establish data base positioning and all enqueues to be certain the requested record is available. To prevent any of the above, the user is forced to re-establish data base positioning.

If the MPP or BMP program references a transaction code, the message queue for that transaction is checkpointed meaning a synchronization point has been reached. After the checkpointing action is completed, a GU call to the input message queue is internally generated, and a new message (if one is available) is returned in the I/O area. When this situation exists, the user should test for the return codes normally received as a result of a GU to the message queue (QC, etc.) in addition to those expected from a checkpoint call.

The CHKP call is used in a message processing region in conjunction with multiple mode scheduling of transactions. It allows the user to determine the grouping of messages for backout and restart purposes. For single mode scheduling or multiple mode scheduling where no grouping is necessary, Program Isolation will handle all checkpoint functions. The grouping IMS/VS uses is either that each message is unique or that all messages read at a given schedule of the program are considered to be connected. The CHKP call allows the user to specify groupings in between.

BASIC RESTART.

1. OS/VS RESTART.

   If the user had specified that an OS/VS checkpoint be taken, then OS/VS restart procedures should be followed. There are certain limitations associated with OS/VS Checkpoint/Restart. These are:

   a) At restart time the same address space must be available.

   b) No program changes are allowed between the time of failure and restarting. Most program failures are the result of code deficiencies and since the code cannot be changed, this form of checkpoint and restart offers no solution.

   c) OS/VS restart does not reposition any direct access files.

2. USER RESTART.

   User restart routines could be coded such that the program logic could recognize a restart situation and reposition input files. One technique might be to use the input key value as the checkpoint ID. A method could be devised to allow the input of that information to the program which then would reposition the input file.

EXTENDED RESTART CALL.

The user has two methods of informing IMS that a restart is occuring. He may code the CKPTID parameter in the EXEC card (this is preferred) or via program logic move the checkpoint ID into the work area pointed to by the XRST call. The first DL/I call in the program must be a XRST call. Upon receiving this call, IMS/VS checks whether a checkpoint ID has been supplied in the PARM field of the EXEC card or in the work area pointed to by the XRST call.

If no ID has been supplied, the call is treated as a NOP, except a flag is set to indicate that any checkpoint calls are the Extended Checkpoint variety. This is how DL/I differentiates between Basic and Extended checkpoint calls.

If the checkpoint ID has been supplied, the IMS/VS batch restart routine reads forward on the log defined in the //IMSLOGR DD statement to locate the checkpoint records.

The user program area (defined in the area parameters in the call) are restored to their status as of checkpoint time.

Each input GSAM data base that is active at the checkpoint is repositioned for sequential precessing by issuing a GU for the last record processed at that point. Output GSAM data bases are repositioned using a POINT call. GSAM data bases being loaded are repositioned if they are defined to use BSAM accessing (in the DBD).

Using the I/O area length parameter, a storage area is obtained. A GU call for each data base is issued using the checkpointed PCB feedback areas. The storage area is used as the call I/O area. The segment data is not moved to the user I/O area, however, the PCB feedback area is restored. If the user has a need for the segment data, he must issue a call using the PCB key feedback. If the data base contains segments which have non-unique or no keys, data base positioning may not be at the exact position which existed at checkpoint time. The user program should always reposition itself on all non-GSAM data bases.

Data base initial load programs (except for GSAM) are not restartable except under control of the Utility Control Facility.

The format of the Extended Restart call is:

```
ASSEMBLER:
CALL ASMTDLI    ([COUNT,] XRST,IOPCB,IOLEN,WORKAREA
                [,1STLEN,1STAREA,.....,NTHLEN,NTHAREA]) [,VL]

PL/I:
CALL PLITDLI    (COUNT,XRST,IOPCB,IOLEN,WORKAREA
                [,1STLEN,1STAREA,.....,NTHLEN,NTHAREA]);

ANS COBOL:
CALL 'CBLTDLI' USING [COUNT,]XRST,IOPCB,IOLEN,WORKAREA
                [,1STLEN,1STAREA,.....,NTHLEN,NTHAREA].
```

where:

COUNT — is the address of a binary fullword containing the number of parameters following (required for PL/I).

XRST — is the address of the call function "XRST".

IOPCB — is the address of the I/O PCB or the "dummy" I/O PCB supplied by CMPAT option in PSBGEN.

IOLEN — is the address of a binary fullword containing the length of the largest I/O area used by the user program. This value determines the amount of storage needed to hold the longest segment accessed by the restart process.

WORKAREA — is the address of a 12 byte area. This area should be set to blanks prior to the call and tested upon return. If the program is being started normally, the area will be unchanged. If the program is being restarted, the ID supplied in the EXEC statement will be placed in this area.

If the user wishes to use his own restart method (he has his own checkpoint file on a non-DL/I file) the XRST call can be used to reposition GSAM data bases by placing the checkpoint ID in this area before issuing the call. This ID can be either the 8 byte left-aligned ID, or the 12 byte YYDDD/HHMMSS time of the checkpoint.

1STLEN — is the address of a binary fullword containing the length of the first area to be restored.

1STAREA — is the address of the first area to be restored.

NTHLEN — is the address of a binary fullword containing the length of the last area to be restored.

NTHAREA — is the address of the last area to be restored.

NOTE:

1. The XRST must contain the same number of parameters and in the identical sequence as the CHKP call.

2. The lengths of the areas must be identical to those specified in the CHKP call.

3. The maximum number of areas specified is 7.

4. The XRST must be issued only once, and must be the first DL/I call issued.

RELEASE COMPATIBILITY.

Those IMS/VS 1.0.1 users who coded checkpoint calls in the old format need not change their programs unless usage of the new features is desired. The old format is supported.

The old format did not provide a restart call so while those programs can be checkpointed, they can be restarted only under OS/VS Checkpoint/Restart or a user written restart routine.

These programs will execute in the same fashion on IMS/VS 1.1.1 and later systems.

Those IMS/VS 1.1.1 users who coded checkpoint calls in the old format are still supported. All new development should use the Extended Checkpoint facilities.

GENERAL CONSIDERATIONS.

If the user has elected to use OS/VS checkpointing and IMS/VS receives other than a return code of 0 or 4 from the OS/VS routine, a U475 Abend occurs. A Batch Message region would be backed out to the previous checkpoint because of the abend. Batch Backout should be executed for batch regions. Once the problem is corrected each can be restarted from the last checkpoint.

Users who have the option of using Basic or Extended must use one interface exclusively. When using the Extended, the first DL/I call must be a XRST.

BMP programs are basically one of two types. One type accesses the message queues, the other does not. The entire discussion above pertains to those BMPs which do not access the message queues.

A checkpoint call acts like a GU call to the message queue in that the next messge is returned in the program I/O area. If a checkpoint call is issued for every message it would result in far too many checkpoint calls, therefore, the logic should be to

issue a series of GU calls between checkpoint calls. When a checkpoint call is issued a message is written to the console. Situations have developed where the calls were issued so frequently that the console printing backed up. This should be avoided.

A BMP which accesses the message queues need not be "restarted" after a failure. The user may simply "cold" start the program and processing will continue with the next message on the queue. The dynamic or batch backout which followed the failure would have backed out to the last synchronization point. Any messages processed since the synchonization point would have been returned to the queue and a "cold" start would begin processing with the first of these.

There may be situations when a "restart" is desirable. The program may be gathering statistics, it may be creating an output GSAM file which should be repositioned, or it may have checkpointed data necessary for continued processing. If the ability to "restart" is desirable or necessary, consideration should be given to the MODE specified for the transaction.

With a MODE=SNGL specification there is no reason to mix GU and checkpoint calls. Each GU or checkpoint call results in a synchronization point. Backout will backout to the last synchronization point which probably will not coincide with a checkpoint call. Restarting from the last checkpoint means restarting from an earlier point in time which could produce spurious results. With this mode of operation the user must issue all GU calls or all checkpoint calls.

With a MODE=MULT specification only the checkpoint call results in a synchronization point. Either type of backout will backout to that point and the program may be restarted at that point.

CHECKPOINT ID.

The user is encouraged to provide unique checkpoint IDs for each checkpoint call issued. At restart the log tape is read forward searching for a X'18' log record matching the supplied ID, and the first matching ID becomes the restart point. If the IDs are not unique the user must supply the date/time of the checkpoint instead of the ID. Also, remember backout was done to the last checkpoint so starting at any previous checkpoint should not be attempted except possibly in a batch environment and there is probably no reason to do so then.

If a given log tape contains two executions of the same BMP which issues the same checkpoint IDs for each execution, the user must specify the date/time as a checkpoint ID in order to restart the second execution.

As an aid to restarting, the user might wish to avoid duplicating

IDs across programs. He might consider using a convention such as:

a) 3 bytes of information which uniquely identifies the program.

b) 5 bytes of information to identify the ID. This could be a value which is incremented by one for each checkpoint call.

Whatever the method chosen, it is recommended that checkpoint IDs be unique within and across programs.

PL/I PROGRAMS.

For the Extended Checkpoint call and the XRST call, the I/O area length parameter (and the first area length, etc.) must be coded in one of the following ways for PL/I.

a)
```
DCL 1 IO-AREA-LEN,
      2 LNTH FIXED BIN(31) INIT(xxx);
```

b)
```
DCL IO-AREA-LEN PTR;
DCL LNTH FIXED BIN(31) INIT(xxx);
IO-AREA-LEN = ADDR(LNTH)
```

This is because IMS/VS expects PL/I parameters to have dope vectors, which add an extra level of pointers. But, variables declared fixed binary (at level 1) don't have dope vectors. The parmcount parameter is the only parameter which may and must be declared fixed binary (31)!

CHECKPOINT CALL FREQUENCY.

Message processing programs.

The only purpose of checkpoint call in a MPP is to determine groupings of messages when MODE=MULT has been specified. Since it is recommended that MODE=SNGL be used for performance reasons, very few users would ever issue checkpoint calls in a MPP. For those who do, the consideration is how many messages does he wish to process between synchronization points.

Batch message programs.

The frequency of checkpoint calls in a BMP is the most critical of all program types. There are no definate rules to aid the user. Some of the things he must consider are Dynamic Log usage, Enqueue space, Enqueue lockout, and Restart time in the event of a program failure or Roll call issued on behalf of a Program Isolation deadlock.

Failure to issue checkpoint calls can cause the Dynamic Log to reach the wrap point. When the user receives DFS228 messages he

must either increase the size of the Dynamic Log or issue more frequent checkpoint calls.

Since Program Isolation enqueues are held to synchronization points, the enqueue space could overrun between synchronization points. If so a U775 abend occurs. The user must increase the available storage or issue more frequent checkpoint calls.

A fact that may be more critical than storage space is Enqueue lockout. For the period of time that a data base resource is held by an enqueue, no other user may access it. This could cause excessive wait times in other regions. If a BMP performs a large number of updates between checkpoint calls it could enqueue on large blocks of the data base. This will have to be evaluated by the user since only he knows what program contention for data might exist.

The last influencing factor is the time to perform backout and reprocessing time after failure. Hopefully program failures are rare and enqueue deadlock situations can be controlled by scheduling techniques, so this is probably the least important facet.

Batch Programs.

Batch programs have no Dynamic Log or Enqueue considerations. The time required to backout and reprocess after a failure is probably the main consideration. It would seem reasonable to checkpoint every ten or fifteen minutes in most cases.

For users who foresee the need to backout an entire batch job a checkpoint call must be the first call in the program (following the XRST). Batch backout will backout to the specified checkpoint or the last checkpoint if none is specified. By issuing a checkpoint call at the beginning of a job the user has a checkpoint ID to specify in order to back out the entire run.

ALL PROGRAMS.

A checkpoint call should never be issued while position within a data base record is important. Since positioning is lost at call time it is only logical to issue checkpoint calls just before issuing a GU ROOT call. For users who issue GN calls through a data base a key must be saved and a GU for the Root specifying a Key Greater or Equal value issued following the checkpoint call. This re-establishes positioning and enqueues the appropriate segments.

Whatever the frequency chosen the user is encouraged to code it such that it can easily be changed by a link edit or simple modification and compile. Some users have elected to issue checkpoints based upon an elapsed time, others on a count of root segments accessed, others on a count of updates performed. If

complex coding structures are involved the effort to change the frequency can be considerable. Advanced planning can eliminate this.

IMS/VS RECOVERY UTILITIES

This section will be devoted to the IMS/VS utilities which play a major role in data base recovery.

IMAGE COPY (DFSUDMPO)

The purpose of this utility is to create a copy of the user's data base. The copying is done on a Dataset basis which allows the user the option of copying a single Dataset or all Datasets for the entire data base. Copying all Datasets at one time is highly recommended.

Multiple datasets can be copied in one Job step as long as the datasets are contained within the same database. Image Copy requires a separate Job Step for each data base. A data base would have a DBD therefore HIDAM, HIDAM INDEXES, and each SECONDARY INDEX, are separate data bases.

The user has the option of copying the index of a KSDS. This becomes advantageous in the event of an index error. The user can recover a track of the index instead of reloading the KSDS and thereby rebuilding the index.

MULTIPLE COPIES:

The user has the option of creating one or two copies of each data set. The advantage of specifying two copies is that if an I/O error occurs on one copy, the job will continue on the other copy. Also if one copy is not readable a Recovery can be performed using the second. Performance is degraded by the amount of time required to physically write the second copy.

JCL AND CONTROL STATEMENTS:

Refer to the Utilities Reference Manual SN20-9029.

ACCESS METHODS:

1. Input Data Base:

   ISAM Data Bases are read using QISAM.

   OSAM Data Bases are read using QSAM.

   KSDS Data Bases are read using the DL/I buffer handler. The first logical record is retrieved via a call requesting retrieval by key beginning at the start of the Data Base. Each succeeding logical record is read using a Get Next Root by Key function.

   ESDS Data Bases are read using the DL/I buffer handler. Each logical record is retrieved via a call requesting retrieval by Relative Record number.

   KSDS INDEX records are read using the DL/I buffer

handler. Each logical record is retrieved via a call requesting retrieval by Relative Record number.

2. Output File(s):

The LRECL size is obtained from the data set being copied. Should the LRECL be less than 42 bytes or 54 bytes in 1.1.3, a minimum LRECL of 42 or 54 is substituted in order to provide space to contain the Header record. To the LRECL value is added from 8 to 11 bytes. The 8 bytes are used for Image Copy control information and 0 to 3 addition bytes are added for alignment to insure that the LRECL value is divisable by 4.

IMS/VS 1.1.1 PTF 3 and later systems:

If the data set block size exceeds 16K that block size is used and the blocking factor is one. If the data set block size is less than 16K the output will be blocked up to 16K or output device maximum.

Earlier systems:

If the calculated LRECL size is greater than 3625, the LRECL size becomes the blocksize. If it is not, a blocksize is calculated by dividing 3625 by the LRECL size and subtracting the remainder. The user could obtain a much more favorable blocksize by changing a constant labeled "KON3625". To do so should improve performance.

The output file(s) is written using QSAM.

OUTPUT FILE CONTENT:

The first record on the output file is a header record with the following format.

HSAM CTRL	ID OUT	DCB NOOUT	DB OUT	IDDN OUT	DATE OUT	TIME OUT	ODDN OUT	IBLKS OUT	ILREC OUT
1	1	2	8	8	4	4	8	2	2

OBLKS OUT	OLREC OUT	IKEY LENG	IKEY POS	DBORG OUT
2	2	2	2	1

where:

HSAMCTRL	X'00'
IDOUT	'D' Identifies this as a Dump header.
DCBNOOUT	X'00' or DCB number.
DBOUT	Data Base Name (DBD Name).
IDDNOUT	Prime ISAM/KSDS DD Name or OSAM/ESDS DD Name
DATEOUT	System Date.
TIMEOUT	System Time (HHMMSSTH).
ODDNOUT	OSAM/ESDS DD Name.
IBLKSOUT	ISAM Blocksize/KSDS CI Size.
ILRECOUT	ISAM/KSDS LRECL.
OBLKSOUT	OSAM Blocksize/ESDS CI Size.
OLRECOUT	OSAM/ESDS LRECL.
IKEYLENG	ISAM/KSDS Key Length.
IKEYPOS	Relative Key Position.
DBORGOUT	Data Base Organization.

01 = HISAM single data set group.
02 = HISAM multiple data set groups.
03 = SSAM.
04 = HSAM multiple segments.
05 = HDAM.
06 = HIDAM.
07 = INDEX
09 = HISAM/VSAM single data set group.
10 = GSAM.
0B = HISAM/VSAM single segment.
0C = INDEX/VSAM single data set group.
0D = HDAM/VSAM.
0E = HIDAM/VSAM.
0F = INDEX/VSAM.

The remaining output records have the following format.

CONT OUT	DSID OUT	BLNK DOUT	USER LRECL
4	1	3	VARIABLE

where:
CONTOUT RBA value if VSAM, else output record count.
DSIDOUT 0 = OSAM/ESDS, I = ISAM/KSDS
BLNKDOUT Binary zeros.
USERLREC Data Set Logical Record.

OUTPUT MESSAGES:

Execution of this utility can result in any of the following messages being issued. Most of the messages are self explanatory. For the remainder, the reason for issuing the message is listed. The "REQUEST" which appears in many messages is the result of UCF. When executing the utility under UCF the user may specify multiple Data Sets be image copied. In that event each would be assigned a request number.

1. DFS301A UNABLE TO OPEN DDNAME xxxxxxxx.

2. DFS302A INVALID OR MISSING FUNCTION IM SPECIFIED ON REQUEST nnnn.

   a) Column 1 of control card must contain a 'D'.

3. DFS303A nn COPIES IS AN INVALID REQUEST FOR FUNCTION IM ON REQUEST nnnn.

   a) 2 copies requested - 2nd DD NAME was not supplied.

   b) 1 copy requested - 2nd DD NAME was supplied.

   c) Column 2 of the control card must contain a 1 or 2.

4. DFS304A DBD NAME NOT SPECIFIED FOR FUNCTION IM ON REQUEST nnnn.

   a) A DBD name must be supplied in columns 4 through 11.

5. DFS305A DBD LIBRARY DOES NOT CONTAIN DBD xxxxxxxx.

6. DFS306A DBD xxxxxxxx DOES NOT CONTAIN DDNAME FOR FUNCTION IM.

a) DD name specified in columns 13-20 is not valid.

7. DFS307A NO DDNAME SPECIFIED FOR DATA BASE xxxxxxxx FOR FUNCTION IM.

a) Control card columns 13-20 are blank.

8. DFS308A NO OUTPUT DDNAME SPECIFIED FOR FUNCTION IM ON REQUEST nnnn.

a) Control card columns 22-29 are blank.

9. DFS310A INPUT RECORD IMAGE FOR FUNCTION IM IS IN ERROR AND FOLLOWS THIS MESSAGE.

a) The control card contains an error (card follows).

10. DFS316A DBD xxxxxxxx CONTAINS AN UNKNOWN ORGANIZATION CODE xx.

a) The valid codes are documented in the DBORGOUT field of the Header Record.

11. DFS319A DDNAME xxxxxxxx HAD A PERMANENT I/O ERROR FOR FUNCTION IM.

12. DFS320I DDNAME xxxxxxxx IS STOPPED BUT DDNAME xxxxxxxx CONTINUES.

a) An I/O error occured on one output copy.

13. DFS321I FUNCTION IM EXPERIENCED AN ERROR AND MUST BE RESTARTED.

a) The last or only output data set experienced an I/O error.

14. DFS341A CATASTROPHIC ERROR OCCURED IN DL/I DURING FUNCTION IM.

a) Buffer handler return code indicates an illegal call or the KSDS key was not found or the end of data was reached prior to a key of FFFF's.

15. DFS391I DATA BASE xxxxxxxx TYPE ddname COPY 1(2) ON VOLUMES ser #, ser #, ...ser#.

a) Standard EOJ message.

16. DFS392A DUPLICATE NAMES xxxxxxxx SPECIFIED ON CONTROL REQUESTS FOR FUNCTION IM.

a) Control card columns 13-20 and 22-29 are duplicates.

17. DFS394A AN RBA SEQUENCE ERROR OCCURED FOR FUNCTION IM.

a) Buffer handler return code was X'04' in PSTRTCDE.

DATA BASE BACKOUT (DFSBB000)

The Data Base Backout utility provides the user with the capability to remove data base changes made during the execution of a DLIBATCH region. It also should be used to backout data base changes made by a BMP region or MPP region which was active at the time of system failure when Emergency Restart also fails. It is also used by the system during Emergency Restart to backout data base changes made by MPP regions and BMP regions which were active at the time of system failure. It is also invoked when a ROLL call is issued by an application program or when a Program Isolation deadlock situation develops.

Data Base Backout is comprised of three modules.

DFSBACK0 analyzes the user control card, reads the input log tape, and passes the appropriate log records to DFSRDBC0.

DFSRDBC0 reads the Data Base records, performs the functions necessary to restore the Data Base to its before image, and logs the changes made.

DFSBACM0 is a module which contains all the messages issued during Backout.

Data Base Backout has certain characteristics which make it unique. Unlike recovery, Backout does not always do a physical replacement of data. If necessary, it calls the DL/I space management routine to free or reserve space in a record for a previously inserted or deleted segment. Since it does not restore the Data Base to its original physical status, a system log tape is created by Backout and that log must be input to any subsequent Data Base Recovery procedure. Backout is executed to backout the effect of a PSB, which means that it backs out changes to all Data Bases that were made by a specific program.

BACKOUT EXECUTION:

Module DFSBACK0 first looks for the optional SYSIN control card. If the card is present, the checkpoint number is saved and the input log is opened. The log is read searching for a type '41' record with a matching checkpoint ID produced by the PSB specified in the EXEC statement. When the user specifies a checkpoint ID it must be found on the log. Once located, backout continues to read the log tape searching for an '08' (scheduling) record for the same PSB. Should one be found, it constitutes an error because the user may backout only the last execution of a JOB. This search continues until EOF is reached.

If no control card was supplied, or if EOF is reached on the input log, the log is opened for read backward. All type '51' (before images) and all type '50' and '52' (after images), are passed to DFSRDBC0 who restores the Data Base to its original status.

Backout ceases when one of the following records is encountered under the following conditions.

TYPE	CONDITION
06	If coded as batch started.
07	If Dynamic backout did not fail. If it did, ignore the record.
08	Always.
37	If MODE=SNGL was specified in the TRANSACT macro for the transaction.
38	If MODE=SNGL was specified in the TRANSACT macro for the transaction.
41	Always with Program Isolation. Without PI, if a checkpoint ID was specified, the ID must match the '41' record.

When the data base is restored to its original condition, a type '50' log record is produced for each change. Because a type '51' log record is never produced the user can never backout a Backout execution. This log tape must be retained and must be input to Data Base Recovery should the need arise to recover a data set.

In the event that Backout is not successful, the user may re-execute Backout. The input log data set must be the same for both executions and both output log data sets must be saved.

DATA BASE ERRORS DURING BATCH BACKOUT.

If an I/O error occurs during backout, message DFS398I will be issued. When this occurs, the following procedure should be executed:

a) Execute Data Base Recovery using all log tapes, including the one created by the unsuccessful backout attempt.

b) Re-execute the backout job against the original PSB, using the same input tapes as the first attempt.

c) Accumulate all log tapes or keep them in creation sequence.

The above procedure recovers the data base up to the point of failure. It then completes the backout process. Those records which were already backed out during the initial execution will, in effect, be ignored and those which were not completed will be backed out.

DATA BASE ERRORS DURING EMERGENCY RESTART BACKOUT.

If an I/O error occurs during Emergency Restart backout, message DFS983 is issued and the data base and program are stopped. Note that even though the data base is stopped, if there is backout to be done for another active PST against the same data base, it will be backed out. The procedure for recovery is to take the damaged data base offline, execute the data base recovery, and then execute the Batch Backout utility to complete the data base backout. The operator must issue the /DBRECOVERY command, execute the two utilities, then restart the data base. It may be necessary to bring IMS/VS down momentarily (/CHKPT FREEZE) to reallocate the recovered data set if it is on a different volume.

I/O ERROR ON LOG TAPES

If the I/O error occurs on the input log the System Log Recovery Utility should be used to correct the error condition. The output log must be terminated normally and retained for input to Change accumulation or Data Base Recovery. Once the input error condition is corrected, re-executed the Backout.

If the I/O error occurs on the output log, terminate the output log correctly, and re-execute Backout. Keep both output log tapes for Change Accumulation or Recovery. Terminating the tape will involve obtaining an IMS dump, copying the tape, and executing the log terminator.

SYSTEM FAILURE.

Close the output log using the System Log Terminator. Re-execute Backout and keep both output log tapes for Change Accumulation or Recovery.

OUTPUT MESSAGES.

Execution of this utility can result in any of the following messages being issued. Most of the messages are self explanatory. For the remainder the reason for issuing the message is listed.

1. U963 ABEND

   a) Backout was unable to open the SYSIN data set or the DEVTYPE macro failed.

2. DFS395I BACKOUT COMPLETE FOR PSB name [TO CHKPT id.]

3. DFS396I SYSTEM ERROR DURING BACKOUT OF DATA BASE dbdname PROGRAM psbname.

   a) Unable to locate a DMB directory entry for the data base specified.

4. DFS397I BACKOUT UNABLE TO OPEN DATA BASE dbdname PROGRAM psbname.

5. DFS398I I/O ERROR DURING BACKOUT IN DATA BASE dbdname PROGRAM psbname.

6. DFS399I BUFFER POOL TOO SMALL FOR BACKOUT OF DATA BASE dbdname PROGRAM psbname.

   a) Increase the buffer pool in the EXEC card.

7. DFS400I BATCH BACKOUT UNABLE TO OPEN THE INPUT LOG TAPE.

8. DFS888I NO DATA BASE RECORDS READ FOR PSB.

   a) The log tape contained no '50' or '51' records for the data base.

9. DFS890I BLKSIZE NOT SUPPLIED FOR IMSLOGR DD CARD

10. DFS894I INVALID RECORD ENCOUNTERED ON INPUT LOG TAPE.

    a) A variable length record was read with a length field of zero.

11. DFS896I INCOMPLETE SPANNED RECORD ON INPUT LOG TAPE.

12. DFS957I NO CHKPT ID IN SYSIN.

a) Columns 1-6 of the control card do not contain 'CHKPT', or if columns 1-6 are correct, the remainder of the card is blank.

13. DFS958I CHKPT NOT FOUND ON LOG.

14. DFS959I CHKPT NOT WITHIN LAST SCHEDULE OF PGM.

15. DFS964I INSERT FAILED ON PRIOR UPDATE. RECOVERY REQUIRED FOR DATA BASE xxxxxxxx.

a) When inserting to a VSAM KSDS a '52' record is produced prior to the insert. After returning from VSAM a '50' is produced. In this case only the '52' is present on the log tape which means the insert failed at some point in time in the VSAM insert process. Conceivably this could occur during a CI split after the VSAM index is updated but prior to the KSDS being updated. The only sure recovery technique is Data Base Recovery.

Do not specify a checkpoint ID.

With Program Isolation in the system backout will cease when the first (last on tape) checkpoint record is encountered, therefore, the specification is meaningless.

Secondly, if multiple schedulings of the same BMP occur on a log tape there probably will be multiple checkpoint records on that tape which contain the same checkpoint ID. The first matching checkpoint ID establishes the checkpointed interval to be backed out. From that point, backout searches forward on the log tape. If a second scheduling record is encountered, DFS959 is issued and the backout terminated.

Example:

where:

S = BMP schedule.
1 = Checkpoint ID 1.
2 = Checkpoint ID 2.
T = Termination of first execution.
F = Point of failure.

The only area that legitimately may be backed out is from the failure (F) back to the last checkpoint (second 2).

If a checkpoint ID of 2 is supplied, the first 2 is located and saved while reading forward. When the second scheduling is encountered, DFS959 is issued and the job is terminated.

CHANGE ACCUMULATION (DFSUCUMO)

The purpose of the Change Accumulation utility is to produce one or two output data sets. The first of these data sets is an accumulation of type '24' and type '50' log records where only the latest changes are retained. In other words if two changes are made to the same data base segment with the same offset and length, only the latest change is retained on the accumulated data set. This data set is sorted by DBD name, data set, record type, record key (RBN) and date/time, and becomes input to succeeding Change Accumulation executions or to Data Base Recovery. Data Base Recovery is much faster with accumulated change input than with log only input since records are in sequence and changes are applied directly to the Image Copy prior to writing the records to disk.

The second output data set is a copy of the log type '24' and type '50' records as they appear on the input log data set. This data set cannot be used as input to the Data Base Backout utility because it does not contain the type '51' (before image) records.

USER SPECIFICATIONS AND OPTIONS.

1. All control statements are optional. The user can supply a //SYSIN DD DUMMY data set in which case the following defaults are taken.

   a) The maximum length sequence field is 10 bytes. If ISAM or VSAM KSDS records exist and the keys are larger than 10 bytes, message DFS371 will result.

   b) All input log records will be accumulated and written to the accumulated data set. No new log data set output is possible.

2. ID Control Statement.

   a) Required if ISAM or VSAM KSDS records exist whose keys exceed 10 bytes.

   b) Required if the number of DBO or DBl data bases exceeds 16 or the number of DDs exceeds 80. Exceeding either of these defaults results in DFS365.

3. DBO Control Statement.

   a) Specifies for which data bases and data sets the log records are to be accumulated. The user may accumulate records for all data bases or selected data bases and selected data sets within data bases.

4. DB1 Control Statement.

a) Specifies for which data bases and data sets a new log tape is to be created. The New Log data set may be used to select records for critical data bases. These data sets may then be input to individual change accumulation executions to obtain an accumulated change data set with only changes for a particular data base/data set. The new log data set may be used for log input to Data Base Recovery.

b) If OTHER is specified, all records not described by a DBO control card are written to the New Log data set. Output to accumulation and the log are mutually exclusive.

PURGE DATES AND TIMES.

The specification of purge dates and times is a user option and must be approached with care and understanding. This specification causes the Change Accumulation utility to drop all input records containing an earlier creation date/time. An incorrect specification could cause needed log records to be dropped or unneeded records to be retained. The unneeded records could, in some cases, cause Data Base Recovery to perform incorrectly thus destroying a data base.

The user is strongly advised to establish procedures which encompass the guidelines and rules as stated in the Recovery section of this chapter.

OPERATIONAL CONSIDERATIONS.

The Change Accumulation utility is run independently of IMS/VS, however, it requires a properly closed and terminated log tape in order to reach EOJ. When executing under MVS, the user may have to specify BLP to avoid enqueue contention on the data set name.

It is strongly recommended that log tapes be input in sequence. DBO output records are sorted therefore the input sequence is not important. DB1 output is not sorted therefore an out of sequence condition on input is carried over to the output. A log out of sequence condition will cause Data Base Recovery to fail.

A missing log tape will not be detected by either Change Accumulation or Recovery yet such a condition will result in data base destruction after recovery. Procedures should ensure that input log tapes are in sequence and all log tapes are accounted for.

OUTPUT DATASETS.

The accumulated output data set consists of a 44 byte header record followed by the data records for the first data set, then a header record followed by the data records for the second data set, etc.

The format of header record is:

HDB LEN	HDB SPACE	HDB CODE	HDB FLG	HDB LRECL	HDB ORG	HDB PURD	HDB PURT	HDB DDNAM	HDB NAM
2	2	1	1	2	1	3	4	8	8

HDB DSID	HDB DATE	HDB TIME	HDB SEQ	HDB ELKS
1	3	4	2	2

where:

HDBLEN Header length - always 44 bytes.
HDBSPACE Zeros
HDBCODE X'25' if X'50' record follows.
X'24' if X'24' record follows.
HDBFLG Data set organization.
'00' = OSAM HS organization.
'02' = OSAM HD organization.
'04' = ISAM
HDBLRECL DBD logical record length if HS organization.
HDBORG DBD organization.
'01' = HISAM single data set group.
'02' = HISAM multiple data set groups.
'03' = SSAM.
'04' = HSAM multiple segments.
'05' = HDAM.
'06' = HIDAM.
'07' = INDEX.
'09' = HISAM/VSAM single data set group.
'10' = GSAM.
'0B' = HISAM/VSAM single segment.
'0C' = INDEX/VSAM single data set group.
'0D' = HDAM/VSAM.
'0E' = HIDAM/VSAM.
'0F' = INDEX/VSAM.
HDBPURD Purge date specified by user.
HDBPURT Purge time specified by user.
HDBDDNAM DD Name.
HDBNAM DBD name.
HDBDSID Data set ID.
HDBDATE Run date (creation date).
HDBTIME Run time (creation time).
HDBSEQ Zeros.
HDBBLKS DBD block size.

The accumulation prccess merges all the log record changes to a logical record (identified by a key if ISAM/KSDS or a RBN/RBA if OSAM/ESDS) into one data record. If two changes were made to the same offset within a block, only the latest change is retained. The data record contains a count of the changes, the offset and length of each change, and the data for each change.

At Recovery time this record simply overlays the Image Copy logical record at the offset and for the length specified.

The format of the data record is:

CLEN GTH	CSP ACE	CCODE	CFLG	CIDLN	CDB NAME	CD SID	CDATE	CTIME
2	2	1	1	2	8	1	3	4

CSEQ	CCOUNT	CDATA ID	CDATA OL	...	CDATA
2	2	VAR	4		VAR

where:

CLENGTH Length of entire record.
CSPACE Zeros.
CCODE X'50' for db changes - X'24' if I/O error record.
CFLG Data set organization.
'00' = OSAM HS organization
'02' = OSAM HD organization.
'04' = ISAM
CIDLN Length of key or RBN field.
CDBNAME DBD name.
CDSID Data set ID.
CDATE Date from log record.
CTIME Time from log record.
CSEQ Sequence field from log record.
CCOUNT Number of elements in CDATA (number of changes).
CDATAID ISAM key or 4 byte OSAM RBN.
CDATAOL 2 byte offset and 2 byte length per data element. (repeated for each element when CCOUNT is greater than one).
CDATA Data elements (changes).

All data sets are read and written using QSAM.

OUTPUT MESSAGES.

Execution of this utility can result in any of the following messages being issued. Most of the messages are self explanatory. For the remainder the reason for issuing the message is listed.

1. DFS301A UNABLE TO OPEN DDNAME xxxxxxxx.

2. DFS305A DBD LIBRARY DOES NOT CONTAIN DBD xxxxxxxx..

3. DFS306A DBD xxxxxxxx DOES NOT CONTAIN DDNAME xxxxxxxx FOR FUNCTION CA.

4. DFS310A INPUT RECORD IMAGE FOR ID/DBx CARD IS IN ERROR AND FOLLOWS THIS MESSAGE.

   a) Columns 1-3 do not contain 'ID' or 'DBx', the number of data bases or data sets or key length is not within the limits, a date is invalid, etc.

5. DFS356A HEADER RECORD NOT FOUND ON DDNAME xxxxxxxx FOR FUNCTION CA.

   a) The DBD name in the change accumulation record does not match the change accumulation header.

6. DFS363A *ALL/OTHER SPECIFIED MORE THAN ONCE FOR FUNCTION CA.

7. DFS364W DDNAME xxxxxxxx SPECIFIED MORE THAN ONCE.

8. DFS365A DDNAME/DBNAME TABLE HAS OVERFLOWED.

9. DFS368I ERLG RECORD ENCOUNTERED FOR DATABASE xxxxxxxx DATASET xxxxxxxx ON VOLUME serial FOR FUNCTION CA.

   a) A type '24' log record was found on the input file. No operations should be attempted against the data base prior to a Track or Full Recovery of the data set.

10. DFS371A A RECORD KEY LENGTH nnn IS LONGER THAN SPECIFIED FOR FUNCTION CA.

11. DFS372A DBD xxxxxxxx CONTAINS A BLANK DDNAME.

12. DFS373A DDNAME DFSUCUMO IS NOT A VALID OLD CHANGE ACCUMULATION DATA SET.

13. DFS374W WARNING - NO CHANGE RECORDS FOUND ON LOG(S)

14. DFS376A AN APPARENT SORT SEQUENCE ERROR HAS BEEN DETECTED.

15. DFS383A INPUT RECORDS ARE NOT IN SEQUENCE FROM DDNAME xxxxxxxx FOR FUNCTION CA.

16. DFS390A DDNAME xxxxxxxx HAS AN INVALID LOGICAL RECORD FOR FUNCTION CA.

17. DFS391I (CONTROL CARD IMAGE).

    a) Will follow a previous error message.

18. DFS896I INCOMPLETE SPANNED RECORD ON INPUT LOG TAPE.

19. DFS339I FUNCTION CA HAS COMPLETED NORMALLY/ ABNORMALLY RC-nn.

DATA BASE RECOVERY (DFSURDBO)

The Data Base Recovery utility program is designed to restore a physically damaged data set within an IMS/VS data base. Only one data set may be recovered in any one execution of the utility. Data sets from all IMS/VS access methods except GSAM and HSAM are recoverable.

The Data Base Recovery utility is executed in a special IMS/VS batch region. This allows the utility to be run independently of the IMS/VS system, however, there may be restrictions imposed by the Operating system. The SVS user may execute the recovery utility then re-name the data set. MVS does not allow the re-name facility. If the data set organization is VSAM and the user specified VSAM share options 1 or 2, recovery may be performed in the following manner. First, issue the /DBRECOVERY command which closes the data base. Second, execute the Recovery utility which then becomes the update user of the data set. When Recovery completes the data bases are closed. Third, a /START DATABASE command will then allow IMS/VS to again open the data sets as the update user. When using this technique the user is responsible for ensuring that no other user in the system opens the closed data set with update intent between points one and three.

With the Dynamic De-allocation capability which was announced with IMS/VS 1.1.1 PTF 3 (for MVS), the user will now de-allocate the data bases when he issues the /DBRECOVERY command. Should the IMS Control region fail during the time Recovery is being performed an Emergency Restart can not be accomplished until Recovery finishes. To circumvent this possibility it is suggested the user perform the following:

a) /DBRECOVERY DATABASE

b) RENAME DATABASE

c) Execute Recovery

d) RENAME DATABASE

e) /START DATABASE

RECOVERY basically consists of two phases when recovering a complete data set or optionally allows the user to recover one or more tracks of a data set (see Track Recovery). The first phase of a full recovery consists of reading an Image Copy record, merging any Change Accumulation records, and writing each logical record to the data set. For HISAM data bases the image copy input may be the output of the HISAM REORGANIZATION UNLOAD utility. If the latter is used the only log records permissable as input are those created after the data base was reloaded. The user must control this and cannot depend upon the date and times

stored within the Unload, Change Accumulation, and Log records. For example:

a) On day 1 the user executes HISAM UNLOAD.

b) On day 2 he updates the data base.

c) On day 3 Recovery is attempted.

In this case the log records created on day 2 are acceptable to recovery (date wise) because the unload occurred earlier. When recovery reloads the data set the records will be placed into a different physical location thereby invalidating the log records and destroying the recovered data set. If this form of Image Copy is used the user should immediately follow the unload with a reload.

The second phase consists of randomly updating the partially restored data set with log input. The log input data sets may be concatenated and they must be in date/time sequence.

The Image Copy and Change Accumulation input files each contain a creation date/ time in their header records and each log record contains a creation date/time. Change Accumulation contains a PURGE date/time if the user specified a purge date/time during Change Accumulation execution, or zeros in the field if no purge date/time was specified. These date/times are used as follows:

a) Image Copy and Change Accumulation creation date/times are matched when Change Accumulation is input.

   1. If Image Copy is high DFS326 is issued and the Change Accumulation input is ignored since any records present would also be on the Image Copy.

   2. If Image Copy is equal to or lower go to b).

b) Image Copy creation date/time is matched to Change Accumulation purge date/time.

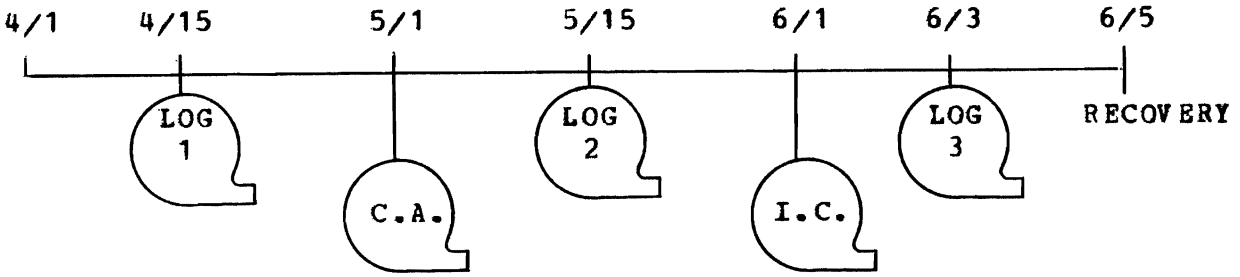
   1. If Image Copy is low DFS325 is issued since potential log records produced between those times would have been purged (dropped) and data base integrity can not be guaranteed.

c) The creation date/time in the first log record (from the log input, not those on the change accumulation input) is matched to the Image Copy creation date/time if no Change Accumulation was present, or to the Change Accumulation purge date/time. If the log date/time is low the log records are ignored until a record containing an equal or higher condition is encountered. At this point each log record is compared to the following log record. If the

following one is low, message DFS330 is issued indicating the log records are not in sequence.

The following examples show the effect of the date/time comparisons.

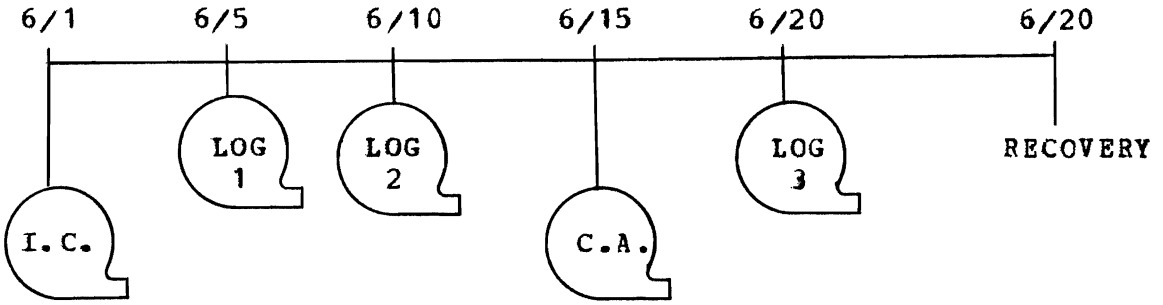
CASE 1.



Recovery input = C.A. and I.C., LOG 2 and LOG 3.

Since the Image Copy creation date is later than the Change Accumulation creation date, message DFS326 is issued and the whole Change Accumulation tape is ignored. Since the Change Accumulation was ignored, LOG 2 records are matched to the Image Copy and ignored because their creation date is earlier. LOG 3 records will be used.

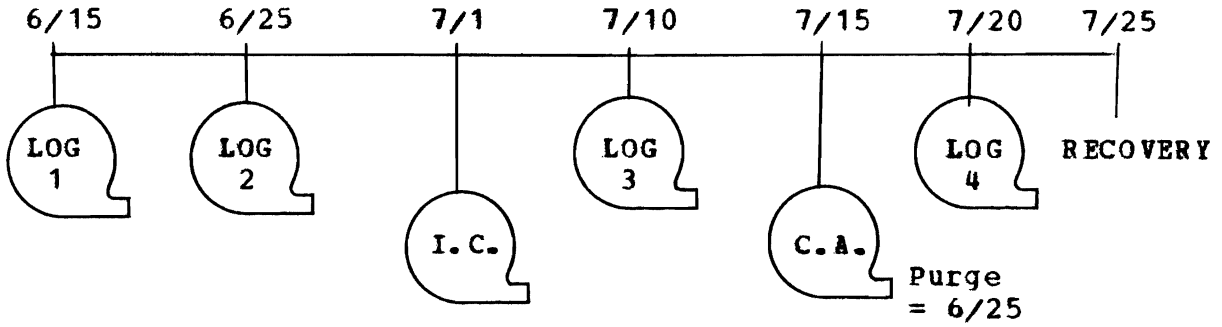
CASE 2.



Recovery input = I.C., C.A., and LOG 3.

This is the normal or expected input. If the user were to input LOG 1, LOG 2 or both, Recovery will accept the tapes, apply the records, and Recovery will be successful. The user should be aware that so doing causes much unnecessary processing. The first phase would merge the Image Copy and Change Accumulation logical records and create the partially restored data set. The second phase would read each log record and randomly update the data set. Since the log records on LOG 1 and LOG 2 were also on the Change Accumulation this is needless double updating. Only LOG 3 should be input.

CASE 3.

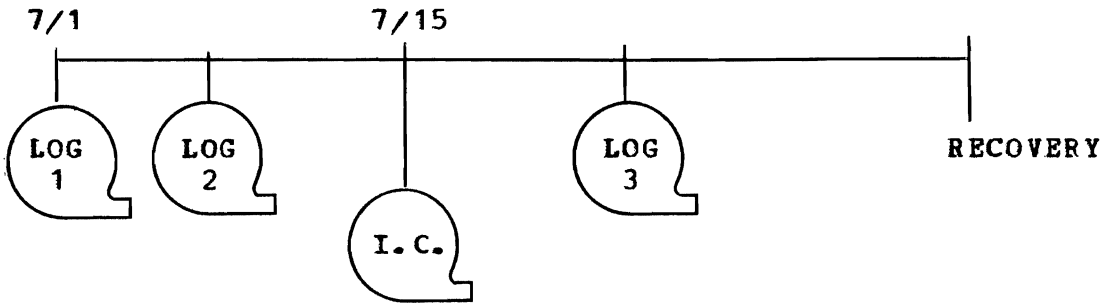


Recovery input = I.C., C.A., and LOG 4.

The Change Accumulation tape contains records only from LOG 2 and LOG 3 because of the 6/25 purge date. The first phase will merge these with the Image Copy which means some extra unnecessary processing is involved in handling records from LOG 2 which also exists on the Change Accumulation tape. The purge date should have been 7/1.

WARNING: If a date base reorganization had occurred between 6/25 and 7/1 the data base would be destroyed by Recovery because the records on LOG 2 would reflect the old physical location which changed with the reorganization.

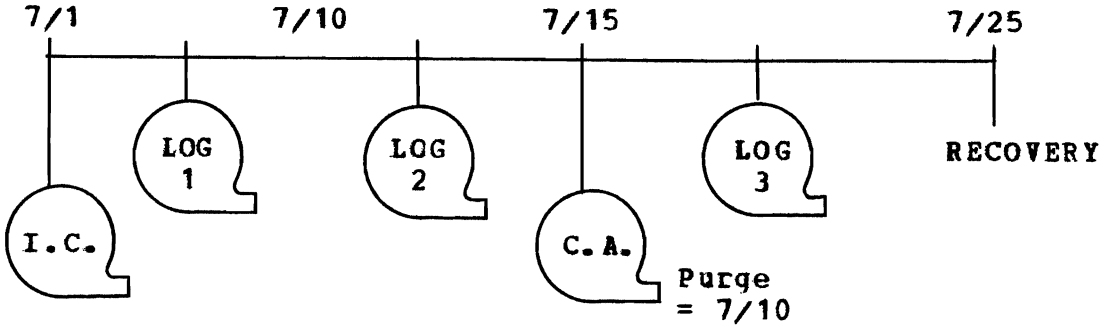
CASE 4.



Recovery input = I.C., LOG 1, LOG 2, LOG 3.

Because there was no Change Accumulation input, each log record is matched to the Image Copy creation date until an equal or later date is encountered. In this example LOG 1 and LOG 2 will be ignored, however it does result in needless processing.

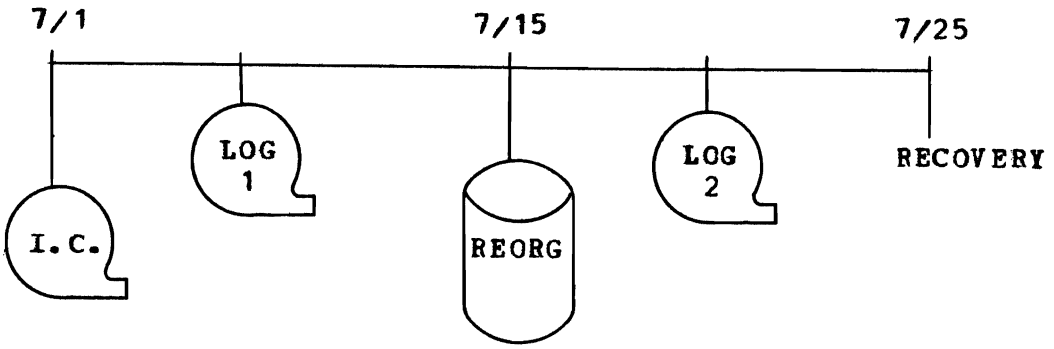
CASE 5.



Recovery input = I.C., C.A., and LOG 3.

Since the Change Accumulation purge date (7/10) is higher than the Image Copy creation date, message DFS325 is issued indicating potential missing log records, which are missing in this example. The purge would drop all records from LOG 1 and recovery can not complete successfully.

CASE 6.



This example shows a situation where recovery is very difficult at the least. If the user inputs the Image Copy and both log tapes it might appear to recover, however, the data base is not usable. LOG 2 reflects the new physical location of the data base segments while the Image Copy and LOG 1 reflect the old location.

The user must recover to the beginning of the Reorganization by executing Recovery using the Image Copy and LOG 1. Reorganization must then be re-executed. Recovery must then be executed using only LOG 2 as input. If an Image Copy were taken immediately following the Reorganization, Recovery would have been possible by using that copy and LOG 2.

From the above examples some basic rules can be developed and the user should ensure that his operating procedures adhere to these rules.

a) Always Image Copy the entire data base immediately after a reorganization or a data base load (see case 6). This establishes a starting point should recovery be necessary.

b) Never supply a log tape to Recovery containing log records whose date/time is earlier than the Image Copy creation date/time if possible.

   1) Some users create Image Copies while a Log tape is being created via the /DBDUMP command. If at all possible a Change Accumulation should be executed specifying a PURGE to remove the earlier records.

c) The specified PURGE date/time should never be later than the last Image Copy creation date/time.

   1) If the PURGE date/time is later it does not necessarily mean recovery will be unsuccessful, however, the possibility exists and it should be treated as an integrity exposure (see case 5).

The user should be especially cautioned to understand the function and limitations of Recovery if he is using some form of backup other than Image Copy or HISAM Unload. A user copy of the data set or HD Reorganization Reload input may be used to recover a data set by first restoring the data set and using log only input to the Data Base Recovery program. Change Accumulation input can be used only with Image Copy or HISAM Unload data sets. The time required tc process individual log records and tapes will usually be considerably longer than processing the Change Accumulation output. The time saved in doing a DASDR vs Image Copy might well be lost during a Recovery cycle.

EXECUTION OF DATA BASE RECOVERY.

The JCL EXEC card must contain a DBD name. This DBD is used to establish the DL/I environment when calls are executed to the buffer handler and must be the same DBD name supplied in the control card.

Recovery will accept two control cards. If the first control card contains ABEND in columns 1 through 5, a switch is turned on. If a serious error occurs during execution the utility will ABEND at the point of failure. Without this card the utility will continue to process (after issuing a DFSxxx message) and will perform those functions which are error free. There are some ABENDS which are not optional. For example, an I/O error on a data base will immediately ABEND.

The DBD name specified in columns 4 through 11 of the control card is used to obtain the data set characteristics such as access method, blocksize, logical record size, etc. A BLDL is issued and the information extracted from the DBD.

The Image Copy, the Change Accumulation, and the Log input data sets are read using QSAM. During phase 1, ISAM and OSAM data sets are written using QISAM and QSAM respectively, while VSAM data sets are written using the buffer handler. During phase 2 when applying the log records, all data bases are read and written using the buffer handler.

TRACK RECOVERY.

Track recovery is a feature of IMS/VS 1.1.1 and later versions which allows the recovery of one or more tracks of a data set. This feature applies to VSAM only.

When an I/O error is detected by the buffer handler a type '24' log record is produced. This log record contains such information as DBD name, data set DD name, program name, transaction name, the RBA of the error CI, the CCHHR of the physical error record, type of error (read or write), date and time, etc.

To perform track recovery all log tapes must be input to a Change Accumulation execution. The Change Accumulation sort will place the '24' records in front of any '50' log records for the data set. The user must also place a 'T' in column 2 of the control card. Absence of the 'T' implies full recovery.

If alternate tracks are to be assigned IEHATLAS is invoked to perform the function. The user can force alternate track assignment by placing a 'M' in column 3 of the control card or alternate tracks will be assigned should an I/O error occur during the track recovery process.

When track recovery is requested the type '24' log records are saved in core in an area obtained with a GETMAIN. The records are sorted via chains by DBD name, DD name, and RBA.

Every CI on the error track is then read into an area obtained with a GETMAIN. Every CI is then rewritten to the data set. Any read or write errors or a 'M' in the control card results in alternate tracks being assigned via IEHATLAS.

The Image Copy and Change Accumulation records are then read and matched to the error tracks. Only matching records are used to recover the tracks. The error tracks are retrieved and overlayed with the matching records. If log tapes are input to track recovery they are ignored.

Track recovery is usually faster than a full data set recovery therefore users are encouraged to investigate this facility.

OUTPUT MESSAGES.

Execution of this utility can result in any of the following messages being issued. Most of the messages are self explanatory. For the remainder the reason for issuing the message is listed.

1. DFS301A UNABLE TO OPEN DDNAME xxxxxxxx.

2. DFS302A INVALID OR MISSING FUNCTION RV SPECIFIED ON REQUEST xxx.

   a) Control card column 1 does not contain an 'S'.

3. DFS304A DBD NAME NOT SPECIFIED FOR FUNCTION RV ON REQUEST xxx.

   a) Control card columns 4-11 are blanks.

4. DFS305A DBD LIBRARY DOES NOT CONTAIN DBD xxxxxxxx.

5. DFS306A DBD xxxxxxxx DOES NOT CONTAIN DDNAME xxxxxxxx FOR FUNCTION RV.

6. DFS307A NO DDNAME SPECIFIED FOR DATA BASE xxxxxx FOR FUNCTION RV.

   a) Control card columns 13-20 are blanks.

7. DFS312A DDNAME xxxxxxxx DOES NOT CONTAIN A VALID IMAGE OR REORGANIZED COPY FOR FUNCTION RV.

   a) The Image Copy or HISAM Unload header record does not contain 'D' or 'R' in byte 2.

8. DFS313A DDNAME xxxxxxxx DOES NOT APPLY TO DATABASE xxxxxxxx FOR FUNCTION RV.

   a) The control card DBD does not match the Image Copy header DBD.

9. DFS314A CHANGE ACCUMULATION SUPPLIED BUT WITHOUT IMAGE COPY FOR FUNCTION RV.

10. DFS315A DEVTYPE MACRO FAILED ON DDNAME xxxxxxxx FOR FUNCTION RV.

11. DFS316A DBD xxxxxxxx CONTAINS AN UNKNOWN ORGANIZATION CODE nn.

12. DFS317A DDNAME xxxxxxxx NOT FOUND ON DDNAME xxxxxxxx HEADER RECORD FOR FUNCTION RV.

a) The control card DD name does not match the Image Copy header record DD name.

13. DFS319A DDNAME xxxxxxxx HAD A PERMANENT I/O ERROR FOR FUNCTION RV.

a) The buffer handler received a PSTRTCDE of '08' (Permanent read error).

14. DFS322W FUNCTION RV WAS NOT SUPPLIED AN IMAGE COPY INPUT.

15. DFS323W FUNCTION RV WAS NOT SUPPLIED A CHANGE ACCUMULATION INPUT.

16. DFS324W FUNCTION RV WAS NOT SUPPLIED AN INPUT LOG FILE.

17. DFS325A PURGE DATE ON DDNAME xxxxxxxx IS LATER THAN DATE ON DDNAME xxxxxxxx FOR FUNCTION RV.

a) Image Copy date or time is lower than Change Accumulation PURGE date or time.

18. DFS326A DDNAME xxxxxxxx INPUT IGNORED DUE TO PURGE DATE OR DDNAME xxxxxxxx.

a) Image Copy date or time is higher than Change Accumulation creation date or time.

19. DFS327I NO RECORDS ON DDNAME xxxxxxxx FOR RECOVERED DATA SET.

a) No log records existed on the log data set or no Change Accumulation records existed for the data set being recovered.

20. DFS329A CHANGE ACCUMULATION HEADER ON DDNAME xxxxxxxx IS NOT CONSISTENT WITH DBD xxxxxxxx FOR FUNCTION RV.

a) Change Accumulation header data set organization code does not match the DBD data base organization code.

21. DFS330A DDNAME xxxxxxxx IS OUT OF SEQUENCE FOR FUNCTION RV REASON = x.

a) Reason = 1. Image Copy RBN is higher than Change Accumulation RBN.

b) Reason = 2. Image Copy key is higher than Change Accumulation key.

c) Reason = 3. A log record contains a date/time which is lower than a previously read log record.

22. DFS332A OSAM (ESDS) RECORD FROM DDNAME xxxxxxxx HAS RBN nnnnnnnn AND IS BEYOND THE CURRENT END OF THE DATA SET.

a) The Image Copy or Change Accumulation RBN is greater than the current EOF + 1 block, which indicates missing blocks on the Image Copy input data set.

b) The buffer handler received a PSTRTCDE of '04' (RBN beyond data set) when applying the log record to the set. This would indicate missing log records.

23. DFS333A KEY ID ON DDNAME xxxxxxxx DOES NOT EXIST IN DDNAME xxxxxxxx DATA SET.

a) A Log, a Change Accumulation, or an Image Copy record contains a key which cannot be found in the data set. Change Accumulation and Image Copy errors only occur during Track Recovery.

24. DFS337A BUFFER POOL TOO SMALL FOR FUNCTION RV.

25. DFS338A DATA SET DDNAME xxxxxxxx IS TOO SMALL.

26. DFS339I FUNCTION RV HAS COMPLETED NORMALLY (ABNORMALLY) RC=00(16).

27. DFS341A CATASTROPHIC ERROR OCCURED IN DL/I DURING FUNCTION RV.

a) An attempt was made to find a record in the data set in order to apply the log tape changes. The attempt was unsuccessful because of a DL/I failure.

28. DFS356A HEADER RECORD NOT FOUND ON DDNAME xxxxxxxx FOR FUNCTION RV.

a) The Change Accumulation header record DBD name does not match the DBD name in the Change Accumulation record.

29. DFS391I DATABASE DATA SET RECOVERY.

a) Heading of output report.

OTHER BACKUP/RESTORE UTILITIES

The following chart lists the various utilities available for copying and restoring data sets. The user is free to use any utility where yes is indicated for that type of data set.

If the user elects to copy the data sets with other than IMAGE Copy, when recovery is to be performed, he must reload the data set and apply log only input to the IMS/VS RECOVERY module. If the log tape volume is considerable this will negate any performance gains made during the copy phase. Also, this method prevents usage of the track Recovery Facility.

Whatever the method chosen for backup it is strongly recommended that the data bases be copied immediately following an initial load, a reorganization, or a recovery operation.

DATA SET	IMS REL	EXPORT IMPORT	IEHD-ASDR	IMAGE COPY	RECOVERY	TRACK RECOVERY
INDEX SET	1.0.1 1.1 +	NO NO	YES(7) YES(7)	NO YES(1)	NO NO	NO YES(6)
KSDS	1.0.1 1.1.+	YES(2,3) YES(2,3)	YES(7) YES(7)	YES(2) YES(2)	YES(3) YES(3)	NO YES(6)
ESDS	1.0.1 1.1+	YES(4,5) YES(4,5)	YES(7) YES(7)	YES(4) YES(4)	YES(5) YES(5)	NO YES(6)
ISAM	1.0.1 1.1+	NO NO	YES(4,5) YES(4,5)	YES(4) YES(4)	YES(5) YES(5)	NO NO
OSAM	1.0.1 1.1+	NO NO	YES(4,5) YES(4,5)	YES(4) YES(4)	YES(5) YES(5)	NO NO

(1) User must code 'I' in DUMP Control card. The DUMP is usable only for Track Recovery.

(2) Dumps in logical sequence by key.

(3) Restores in physical = logical sequence (physical rearrangement).

(4) Dumps LRECLS in physical sequence.

(5) Restores LRECLS in same physical sequence (No change to physical locations).

(6) Uses Image Copy and Change Accumulation records for error tracks only.

(7) Care must be exercised to insure data set and catalog compatibility.

SYSTEM LOG TERMINATOR (DFSFLOTO)

DFSFLOTO is a IMS/VS utility program which can be used to complete and close the IMS log data set. This utility is necessary only when the system failure is such that the IMS STAE routines do not gain control. Hardware failures, operating system ABENDS or power failures where main memory is not lost are examples of these system failures.

DFSFLOTO is executed as a standard job under all operating systems for which IMS/VS is supported. A storage dump taken at the time of failure is required. This dump may be either a SYS1.DUMP or a Standalone dump.

DFSFLOTO EXECUTION:

The program first issues a DEVTYPE macro to determine whether the dump resides on tape or a disk device. The dump data set is opened, the header record is read and saved.

The log tape is then opened and each record is read until the type '42' record is located. Any log record which contains a bad block size or bad logical record size results in message DFS943 and a U0005 ABEND. The '42' record is saved and the log tape is closed.

Should a '42' not be found or if an I/O error occurs prior to finding the '42' record, message DFS645 is issued and the user must reply with the buffer address. These buffer addresses may be obtained from the DFS810 messages which are issued at IMS start up time. The entered addresses are converted and stored as if they were obtained from the '42' record. (1.)

In a V=V system the real addresses are sorted into ascending sequence. In a V=R system the buffer area is continueous.

The dump data set is then read until a record is located which matches the first or only real address. This dump record is saved along with any following records necessary to cover the size of the log buffers.

The log tape is then opened for input/output and the DCB address is validated. The first 4 bytes of the log work area in the dump must contain a pointer to the log DCB within the log work area. (2.)

The log DCB is then checked to determine whether or not it was open at failure time. If the log was closed prior to failure there is no need to execute DFSFLOTO.

From the dump record each buffer is located using the '42' or input addresses. The buffer address is saved and the sequence number of the first log record in the buffer is saved. These entries are then sorted by the log sequence number. (3.)

The log tape is then read and the last log record within the block is located. A value of one is added to the sequence number and this is compared to the log sequence numbers in the dumped buffers. A low condition causes the next tape record to be read and compared. A high condition causes the next buffer to be compared. An equal condition indicates the tape is positioned. The remaining buffers from the dump are then written to the log tape. (4.)

A type '06' log record is then written and log tape is closed. At this point message DFS941 is issued.

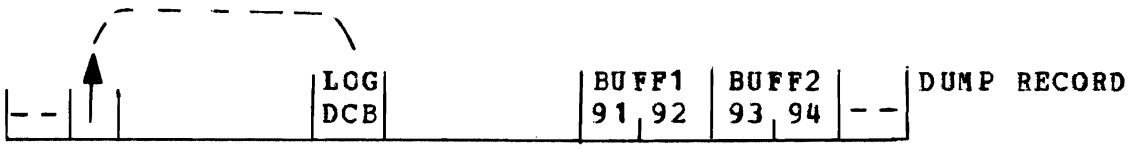
EXAMPLE

1.

BUFF1 ADDR	BUFF2 ADDR

'42' RECORD OR ENTERED ADDR.

2.



3.

BUFF1	91
BUFF2	93

4.

EXISTING LOG RECORDS

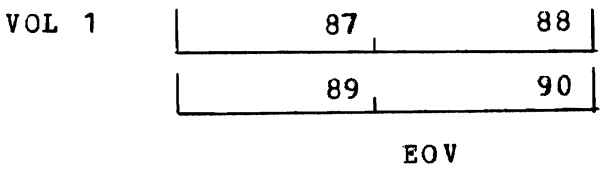
87	88
89	90
91	92

88 + 1 VS 91 = READ NEXT RECORD

90 + 1 VS 91 = TAPE POSITIONED

The two buffers containing log sequence numbers 91 and 92, 93 and 94 are written to the tape. The first buffer will overlay the log record containing sequence numbers 91 and 92. The second buffer will add one block to the log tape.

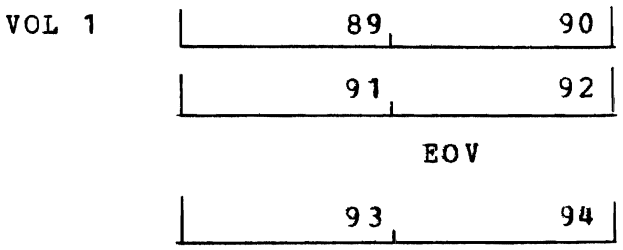
Suppose the log data set had been created on two volumes containing the following records:



92 + 1 VS 91 = GET NEXT BUFFER
92 + 1 VS 93 = TAPE POSITIONED

Since only volume two is input, the first record read is higher than the first (lowest sequence number) buffer. In this case buffer 1 is disregarded and buffer 2 is tested.

Suppose the log data set had contained the following format:



94 + 1 VS 91 = GET NEXT BUFFER
94 + 1 VS 93 = GET NEXT BUFFER

In this example all the buffers had been written prior to the failure. Since both buffers are lower than the first record read, a backspace tape is issued. The sequence number of the last buffer (93) is compared to the first sequence number in the log tape block (93). An equal condition indicates the tape is positioned and the last buffer is then written.

Any failure to reposition the tape using these methods results in message DFS942 with a return code of 24, followed by a U0005 ABEND. An EOF will never be written to the log tape unless the tape was repositioned and the buffers were flushed.

OUTPUT MESSAGES.

1. DFS941 SYSTEM LOG CLOSED

   A. DFSFLOTO execution was successful. This is the normal end of job message.

2. DFS942 SYSTEM LOG NOT CLOSED - 08

   A. The first four bytes of the IMS Log Work Area contains a pointer to the LOG DCB. This pointer is validated in the record from the Dump data set to insure it points to the DCB. If it does not, it is assumed the dumped Log Work Area is invalid which includes the buffers.

   B. Before or during the failure of the system the core location of the Log Work Area was altered, or the wrong dump data set was used as input.

3. DFS942 SYSTEM LOG NOT CLOSED - 16

   A. The log DCB was already closed at the time the dump was taken. This is not an error but indicates that DFSFLOTO is unnecessary.

4. DFS942 SYSTEM LOG NOT CLOSED - 20

   A. The buffer number in the Log Work Area is a negative value, or an EOF was reached on the Dump data set prior to locating the dump record containing the Log Work Area.

5. DFS942 SYSTEM LOG NOT CLOSED - 24

   A. When reading the log tape, a log record containing a sequence number one less than the first record in the dumped log buffers can not be located.

6. DFS943I ERROR ON LOG TAPE READ/WRITE ERROR

   A. An I/O error occured.

   B. The log record read contained an invalid block size or invalid record length.

   C. An EOF occured on the log tape while trying to locate a record with a sequence number one less than the buffer.

   D. The user failed to specify 'BUF=' when responding to a DFS645 message.

7. DFS645I ENTER BUFFER ADDRESS(ES) IN HEX

   A. A '42' record could not be located on the input log data set. The operator must supply the buffer address. The address(es) may be obtained from the DFS810 message produced at IMS/VS start up time and should be entered as BUF=xxxxxx.

LOG RECOVERY UTILITY (DFSULTRO)

This utilitiy enables the user to produce a valid log data set from either an unclosed data set, a data set which contains erroneous data, or results in I/O errors when reading.

The utility probably functions best when the user has dual logs and reads both data sets during the DUP phase. During this phase the primary log data set is read and copied to an interim data set until an I/O error occurs, a block length or record size error is found, or an out of sequence condition occurs within the log records. When this happens, the secondary log data set is read until the corresponding block is found. This block is tested and if found to be good, the secondary block is copied to the interim log, and reading continues on the secondary log data set until another error condition is detected. A second error will cause the primary to be positioned to the corresponding block and the primary data set will be copied. In this flipflop fashion a new log data set is created using the good blocks from one or the other of the input logs.

In the event the same block is in error on both log data sets, the user receives an error message, both blocks are copied to the interim data set, and both blocks are printed. The user may then correct either of the blocks on the interim data set using the REP mode of operation.

If only a single log data set is input, the utility reads and copies this to an interim data set. I/O errors, block or record length errors, or incomplete spanned records cause the block to be printed following the appropriate error message. The user must then correct the block using the REP mode of operation.

The REP phase reads the input control cards, locates the specified block and replaces the erroneous data within the block. The resulting output is a good data set.

EXECUTION OF LOG RECOVERY.

DUP PHASE

1. Each record is read from the input log data set. At the end of file all data sets are closed and the job terminated.

   A. Dual input - I/O errors. An attempt is made to position and use the alternate tape.

   B. Single input - I/O errors.

      1. The block length is validated and replaced with the DCB blocksize if in error.

      2. An error ID record is written to the interim data set(s).

      3. Error message 1 is printed.

      4. The input record is written to the interim data set(s) and a return is made to read the next record.

2. The sequence of the log records is validated. The sequence number of the first record in the current block must be exactly one higher than the sequence number of the last record in the previously read block. If it is not:

   A. Dual input - an attempt is made to position and use the alternate tape.

   B. Single input.

      1. An error ID record is written to the interim data set(s)

      2. Error message 2 or 3 is printed.

3. Spanned records are validated. If the previous block contained the first portion of a spanned record, the current block should contain the last portion. If not:

   A. The interim data set(s) is backspaced.

   B. An error ID record is written to the interim data set(s).

   C. The previous block is re-written to the interim data set(s). This insures that an error ID record proceeds the bad data record.

D. Error message 4 is printed.

4. The block length is validated and each log record is inspected for a valid length. If errors exist:

   A. Dual input - an attempt is made to position and use the alternate tape.

   B. Single input.

      1. The blocksize is replaced with the DCB blocksize.

      2. An error ID record is written to the interim data set(s).

      3. Error message 5 is printed.

5. The sequence number of the last log record within the block is saved and the block is written to the interim data set(s). A return is made to read the next record.

POSITIONING THE ALTERNATE TAPE

1. The currently used data set is closed and the alternate data set is opened.

2. The date and time from the header label of the alternate is compared to the currently used data set label.

   A. If the alternate is higher, the volume is closed, the previous volume is opened, and the date/time test is repeated.

   B. If the alternate is lower, the volume is closed, the next volume is opened, and the date/time test is repeated.

   C. An equal condition indicates matching volumes.

3. The alternate data set is read.

   A. Any I/O errors, block length errors or record size errors are ignored until a record is read which matches the bad record on the primary log data set.

   B. The block length and the record size of this record is validated. Errors result in a return code being passed which will generate a DFS45x message and cause job termination.

   C. The DD names are switched so that the alternate data set now becomes the primary.

D. If the desired block can be located, a return is made to begin checking the sequence number. If not:

1. An error ID record is written to the interim data set(s).

2. The error record from the primary log data set is copied to the interim data set(s).

3. Error message 6 is printed.

4. An error ID record is written to the interim data set(s).

5. Error message 7 is printed. This message will contain the same sequence number as number 6 however the sequence prefix will be 'B'.

6. The error record from the alternate log data set is copied to the interim data set.

7. A return is made to begin checking the sequence number.

THE INTERIM LOG DATA SET

This data set is a copy of the log data set with inbeded error ID records. Any blocks which contain errors are preceeded by a 32 byte error ID block with the following format.

BLKLEN		RECLEN			SEQNO
32	00	28	00	DFSULG10 ERROR ID =	(A) (B) XXXXX
2	2	2	2	24	

The sequence number begins with 00001 and is incremented by one for each error ID block written. This sequence number must match a printed error message which identifies the error condition.

Two copies of the interim data set may be obtained by including the NEWRDER2 DD card in the JCL.

EXECUTION OF LOG RECOVERY.

REP PHASE

1. The first control card is read and the contents verified. Any errors result in message DFS452 and job termination.

2. The interim log data set is then read. If the record is not an error ID record, the record is copied to the output log data set and the next record is read.

3. When an error ID record is encountered it is matched to the card using the error sequence number beginning in column 9. Unmatched ID records result in message DFS456 and job termination, therefore, a card must exist for every error ID record produced and the cards must be in the sequence of the error ID records.

4. When the matching error ID record is located, the next record is read from the interim tape. If the card operation specified 'CLOSE' the job is terminated which will close the new log data set. This means 'CLOSE' always closes just prior to the record which was found to be in error during the DUP phase.

   When 'CLOSE TRUNC' is specified the last data record preceeding the error ID record is inspected for a spanned record. If a spanned record exists, it is dropped by shortening the blocksize and message 9 is printed.

5. If the card operation specified 'SKIP', the next control card is read, the next interim tape record is read (thus dropping the error record), and the matching is continued.

6. If the card operation specified 'POS=', the error record is overlayed with the card data and the next card is read. If it contains the same sequence number the process is repeated. When a higher card sequence number is read, message 8 is printed indicating the record data was replaced. The updated record is then written, the next input record is read from the interim data set and the matching process continues.

CLOSING AN UNCLOSED LOG TAPE.

The user should DUP the unclosed log tape and look for I/O errors, a bad block size, or a sequence error where the following log tape record contains an earlier date/time.

Execute the REP phase including any corrections for earlier errors (if any were found) and supply a 'CLOSE TRUNC' card specifying the sequence number from the message printed when the end of the tape was reached.

OUTPUT MESSAGES.

The following messages are generated by the SYSTEM LOG RECOVERY utility. Messages 1 through 9 will be followed by a printed image of the error block. Message 10 will be followed by a print of the control card image. Messages 12 through 25 will be written to the console.

1. I/O ERROR ON IEFRDER BLOCK #xxxxxx-ID=Axxxxxx LENGTH CHANGED FROM 'xxxx'*** (if BLKLEN error)

A. A SYNAD exit was taken due to an I/O error on the input log tape when only one log tape was input.

2. RECORD IN ERROR ON IEFRDER BLOCK #xxxxxx-ID=Axxxxx *SEQ ERROR xxxx TO xxxx

A. The sequence number of the first log record in this block is not one higher than the sequence number of the last log record in the previous block. The sequence numbers found are included in the message.

3. RECORD IN ERROR ON IEFRDER BLOCK #xxxxxx-ID=Axxxxx *SEQ ERROR I/O ERROR TO xxxx

A. The sequence number of the first log record in this block is not one higher than the sequence number of the last log record in the previous block because the the previous block had an I/O error.

4. RECORD IN ERROR ON IEFRDER BLOCK #xxxxxx-ID=Axxxxx INCOMPLETE LAST RECORD

A. The last log record in this block is coded as the first segment of a spanned record. The first log record in the next block is not coded as the second segment of a spanned record.

5. RECORD IN ERROR ON IEFRDER BLOCK #xxxxxx-ID=Axxxxx LENGTH CHANGED FROM 'xxxx'*** (If BLKLEN error)

A. A block length or record length error was detected.

6. RECORD IN ERROR ON IEFRDER BLOCK #xxxxxx-ID=Axxxxx LENGTH CHANGED FROM 'xxxx'*** (If BLKLEN error)

7. RECORD IN ERROR ON IEFRDER BLOCK #xxxxxx-ID=Bxxxxx LENGTH CHANGED FROM 'xxxx'*** (If BLKLEN error)

A. A SYNAD exit was taken due to an I/O error on the input log tape. When the alternate input tape was positioned to the same block the same error occured. Both messages and blocks are printed. During the REP phase only one should be corrected. The other will automatically be dropped.

8. DATA REPLACED IN RECORD Axxxxx
B

A. The interim log record data has been replaced by the card input data.

9. SPANNED RECORD TRUNCATED

   A. The user specified 'CLOSE TRUNC' and when closing the new log data set, a spanned record was found and eliminated by shortening the block size.

10. ERROR IN CONTROL CARD FORMAT

   A. Column 4 was not blank.

   B. Columns 5-8 did not contain 'SEQ='.

   C. Column 9 did not contain a 'A' or 'B'.

   D. Columns 10-14 did not contain numeric digits.

   E. Column 15 was not blank.

   F. Columns 16-20 did not contain 'POS= ', 'CLOSE', or 'SKIP '.

   G. POS= value was not numeric.

   H. When POS= was specified, columns 26-30 did not contain ' DAT='.

11. U0031 ABEND (While positioning the alternate tape).

   A. EOF reached on the alternate input log data set prior to locating the matching block.

   B. Point error or invalid read.

   C. A log volume is missing.

12. DFS452I CONTROL CARD INPUT NOT PRESENT

13. DFS452I ERROR IN CONTROL CARD FORMAT (See message 10)

14. DFS453I UNABLE TO OPEN INPUT LOG-DD IEFRDER

15. DFS453I UNABLE TO OPEN CARD INPUT-DD SYSIN

16. DFS453I UNABLE TO OPEN PRINT OUTPUT-DD SYSPRINT

17. DFS454I UNSUCCESSFUL ALTERNATE TAPE POSITION-REASON 1

   A. OPEN failed when opening the alternate tape.

18. DFS454I UNSUCCESSFUL ALTERNATE TAPE POSITION-REASON 2

    A. A device error occured on the alternate tape.

19. DFS455I I/O ERROR ON INPUT TAPE DURING REP OPERATION

20. DFS456I CARD AND TAPE ID'S DO NOT MATCH

    A. A REP card is missing or out of sequence

21. DFS457I CONTROL CARD SPECIFIED B RECORD-NO B RECORD ON TAPE

22. DFS458I CONTROL CARDS OUT OF SEQUENCE

23. DFS459I EOF ON INPUT TAPE WITH CONTROL CARDS YET TO PROCESS

24. DFS460I MOVE WILL NOT FIT IN RECORD

    A. The offset starting point plus the data length would equal a point beyond the end of the record.

25. DFS461I USER HAS SPECIFIED REP IN BOTH A AND B RECORDS.

    A. Only one record can be corrected. The other must be dropped and will be automatically.

OPERATIONAL PROCEDURES

This section contains the initialization, termination and failure procedures presented in flow chart form.

All of the procedures assume the presence of Program Isolatio n and Log Tape Write Ahead.

All of the procedures are based on IMS/VS 1.1.3. In other words the error messages and etc. are those produced by 1.1.3.

Program Isolation allows multiple dependent regions concurrent access to the same data base and segment types but not to the same data base record. In other words, only one program may have position established within a data base record at any given point in time, and access to any updated segments by a second program is restricted until the updating program reaches a synchronization point. Backout of an online program (dynamic or batch backout) ceases when the last synchronization point is reached.

The Log Tape Write Ahead feature ensures that all data base updates which are physically written to the disk are first physically written to the log tape. This allows the user to perform the backout operation and remove all data base updates even when the log buffers are lost during the closing of the log tape. Without the feature a data base update could be written to the disk and the corresponding log record placed into the log buffer. Loss of that buffer when closing the log tape would mean that backout could not remove the update. In this situation the user must perform data base recovery.

The symbols used in the flowcharts and their meanings are:

Entry point of a procudure. Each procedure was assigned a number which appears here. If the procedure has multiple pages the next page would begin with xA, etc.

Leave this procedure and enter another whose number appears within.

Enter another procedure and perform the instructions within it. If you reach the FINISHED point return here.

Output to a terminal - usually the Master Terminal.

Terminal or Console input - usually commands.

Exit of a procedure. Return to the original procedure from which you came.

STARTING IMS/VS

The normal restart facility of IMS/VS is used to initialize the DC system with no previous log for input (COLD START) or to restart a system which has been terminated by a checkpoint command (WARM START).

The emergency restart facilities of IMS/VS are used to extend the normal restart facilities to be able to restart IMS/VS from system failures (EMERGENCY RESTART). Basically this involves resetting each active region back to its last synchronization point, restoring the data bases to that point, and restoring the message queues and SPAs to their condition at time of failure. Each dependent message region which was active at the point of failure will be rescheduled. Batch Message regions must be restarted by the user, however, IMS/VS will inform the user of the last BMP synchronization point in order that he may specify that point in the restart.

Normal system checkpointing and restart procedures are easily implemented and are of little concern from a recovery point of view. There are, however, a few situations where an operator error could adversly effect the integrity of the system.

a) Specification of COLD start after a Checkpoint Freeze or Dumpq, or after a Purge when all messages were not completely processed.

b) Normal restart will execute when entered with a termination checkpoint number and the operator intended to do an Emergency Restart.

c) Failure to initialize the security tables properly.

COLD START

A COLD start is required only when the IMS/VS system is initialized for the first time, or when some error condition prevents a WARM or EMERGENCY restart.

A COLD start may optionally be performed after a CHECKPOINT PURGE when all messages have been completely processed. Since this is usually difficult to ascertain, it is not recommended.

A COLD start assumes empty message queues. If any messages exist, they are discarded.

A previously created log tape is never input to a COLD start.

During a COLD start all control blocks are loaded from the libraries. The Master Terminal Operator may specify whether or not Password and/or Terminal security defined by the Security Maintenance Program is to be in effect.

a) MTO specifies YES = Security is in effect.

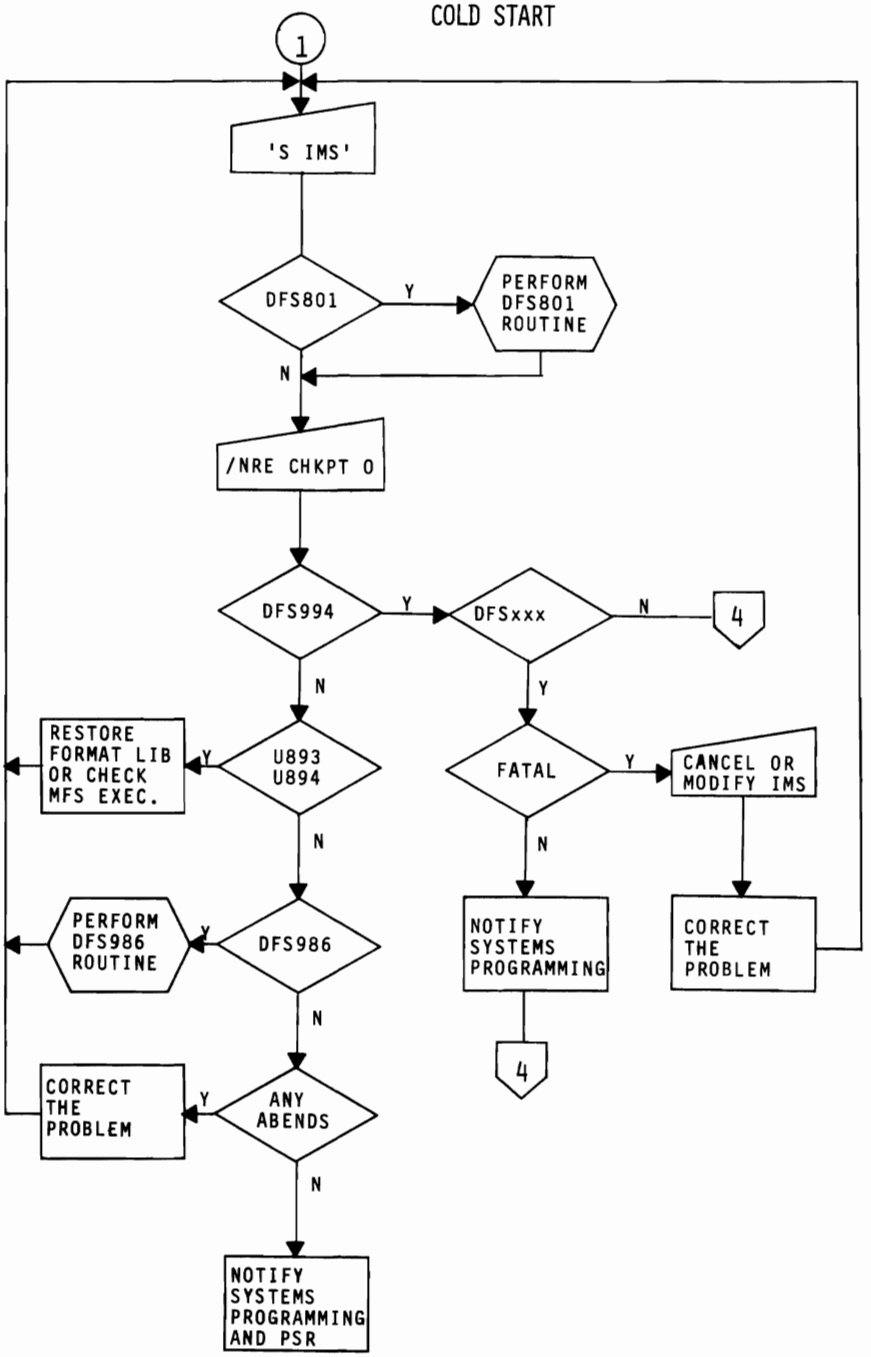
b) MTO specifies NO = No Security is in effect.

c) MTO specifies neither, SYSGEN specifies YES, Security is in effect.

d) MTO specifies neither, SYSGEN specifies NO, No security is in effect.

e) MTO specifies neither, SYSGEN specifies FORCED, Security is in effect.

COLD START


1
'S IMS'
DFS801
Y
PERFORM DFS801 ROUTINE
N
/NRE CHKPT 0
DFS994
Y
DFSxxx
N
4
N
Y
RESTORE FORMAT LIB OR CHECK MFS EXEC.
Y
U893 U894
FATAL
Y
CANCEL OR MODIFY IMS
N
N
PERFORM DFS986 ROUTINE
Y
DFS986
NOTIFY SYSTEMS PROGRAMMING
CORRECT THE PROBLEM
N
4
CORRECT THE PROBLEM
Y
ANY ABENDS
N
NOTIFY SYSTEMS PROGRAMMING AND PSR

PROCEDURE 1

The System Operator must enter a START IMS command from the system console. The response to this will be message DFS810A which contains the log buffer addresses. These addresses should be recorded in the Operations Master Log.

1) DFS801 TYPE 1 (2) SVC VECTOR NOT SET

If the vector is in use reply 'N' to abend the user, else reply 'U' to reset it.

2) The user must specify CHECKPOINT 0. The user may format any or all of the system data sets. All data sets must be formatted when IMS/VS is initialized the very first time and each must be formatted thereafter when I/O errors occur or the data set is reallocated.

The operator may optionally specify whether or not security is to be loaded.

3) DFS994 COLD START COMPLETED

When this message is received the IMS/VS Control Region is initialized.

It is possible that other DFS messages were received. For example, messages DFS827 through DFS838 may be received and may or may not represent a fatal error condition. If necessary, the System Operator should enter 'MODIFY IMS, STOP' to terminate IMS/VS. The error condition must be corrected before restarting.

If no other messages are received, or the condition has no adverse effect, Procedure 4 should be entered to start the dependent regions.

4) U893 or U894 I/O ERRORS ON FORMAT LIBRARY

It will be necessary to restore the Message Format Library from a backup copy or to re-execute the Message Format Services utility to create a new library.

5) DFS986 CANNOT OPEN SYSTEM DATA SET-DD NAME name

The DFS986 Procedure should be entered. A DD card must be added or the data set requires re-formating during the next restart.

6) If any ABENDS occur, the condition must be corrected prior to the next restart. An example of these would be U0071, U0821, etc. These would be followed by a DFS message which further explains the problem (see DFS822, DFS825).

7) If COLD start fails without an ABEND, any DFS messages must be analyzed and corrected if possible. If the condition cannot be corrected, Systems Programming should be notified.

WARM START

Warm START is the most common method of reinitializing IMS/VS. It is the recommended method of restarting after an orderly shutdown of IMS/VS and can be attempted only after such an orderly shutdown.

Before entering the restart command the Master Terminal Operator must know:

a) The type of CHECKPOINT command used to terminate IMS/VS.

b) The identification of the appropriate restart checkpoint.

c) The volume serial numbers in sequence of all log tapes to be used for restart.

d) Password and Terminal security requirements.

This information should be obtainable from the Operations Master Log.

The security tables that will be loaded are from:

a) MTO specifies YES = RESLIB*

b) MTO specifies NO = OLD LOG

c) MTO specifies neither, SYSGEN specifies YES = RESLIB*

d) MTO specifies neither, SYSGEN specifies NO = OLD LOG

e) MTO specifies neither, SYSGEN specifies FORCED = RESLIB*

* If a new security generation was done between the time of shutdown and restart, the new security will of course be used.

The system will not allow:

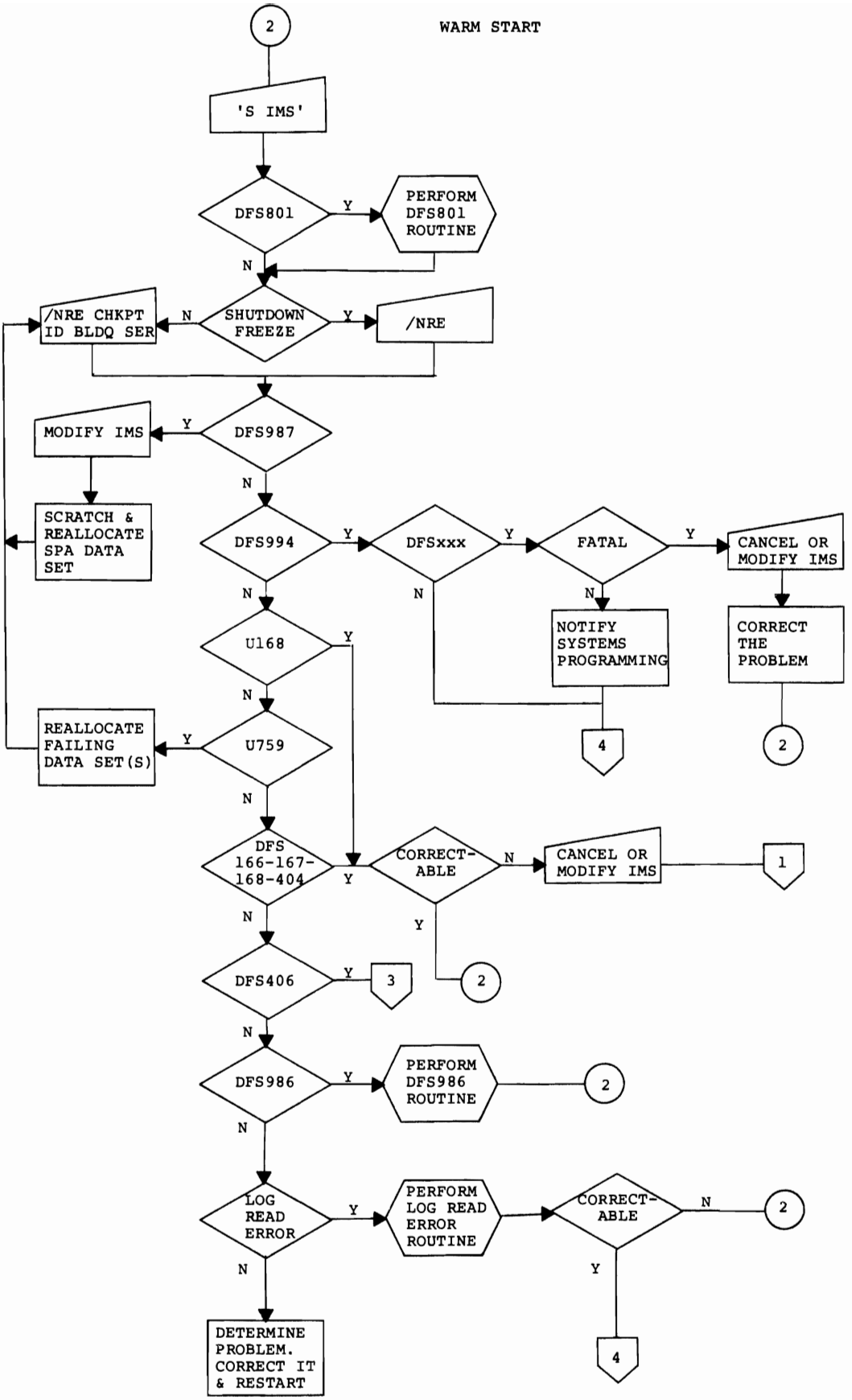
a) Normal restart from a simple checkpoint.

b) BLDQ restart from a simple checkpoint, or shutdown checkpoint which did not include dumping the queues.

c) Non-BLDQ restarts from a dump queue checkpoint.

d) Non-BLDQ restarts from any shutdown checkpoint other than the most recent.

WARM START consists of re-establishing the control program status using the control blocks logged at termination, and optionally restoring disk queues if they were logged as part of the shutdown procedures. The terminal and password security tables are initialized as stated above. The effect of a warm start is to restore the system to its status at the time of termination.


2
'S IMS'
DFS801
Y
PERFORM DFS801 ROUTINE
N
/NRE CHKPT ID BLDQ SER
N
SHUTDOWN FREEZE
Y
/NRE
MODIFY IMS
Y
DFS987
N
SCRATCH & REALLOCATE SPA DATA SET
DFS994
Y
DFSxxx
Y
FATAL
Y
CANCEL OR MODIFY IMS
N
N
N
NOTIFY SYSTEMS PROGRAMMING
CORRECT THE PROBLEM
U168
Y
N
REALLOCATE FAILING DATA SET(S)
Y
U759
4
2
N
DFS 166-167-168-404
Y
CORRECT-ABLE
N
CANCEL OR MODIFY IMS
1
N
Y
DFS406
Y
3
2
N
DFS986
Y
PERFORM DFS986 ROUTINE
2
N
LOG READ ERROR
Y
PERFORM LOG READ ERROR ROUTINE
CORRECT-ABLE
N
2
N
Y
DETERMINE PROBLEM. CORRECT IT & RESTART
4

PROCEDURE 2

The System Operator must enter a START IMS command from the system console. The response to this will be message DFS810A which contains the log buffer addresses. These should be recorded in the Operations Master Log.

1) DFS801 TYPE 1 (2) SVC VECTOR NOT SET

If the vector is in use reply 'N' to abend the user, else reply 'U' to reset it.

2) A WARM start following a FREEZE requires a log serial number only if the tape serial number has been changed since written (copied or recovered) or the restart is performed using the secondary log. The only data set which may be formatted is the Dynamic Log and that would be necessary only if the DBLLOG cannot be opened, has been reallocated, or changed in size. The appropriate restart checkpoint ID is obtained by IMS/VS from the checkpoint ID table located in the QBLKS data set.

A WARM start following a PURGE or DUMPQ requires a checkpoint ID, a BLDQ parameter, a log serial number, and optionally a FORMAT parameter. Obtain these from the Operations Master Log. Any or all of the system data sets may be formatted. This is required only if the data set cannot be opened, had been reallocated, or changed in size.

3) DFS987 WRITE ERROR ON SPA

The user may elect to ignore this message and the referenced conversation will be terminated. When this is done future conversations may be terminated because the disk SPA is experiencing I/O errors. The user could modify IMS/VS, scratch and re-allocate the SPA and perform a BLDQ restart. This wil l recover the current conversation and prevent future conversation terminations.

4) DFS994 WARM START COMPLETED

When this message is received the IMS VS Control Region is initialized.

It is possible other DFS messages were received. For example, messages DFS827 through DFS838 may be received and may or may not represent a fatal error condition. If necessary, the System Operator should enter 'MODIFY IMS, STOP' to terminate IMS/VS. The error condition must be corrected before restarting.

If no other messages are received or if the condition has no adverse effect, Procedure 4 should entered to start dependent regions.

5) U759 ERRORS IN THE MESSAGE QUEUE

If a dump is available, register 9 points to the OSAM DECB. The user may re-allocate the failing data set and restart using the BLDQ and FORMAT option for that data set.

If no dump is available, the user may re-allocate all data sets and restart using the BLDQ and FORMAT ALL option.

6) DFS166, 167, 168, 404 and U168

These messages are received when the operator selects the wrong checkpoint ID, the wrong type of restart, the wrong log serial number, etc. Message DFS168 may require a COLD start.

7) DFS406 INVALID, STATE OF SYSTEM DATA SET REQUIRES BLDQ

A previous EMERGENCY restart failed, therefore, a simple restart is not possible. Enter Procedure 3 and specify a BLDQ operand.

8) DFS986 CANNOT OPEN SYSTEM DATA SET-DD NAME name

The DFS986 Procedure should be entered. A DD card must be added or the data set requires re-formatting during the next restart.

9) S001 - LOG read error

If a log read error cccurs, enter the LOG READ Procedure. If OS/VS recovered, or swapping to a new drive corrected the problem, Procedure 4 should be entered to start the dependent regions. If a backup log tape was made available or if the log tape is recovered, WARM start must be executed again.

EMERGENCY RESTART

Whenever the IMS/VS system is to be restarted from a prior execution which was not terminated with a successful checkpoint, EMERGENCY RESTART must be used. Entering the /ERESTART command results in the same processing as normal restart, but since the system failure normally occurs sometime after the restart checkpoint, the system must be further updated to reflect the status at the time of failure.

Updating the system is accomplished in the following fashion.

a) The old log data set is opened.

b) Areas are obtained in which to reprocess message records from the log.

c) The system data sets are opened.

d) If no BLDQ was specified, the Checkpoint ID table is read from QBLKS to obtain the last system checkpoint. BLDQ restart specifies the restart checkpoint.

e) The old log is read forward until the system checkpoint is located.

f) BLDQ restarts cause the system blocks, the queues and security tables to be loaded from this checkpoint.

g) The beginning checkpoint ID is recorded in the Checkpoint ID table.

h) The old log is read forward and the records are reprocessed until EOF is reached.

   1) All commands are reprocessed.

   2) Start or stop all conversations.

   3) SPAs are written to the disk data set.

   4) Messages are placed into the temporary area and later removed if a message dequeue record is later found.

   5) If no BLDQ was specified, the system blocks are loaded from the last system checkpcint on the log.

i) Any messages remaining in the temporary areas are then written to the message queue and the area is released.

j) The old log is then opened and read backward to the last system checkpoint. All data base records are placed into the Dynamic Log data set.

k) The old log is closed and the new log is opened.

l) The Dynamic Log is read and the records are passed to Backout. This backs out data base updates to the last synchronization point for each active dependent region.

m) The backout complete message is issued for BMP regions. This provides the user with the restart point for these regions.

n) The queue fix module DFSQFIX0 is called to check the condition of the message queues.

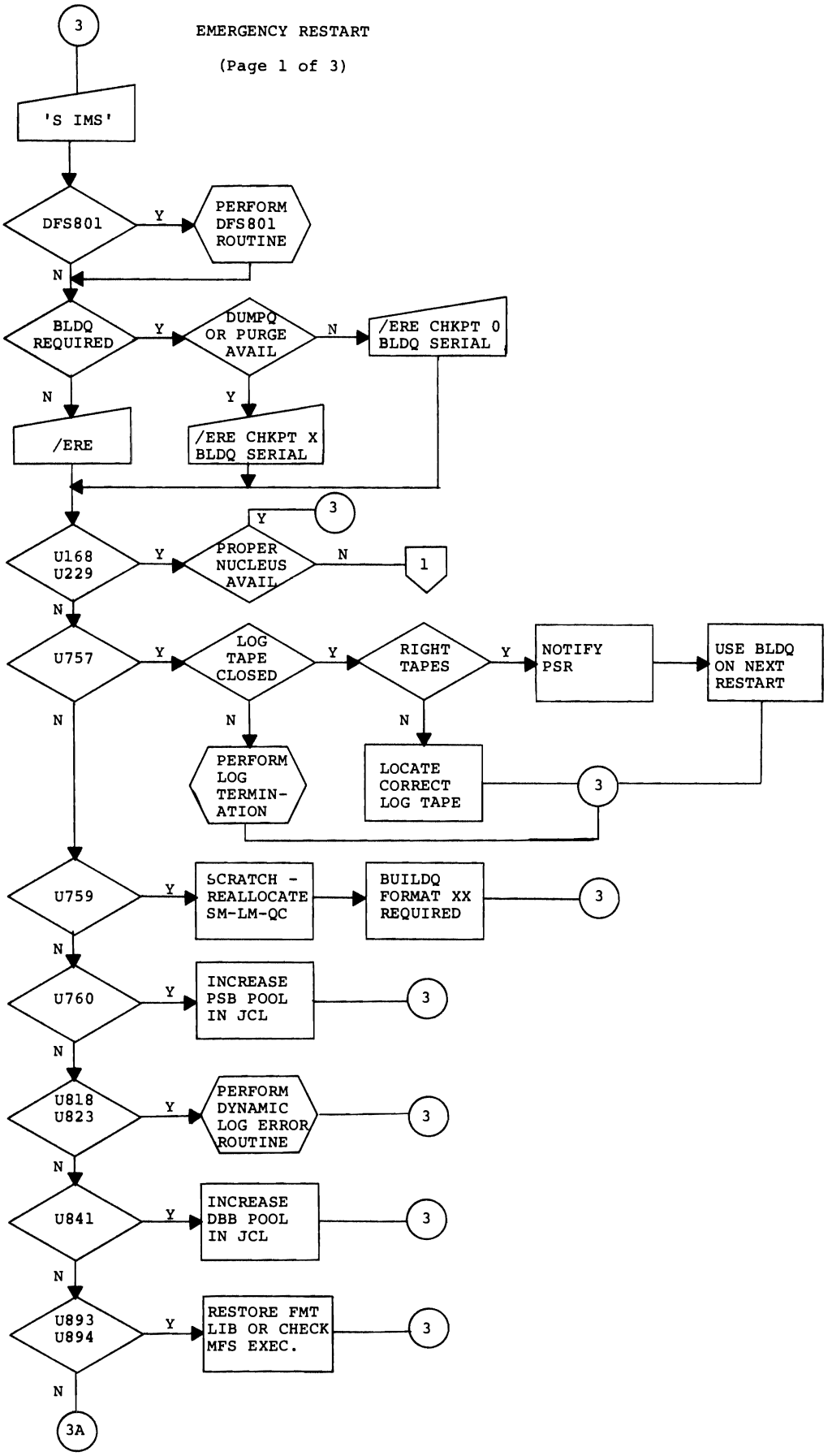
o) All lines except the line for the master terminal are stopped.

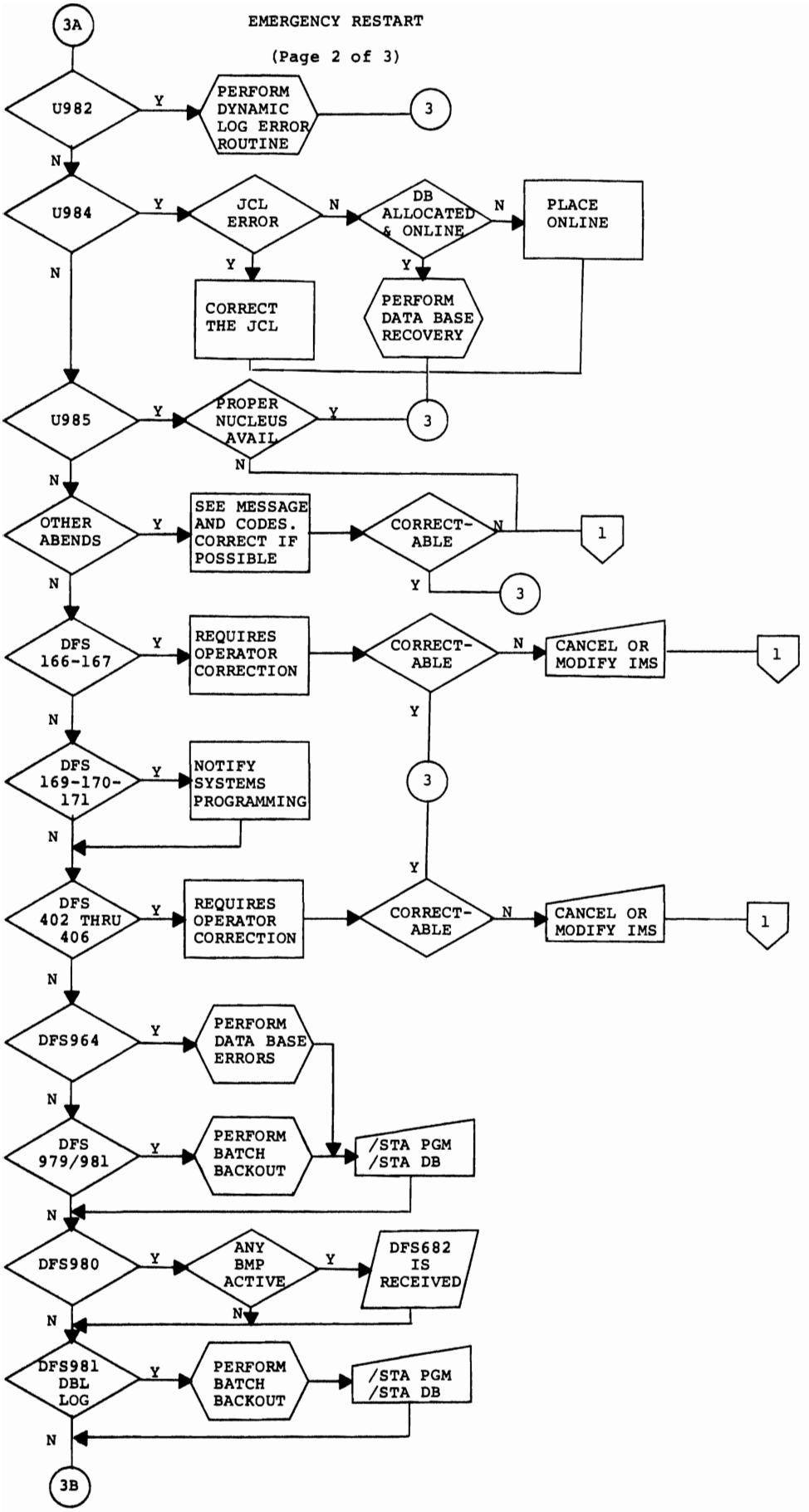
p) A system checkpoint is taken.

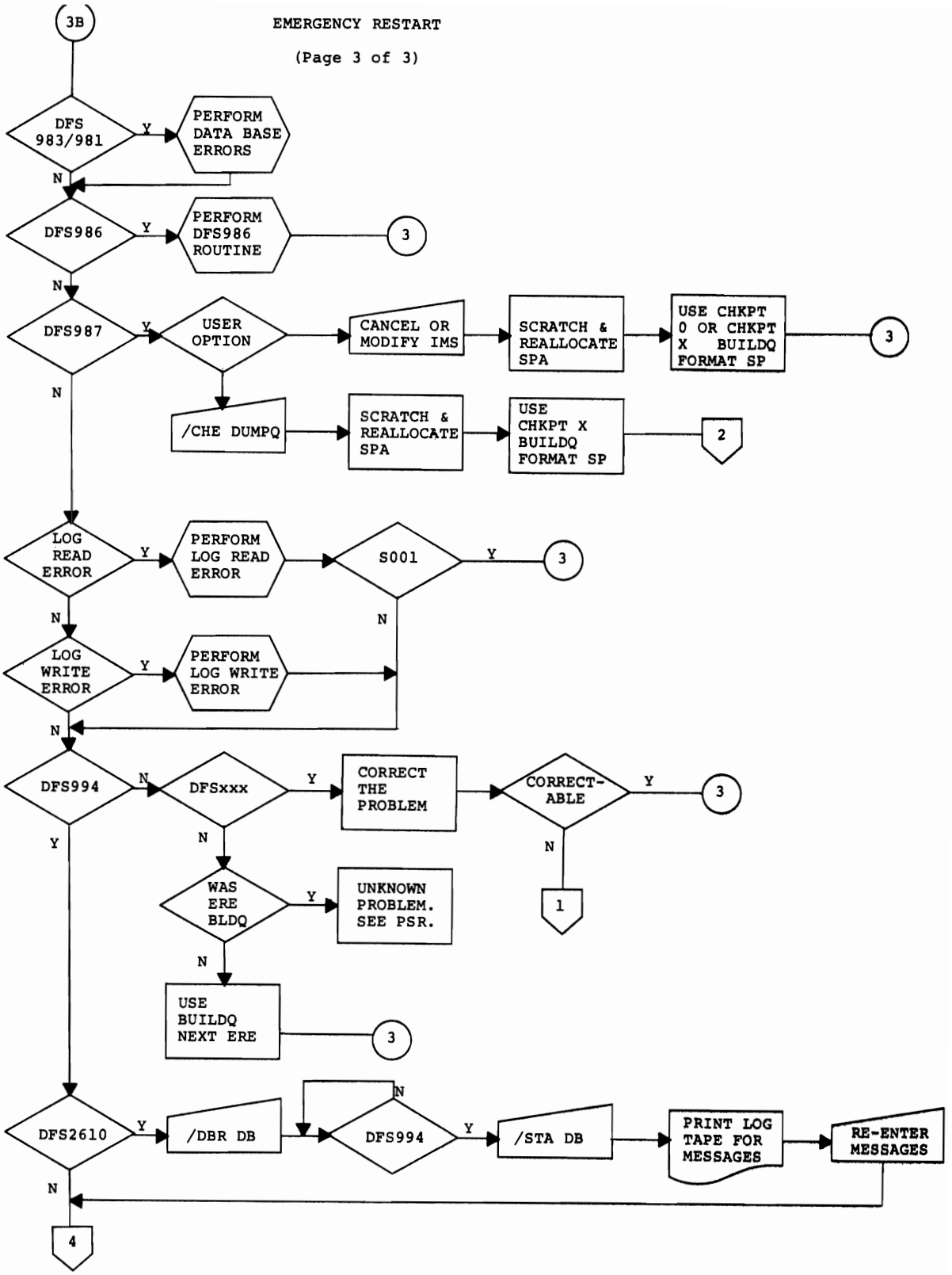
At this point the status of the system has been restored to its status as of the last system checkpoint. The data bases have been backed out to the last synchronization point of each dependent region, the message queues reflect their contents as of the failure, and the system has just taken a checkpoint. When the user starts the dependent regions the previously processing programs will be rescheduled.

EMERGENCY RESTART

(Page 1 of 3)




3A
U982
Y
PERFORM DYNAMIC LOG ERROR ROUTINE
3
N
U984
Y
JCL ERROR
N
DB ALLOCATED & ONLINE
N
PLACE ONLINE
Y
CORRECT THE JCL
Y
PERFORM DATA BASE RECOVERY
N
U985
Y
PROPER NUCLEUS AVAIL.
Y
3
N
OTHER ABENDS
Y
SEE MESSAGE AND CODES. CORRECT IF POSSIBLE
CORRECT-ABLE
N
1
Y
3
N
DFS 166-167
Y
REQUIRES OPERATOR CORRECTION
CORRECT-ABLE
N
CANCEL OR MODIFY IMS
1
Y
3
N
DFS 169-170-171
Y
NOTIFY SYSTEMS PROGRAMMING
N
Y
N
DFS 402 THRU 406
Y
REQUIRES OPERATOR CORRECTION
CORRECT-ABLE
N
CANCEL OR MODIFY IMS
1
N
DFS964
Y
PERFORM DATA BASE ERRORS
N
DFS 979/981
Y
PERFORM BATCH BACKOUT
/STA PGM /STA DB
N
DFS980
Y
ANY BMP ACTIVE
Y
DFS682 IS RECEIVED
N
N
DFS981 DBL LOG
Y
PERFORM BATCH BACKOUT
/STA PGM /STA DB
N
3B


3B
DFS 983/981
Y
PERFORM DATA BASE ERRORS
N
DFS986
Y
PERFORM DFS986 ROUTINE
3
N
DFS987
Y
USER OPTION
CANCEL OR MODIFY IMS
SCRATCH & REALLOCATE SPA
USE CHKPT 0 OR CHKPT X BUILDQ FORMAT SP
3
N
/CHE DUMPQ
SCRATCH & REALLOCATE SPA
USE CHKPT X BUILDQ FORMAT SP
2
LOG READ ERROR
Y
PERFORM LOG READ ERROR
S001
Y
3
N
N
LOG WRITE ERROR
Y
PERFORM LOG WRITE ERROR
N
DFS994
N
DFSxxx
Y
CORRECT THE PROBLEM
CORRECT-ABLE
Y
3
Y
N
N
WAS ERE BLDQ
Y
UNKNOWN PROBLEM. SEE PSR.
1
N
USE BUILDQ NEXT ERE
3
N
DFS2610
Y
/DBR DB
DFS994
Y
/STA DB
PRINT LOG TAPE FOR MESSAGES
RE-ENTER MESSAGES
N
4

PROCEDURE 3

The System Operator must enter a START IMS command from the system console. The response to this will be message DFS810A which contains the log buffer addresses. These addresses should be recorded in the Operations Master Log.

1) DFS801 TYPE 1 (2) SVC VECTOR NOT SET

If the vector is in use reply 'N' to abend the user, else reply 'U' to reset it.

2) A BLDQ operand is required to initiate rebuilding of message queue or scratchpad data sets during EMERGENCY restart only when an error has occured in using one of the data sets or one has become full. The IMS/VS 1.1.1 and later user has more flexibility in deciding whether or not a BLDQ is required. Those releases of IMS/VS contain module DFSQFIXO which will verify the state of the message queues. These users may try restarting with out a BLDQ and if message DFS2610 is received, they may then try restarting with a BLDQ.

If a BLDQ restart is required, the latest DUMPQ or PURGE checkpoint should be specified. If none exist since the last COLD start, checkpoint 0 must be specified.

When BLDQ is specified, the operator may optionally format any or all of the system data sets. This is required only if the data sets cannot be opened, or had been reallocated.

3) U168 - U169 BLOCKS DO NOT MATCH - TOO MANY REGIONS

If a new SYSGEN were done between the failure and the restart, the user must COLD start. If this is not the case the user probably specified the wrong nucleus.

4) U757 QUEUE MANAGER ERROR

An unclosed log tape, a missing volume, or using a tape other than the one in use when the failure occured, could be responsible for this abend. If the log tapes are correct, turn the problem over to the PSR.

5) U759 MESSAGE QUEUE ERROR

If this is not a real I/O error perform the actions for U757. If the problem is an I/O error, scratch and reallocate the data set. A BLDQ and a FORMAT specifying the data set is required when attempting the next restart.

6) U760 PSB POOL TOO SMALL

Increase the PSB pool size in the PARM field of the EXEC card.

7) U818 - U823 I/O ERROR ON DYNAMIC LOG

A write error or out of space condition occured on the Dynamic log. Scratch and reallocate the data set adding space when the condition is U823.

8) U841 DATA BASE BUFFER POOL TOO SMALL

Increase the pool size in the PARM field of the EXEC card.

9) U893 - U894 I/O ERROR ON FORMAT DATA SET

If a backup copy of the Format Library exists, the data set may be restored. Otherwise check the last execution of the Message Format Utility for errors and re-execute the utility. DFS893 will also be issued.

10) U982 I/O ERROR ON DYNAMIC LOG

A read error occured on the Dynamic log. Scratch and reallocate the data set. DFS982 and DFS981 will also be issued.

11) U984 UNABLE TO OPEN DATA SET

See message DFS984 which is also issued for the data base name. If the JCL is good and the data base was online EMERGENCY restart could have damaged the data base. It must be recovered before the next restart.

12) U985 UNABLE TO LOCATE DMB

If a SYSGEN was done between the failure and the restart the system must be COLD started. Otherwise the operator probably specified the wrong nucleus.

13) DFS166 - DFS167 INVALID CHKPT or RESTART TYPE

These are probably operator errors. If the condition cannot be corrected, the operator should modify IMS and perform a COLD start.

14) DFS169 - DFS170 - DFS171 SECURITY FAILED

Systems Programming should be notified of security problems.

15) DFS402 thru DFS406 INVALID RESTART, LOG, CHKPT

These are operator errors. Check the Messages and Codes and correct the failure. If the condition is uncorrectable the system must be COLD started.

16) DFS964 INSERT FAILED ON PRIOR UPDATE. RECOVERY REQUIRED FOR DATABASE XXXXXXX.

The Failure occured during an insert to a VSAM database. The '50' record is not available to backout the change therefore the database must be recovered and Batch Backout executed. Enter the DATABASE ERRORS Procedure to accomplish this.

17) DFS979 and DFS981 BMP name WAS ACTIVE AT TIME OF FAILURE

The system is executing without Program Isolation or the operator specified NOBMP in the /ERE command therefore the listed BMPs were not backed out. These must be backed out prior to restarting the data bases and programs.

18) DFS980 BACKOUT COMPLETE FOR PSB name

This message should be expected for any region which had updated data bases between the last synchronization point and the failure. If the region is a BMP message DFS682 should follow and contains the restart checkpoint ID.

19) DFS981 I/O ERROR ON DYNAMIC LOG WHILE BACKING OUT

An I/O error occured on the Dynamic Log. Perform a Batch Backout using the log tape which was input to Emergency Restart and specify the PSB and data bases listed in the following DFS981 message.

20) DFS983 and DFS981 I/O ERROR ON DATA BASE DURING BACKOUT

The data base must be recovered and Batch Backout executed prior to starting the data base and the program. The Procedure DATA BASE ERRORS should be followed.

21) DFS986 CANNOT OPEN SYSTEM DATA SET-DD NAME name

The DFS986 Procedure should be entered. A DD card must be added or the dat set requires re-formatting during the next restart.

22) DFS987 WRITE ERROR ON SPA

The user may elect to ignore this message and the referenced conversation will be terminated. When this is done future conversations may be terminated because the disk SPA is experiencing I/O errors.

The user has the option of terminating IMS/VS with the DUMPQ, scratching and reallocating the SPA and restarting with a BLDQ and FORMAT from that point. The cancelled conversation is lost.

Or IMS/VS may be cancelled, the data set scratched and reallocated, and a BLDQ and FORMAT done from an earlier DUMPQ. This will recover the cancelled conversation.

23) LOG READ ERROR

If a log read error occurs, enter the LOG READ ERROR Procedure. If OS/VS recovered, or swapping to a new drive solved the problem, continue. If not, a S001 will occur.

24) LOG WRITE ERROR

Enter the LOG WRITE ERROR Procedure to handle these errors. If swapping to a new drive or usage of dual logs corrects the problem IMS/VS continues. If not, IMS/VS will abend and the log tape must be closed using DFSFLOTO.

25) DFS994 EMERGENCY RESTART COMPLETED

When this message is received the IMS/VS Control Region is initialized. THIS MESSAGE SHOULD BE RECEIVED. If not other DFS messages may explain the problem. If the problem is correctable EMERGENCY restart may be tried again. If not a COLD start is required.

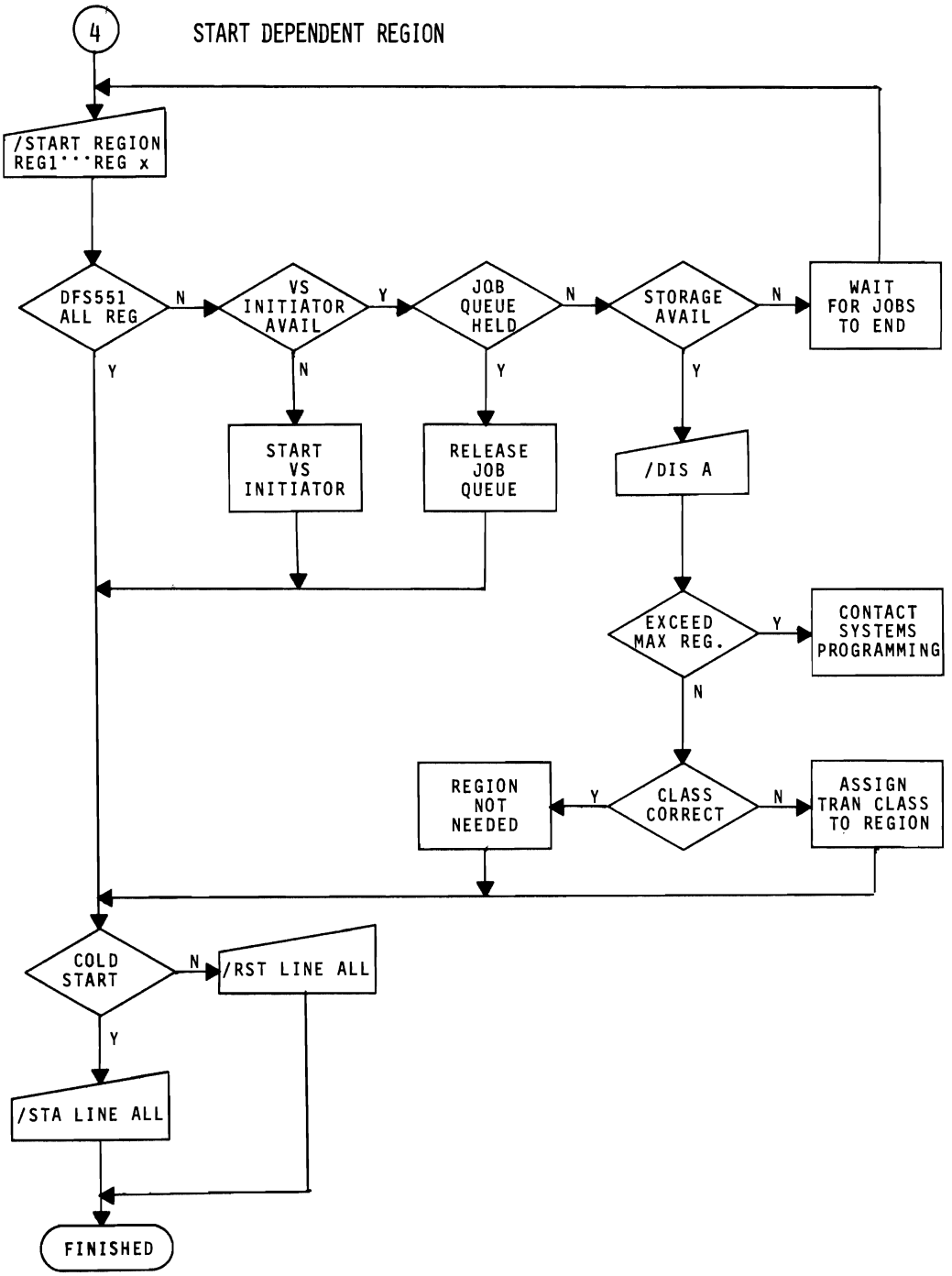
Many EMERGENCY restart problems can be corrected by specifying a BLDQ and trying again. This is especially true when the problem involves the system data sets.

26) DFS2610 MSG Q ERRORS. MSGS MAY BE LOST.

This message is generated when DFSQFIXO finds errors in the message queues. In the process of verifying the message queues, errors were encountered and various messages were lost during the process of restoring the queue pointers.

If the restart was not a BLDQ, the operator could cancel IMS/VS and perform a BLDQ restart from a previous DUMPQ. If the restart was a BLDQ, entering a /DBR command will free the current log tape. That tape may be printed using DFSERA10. The log record type '3x' represent the messages which were discarded. These messages may then be re-entered for processing.

At this point EMERGENCY restart is completed. The operator should enter Procedure 4 to restart the dependent regions.


4
START DEPENDENT REGION
/START REGION REG1'''REG x
DFS551 ALL REG
N
VS INITIATOR AVAIL
Y
JOB QUEUE HELD
N
STORAGE AVAIL
N
WAIT FOR JOBS TO END
Y
N
Y
Y
START VS INITIATOR
RELEASE JOB QUEUE
/DIS A
EXCEED MAX REG.
Y
CONTACT SYSTEMS PROGRAMMING
N
REGION NOT NEEDED
Y
CLASS CORRECT
N
ASSIGN TRAN CLASS TO REGION
COLD START
N
/RST LINE ALL
Y
/STA LINE ALL
FINISHED

PROCEDURE 4. START DEPENDENT REGION

Once the IMS/VS Control Region has been initialized via a COLD, WARM, or EMERGENCY restart, the dependent regions must b e started.

1) /START REGION REG1 . . . . REGx

The Master Terminal Operator should enter the start region command specifying each dependent region. There is no start region ALL command.

The response to this command will be DFS058 START COMMAND IN PROGRESS.

2) DFS551 MESSAGE REGION name STARTED

This message should be received for each dependent region specified in the start command.

If one or more DFS551 messages is not received, check the OS/VS initiators to be sure one is available with the proper class. If not, start that initiator.

If the initiator is available, check the JOB QUEUE and release it if it is in HELD status.

If sufficient storage is not available, wait for active jobs to end and then start the dependent region again.

Enter a /DISPLAY ACTIVE command from the Master Terminal. If the number of active regions equals the maximum allowable for this IMS/VS system no more regions can be started. This should be reported to Systems Programming for resolution. The operator may continue with the number of regions available or terminate IMS/VS.

If the number of regions specified is within the allowable range, check the transaction classes assigned to the regions. If all classes are accounted for, the additional requested region(s) is not needed. Otherwise use the /ASSIGN command to assign the missing classes to the started regions.

3) COLD START

A /START LINE ALL must be entered for all COLD starts.

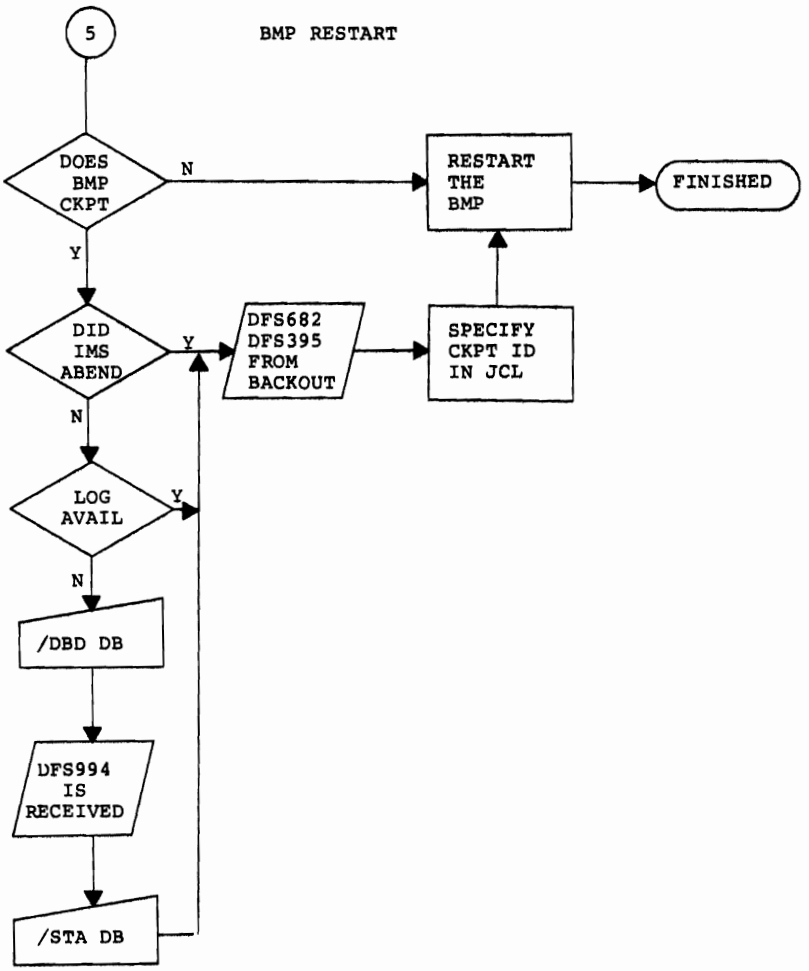
A RESTART LINE ALL should be entered for all WARM and EMERGENCY restarts. This command will not reset conditions such as conversations or special operating modes.

4) FINISHED

The IMS/VS system should now be fully operative. The Master Terminal Operator could optionally inform the users of the availability of the system by entering:

```
/BROADCAST ACTIVE
IMS IS OPERATIONAL    TIME=xx:xx
```

If BMP regions are to be activated, these regions must be started. If the system was started via EMERGENCY RESTART, enter the BMP RESTART procedure for each of these regions.


5
BMP RESTART
DOES BMP CKPT
N
RESTART THE BMP
FINISHED
Y
DID IMS ABEND
Y
DFS682 DFS395 FROM BACKOUT
SPECIFY CKPT ID IN JCL
N
LOG AVAIL
Y
N
/DBD DB
DFS994 IS RECEIVED
/STA DB

PROCEDURE 5. BMP RESTART

This procedure should be entered to restart BMP regions following a DEPENDENT REGION ABEND, a WARM START necessitated by a failure condition, or an EMERGENCY RESTART.

1) DOES BMP CKPT

A BMP which issues DL/I checkpoint calls generates "DFS681 CHKPT BMP PSB psbname ID number SER number" messages on the Master Terminal. If the BMP does not issue checkpoint calls there is no "restart" point other than the beginning of the program. These programs should be restarted exactly as they were started intitially.

2) DID SYSTEM ABEND

If the IMS/VS Control Region abended, Emergency Restart would have backed out the BMP region and freed the log tape which was being created at the time of failure.

3) LOG AVAILABLE

Is the log tape which was being created at the time of the BMP failure available for usage during the restart process? It is if a /DBR or /DBD command has been entered since the failure.

If not, enter a /DBD command to force an end of volume on the log tape. Message DFS994 will be received following the mounting of a new volume. The named data base may then be restarted.

4) DFS682 - DFS395

A dynamic backout at program abend or during Emergency Restart will produce message "DFS682 BATCH-MSG PROGRAM pgmname MAY BE RESTARTED FROM CHKPT ID value". If the backout was performed by Batch Backout, message "DFS395 BACKOUT COMPLETE FOR PSB psbname (TO CHKPT value)" will refer to the last DFS681 message issued to the Master Terminal for the CHKPT id.

5) SPECIFY CHKPT ID IN JCL

Add the following to the BMP JCL:

```
EXEC PGM=.......,PARM=(.......,ckptid)
//IMSLOGR  DD  DSN=IMSLOG,DISP=(OLD,KEEP),
//    VOL=SER=number,UNIT=TAPE,LABEL=(2,BLP)
```

The ckptid is obtained from the DFS682, the DFS395, or the last DFS681 issued prior to the failure.

The VOL=SER=number is the serial number of the log volume being created when the failure occured and may be obtained from the last DFS681 message issued prior to the failure.

The label parameter is required for MVS systems.

6) RESTART BMP

Using the original or modified JCL, restart the BMP.

NORMAL TERMINATION

IMS/VS uses the checkpoint facility for orderly system termination. The /CHECKPOINT command may be entered by the Master Terminal Operator with a FREEZE, DUMPQ, or PURGE operand to initiate system termination. This is the recommended procedure for terminating IMS/VS and the selection of the operand is dependent upon operational requirements and the time allowed for termination.

The following chart shows the system actions performed for each type of checkpoint.

ACTION	SIMPLE	FREEZE	DUMPQ	PURGE
Stop terminal input		X	X	X
Process queued transactions				X
Free MSG regions		X	X	X
Purge incore MSG queue to disk	X	X	X	X
Log blocks and tables	X	X	X	X
Write CHKPT ID to Master Terminal	X	X	X	X
Purge data base buffer pool*		X	X	X
Terminate MSG regions		X	X	X
Send queued output				X
Wait for BMPs sync point		X	X	X
Send shutdown MSG to terminals		X	X	X
Dump MSG queues and SPAs to log			X	X
Close data bases		X	X	X
Close queues		X	X	X
Close log		X	X	X
Terminate IMS Control Region		X	X	X

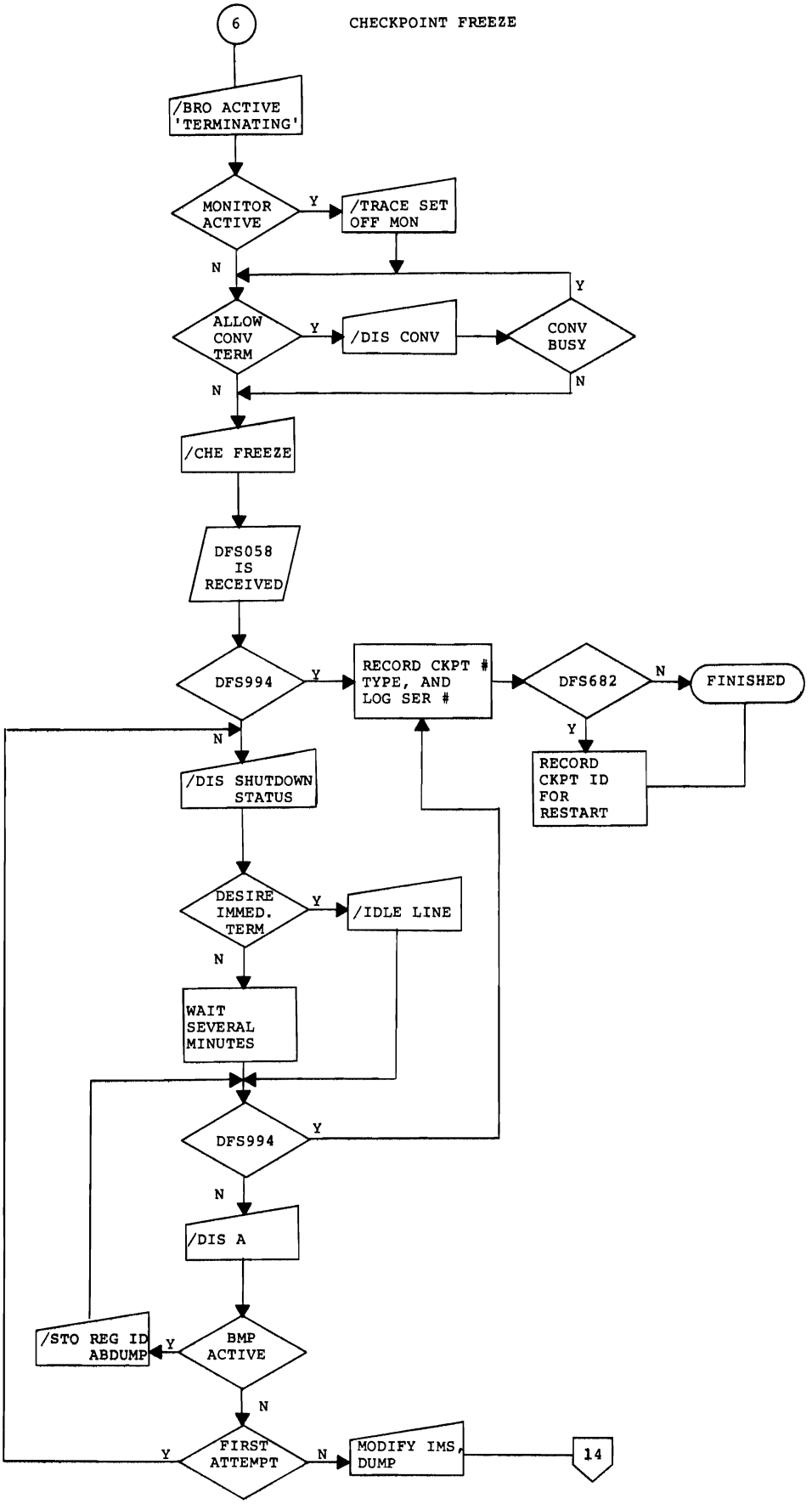
* Data base buffers are flushed for each PSB as it terminates. This is part of program termination rather than checkpoint or system termination.

CHECKPOINT FREEZE

CHECKPOINT FREEZE is the most expedient means of causing orderly IMS/VS termination. Input and output messages in transit are allowed to complete, then each communication line is stopped. Message regions are terminated as soon as the current messages being processed have been completed. Batch Message regions are presented a status code at their next DL/I CHECKPOINT call.

Message queues and data bases are closed. Checkpoint data is written to the system log just as during simple checkpoint. Message DFS994 containing the last checkpoint ID is sent to the master terminal, the system log is closed, and the IMS/VS Control Region is terminated.

Although it is the fastest way to accomplish an orderly shutdown, CHECKPOINT FREEZE should not be used regularly for normal system termination. It may cause some terminal operators to not receive responses for extended periods of time (until IMS/VS again becomes active). In the event the restart requires a BLDQ, it may be necessary to back up many log tapes (to the last PURGE or DUMPQ). The cost of executing a DUMPQ vs FREEZE is the time it takes to dump the queues, and will vary depending upon the system design, transaction volumes, and activity at a particular time of day.


6
CHECKPOINT FREEZE
/BRO ACTIVE 'TERMINATING'
MONITOR ACTIVE
Y
/TRACE SET OFF MON
N
Y
ALLOW CONV TERM
Y
/DIS CONV
CONV BUSY
N
N
/CHE FREEZE
DFS058 IS RECEIVED
DFS994
Y
RECORD CKPT # TYPE, AND LOG SER #
DFS682
N
FINISHED
Y
N
/DIS SHUTDOWN STATUS
RECORD CKPT ID FOR RESTART
DESIRE IMMED. TERM
Y
/IDLE LINE
N
WAIT SEVERAL MINUTES
DFS994
Y
N
/DIS A
/STO REG ID ABDUMP
Y
BMP ACTIVE
N
Y
FIRST ATTEMPT
N
MODIFY IMS DUMP
14

PROCEDURE 6

If at all possible, the Master Terminal operator should notify all users that IMS/VS is to be terminated and when it will occur. This allows the users to terminate conversations and warns them that the response to any further input may be delayed. The /BROADCAST ACTIVE should be issued as much in advance of termination as possible.

If the monitor is active, it should be turned off. Failure to do so will result in loss of statistics when the monitor reports are printed.

If active conversations are to be allowed to terminate, the Master Terminal Operator should display the conversations. If busy conversations exist, he should wait a reasonable length of time and display them again.

Entering the /CHECKPOINT FREEZE command will result in message DFS058 CHECKPOINT COMMAND IN PROGRESS being received at the master terminal.

When termination is completed, message DFS994 which contains the checkpoint number (date/time), the log volume serial number, and the type of checkpoint is received. This information should be recorded in the Operations Master Log. If any BMPs were active, message DFS682 BATCH-MSG PROGRAM pgmname MAY BE RESTARTED FROM CHKPT ID id-value is issued for each. This information should be recorded in the Operations Master Log and used to restart each BMP at the next warm start.

If termination is not successful (no DFS994 message) a /DISPLAY SHUTDOWN STATUS command will display the messages in process. If the operator can not wait for the messages to finish processing, a /IDLE LINE x will cause any input messages to be discarded and output messages to be returned to the message queue.

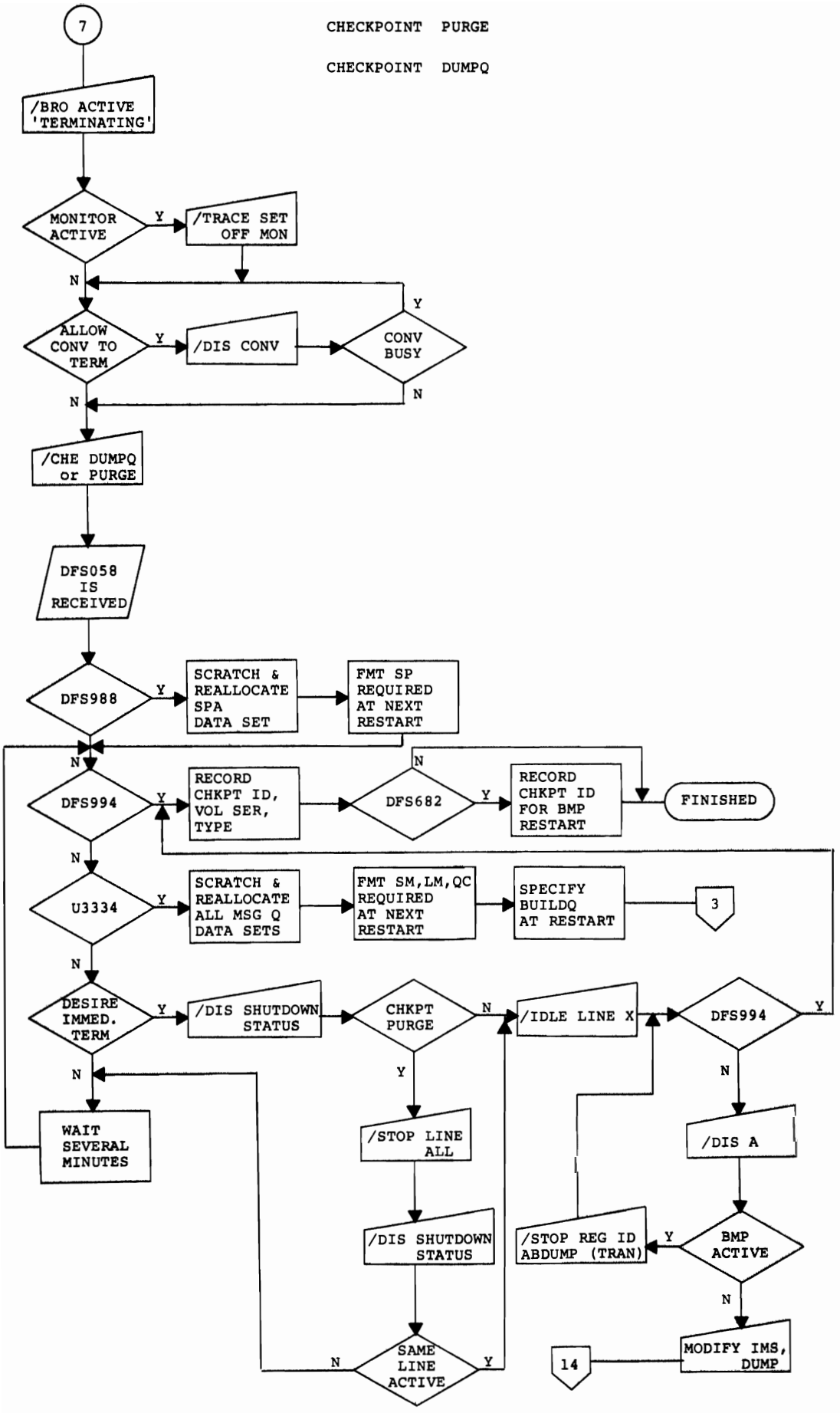
If after several minutes the DFS994 is not received, the MTO should enter a /DISPLAY ACTIVE command and look for BMP programs. If these programs do not access the message queues or issue DL/I checkpoint calls they will not terminate until they reach EOJ. The MTO may cancel BMPs with a /STOP REGION id ABDUMP command.

If no BMPs are active the lines may still be active, therefore, the operator should once more display the shutdown status and react accordingly. If termination does not complete, it may be assumed IMS/VS cannot be terminated normally. When this is the case the operator should enter a MODIFY IMS,DUMP or MODIFY IMS,STOP command from the system console. This will cause OS/VS to terminate IMS/VS. At this point procedure 14 should be entered to insure the log tape is properly closed.

CHECKPOINT PURGE or DUMPQ

DUMPQ requests an immediate shutdown. Input and output messages in transit are allowed to complete, then each communication line is stopped. Message regions are terminated as soon as the current messages being processed complete. Batch Message regions are terminated at their next DL/I checkpoint call. The contents of the message queues and scratchpad data sets are written to the system log along with the checkpoint data.

PURGE is the most time consuming method of terminating IMS/VS. The input lines are stopped as soon any messages in transit are received. All messages in the input queue are processed if possible (transaction or program not stopped) and all output is transmitted if possible (line or terminal not stopped). When all active regions have terminated, the output communication lines are stopped. Any unprocessed input messages or any untransmitted output messages are written to the system log along with the checkpoint data.


7
CHECKPOINT PURGE
CHECKPOINT DUMPQ
/BRO ACTIVE 'TERMINATING'
MONITOR ACTIVE
Y
/TRACE SET OFF MON
N
ALLOW CONV TO TERM
Y
/DIS CONV
CONV BUSY
Y
N
N
/CHE DUMPQ or PURGE
DFS058 IS RECEIVED
DFS988
Y
SCRATCH & REALLOCATE SPA DATA SET
FMT SP REQUIRED AT NEXT RESTART
N
DFS994
Y
RECORD CHKPT ID, VOL SER, TYPE
N
DFS682
Y
RECORD CHKPT ID FOR BMP RESTART
FINISHED
N
U3334
Y
SCRATCH & REALLOCATE ALL MSG Q DATA SETS
FMT SM,LM,QC REQUIRED AT NEXT RESTART
SPECIFY BUILDQ AT RESTART
3
N
DESIRE IMMED. TERM
Y
/DIS SHUTDOWN STATUS
CHKPT PURGE
N
/IDLE LINE X
DFS994
Y
N
Y
N
WAIT SEVERAL MINUTES
/STOP LINE ALL
/DIS A
/DIS SHUTDOWN STATUS
/STOP REG ID ABDUMP (TRAN)
Y
BMP ACTIVE
N
N
SAME LINE ACTIVE
Y
14
MODIFY IMS, DUMP

PROCEDURE 7

The Master Terminal Operator should notify all users that IMS/VS will terminate in five to ten minutes. This allows the terminal operators to complete their work or at least to reach a convenient stopping point.

1) MONITOR ACTIVE

If the monitor is active, it should be turned off prior to entering the checkpoint command. Failure to do so will result in loss of statistics when the monitor reports are printed.

2) ALLOW CONV TO TERM

If active conversations are to be allowed to terminate, the Master Terminal Operator should display conversations. If busy conversations exist, he should wait a reasonable length of time and display them again.

3) /CHE DUMPQ or PURGE

Entering the /CHECKPOINT DUMPQ or PURGE command will result in message DFS058 CHECKPOINT COMMAND IN PROGRESS being received at the master terminal.

4) DFS988 READ ERROR OCCURRED READING SPA

If message DFS988 is received, an I/O error occured when trying to read the SPA disk data set in preparation to dumping it to the log. This can occur only if some conversations had not terminated prior to entering the checkpoint command. Instead of the SPA, the DFS988 message is written to the log tape which will cause this specific conversation to be terminated during the next restart.

5) DFS994

When termination is completed, message DFS994 which contains the checkpoint number (date/time) , the log volume serial number, and the type of checkpoint, is received. This information should be recorded in the Operations Master Log. If any BMPs were active and DUMPQ was specified, message DFS682 BATCH-MSG PROGRAM pgmname MAY BE RESTARTED FROM CHKPT ID id-value is received for each. This information should be recorded in the Operations Master Log and used to restart each BMP at the next warm start.

6) U3334 QUEUE BUFFER IN USE

If a U3334 ABEND occurs, there is probably some logic error within IMS/VS. The user must scratch and re-allocate the three message queue data sets. An Emergency Restart must be performed with the BLDQ and FORMAT SM,LM,QC options specified.

7) DESIRE IMMEDIATE TERM

The Master Terminal Operator should allow IMS/VS time to process the current messages if DUMPQ was specified, or time to process all the messages in the queue when PURGE was specified. If he is unable to wait, he may displa y the shutdown status. If termination was a PURGE, he may stop the lines which will stop the sending of all output messages (after the current one is sent). The current message is allowed to complete but any more messages produced will be queued. This should soon free all the lines. Another /DISPLAY SHUTDOWN STATUS would indicate if IMS/VS were hung up while trying to send a current message to a line.

If a DUMPQ was specified or if IMS/VS is hung up on a line, a /IDLE LINE x command will cause any input messages to be discarded and the current output message to be returned to the message queue. This should allow termination to complete shortly.

If it does not, a /DISPLAY ACTIVE command will inform the operator of any BMPs which might be processing. If these programs do not access the message queues or issue DL/I checkpoint calls they will not terminate until they reach EOJ. The operator may cancel such BMPs with a /STOP REGION id ABDUMP command.

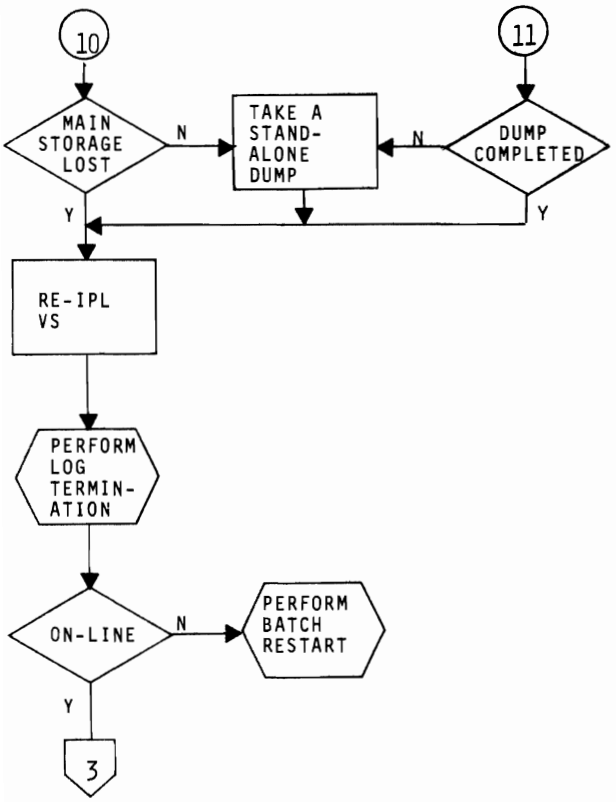
If no BMPs are active it may be assumed IMS/VS cannot be terminated normally. When this is the case the user should enter MODIFY IMS,DUMP or MODIFY IMS,STOP from the system console. This will cause OS/VS to terminate IMS/VS. At this point procedure 14 should be entered to insure the log tape is properly closed.

SYSTEM FAILURES

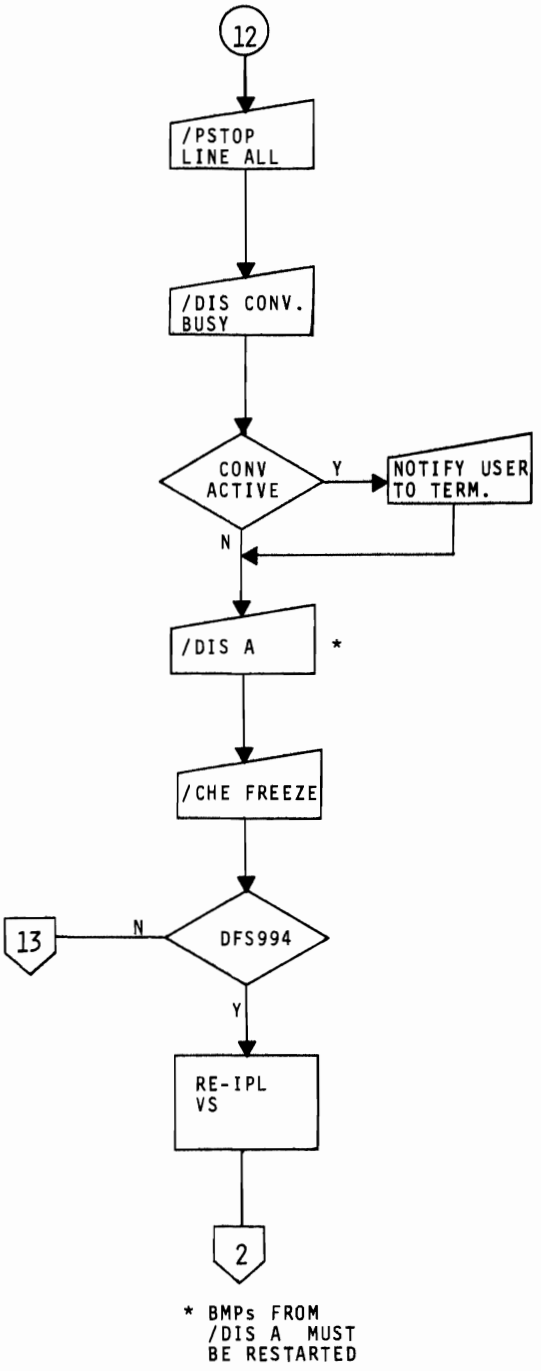
When a failure occurs in the OS/VS-IMS/VS environment it is important that operations perform the correct recovery and restart procedures to insure system integrity. The procedures in this section were developed to assist the user to that end.

VS LOOP/WAIT
HARDWARE OR POWER FAILURE

VS ABEND



VS PROBLEMS
REQUIRING IPL



* BMPs FROM /DIS A MUST BE RESTARTED

PROCEDURE 10. OS/VS LOOP, WAIT STATE, HARDWARE OR POWER FAILURE.

When the OS/VS Operating System experiences a loop or wait condition, a power failure, or a hardware failure, both the operating system and IMS/VS must be restarted.

If the failure did not result in a dump being taken and if memory is still intact, the operator should take a standalone dump. This dump will allow the log termination procedure to properly close the log tape and guarantees complete integrity of the IMS/VS system. The log tape input to the log termination procedure must be the unclosed log tape which was being created at the time of failure.

If the IMS/VS system is an online system, the Emergency Restart procedure will backout all activity to the last synchronization point for each dependent region. The user will be notified of the restart checkpoint ID for each BMP region and he must restart these regions at that point.

If the IMS system is a batch system, the batch restart should allow the user the option of performing a batch backout for updated data bases and restarting the job, or recovering all updated data bases to the beginning of the job and re-executing the entire job.

PROCEDURE 11. OS/VS ABEND.

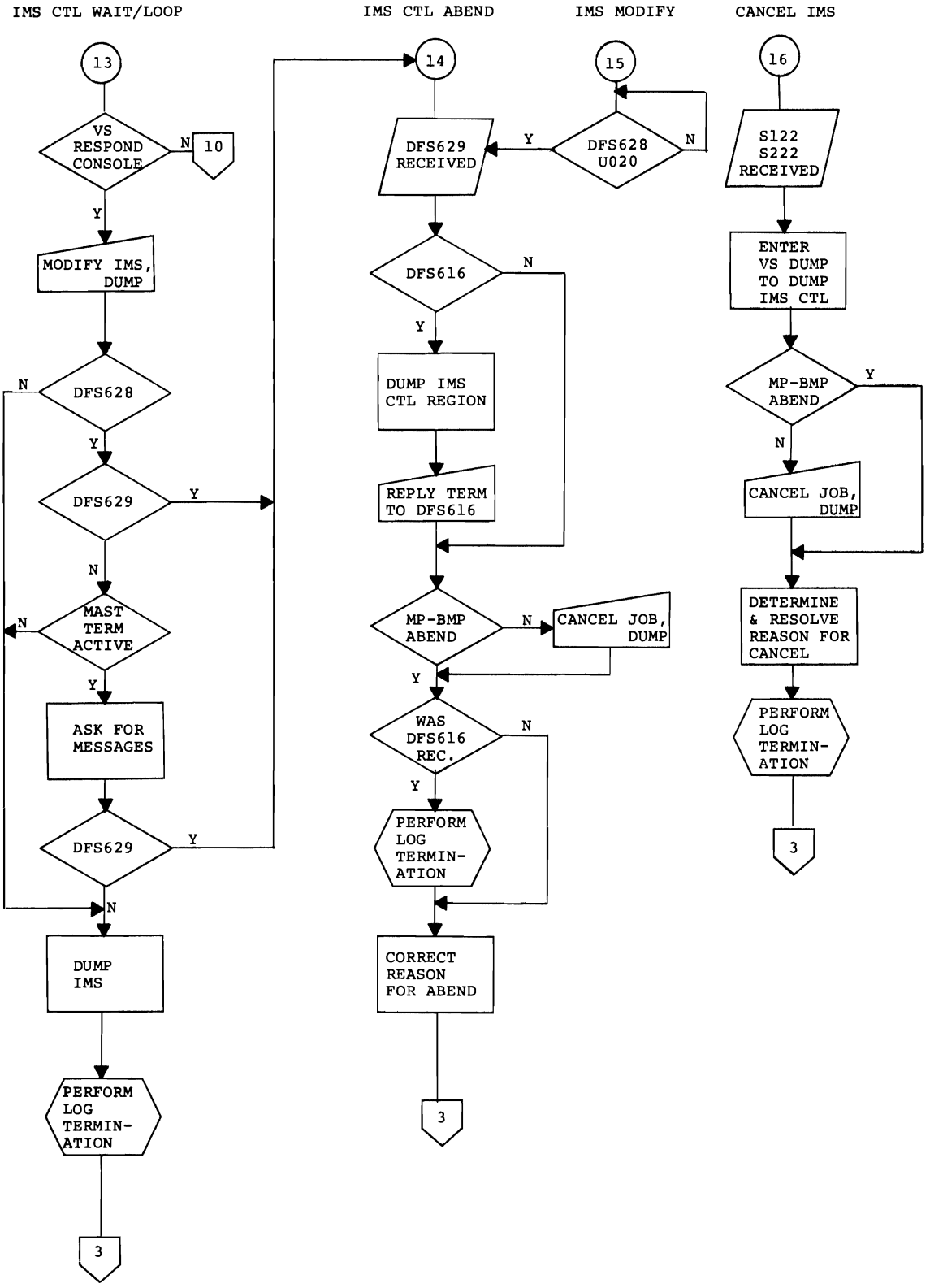
When OS/VS abends a dump will be taken and placed on the SYS1.DUMP data set. If the dump completes OS/VS message IEA023I is issued. If this message is not received, the operator must execute a standalone dump.

PROCEDURE 12. OS/VS REQUIRES AN IPL.

Certain OS/VS problems could be encountered which require a re-IPL of the OS/VS system.

The operator might wish to broadcast a message to all active terminals informing them the system is to be shut down. After allowing time for terminal activity to quiesce, display conversations. If any busy conversations exist, the user should be notified to terminate them. A display active will indicate BMP regions which are processing. The operator may elect to wait for these to terminate before proceeding. If he does, a DFS552 message will be received for each BMP region as it terminates.

Entering the /CHECKPOINT FREEZE command will terminate the IMS/VS Control Region and should result in a DFS994 message being received. If the message is not received refer to Procedure 13. After the re-IPL of OS/VS the IMS/VS system may be restarted with a WARM start. Any BMPs which were still active when the Control Region was terminated will have to be restarted.


IMS CTL WAIT/LOOP
IMS CTL ABEND
IMS MODIFY
CANCEL IMS
13
VS RESPOND CONSOLE
N
10
Y
MODIFY IMS, DUMP
N
DFS628
Y
DFS629
Y
N
MAST TERM ACTIVE
N
Y
ASK FOR MESSAGES
DFS629
Y
N
DUMP IMS
PERFORM LOG TERMIN-ATION
3
14
DFS629 RECEIVED
15
DFS628 U020
Y
N
DFS616
N
Y
DUMP IMS CTL REGION
REPLY TERM TO DFS616
MP-BMP ABEND
N
CANCEL JOB, DUMP
Y
WAS DFS616 REC.
N
Y
PERFORM LOG TERMIN-ATION
CORRECT REASON FOR ABEND
3
16
S122 S222 RECEIVED
ENTER VS DUMP TO DUMP IMS CTL
MP-BMP ABEND
Y
N
CANCEL JOB, DUMP
DETERMINE & RESOLVE REASON FOR CANCEL
PERFORM LOG TERMIN-ATION
3

PROCEDURE 13. IMS/VS CONTROL REGION WAIT STATE OR LOOP

This procedure should be entered when the IMS/VS Control Region appears to be in a wait state, a loop, or fails to respond to master terminal commands.

1) VS RESPOND TO CONSOLE

   If OS/VS will not respond to the OS/VS console, it would appear the problem exists within OS/VS, therefore, procedure 10 should be followed.

2) MODIFY IMS DUMP

   Modify IMS/VS with a dump from the system console. This should cause an ABEND of the IMS/VS Control Region

3) DFS628 ABNORMAL TERMINATION SCHEDULED
   DFS629 IMS CTL REGION ABEND

   Both of these messages should be received on the system console and the master terminal. If both messages appear the Control Region has been terminated and procedure 14 should be followed.

4) MASTER TERMINAL ACTIVE

   If the master terminal is active the Master Terminal Operator should ask for the next message until all messages are received.

5) DFS629 IMS CTL REGION ABEND

   When all the messages have been received this message will probably have been received. If so procedure 14 should be followed.

6) DUMP IMS

   The System Operator should obtain a dynamic or standalone dump of IMS/VS. If a standalone dump was taken a re-IPL is necessary.

   Perform the LOG TERMINATION procedure to insure correct closing of the log tape. Under these conditions IMS/VS would not have closed the log tape.

   When the problem is resolved IMS/VS may be restarted by EMERGENCY RESTART. Enter procedure 3.

PROCEDURE 14. IMS/VS CONTROL REGION ABEND

This procedure should be entered whenever the IMS/VS Control Region ABENDS and message DFS629 is received on the system console.

1) DFS616 SYSTEM LOG DATA SET NOT CLOSED

This message indicates IMS/VS was unable to flush the log buffers and close the data set.

The System Operator should enter the OS/VS DUMP command to obtain a dump of the IMS/VS Control Region. This dump must be used in the LOG TERMINATION procedure.

After the dump completes, the System Operator should reply TERM to the DFS616. This allows IMS/VS to complete its abnormal termination.

2) MP-BMP ABEND

All active MPP and BMP regions should abend with a U002 ABEND. Any regions remaining active should be terminated by entering a C jobname, DUMP for each from the system console.

3) DFS616 SYSTEM LOG DATA SET NOT CLOSED

If this message was received and a dump taken, perform the LOG TERMINATION procedure to close the log tape.

4) CORRECT REASON FOR ABEND

Any IMS/VS and/or OS/VS messages should be analyzed and corrective action taken. Once corrected, IMS/VS may be restarted by EMERGENCY RESTART. Enter procedure 3.

PROCEDURE 15. IMS/VS CONTROL REGION TERMINATED BY A MODIFY COMMAND

If IMS/VS is terminated by an OS/VS 'MODIFY IMS' command message DFS628 will be received on the system console and the master terminal.

Once the MODIFY command has been entered the System Operator should wait for the DFS628 before proceeding. If the message is not received, enter procedure 13.

PROCEDURE 16. IMS/VS CONTROL REGION TERMINATED BY A CANCEL COMMAND

This procedure should be entered when IMS/VS is terminated by an OS/VS cancel command and a S122 or S222 ABEND results.

1) S122 OR S222 IS RECEIVED.

When a CANCEL command terminates IMS/VS, MVS (E)STAE and VS 1.4+ STAE TERM=YES systems will close the log tape. Other systems will not, therefore, it is recommended that IMS/VS be operated as a systems task in those environments and terminated with the VS MODIFY command.

The System Operator should enter the OS/VS DUMP command to obtain a dump of the IMS/VS Control Region. This dump may be used for problem determination and for LOG TERMINATION if required.

2) MP-BMP ABEND.

All active MPP and BMP regions should abend with a U002 ABEND. Any regions remaining active should be terminated by entering a C jobname, DUMP command for each from the system console.

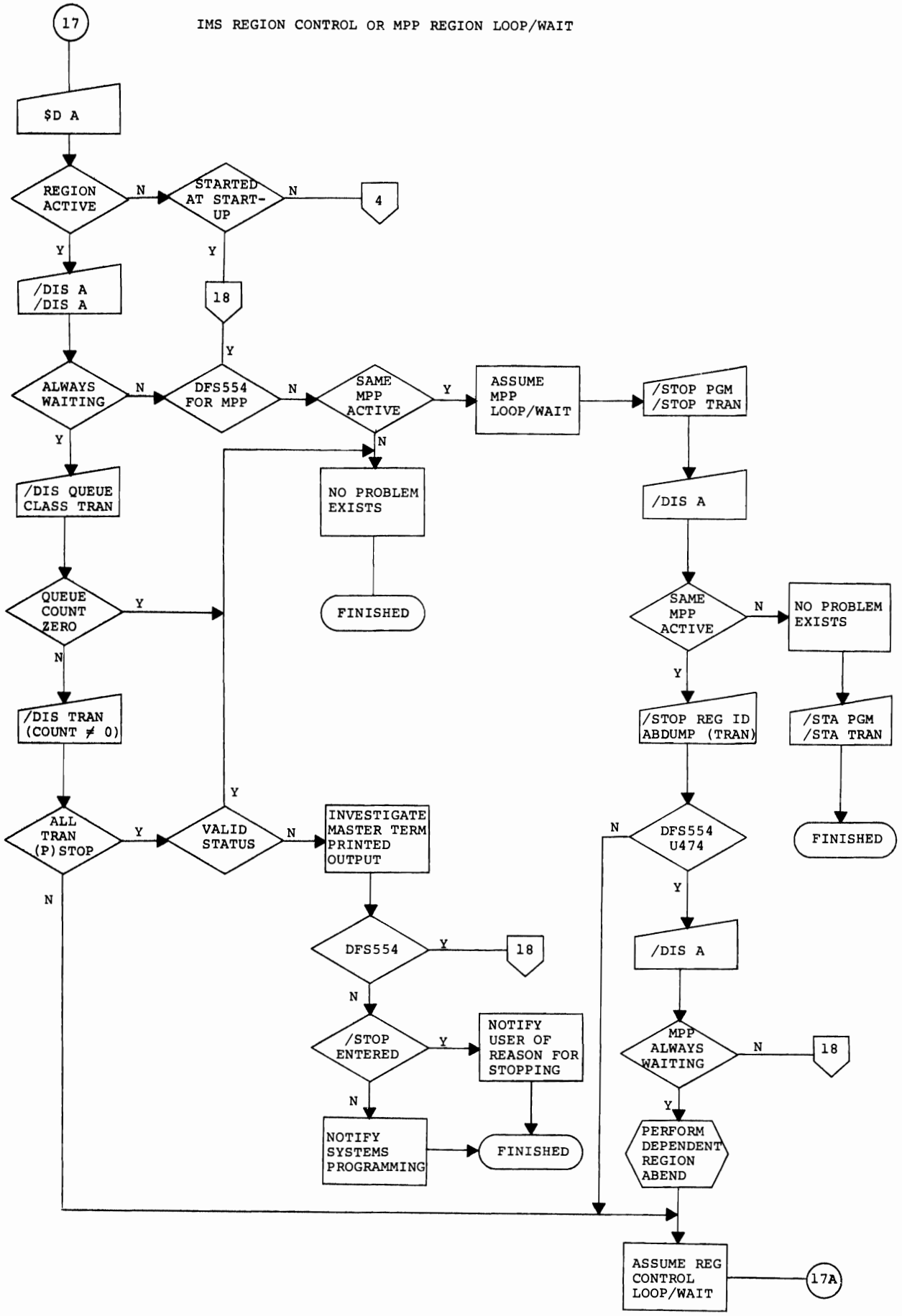
3) DETERMINE AND RESOLVE REASON FOR CANCEL

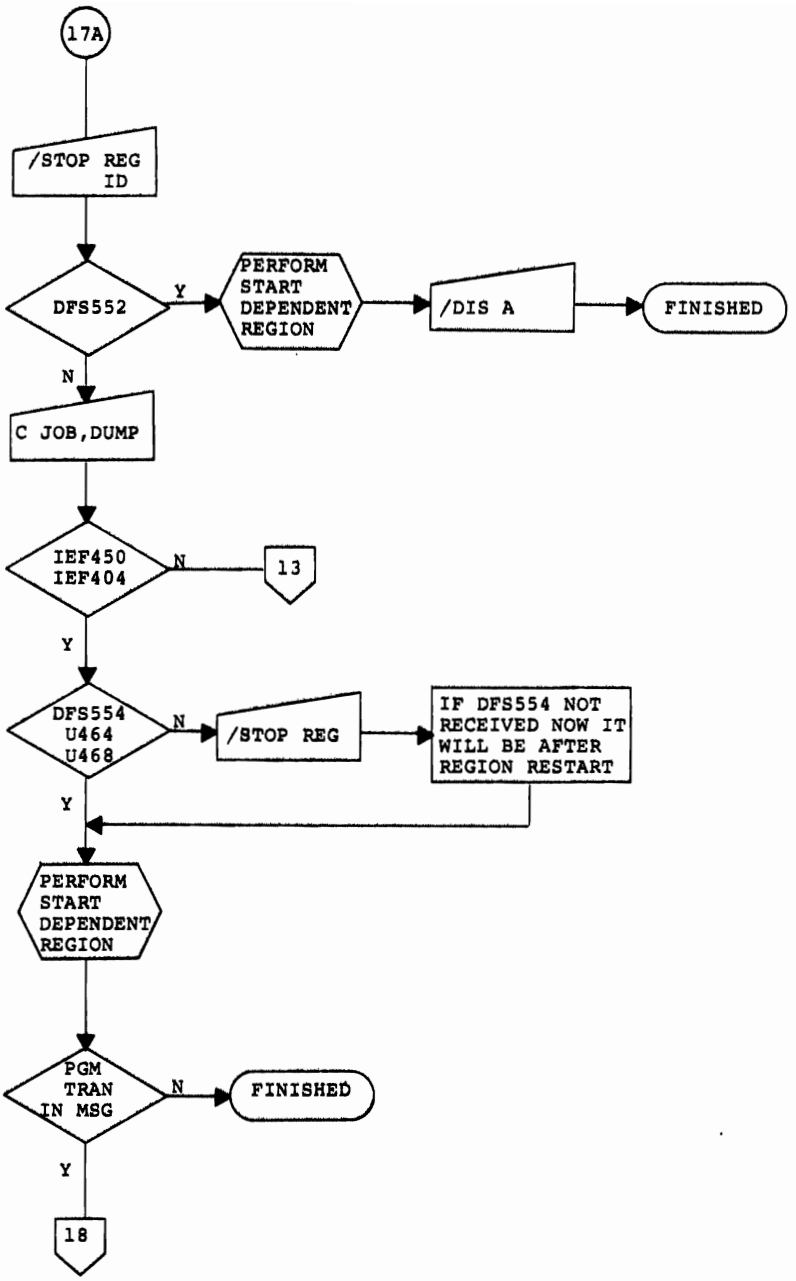
Any IMS/VS and/or OS/VS messages should be analyzed and corrective action taken.

4) PERFORM LOG TERMINATION

If the operating system did not allow the IMS/VS STAE routines to close the log or if message DFS616 was received, the operator must close the log data set. Perform the LOG TERMINATION procedure to do this.

Once the reason for CANCELing IMS/VS is corrected, IMS/VS may be restarted by EMERGENCY RESTART. Enter procedure 3.


17
$D A
REGION ACTIVE
N
STARTED AT START-UP
N
4
Y
Y
/DIS A
/DIS A
18
Y
ALWAYS WAITING
N
DFS554 FOR MPP
N
SAME MPP ACTIVE
Y
ASSUME MPP LOOP/WAIT
/STOP PGM
/STOP TRAN
Y
N
/DIS QUEUE CLASS TRAN
NO PROBLEM EXISTS
/DIS A
QUEUE COUNT ZERO
Y
FINISHED
SAME MPP ACTIVE
N
NO PROBLEM EXISTS
N
Y
/DIS TRAN (COUNT ≠ 0)
/STOP REG ID ABDUMP (TRAN)
/STA PGM
/STA TRAN
Y
ALL TRAN (P)STOP
Y
VALID STATUS
N
INVESTIGATE MASTER TERM PRINTED OUTPUT
N
DFS554 U474
FINISHED
N
Y
DFS554
Y
18
/DIS A
N
/STOP ENTERED
Y
NOTIFY USER OF REASON FOR STOPPING
MPP ALWAYS WAITING
N
18
N
Y
NOTIFY SYSTEMS PROGRAMMING
FINISHED
PERFORM DEPENDENT REGION ABEND
ASSUME REG CONTROL LOOP/WAIT
17A


17A
/STOP REG ID
DFS552
Y
PERFORM START DEPENDENT REGION
/DIS A
FINISHED
N
C JOB,DUMP
IEF450 IEF404
N
13
Y
DFS554 U464 U468
N
/STOP REG
IF DFS554 NOT RECEIVED NOW IT WILL BE AFTER REGION RESTART
Y
PERFORM START DEPENDENT REGION
PGM TRAN IN MSG
N
FINISHED
Y
18

PROCEDURE 17. IMS REGION CONTROLLER or MPP REGION LOOP OR WAIT STATE

This procedure should be entered when a dependent region appears to cease functioning. This might be indicated by a message queue build up of certain classes of transactions or no response at the user terminal to certain input transactions.

1) $D A

Entering this command from the system console will cause a display of active jobs in the system.

2) REGION ACTIVE

If the region in question is not active and was not started when IMS/VS was initialized, enter procedure 4 to start the region. If the region was started but is no longer active it must have abended, therefore, procedure 18 should be entered.

3) /DIS A

Enter several /DIS A commands several seconds apart. This provides a comparison of region activity over a period of time.

4) ALWAYS WAITING

If the region in question is always waiting go to 10.

If the region is not waiting it is possible an application program abended. If a DFS554 for that program can be found on the master terminal, enter procedure 18.

If the region is active and processing different programs and/or transactions there is no problem. If the same program is always active, assume the program is in a loop or wait state.

5) /STOP PGM - /STOP TRAN

Under normal conditions stopping the program and all transactions it can process should shortly terminate the program.

Enter a /DIS A to see the result of the /STOP command.

6) SAME MPP ACTIVE

If the program does terminate there is no problem. The program and transactions should be again started.

7) /STOP REG id ABDUMP trancode

Attempt to abend the program by entering this command.

The id and trancode code can be obtained from the /DIS A response.

8) DFS554 and U474

If this message is not received the program can not be abended therefore, assume the Region Controller is in a loop or wait state. Go to 16.

If the program does abend, enter a /DIS A.

9) MPP ALWAYS WAITING

If the /DIS A shows the region not to be active, enter procedure 18 to make sure the abended program is backed out if necessary.

If the region is still active, perform procedure 18 to make sure the abended program is backed out if necessary, then assume the Region Controller is in a loop or wait state. Go to 16.

10) /DIS QUEUE CLASS xx . . . TRAN

This command will display the counts of messages for the requested classes.

11) QUEUE COUNT ZERO

If the value under the heading MSG COUNT is zero for all the requested classes there are no transactions for that region to process, therefore, no problem exists for the region.

12) /DIS TRAN t1 ..tx

Enter this command specifying a transaction for each non-zero value in the response to the /DIS QUEUE entered above. This displays the status of the transactions associated with the non-zero queue counts.

13) ALL TRAN (P)STOP

If all the transactions are <u>not</u> STOPPED or PSTOPPED assume the Region Controller is in a loop or wait state.

If all transactions are STOPPED or PSTOPPED and the status is valid, there is no problem with the region.

If the status is <u>not</u> valid, check the master terminal output for DFS554 messages or /STOP commands.

14) DFS554

This message was produced when a program abended and the transaction was STOPPED. If found, enter procedure 18 to make sure the program was backed out correctly and make any corrections to the program before restarting it.

15) /STOP ENTERED

If the /STOP TRAN command was entered, notify the terminal user of the reason for stopping.

If the transaction is stopped for no apparent reason, notify Systems Programming.

ASSUME REGION CONTROLLER LOOP OR WAIT STATE

16) /STOP REGION

This is an attempt to stop the region in question. The stop will be effective when the region becomes inactive.

17) DFS552 MESSAGE PROCESSING REGION STOPPED

If this message is received the region could not have been in a loop or wait state.

The operator should then perform procedure 4 to start a region to replace the stopped region.

A display active will verify the presence of the region and the class assignments. Exit at this point since the region is functioning.

18) C JOB,DUMP

Since the region won't stop, attempt to cancel it.

Note: IMS/VS 1.1.1 and later users should use the /STOP REGION CANCEL to perform this function if a previous /STOP REGION ABDUMP has been issued for this specific region.

19) IEF450 - IEF404

If these messages are not received the region won't cancel. This could indicate a loop or wait state in the IMS/VS Control Region. Enter procedure 13 to determine if this is the case.

20) DFS554 - U464 or U468

If this message is not received, enter a /STOP REGION id command. This may cause the DFS554 message to be received now. If not, the message will be received after the region is re-started.

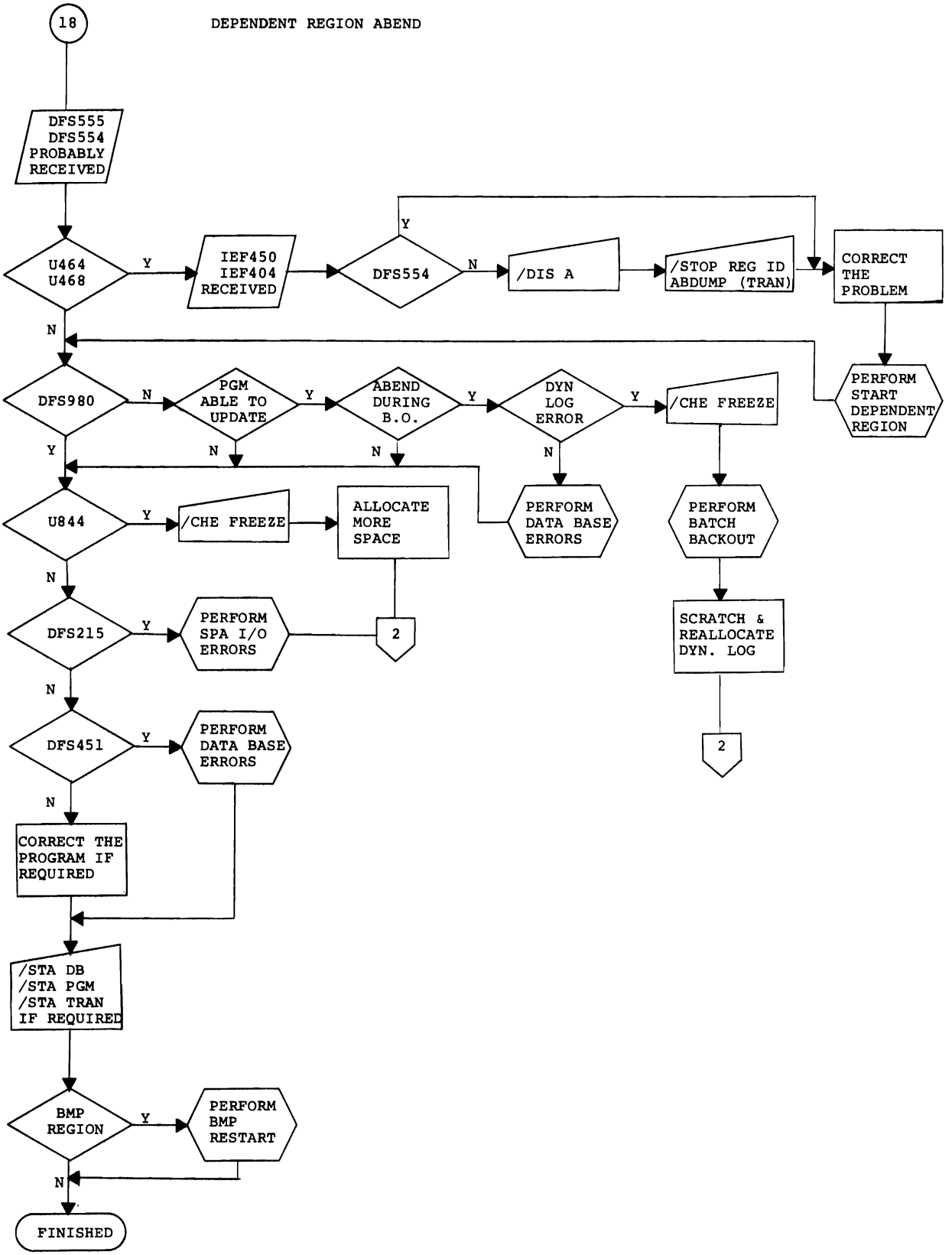
21) PERFORM START DEPENDENT REGION

Enter procedure 4 to start a new dependent region to replace the one cancelled above.

22) PGM TRAN IN MSG

Refer back to the DFS554 message which was received. If the program name and the transaction code fields were filled in the message, an application program was abended. Enter procedure 18 to complete this situation.

If the fields are not filled in, an application program abend did not occur. The region has now been restarted so this procedure is finished.


18
DEPENDENT REGION ABEND
DFS555
DFS554
PROBABLY
RECEIVED
U464
U468
Y
IEF450
IEF404
RECEIVED
DFS554
Y
N
/DIS A
/STOP REG ID
ABDUMP (TRAN)
CORRECT
THE
PROBLEM
N
DFS980
N
PGM
ABLE TO
UPDATE
Y
ABEND
DURING
B.O.
Y
DYN
LOG
ERROR
Y
/CHE FREEZE
PERFORM
START
DEPENDENT
REGION
Y
N
N
N
U844
Y
/CHE FREEZE
ALLOCATE
MORE
SPACE
PERFORM
DATA BASE
ERRORS
PERFORM
BATCH
BACKOUT
N
DFS215
Y
PERFORM
SPA I/O
ERRORS
2
SCRATCH &
REALLOCATE
DYN. LOG
N
DFS451
Y
PERFORM
DATA BASE
ERRORS
2
N
CORRECT THE
PROGRAM IF
REQUIRED
/STA DB
/STA PGM
/STA TRAN
IF REQUIRED
BMP
REGION
Y
PERFORM
BMP
RESTART
N
FINISHED

PROCEDURE 18. DEPENDENT REGION ABEND

The procedure should be entered whenever a dependent region is cancelled, abends, or a DFS554 message is received.

1) U464 - U468 DEPENDENT REGION CANCELLED

If the abend was one of the above the dependent region has been cancelled and the OS/VS messages IEF450 and IEF404 will be received on the system console.

A DFS554 would have been received if the IMS/VS Control Region realized the dependent region had been cancelled. If no DFS554 is received, enter a /DIS A command from the master terminal to display the dependent regions known to IMS/VS.

The cancelled region must be abended. Enter a /STOP REGION id ABDUMP tran command to abend the region and notify the IMS/VS Control Region of the abend.

Correct the reason for the cancellation of the dependent region.

Perform procedure 4 START DEPENDENT REGION to start a region to replace the cancelled one.

2) DFS980 BACKOUT COMPLETE FOR PSB psbname

This message should be received whenever a user program has updated a data base and was abended for any reason. If the program is inquiry only, no dynamic backout will be performed.

If an abend occurs during backout the cause could be a Dynamic Log error or Data Base error.

A Dynamic Log I/O error results in message DFS981 I/O ERROR ON DYNAMIC LOG WHILE BACKING OUT .....

A Data Base I/O error results in message DFS983 I/O ERROR IN DATA BASE dbdname ..... Followed by DFS981 DATA BASE dbdname HAS BEEN STOPPED.

If neither message is received it may be assumed the abend occured before any updating was done therefore backout was unnecessary.

If the error is a data base I/O error (DFS983 and DFS981) perform the DATA BASE ERRORS procedure to correct the problem.

If the error is a Dynamic Log I/O error, enter a /CHECKPOINT FREEZE command to terminate IMS/VS.

Perform the BATCH BACKOUT procedure specifying the PSB and data base(s) listed in the DFS981 message.

Scratch and reallocate the Dynamic Log data set.

Enter the WARM START procedure to restart IMS/VS and specify /NRE FORMAT DL in that procedure. The data base, program, and transaction may be restarted after the WARM START. It may be necessary to correct program errors prior to restart.

3) U844 NO AVAILABLE SPACE IN DATASET

Enter a /CHECKPOINT FREEZE command to terminate IMS/VS. OSAM data sets may be extended by supplying additional volumes and changing the JCL. VSAM data sets require an AMS execution to add volumes and JCL changes. Enter the WARM START procedure to restart IMS/VS.

4) DFS215 WORK AREA I/O ERROR, CONVERSATION TERMINATED

An I/O error occured on a SPA. To prevent future conversation forced terminations the SPA should be scratched and reallocated. Enter the SPA I/O ERRORS procedure to accomplish this.

An application program is presented an X1 status code when a SPA I/O error occurs. That program in turn abended or issued a ROLL call thus accounting for the dependent region abend.

5) DFS451 dbdname STOPPED - I/O ERROR ddname

A read I/O error occured on the named data base, the application program was presented an AO status code, and in turn abended or issued a ROLL call. This accounts for the dependent region abend. Perform the DATA BASE ERRORS procedure to recover the data base.

6) CORRECT PROGRAM IF REQUIRED

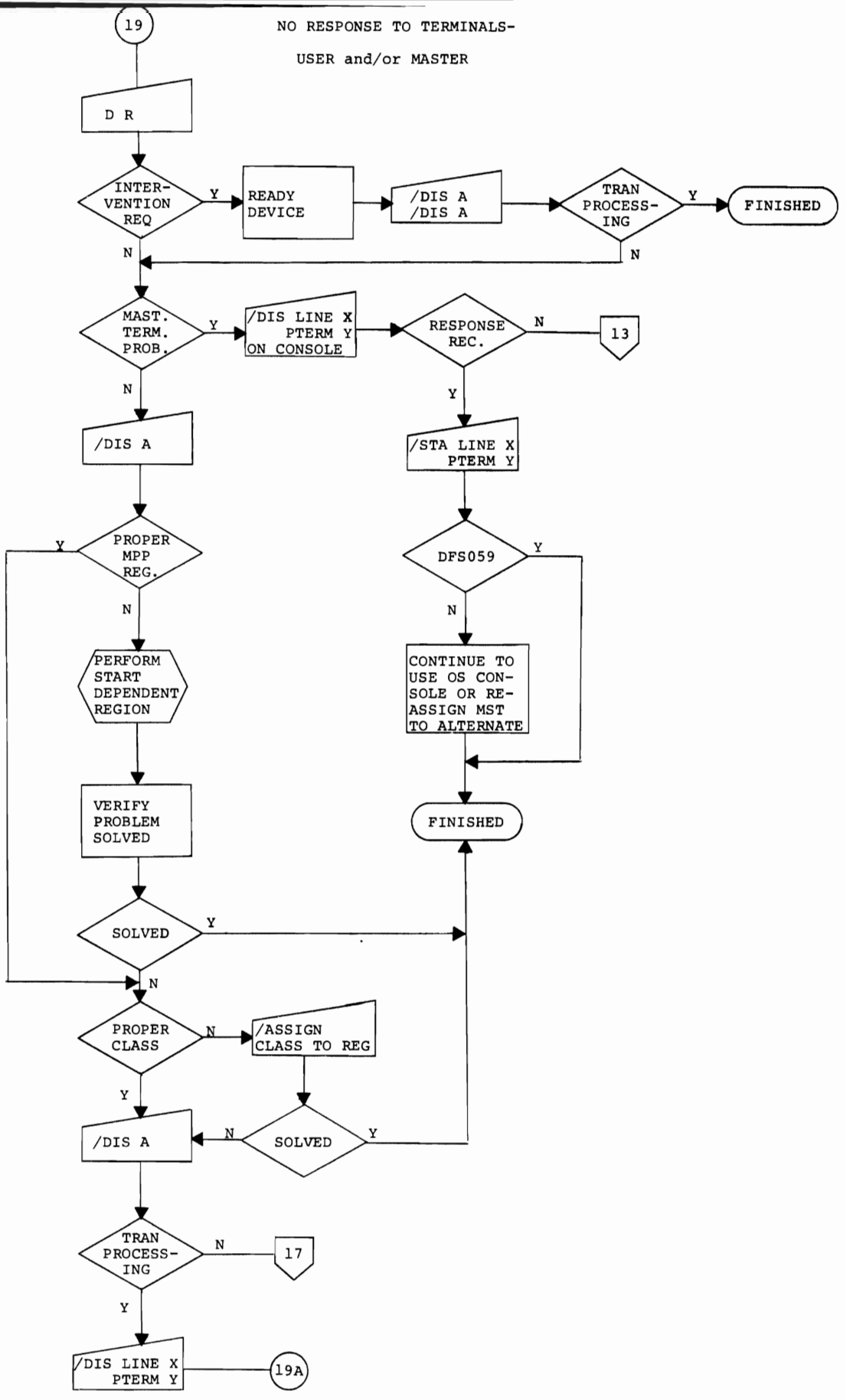
If the Abend was caused by faulty program logic the program must be corrected befcre restarting it. Failure to do so may result in repeated abends.

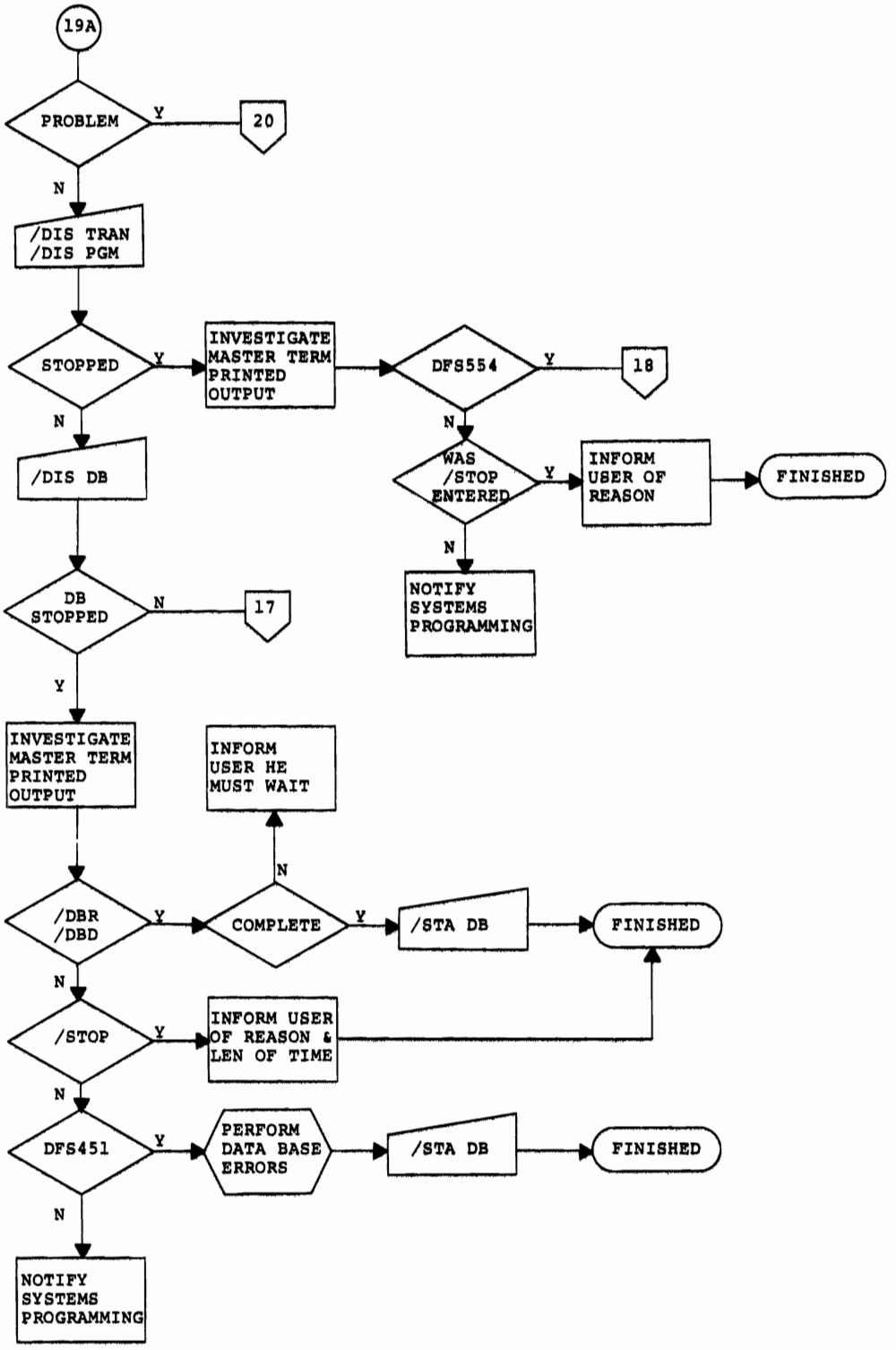
7) /START DB, PGM, TRAN

Depending upon the nature of the abend any or all of these resources must be restarted.

8) BMP REGION

Enter Procedure 5 BMP RESTART to restart BMP regions. MPP regions will automatically be rescheduled.


19
NO RESPONSE TO TERMINALS-
USER and/or MASTER
D R
INTER-VENTION REQ
Y
READY DEVICE
/DIS A /DIS A
TRAN PROCESS-ING
Y
FINISHED
N
N
MAST. TERM. PROB.
Y
/DIS LINE X PTERM Y ON CONSOLE
RESPONSE REC.
N
13
N
Y
/DIS A
/STA LINE X PTERM Y
PROPER MPP REG.
Y
DFS059
Y
N
N
PERFORM START DEPENDENT REGION
CONTINUE TO USE OS CON-SOLE OR RE-ASSIGN MST TO ALTERNATE
VERIFY PROBLEM SOLVED
FINISHED
SOLVED
Y
N
PROPER CLASS
N
/ASSIGN CLASS TO REG
Y
/DIS A
N
SOLVED
Y
TRAN PROCESS-ING
N
17
Y
/DIS LINE X PTERM Y
19A


19A
PROBLEM
Y
20
N
/DIS TRAN
/DIS PGM
STOPPED
Y
INVESTIGATE MASTER TERM PRINTED OUTPUT
DFS554
Y
18
N
N
/DIS DB
WAS /STOP ENTERED
Y
INFORM USER OF REASON
FINISHED
N
NOTIFY SYSTEMS PROGRAMMING
DB STOPPED
N
17
Y
INVESTIGATE MASTER TERM PRINTED OUTPUT
INFORM USER HE MUST WAIT
N
/DBR /DBD
Y
COMPLETE
Y
/STA DB
FINISHED
N
/STOP
Y
INFORM USER OF REASON & LEN OF TIME
N
DFS451
Y
PERFORM DATA BASE ERRORS
/STA DB
FINISHED
N
NOTIFY SYSTEMS PROGRAMMING

PROCEDURE 19. NO RESPONSE TO TERMINALS

This procedure should be entered when users complain that IMS/VS is not responding to their terminals, it appears IMS/VS is not responding to the Master Terminal, or the entire system appears to be in a wait state.

1) D R

Enter the DISPLAY REQUEST command from the system console. This will display any outstanding requests applicable to IMS/VS.

2) INTERVENTION REQ

If any Intervention Required requests are displayed perform the functions necessary to ready the device. If the device is the log tape this could account for the IMS/VS wait state situation.

Enter several /DIS A commands several seconds apart. A comparison of the output will indicate whether or not transactions are being processed. Look for different PSBs and transactions. If the system is now functioning this procedure is finished.

3) MASTER TERMINAL PROBLEM

If the problem is no response at the Master Terminal, from the system console enter a /DIS LINE x PTERM y command specifying the master terminal line and physical terminal.

If no response is received, the IMS/VS Control Region is probably in a loop or wait state. Enter procedure 13 IMS CONTROL REGION WAIT/LOCP to handle that situation.

Enter a /START LINE x PTERM y command from the system console which specifies the master terminal line and physical terminal.

If the DFS059 TERMINAL STARTED message is received at the Master Terminal, the terminal is now active and this procedure is finished.

If the message is not received, the operator may continue to use the OS/VS console as the Master Terminal or he may reassign the master LTERM to another device. He might also check the power source, any switches, or the cables connected to the terminal.

4) /DIS A

This command will display the status of the dependent regions.

5) PROPER MPP REGIONS

If the proper dependent regions are not displayed, perform procedure 4 START DEPENDENT REGIONS to start any missing regions.

Enter a /DIS A to verify that the region(s) was started and is processing transactions.

If this solves the problem this procedure is finished.

6) PROPER CLASS

If the classes assigned to each region are not correct, enter the /ASSIGN CLASS x TO REGION x command to assign the proper classes to the regions.

If this solves the problem and the transactions begin to process, this procedure is finished.

7) /DIS A

Enter several of these commands several seconds apart.

8) TRANS BEING PROCESSED

If the displays show no transactions are being processed, enter procedure 17 IMS REGION CONTROL OR MPP REGION LOOP/WAIT since IMS/VS appears to be in a loop or wait state.

9) /DIS LINE x PTERM y

This command will display the status of the user terminal.

10) PROBLEM

If the display shows the terminal to be in an exceptional status such as stopped, inoperative or anything of that nature enter procedure 20 TERMINAL PROBLEMS to correct the problem.

11) /DIS TRAN - /DIS PGM

These two commands will display the status of the transaction and the program. If either of these are stopped, investigate the master terminal output. If a DFS554 message is found for the program enter procedure 18 DEPENDENT REGION ABEND to insure proper backout and problem determination.

If a /STOP has been entered inform the user of the fact, why, and when he may expect service to resume.

If no /STOP was entered notify Systems Programming to resolve the problem.

12) /DIS DB

The data bases named in this command should be those used by the program in question.

13) DB STOPPED

If the data bases are not stopped, enter procedure 17 IMS REGION CONTROL OR MPP REGION LOOP/WAIT. Since the lines, terminals, transactions, programs, and data bases are all active the problem is probably an IMS/VS loop or wait state.

If a stopped data base is found, look at the master terminal output to determine why.

14) /DBR - /DBD

If the data base is being Image Copied or recovered and not yet completed, inform the user he must wait. When the function is completed issue a /START DATABASE command and this procedure is finished.

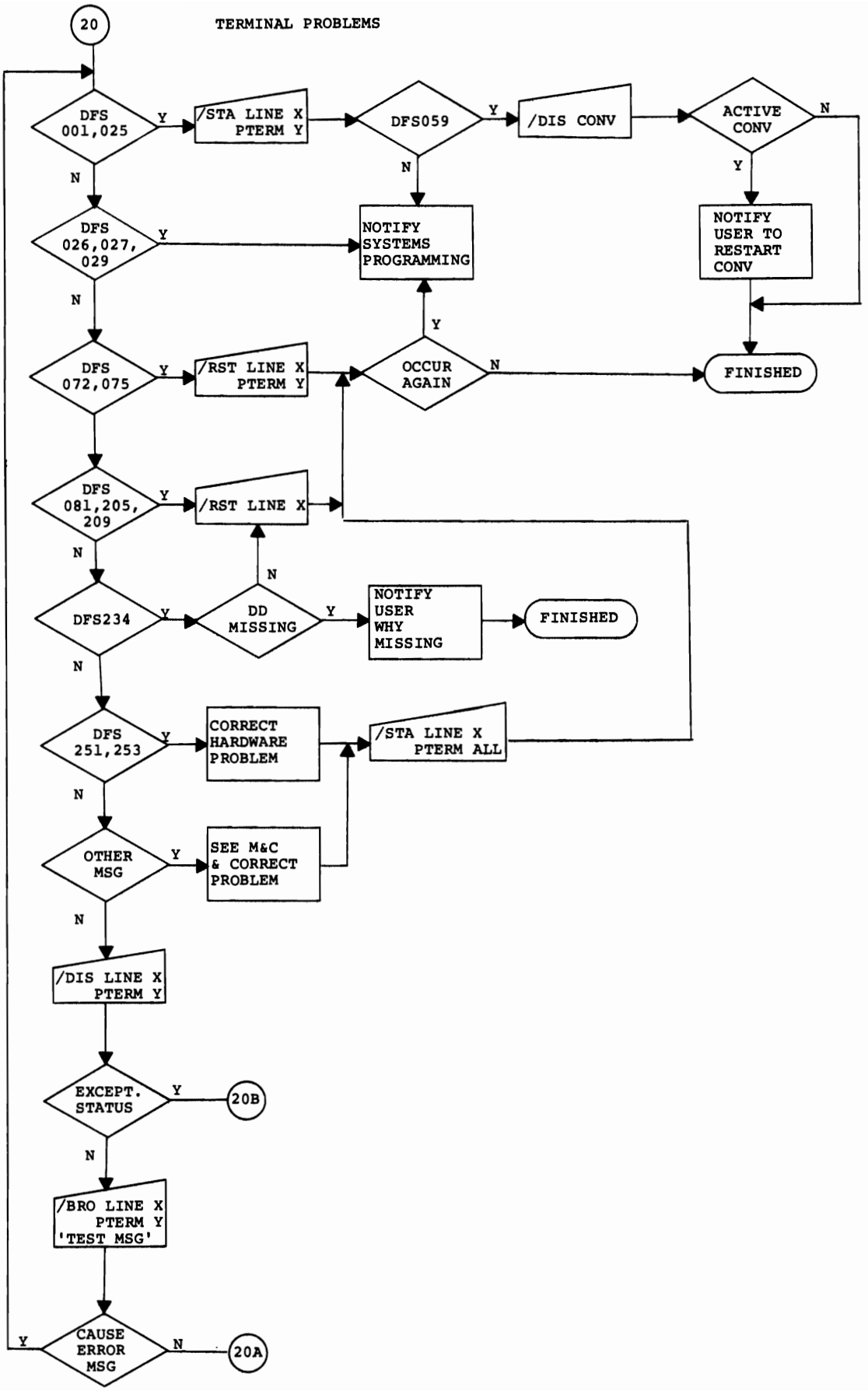
15) /STOP

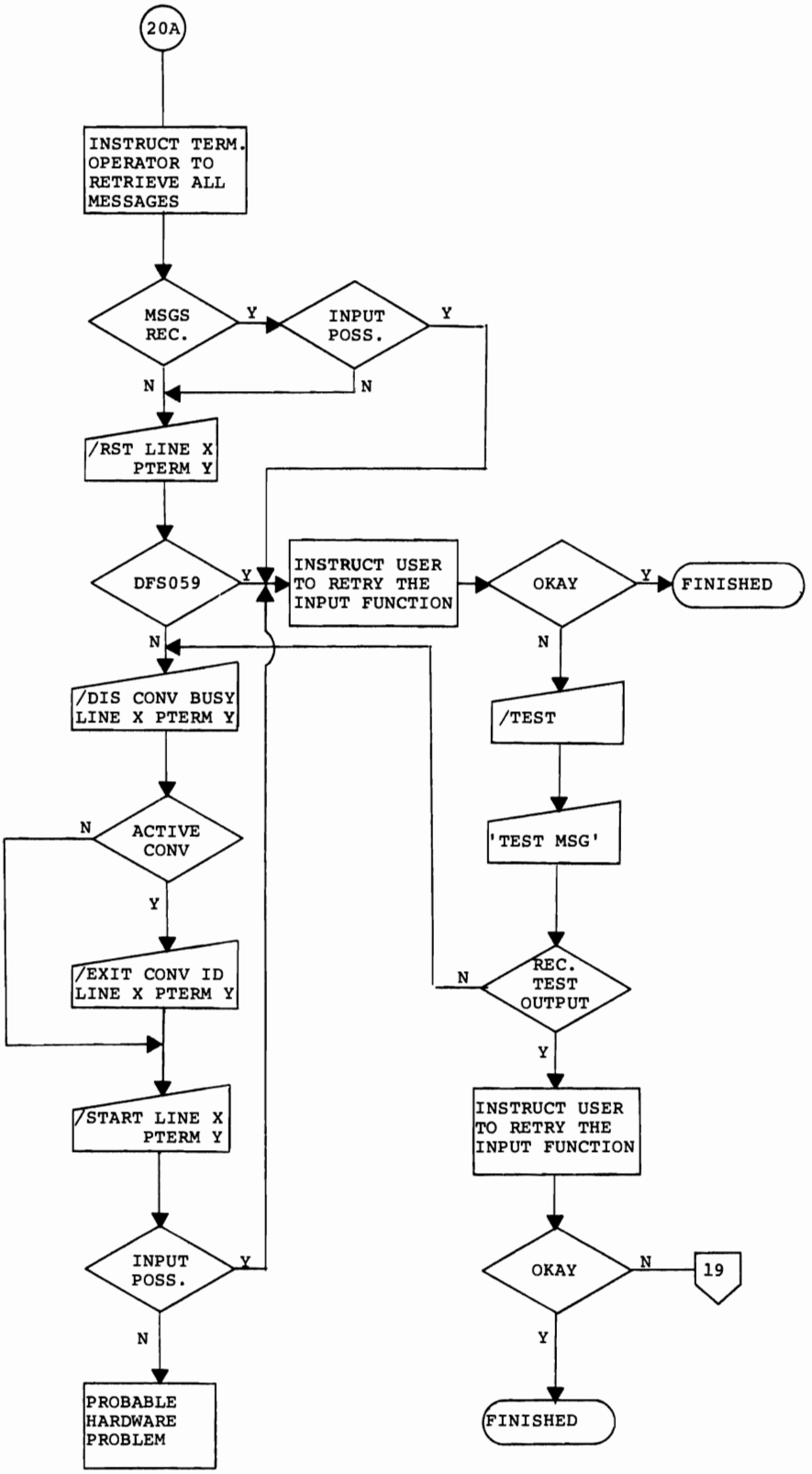
If the data base has been stopped by a command inform the user of the reason and the length of time it will remain stopped.

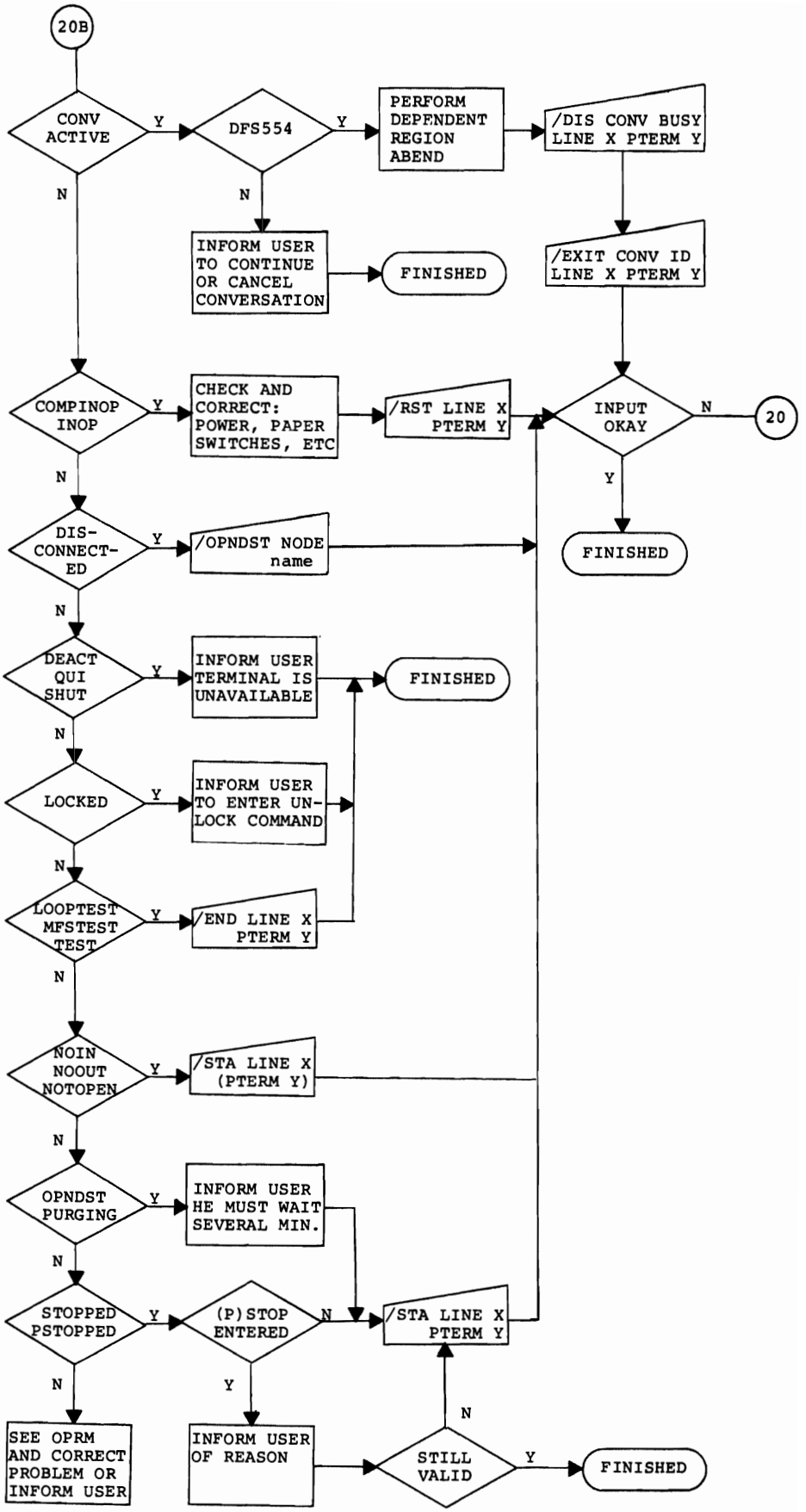
16) DFS451

If a DFS451 message is found an I/O error occured on the named data base. Perform procedure 40 DATA BASE ERRORS and then enter a /START DATABASE command to make the data base usable.

If no DFS451 can be located notify Systems Programming to determine why the data base was stopped.


20
DFS 001,025
Y
/STA LINE X PTERM Y
DFS059
Y
/DIS CONV
ACTIVE CONV
N
Y
N
DFS 026,027, 029
Y
NOTIFY SYSTEMS PROGRAMMING
NOTIFY USER TO RESTART CONV
N
Y
DFS 072,075
Y
/RST LINE X PTERM Y
OCCUR AGAIN
N
FINISHED
DFS 081,205, 209
Y
/RST LINE X
N
DFS234
Y
DD MISSING
Y
NOTIFY USER WHY MISSING
FINISHED
N
DFS 251,253
Y
CORRECT HARDWARE PROBLEM
/STA LINE X PTERM ALL
N
OTHER MSG
Y
SEE M&C & CORRECT PROBLEM
N
/DIS LINE X PTERM Y
EXCEPT. STATUS
Y
20B
N
/BRO LINE X PTERM Y 'TEST MSG'
CAUSE ERROR MSG
Y
N
20A


20A
INSTRUCT TERM. OPERATOR TO RETRIEVE ALL MESSAGES
MSGS REC.
Y
INPUT POSS.
Y
N
N
/RST LINE X PTERM Y
DFS059
Y
INSTRUCT USER TO RETRY THE INPUT FUNCTION
OKAY
Y
FINISHED
N
N
/DIS CONV BUSY LINE X PTERM Y
/TEST
ACTIVE CONV
N
'TEST MSG'
Y
/EXIT CONV ID LINE X PTERM Y
REC. TEST OUTPUT
N
Y
/START LINE X PTERM Y
INSTRUCT USER TO RETRY THE INPUT FUNCTION
INPUT POSS.
Y
OKAY
N
19
N
Y
PROBABLE HARDWARE PROBLEM
FINISHED


20B
CONV ACTIVE
Y
DFS554
Y
PERFORM DEPENDENT REGION ABEND
/DIS CONV BUSY LINE X PTERM Y
N
N
INFORM USER TO CONTINUE OR CANCEL CONVERSATION
FINISHED
/EXIT CONV ID LINE X PTERM Y
COMPINOP INOP
Y
CHECK AND CORRECT: POWER, PAPER SWITCHES, ETC
/RST LINE X PTERM Y
INPUT OKAY
N
20
Y
N
DIS-CONNECT-ED
Y
/OPNDST NODE name
FINISHED
N
DEACT QUI SHUT
Y
INFORM USER TERMINAL IS UNAVAILABLE
FINISHED
N
LOCKED
Y
INFORM USER TO ENTER UN-LOCK COMMAND
N
LOOPTEST MFSTEST TEST
Y
/END LINE X PTERM Y
N
NOIN NOOUT NOTOPEN
Y
/STA LINE X (PTERM Y)
N
OPNDST PURGING
Y
INFORM USER HE MUST WAIT SEVERAL MIN.
N
STOPPED PSTOPPED
Y
(P)STOP ENTERED
N
/STA LINE X PTERM Y
N
Y
N
SEE OPRM AND CORRECT PROBLEM OR INFORM USER
INFORM USER OF REASON
STILL VALID
Y
FINISHED

PROCEDURE 20 TERMINAL PROBLEMS

This procedure should be entered when terminal malfunctions occur. These malfunctions may be terminal hardware errors, remote transmission control unit problems, the lines or terminals are stopped, or the terminal operator complains of no service and procedure 19 has determined the problem resides with the terminal.

This procedure assumes voice contact between the Remote Terminal Operator and the Master Terminal Operator.

When notified of terminal problems the Master Terminal Operator should investigate the master terminal output for error messages relative to the line and physical terminal. It might be necessary to enter a /DIS ASMT LTERM command to determine the location of the remote terminal if only the LTERM name is known.

1) DFS001 UNDETERMINED ERROR ON 3270 LINE x PTERM y
DFS025 UNABLE TO FIND INPUTTING TERMINAL

When either of these messages is received the line has been stopped. The Master Terminal Operator may attempt to restart the lines using a /START LINE x PTERM ALL command.

If the remote terminal is successfully restarted it will receive a DFS059 TERMINAL STARTED message. If the terminal does not restart notify Systems Programming for assistance.

When a terminal is successfully restarted the Master Terminal Operator should enter a /DIS CONV command. If the terminal had an active conversation the user should be notified to restart the conversation using a /HOLD and /RELEASE commmand to reformat the screen. The Remote Terminal Operator may then continue normal operations.

2) DFS026 UNABLE TO FIND INPUTTING TERMINAL
DFS027 I/O ROUTINE ERROR
DFS029 DDM LOGIC ERROR DETECTED

For these error conditions IMS/VS must be shut down and restarted before the line or terminal can be reactivated. Inform the user of the situation.

3) DFS072 UNABLE TO OUTPUT LINE x PTERM y
DFS075 UNABLE TO RECEIVE LINE x PTERM y

Either of these messages indicates the named physical terminal has been stopped. An attempt may be initiated to restart the terminal via a /RESTART LINE x PTERM y command. It the error is repetitive and the Remote Operator has checked the terminal for power on, paper loaded, etc. report the malfunction to the Systems Programming.

4) DFS081 BTAM ERROR 36
DFS205 BTAM ERROR 40
DFS209 BTAM ERROR 44

In each of these cases the line is stopped and the terminal is marked inoperative. The line may be restarted using a /RESTART LINE x command.

5) DFS234 INVALID RC FROM ACCESS METHOD

This error could be caused by a missing DD card or by BTAM. If a DD card is missing notify the user of the reason. Otherwise attempt to restart the line.

6) DFS251 REMOTE CONTROL UNIT INOPERABLE
DFS253 TCU INOPERABLE LINE x PTERM y

These errors are caused by hardware malfunctions which must be corrected. After the corrections the line may be restarted using a /START LINE x PTERM ALL command.

7) OTHER MESSAGES

If other DFS messages are found take any corrective action noted in the Messages and Codes manual. In most cases a terminal or line can be restarted using a /START LINE x PTERM y or ALL command.

8) /DIS LINE x PTERM y

If no DFS messages were found this command will display the status of the remote terminal. For any exceptional status conditions refer to item 16 of this procedure.

9) /BRO LINE x PTERM y THIS IS A TEST MESSAGE

Broadcasting a message to the remote terminal might cause a DFS message to appear on the Master Terminal. If a message does appear take the appropriate corrective action.

Instruct the Remote Terminal Operator to retrieve all messages and look for the test message.

10) MESSAGES RECEIVED

If the remote terminal is able to receive messages instruct the operator to try some form of input. If that is successful, retry the input function.

11) /RESTART LINE x PTERM y

If the remote terminal is unable to receive output or enter input, try to restart the device.

12) DFS059 TERMINAL STARTED

If the remote terminal starts, go retry the input function.

13) /DIS CONV BUSY LINE x PTERM y

If a conversation is active on the terminal this command will display the conversation ID.

If an active conversation is found, the terminal may be awaiting that output.

Enter a /EXIT CONV id LINE x PTERM y to terminate the conversation. Notify the user of this action so he may restart the conversation later.

14) /START LINE x PTERM y

Attempt to restart the remote terminal and instruct the remote operator to attempt some form of input. If that is successful retry the input function.

If the remote terminal is still inoperative the problem is probably a hardware malfunction. Inform the user of the terminal unavailability and report the problem to Systems Programming.

15) INSTRUCT USER TO RETRY

Since the terminal can now both input and receive output, the user should retry his input function. If the expected result is achieved the problem has been solved.

If the user function still fails, instruct the operator to place the terminal in test mode using a /TEST command. After receiving the DFS058 message the operator should enter a test message. The system will respond by sending the message back to the device. Several test messages should be echoed.

If no response is received go to item 13 to check for active conversations.

If the "echo" test is successful, the operator should enter a /END command to take the terminal out of test mode, and retry the input function. A failure at this point would indicate a possible failure within IMS/VS. Enter procedure 19 NO RESPONSE TO TERMINALS to determine the cause.

RESPONSES TO /DIS LINE x PTERM y - EXCEPTIONAL CONDITIONS

16) CONV ACT

A conversation is active on the terminal. If the application program abended message DFS554 and possibly DFS555 would have

appeared on the Master Terminal. The Master Terminal Operator should perform procedure 18 DEPENDENT REGION ABEND to insure proper abend cleanup.

Enter a /DIS CONV BUSY LINE x PTERM y command to obtain the conversation ID. Enter a /EXIT CONV id LINE x PTERM y to terminate the conversation and inform the user that the conversation must be restarted.

If input is still not possible, return to item 1 to check for messages and again display the terminal status.

17) COMPINOP - INOP

A terminal or a component is inoperative. Instruct the remote operator to check power sources, and switches, paper supply, etc.

After any corrections have been made attempt to restart the terminal using a /RESTART LINE x PTERM y command. If the problem is not solved return to item 1.

18) DISCONNECTED

The VTAM node is not connected to IMS/VS. Attempt to make the connection using a /OPNDST NODE command.

19) DEACT - QUI - SHUT

The terminal has been permanently deactivated, VTAM has sent an indicator to suspend output or processing is completed for the node and the VTAM shutdown complete was returned. Inform the user this terminal is unavailable for these reasons.

20) LOCKED

The remote terminal has entered a /LOCK command. Instruct the remote operator to enter a /UNLOCK command and retrieve all output messages.

21) LOOPTEST - MFSTEST - TEST

The terminal has been placed into test mode. When the test is finished enter a /END from the remote terminal or a /END LINE x PTERM y from the Master Terminal.

22) NOIN - NOOUT - NOOPEN

The line or terminal has been stopped for input or output, or the line was not started when IMS/VS was initialized.

Enter a /START LINE x if NOOPEN or a /START LINE x PTERM y for a NOIN or NOOUT condition. 23) OPNDST - PURGING

An OPNDST ACQUIRE is pending or a line, a terminal, or a transaction is purging all of its output.

Inform the user he must wait several minutes. After that time enter a /START LINE x PTERM y command.

24) STOPPED - PSTOPPED

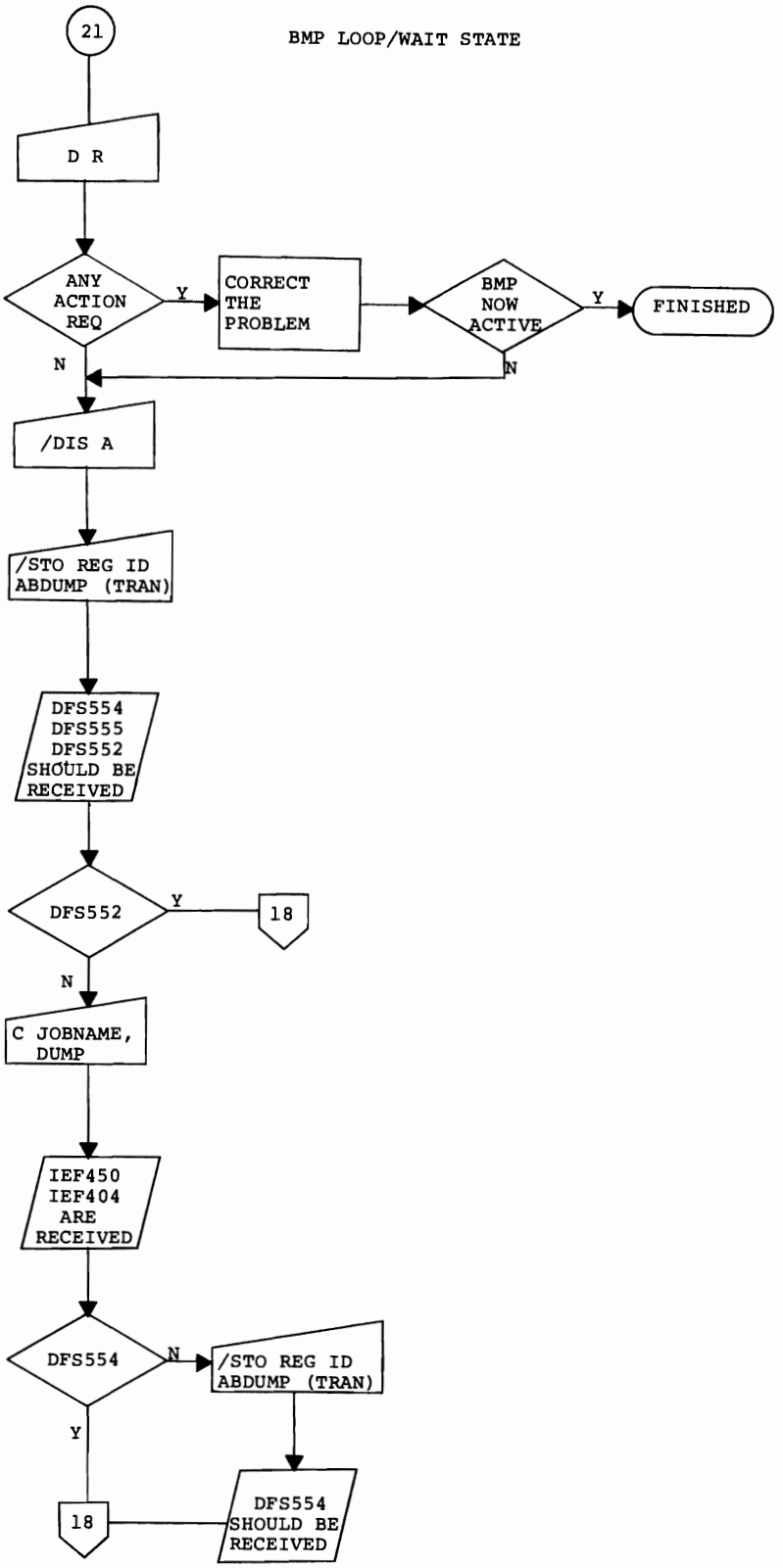
The line or terminal has been stopped or pstopped.

Look for a /STOP or /PSTOP command on the Master Terminal output. If found, determine the reason and inform the user. If the reason is still valid the user may not use the device.

If no command is found or if the stop is no longer valid, enter a /START LINE x PTERM y to restart the terminal.

25) SEE OPRM AND CORRECT PROBLEM

Any other exceptional condition is documented in the Operator's Reference Manual. Take any action necessary to restart the terminal or inform the user and Systems Programming the terminal is unavailable.


21
BMP LOOP/WAIT STATE
D R
ANY ACTION REQ
Y
CORRECT THE PROBLEM
BMP NOW ACTIVE
Y
FINISHED
N
N
/DIS A
/STO REG ID ABDUMP (TRAN)
DFS554 DFS555 DFS552 SHOULD BE RECEIVED
DFS552
Y
18
N
C JOBNAME, DUMP
IEF450 IEF404 ARE RECEIVED
DFS554
N
/STO REG ID ABDUMP (TRAN)
Y
18
DFS554 SHOULD BE RECEIVED

PROCEDURE 21. BMP LOOP/WAIT STATE

This procedure should be entered when a loop or wait state is suspected in a BMP program. This may be indicated by excessive execution time, exceeded lines of output, the BMP appears inactive (no tape movement), or the BMP appears to be in a loop.

1) D R

Enter the DISPLAY REQUEST command from the system console.

2) ANY ACTION REQUIRED

This could be an intervention required for a device used by the BMP or some other request. Satisfy the request and determine if the BMP is now functioning. If so, this procedure is finished.

3) /DIS A

Enter the display active command from the Master Terminal to display the ID of all regions known to IMS/VS.

4) /STOP REGION id ABDUMP (tran)

This command, using the ID from the previous /DIS A, should force the BMP program to abend. Message DFS554 indicating a U474 abend code, and possible DFS555 should be received at the Master Terminal. The BMP region should terminate and message DFS552 should be received at the Master Terminal.

5) DFS552 BATCH PROCESSING REGION STOPPED

If the BMP region did abend, enter procedure 18 DEPENDENT REGION ABEND to ensure proper abend cleanup.

6) C JOBNAME,DUMP

Enter this ccmmand from the system console to cancel the BMP job.

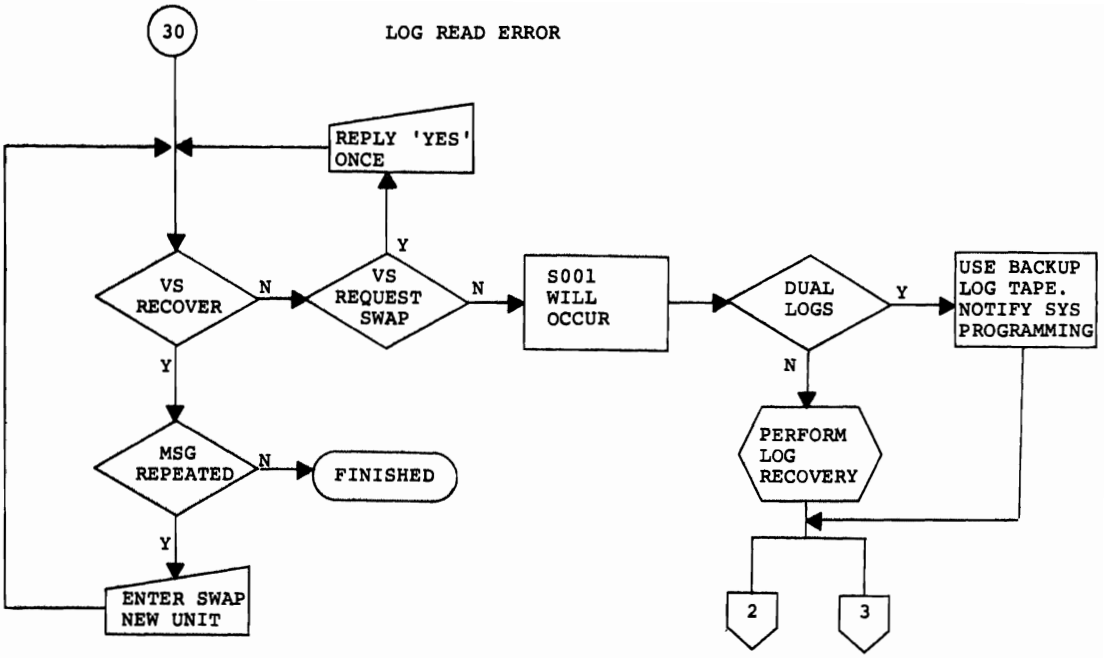
The BMP region will abend and messages IEF450 and IEF404 will be received at the system console.

Note: IMS/VS 1.1.1 and later users should use the /STOP REGION CANCEL to perform this function.

7) DFS554

If this message was received following the above /STOP REGION id ABDUMP (tran), the BMP program has abended. Enter procedure 18 DEPENDENT REGION ABEND.

If the DFS554 was not received, reenter the stop region command. This should force the BMP program to abend and issue a DFS554 message.


30
LOG READ ERROR
REPLY 'YES' ONCE
VS RECOVER
N
VS REQUEST SWAP
N
S001 WILL OCCUR
DUAL LOGS
Y
USE BACKUP LOG TAPE. NOTIFY SYS PROGRAMMING
Y
Y
N
MSG REPEATED
N
FINISHED
PERFORM LOG RECOVERY
Y
ENTER SWAP NEW UNIT
2
3

PROCEDURE 30. LOG READ ERROR

This procedure should be entered whenever a read I/O error occurs on the input system log. A system log is input during restarts, when executing Batch Backout, or when executing the Log Recovery utility.

1) VS RECOVER

OS/VS will attempt to recover from read errors by backspacing and re-reading. If the recovery is successful a message is issued to the system console indicating so.

2) MSG REPEATED

If the recovery messages are issued repeatedly, the tape unit may be malfunctioning. Enter the OS/VS SWAP COMMAND to cause the system to select another tape unit. In most cases this will correct the problem.

3) VS REQUEST SWAP

When OS/VS cannot recover from a read error it will allow the operator to request a different tape unit. A reply of "YES" to the SWAP REQUEST will cause OS/VS to select another tape unit.

If the same error occurs on the second unit, the operator should reply "NO". This will cause the IMS/VS Control Region to abend with a S001 abend.

4) DUAL LOGS

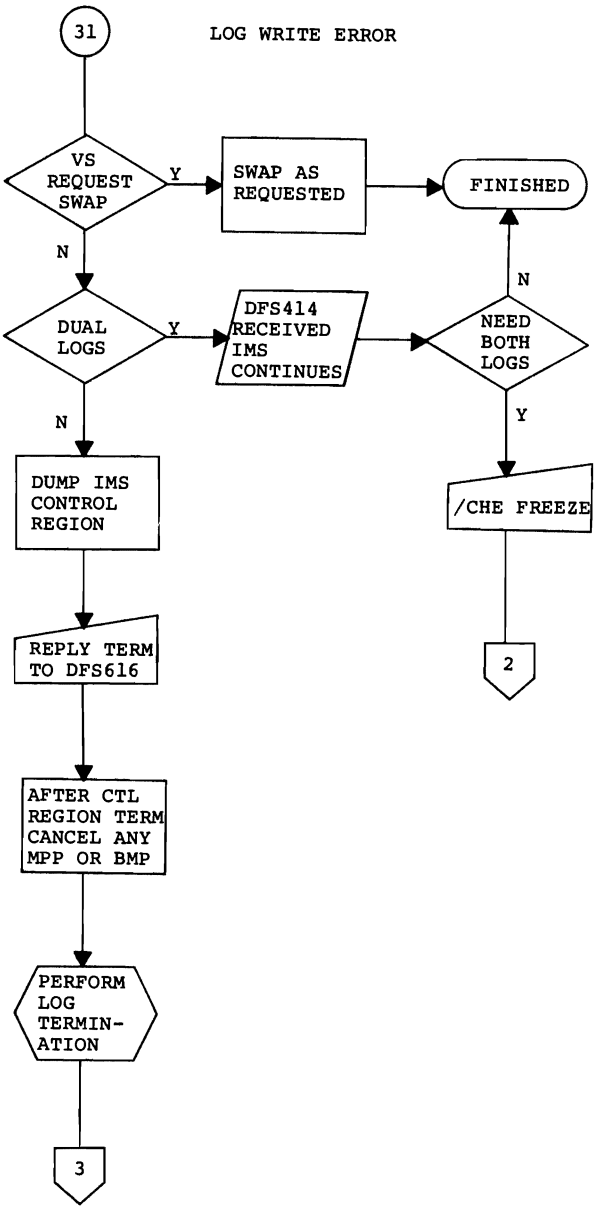
If dual logging was in effect when the input log tape was created, the backup log tape may be used. Re-execute the function using the backup copy.

The operator should also notify Systems Programming of the bad log tape. It might be desirable to execute the Log Recovery Utility to recreate the tape in error.

5) PERFORM LOG RECOVERY

If a backup copy of the log tape is not available or if both copies are not readable, perform Procedure 63 LOG RECOVERY.

Once a usable log tape is created the original function must be re-executed from the beginning.


31
LOG WRITE ERROR
VS REQUEST SWAP
Y
SWAP AS REQUESTED
FINISHED
N
DUAL LOGS
Y
DFS414 RECEIVED IMS CONTINUES
NEED BOTH LOGS
N
Y
N
DUMP IMS CONTROL REGION
/CHE FREEZE
REPLY TERM TO DFS616
2
AFTER CTL REGION TERM CANCEL ANY MPP OR BMP
PERFORM LOG TERMIN-ATION
3

PROCEDURE 31. LOG WRITE ERROR

This procedure should be entered whenever message DFS414 UNCORRECTABLE WRITE ERROR ON IEFRDER(2) is received on the system console.

1) VS REQUEST SWAP

When a write error is detected by OS/VS it may issue a message requesting the operator to switch to another tape unit. If this request is received, switch tape units and continue normal operations.

2) DUAL LOGS

If the Dual Logging feature is being used and one log tape becomes unusable, message DFS414 is issued to the system console. IMS/VS will continue to write to the alternate log tape. Notify Systems Programming to make a decision relative to continued operation with only one log or to resume the dual log mode of operation.

If dual logging is to be resumed, IMS/VS must be terminated and restarted. A Checkpoint FREEZE termination is the fastest method to terminate however PURGE or DUMPQ may be used. Once terminated, IMS/VS must be restarted with a WARM START.

If single log operation is chosen the error log can be later recreated using the Log Recovery Utility.

3) DUMP IMS CONTROL REGION

When a single log is being created and a write error occurs, message DFS616 is issued to the system console followed by a WTOR 'nn DFS616 REPLY TERM TO CONTINUE WITH ABEND'. Do not reply yet!

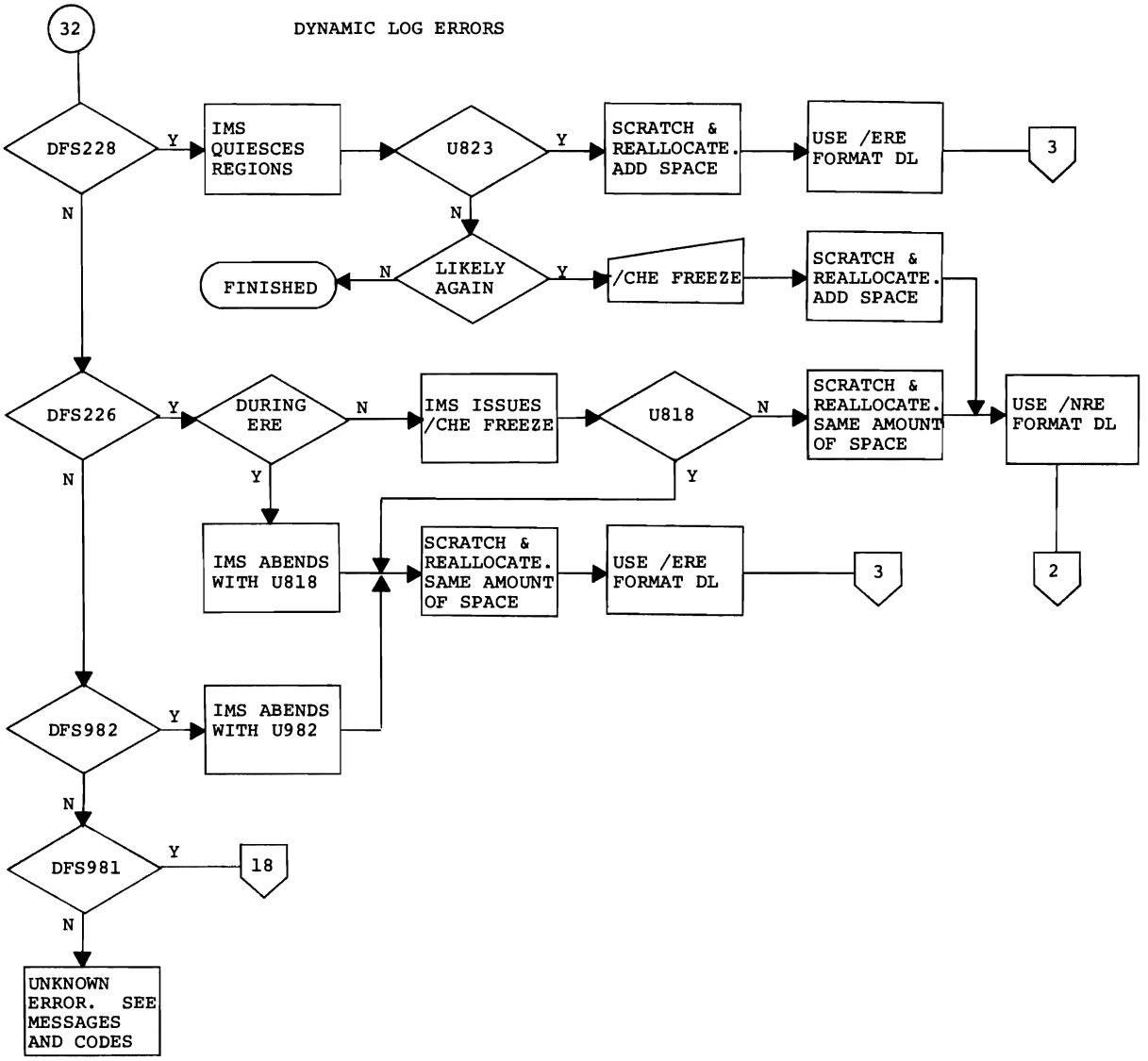
Enter the OS/VS DUMP command to dump the IMS/VS Control Region.

After the dump is completed reply 'nn TERM' to the outstanding WTOR. The IMS/VS Control Region will terminate with a S001 abend.

Dependent regions should all terminate with a U002 abend. If they do not, enter a C jobname,DUMP command from the system console for each dependent region still active.

4) PERFORM LOG TERMINATION

Perform procedure 62 LOG TERMINATION to close the log tape. Upon completion of this procedure the IMS/VS system may be EMERGENCY RESTARTed.


32
DFS228
Y
IMS QUIESCES REGIONS
U823
Y
SCRATCH & REALLOCATE. ADD SPACE
USE /ERE FORMAT DL
3
N
N
LIKELY AGAIN
N
FINISHED
Y
/CHE FREEZE
SCRATCH & REALLOCATE. ADD SPACE
DFS226
Y
DURING ERE
N
IMS ISSUES /CHE FREEZE
U818
N
SCRATCH & REALLOCATE. SAME AMOUNT OF SPACE
USE /NRE FORMAT DL
N
Y
Y
IMS ABENDS WITH U818
SCRATCH & REALLOCATE. SAME AMOUNT OF SPACE
USE /ERE FORMAT DL
3
2
DFS982
Y
IMS ABENDS WITH U982
N
DFS981
Y
18
N
UNKNOWN ERROR. SEE MESSAGES AND CODES

PROCEDURE 32. DYNAMIC LOG ERRORS

This procedure should be entered when the IMS/VS system fails with U818, U823, or U982, or messages DFS226 or DFS228 are received.

1) DFS228 IMSVS.DBLLOG DATA SET TOO SMALL

If the DFS228 is received during Emergency Restart the Control Region is immediately terminated with a U823 abend.

During normal operation IMS/VS will begin to quiesce all dependent regions. If any dependent region should abend or issue a ROLL call prior to being quiesced, the Control Region is terminated with a U823.

2) U823

If the Control Region does not abend dynamic logging has resumed, however, the situation may likely occur again. To prevent a reoccurance, IMS/VS should be terminated, the dynamic log data set should be scratched and reallocated with more space, and a WARM START performed specifying the FORMAT DL parameter.

If the U823 does occur, the Dynamic Log data set must be scratched and reallocated with more space. The system must be EMERGENCY RESTARTed specifying the FORMAT DL parameter.

3) DFS226 WRITE ERROR IN IMS/VS.DBLLOG DATA SET

If the DFS226 is received during Emergency Restart the Control Region is immediately terminated with a U818 abend.

During normal operation a /CHECKPOINT FREEZE is automatically initiated to shut down the system. If a dependent region should abend or issue a ROLL call before the shut down is completed, the Control Region is terminated with a U818.

4) U818

If the Control Region is shut down with the generated checkpoint command, scratch and reallocate the Dynamic Log data set. The system may be restarted with a WARM START specifying the FORMAT DL parameter.

If the U818 does occur, scratch and reallocate the Dynamic Log Data set. The system must be EMERGENCY RESTARTed specifying the FORMAT DL parameter.

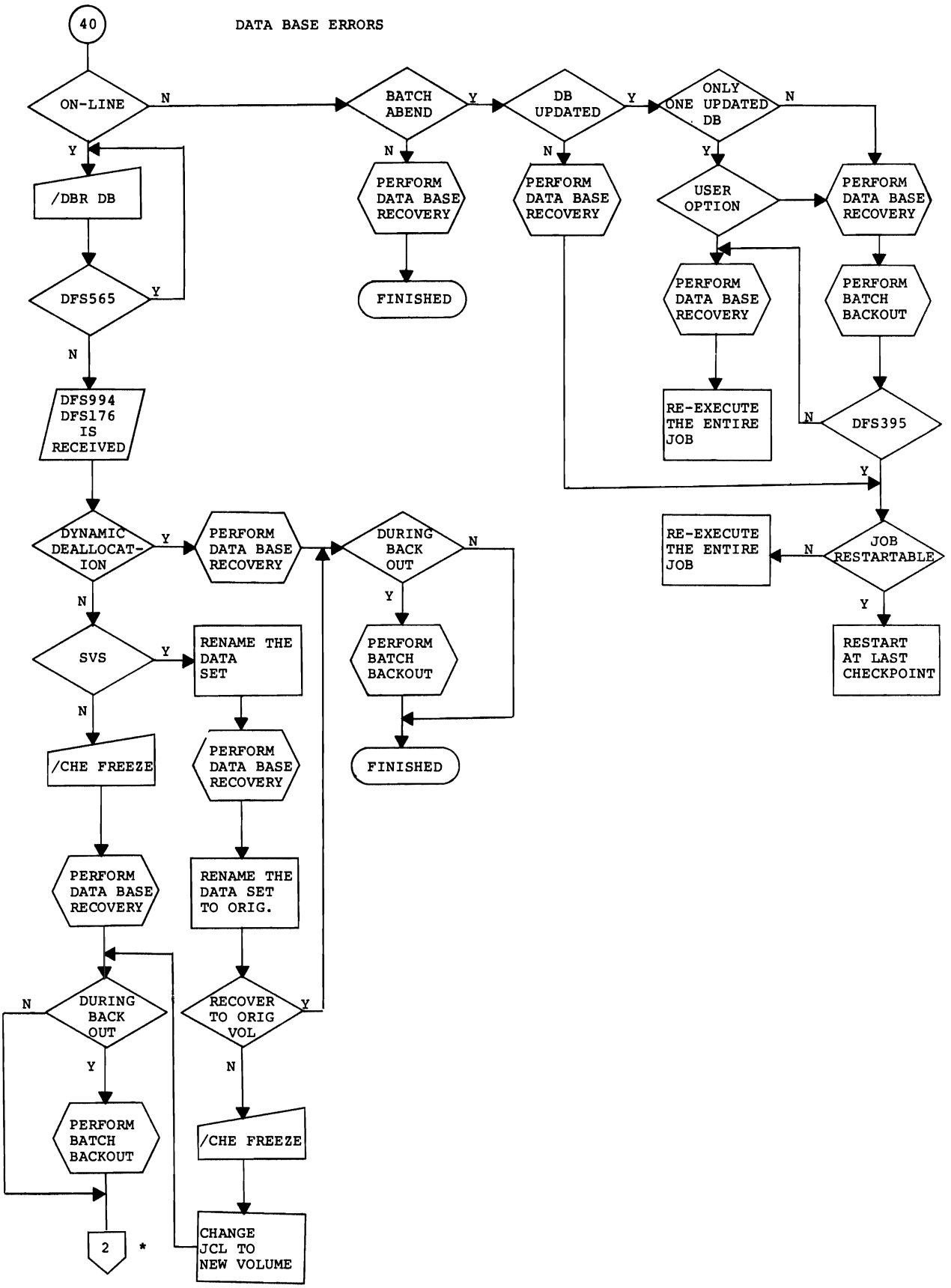
5) DFS982 I/O ERROR READING BACKOUT QUEUE FOR DATABASE ...

This message will be followed by a U982 abend of the Control Region when a read I/O error occurs on the Dynamic Log during Emergency Restart.

Scratch and reallocated the data set. The system must be EMERGENCY RESTARTed using the FORMAT DL parameter.

6) DFS981 I/O ERROR ON DYNAMIC LOG WHILE BACKING OUT . . .

While performing dynamic backout following a dependent region abend, a read I/O error occured. See procedure 18 DEPENDENT REGION ABEND to cover this situation.



* /STA DB, /STA PROG, /STA TRAN MUST BE ISSUED FOLLOWING THE WARM START

PROCEDURE 40. DATA BASE ERRORS

This procedure should be entered in response to a DFS451 message which is produced as the result of a data base read or write error, or any failure of Dynamic or Batch Backout when messages DFS964 and DFS983 are produced. Any of these conditions requires a forward recovery of the data set(s). For read and write errors the data base at fault is stopped from further schedulings but not from access by the currently scheduled program.

Writing to data bases occurs asychronously to the application program therefore no return code is passed to the program. The block in the buffer pool is flagged such that it remains in the pool and is never freed until IMS/VS termination. Any additional updates are performed in the buffer pool and recorded on the log tape. Recovery using that log tape will then restore the data base to the same condition which would have existed had the block(s) been written to disk.

On read errors an 'AO' status code is returned to the application program. The program can ignore that data base segment, it could ABEND (never recommended), or it could issue a ROLL call (recommended if updates had been performed). A Roll call causes online updates to be backed out to the last synchronization point. Batch programs issuing a ROLL must be followed by a Batch Backout step.

An out of space condition results in an application program ABEND and the data base is stopped.

ON-LINE.

1) /DBR name

The Master Terminal Operator should enter a /DBR command to close the log tape. The log tape currently being written must be input to the recovery process. The command will not complete until all executing programs which use the named data base are terminated. A DFS565 is probably caused by a BMP program. Wait for it to terminate and re-enter the command.

The data base should be scratched and reallocated. Allocate additional space if an out of space condition exists.

2) DYNAMIC DEALLOCATION

The dynamic deallocation/allocation feature available in IMS/VS 1.1.1 PTF 3 and later systems operating with MVS allows users to recover data sets while the online region remains active. Perform the DATA BASE RECOVERY procedure to recover the data set. If the failure occured during a backout operation, the operator should perform the BATCH BACKOUT procedure to complete that process. Input to the

backout should be the log volume which was freed as a result of the /DBR command. The data base is now usable.

3) NO SVS

Other MVS or VS1 users must terminate IMS/VS in order to free the data base for recovery.

/CHECKPOINT FREEZE is the quickest way to do this. Once IMS/VS has been terminated, perform the DATA BASE RECOVERY procedure to recover the data set.

4) DURING BACKOUT

If the failure occured during a backout operation the operator should perform the BATCH BACKOUT procedure to complete that process.

5) PERFORM A WARM START

The data base, the program, and the transaction must be restarted following the WARM start.

6) RENAME THE DATA SET

SVS users may rename a data set, perform the DATA BASE RECOVERY procedure, and then rename the data set back to its original name.

7) RECOVER TO ORIGINAL VOLUME

If the data set was recovered to a new volume, IMS/VS must be terminated and the JCL changed before the data set is available for usage. /CHECKPOINT FREEZE is the quickest way to do this.

The data set should be scratched and reallocated. Allocate additional space if necessary.

1) BATCH ABEND

If the batch region did not ABEND, the situation may have been a write error or the program ignored the read error. In either case if the data set is recovered, re-running the program is not necessary. Perform the DATA BASE RECOVERY procedure using all the log tapes and/or Change Accumulation tapes produced since the last Image Copy.

2) DATA BASE UPDATED

If the region did ABEND but no data base updates were being done, the operator should perform the DATA BASE RECOVERY procedure. He may re-execute the entire job or restart from the last checkpoint.

3) ONLY ONE UPDATED DATA BASE

When data bases are updated the operator has two options. He can recover to the beginning of the job and re-execute the entire job, or he can recover to the point of failure and perform Batch Backout. If the program is checkpointable the latter is probably much faster.

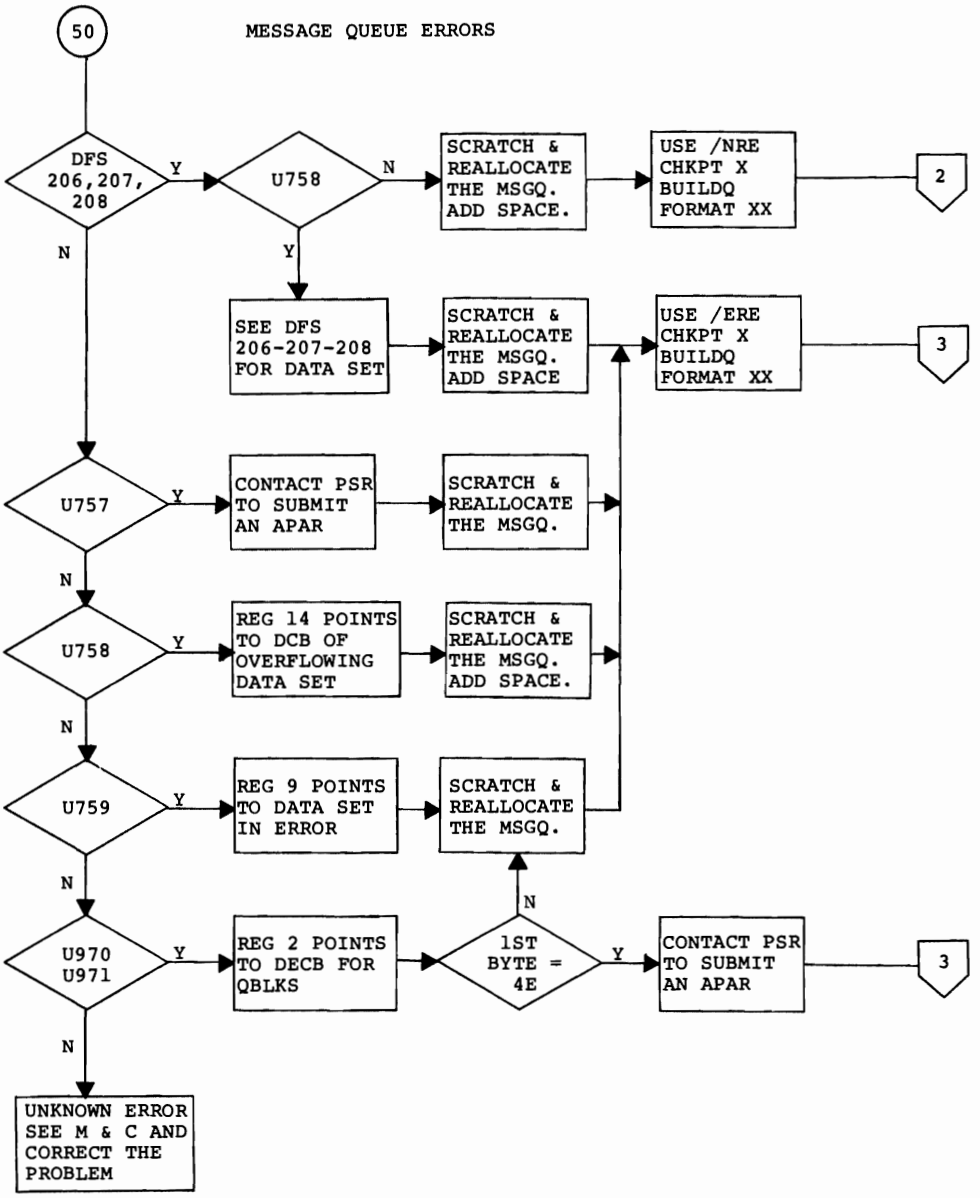
If multiple data bases are updated (Indexes, Secondary Indexes, logically related etc.) the latter is recommended because of the time involved in recovery and the possibilities for error. All data bases must be recovered and and the possibility of overlooking one is great.

To recover to the start of the job use the Image Copy and all Change Accumulation and/or log tapes up to but not including the log produced when the ABEND occured.

To recover to the point of failure, include the log which was being produced when the ABEND occured.

4) DFS395 BACKOUT COMPLETE FOR PSB name

If the Batch Backout does not successfully complete, the operator would have to recover all updated bases to the start of the job and re-execute the entire job.


50
MESSAGE QUEUE ERRORS
DFS 206,207, 208
Y
U758
N
SCRATCH & REALLOCATE THE MSGQ. ADD SPACE.
USE /NRE CHKPT X BUILDQ FORMAT XX
2
N
Y
SEE DFS 206-207-208 FOR DATA SET
SCRATCH & REALLOCATE THE MSGQ. ADD SPACE
USE /ERE CHKPT X BUILDQ FORMAT XX
3
U757
Y
CONTACT PSR TO SUBMIT AN APAR
SCRATCH & REALLOCATE THE MSGQ.
N
U758
Y
REG 14 POINTS TO DCB OF OVERFLOWING DATA SET
SCRATCH & REALLOCATE THE MSGQ. ADD SPACE.
N
U759
Y
REG 9 POINTS TO DATA SET IN ERROR
SCRATCH & REALLOCATE THE MSGQ.
N
U970 U971
Y
REG 2 POINTS TO DECB FOR QBLKS
1ST BYTE = 4E
Y
CONTACT PSR TO SUBMIT AN APAR
3
N
N
UNKNOWN ERROR SEE M & C AND CORRECT THE PROBLEM

PROCEDURE 50. MESSAGE QUEUE ERRORS

This procedure should be entered to correct one of the reasons for an IMS/VS Control Region abend.

1) DFS206 - 207 - 208 D/S LIMIT REACHED. MUST REBUILD

When the space within a message queue data set is used up to the limit imposed by the SHUTDOWN= parameter of the MSGQUEUE macro, IMS/VS initiates a /CHECKPOINT DUMPQ command to shut down the system.

If the shutdown is successful, scratch and reallocate the named data set with additional space. The system may be WARM STARTed using the CHKPT, BUILDQ, and FORMAT parameters.

2) U757 QUEUE MANAGER ERROR

Contact the Program Support Representative to submit an APAR.

It is probably not required but a scratch and reallocate of the data set may help. Attempt to restart the system using EMERGENCY RESTART and specify the BUILDQ and FORMAT parameters.

3) U758 DATA SET OVERFLOWED

Register 14 points to the DCB of the failing data set. Scratch and reallocate the data set with additional space supplied. The system must be restarted using EMERGENCY RESTART.

4) U759 I/O ERROR

Register 9 points to the DECB in error. Scratch and reallocate the data set. The system must be restarted using EMERGENCY RESTART.

5) U970 - U971 QBLKS I/O ERROR

Register 2 points to the DECB of the QBLKS data set. If the first byte of the DECB is "4E" no I/O was involved. Contact the Program Support Representative to submit an APAR. The system must be restarted using EMERGENCY RESTART.

If the first byte is not "4E" scratch and reallocate the data set. The system must be restarted using EMERGENCY RESTART.

6) UNKNOWN ERROR

For any other abends or messages such as DFS074 and DFS082 refer to the Messages and Codes manual. For most messages trace records are produced which the System Programmer can use to resolve the problem.

These errors should be referred to Systems Programming and possibly to the Program Support Representative.


51
/BRO SHUT DOWN MSG
/CHE DUMPQ
SCRATCH & REALLOCATE THE SPA DATA SET
USE /NRE BUILDQ FORMAT SP
2

PROCEDURE 51. SPA I/O ERRORS

This procedure should be entered in response to:

DFS215 WORK AREA I/O ERROR, CONVERSATION TERMINATED
DFS987 WRITE ERROR OCCURRED WRITING SPA FOR CONVERSATION..
DFS988 READ ERROR OCCURRED READING SPA FOR CONVERSATION ...

The DFS215 message is issued when an I/O error occurs while the SPA is being updated from an application program. The DFS987 is issued when a write I/O error occurs during a BUILDQ restart. The DFS988 is issued when a read I/O error occurs during system checkpoint.

In all cases IMS/VS continues operating and the conversation is terminated or will be during a restart operation. When the DFS215 message is issued, the application program is presented an X1 return code in response to the DL/I call. That program may ignore the return code, abend, or issue a ROLL call.

To prevent re-occurrances of the error, IMS/VS should be terminated and a new scratch pad data set obtained.

1) /BRO SHUTDOWN MSG

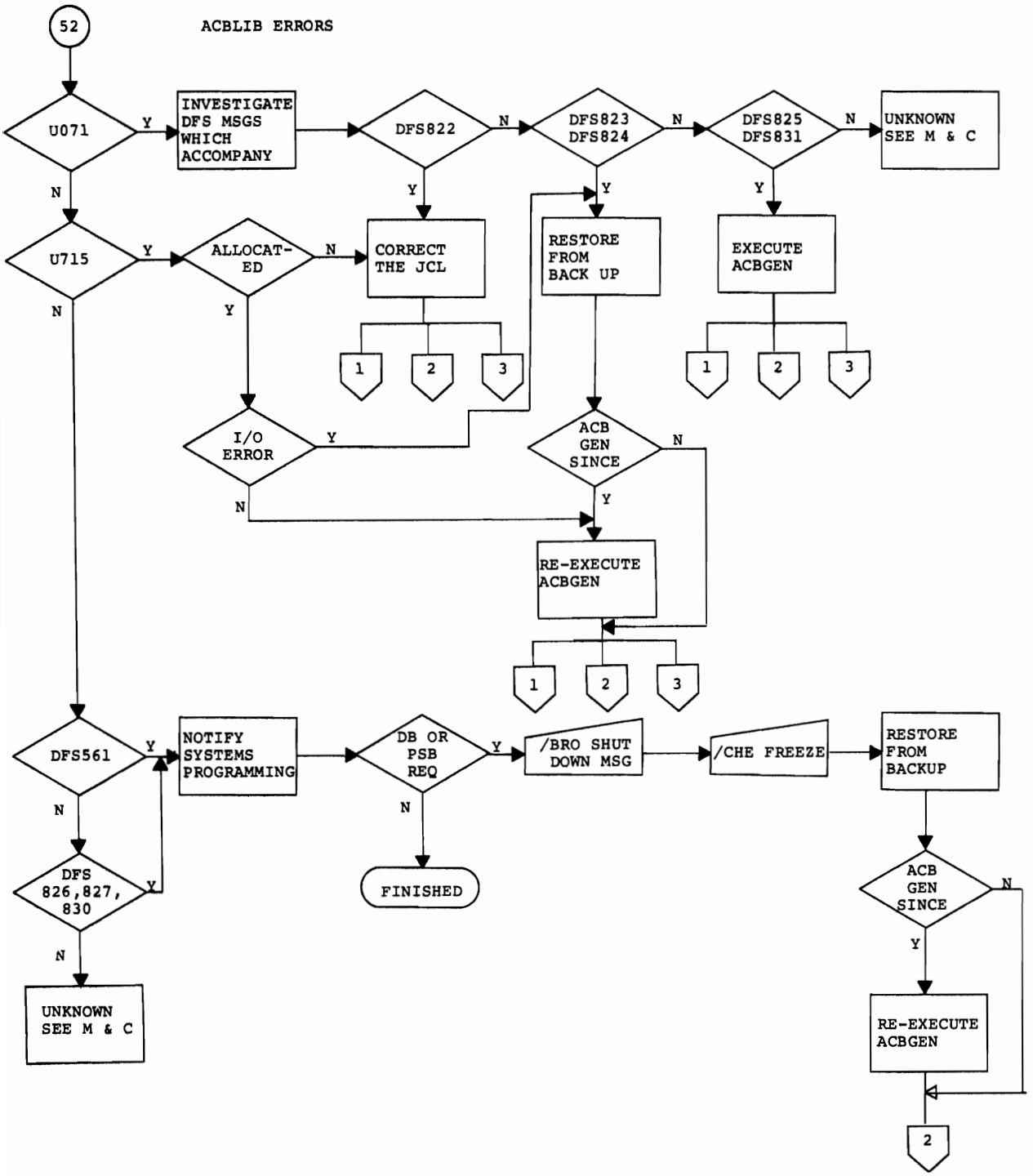
   Operations will probably wish to notify all users that IMS/VS is being terminated.

2) /CHKPT DUMPQ

   This command will terminate IMS/VS.

   The Scratch pad data set must be scratched and reallocated.

   IMS/VS may then be restarted using WARM START and specifying the BUILDQ and FORMAT SP parameters.


52
ACBLIB ERRORS
U071
Y
INVESTIGATE DFS MSGS WHICH ACCOMPANY
DFS822
N
DFS823 DFS824
N
DFS825 DFS831
N
UNKNOWN SEE M & C
N
Y
Y
Y
U715
Y
ALLOCAT-ED
N
CORRECT THE JCL
RESTORE FROM BACK UP
EXECUTE ACBGEN
N
Y
1
2
3
1
2
3
I/O ERROR
Y
ACB GEN SINCE
N
N
Y
RE-EXECUTE ACBGEN
1
2
3
DFS561
Y
NOTIFY SYSTEMS PROGRAMMING
DB OR PSB REQ
Y
/BRO SHUT DOWN MSG
/CHE FREEZE
RESTORE FROM BACKUP
N
N
DFS 826,827, 830
Y
FINISHED
ACB GEN SINCE
N
N
Y
UNKNOWN SEE M & C
RE-EXECUTE ACBGEN
2

PROCEDURE 52. ACBLIB ERRORS

I/O errors or inconsistencies in the ACBLIB may result in a Control Region abend or a message being issued and processing continued.

1) U071 INITIALIZATION ERROR

A U071 abend will be preceeded by a DFS error message which should explain the failure. Investigate the Master Terminal printed output.

DFS822 NO ALLOCATION FOR DDN 'IMSACB'.

This is a JCL error. Add the appropriate DD card and perform the restart which failed.

DFS823 UNABLE TO SUCCESSFULLY OPEN DDN 'IMSACB'.
DFS824 PDS DIRECTORY READ ERROR DDN 'IMSACB'.

These are probably caused by I/O errors when reading ACBLIB. The operator should check the JCL for proper allocation of the data set. If the JCL is correct the data set must be restored from a backup copy or a complete ACBGEN must be executed. When restoring from a backup copy, any ACBGENs performed since the copy was created must be re-executed. Perform the restart which failed.

DFS825 BLDL FAILED FOR ALL DEFINED DATA BASE
DIRECTORIES.
DFS831 BLDL FAILED FOR ALL DEFINED PROGRAM
DIRECTORIES.

Either of these messages would indicate the required PSBs or DBDs were never processed by ACBGEN. Locate the missing members and execute an ACBGEN before attempting the restart.

2) U715 INITIALIZATION ERROR

This abend will occur if there is no allocation for IMSVS.ACBLIB in the Control Region job step. Check for the presence of the DD card and provide it if necessary.

This abend will also occur if none of the required DBD names could be found in ACBLIB. This condition requires an execution of ACBGEN to add the members.

The third reason for the Abend is an I/O error on the PDS directory. This condition requires a restore of ACBLIB.

3) DFS561 ERROR READING ACBLIB, (PSB=) (DMB=)

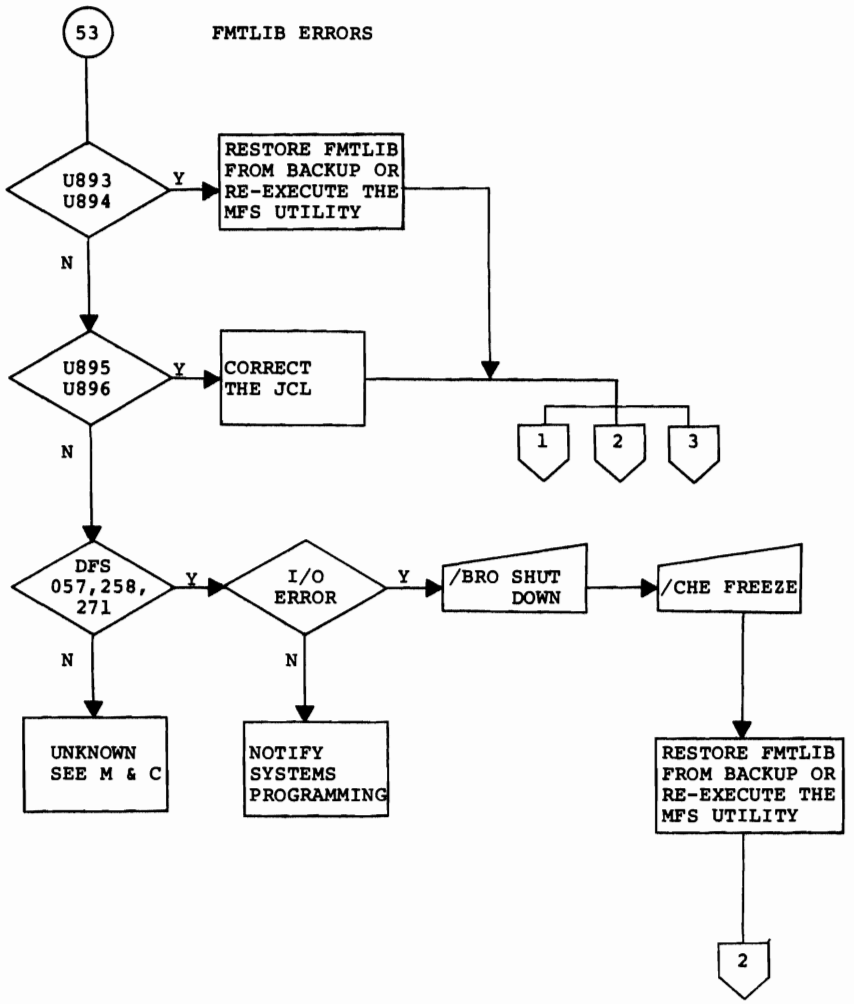
An I/O error has occured when retrieving the named member from ACBLIB. If the system can function without the data base or program no action is required.

If the data base or program is required, enter a broadcast command to notify all users of the intention to terminate IMS/VS. Terminate the system with a CHECKPOINT FREEZE. The ACBLIB data set must be restored from a copy and any ACBGENs performed since the copy was created must be re-executed. The system can then be WARM STARTed.

4) DFS826 BLDL FAILED FOR FOLLOWING DBDS.
DFS827 DYNAMIC PSB name BLDL FAILED.
DFS830 BLDL FAILED FOR FOLLOWING PSBS.

A reason is listed for issuing the above messages. Notify Systems Programming so a decision can be made relative to the need for the member(s).

5) For any other messages refer to the Messages and Codes manual and perform any indicated actions.


53
FMTLIB ERRORS
U893
U894
Y
RESTORE FMTLIB FROM BACKUP OR RE-EXECUTE THE MFS UTILITY
N
U895
U896
Y
CORRECT THE JCL
1
2
3
N
DFS 057,258, 271
Y
I/O ERROR
Y
/BRO SHUT DOWN
/CHE FREEZE
N
N
UNKNOWN SEE M & C
NOTIFY SYSTEMS PROGRAMMING
RESTORE FMTLIB FROM BACKUP OR RE-EXECUTE THE MFS UTILITY
2

PROCEDURE 53. FMTLIB ERRORS

This procedure should be entered when IMS/VS intitialization abends or when users complain that errors are occuring and messages are being received at the remote terminals.

1) U893 - U894 I/O ERRORS ON FMTLIB

DFS893 I/O ERROR READING THE DIRECTORY will precede these abends. Restore the FORMAT data set from a backup copy if one exists. In the absence of such a copy the Message Format Services Utility must be executed to create a new data set. The system can then be started using the same form of start as the one which failed.

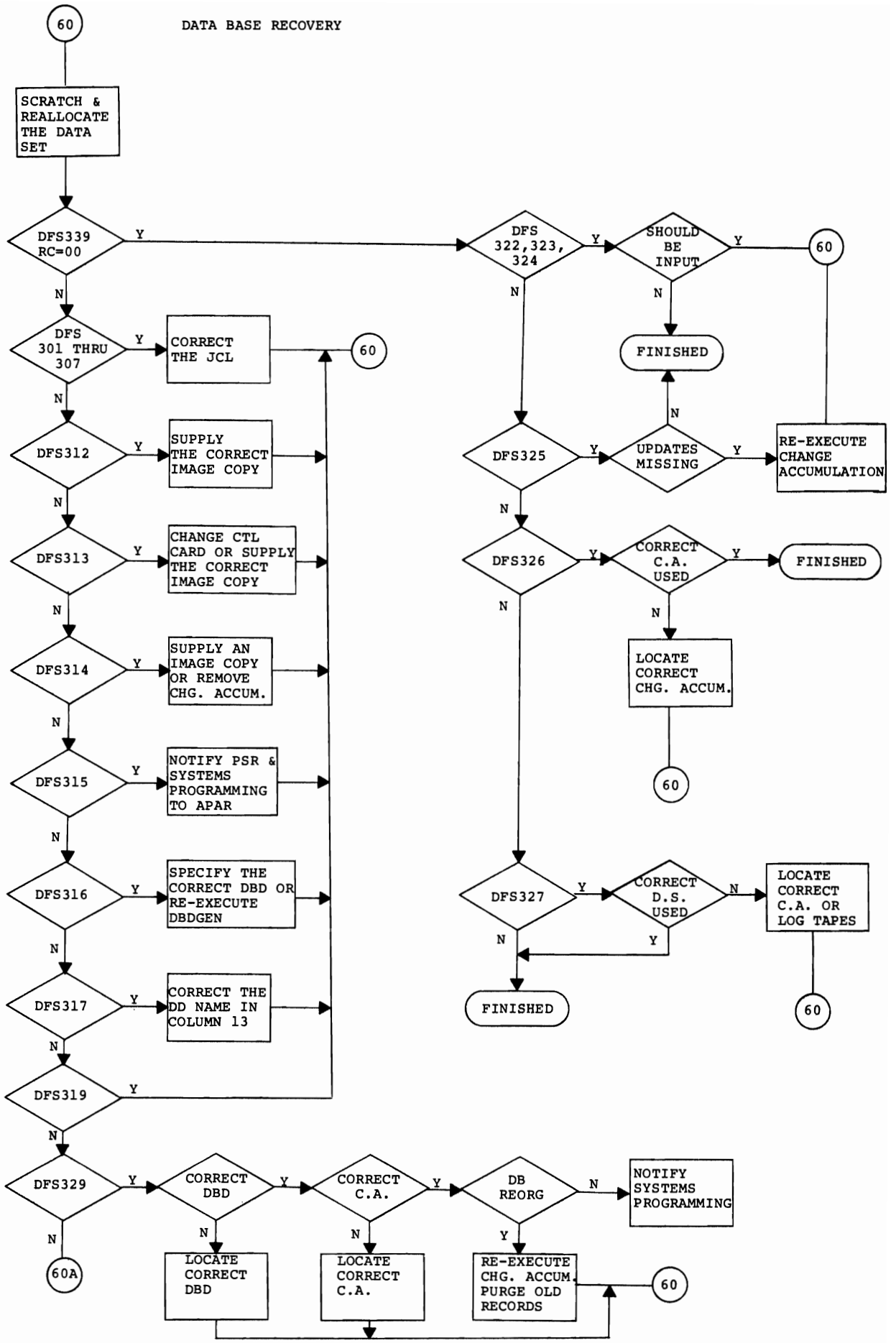
2) U895 - U896 JCL ERRORS

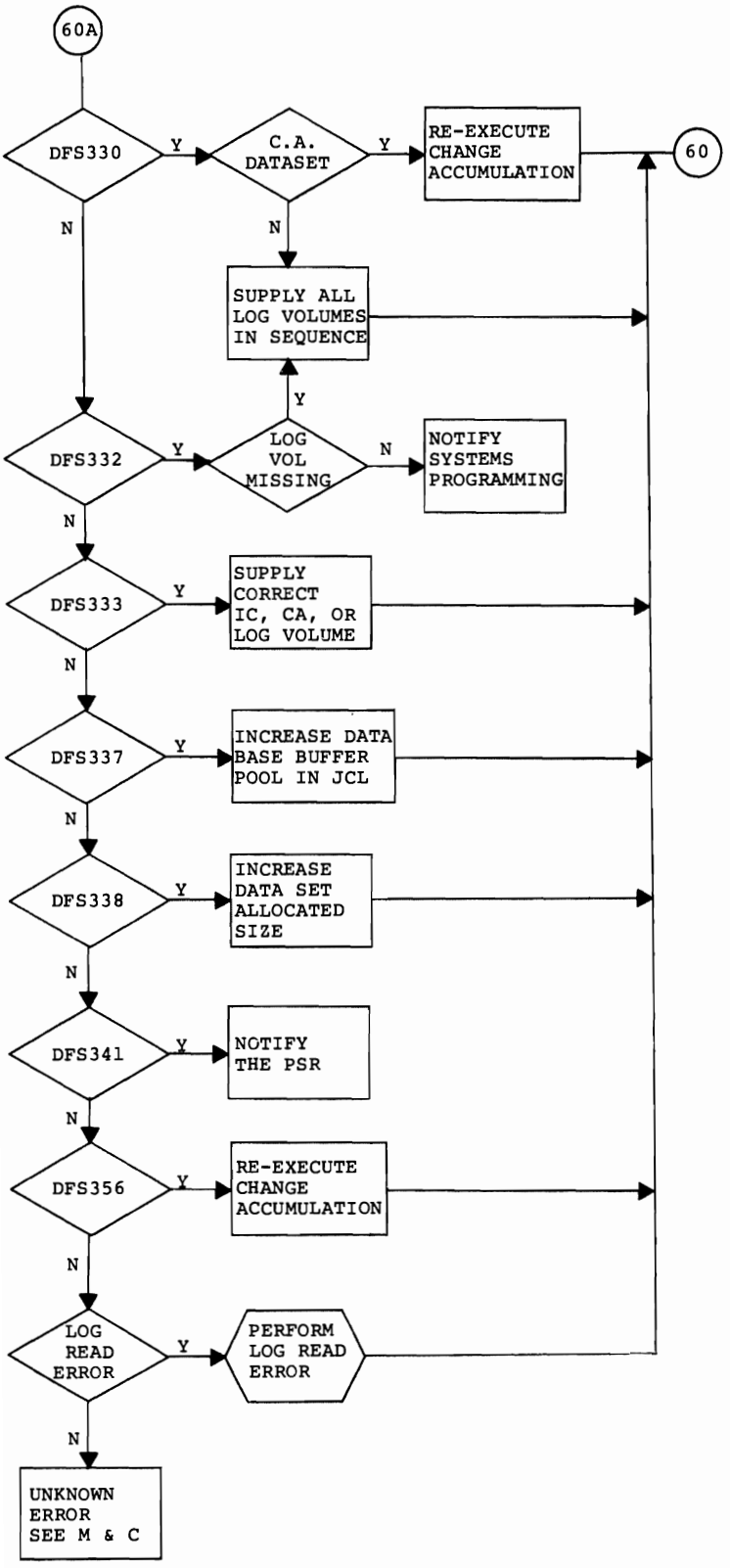
The user has specified DISP=SHR or DD DUMMY for the ddname IMSDILIB. Neither condition is allowed. Correct the JCL and restart using the same form of start as the one which failed.

3) DFS057 REQUESTED FORMAT BLOCK NOT AVAILABLE
DFS258 UNABLE TO LOCATE MESSAGE DESCRIPTION - INPUT IGNORED
DFS271 UNABLE TO LOAD ERROR MESSAGE OUTPUT DESCRIPTION

These messages are issued to the user terminal. When notified by the user that a problem exists, the Master Terminal Operator should check the OS/VS console for OS/VS messages indicating I/O errors on the FORMAT library. If no messages are found Systems Programming should be notified to resolve the problem.

When I/O error messages are found, broadcast a message informing all users that IMS/VS is being terminated. Terminate the system with a CHECKPOINT FREEZE command. Restore the FORMAT data set from a backup copy if one exists. In the absence of such a copy the Message Format Services Utility must be executed to create a new data set. The system can then be WARM STARTed.


60
SCRATCH & REALLOCATE THE DATA SET
DFS339 RC=00
Y
N
DFS 322,323, 324
SHOULD BE INPUT
60
DFS 301 THRU 307
CORRECT THE JCL
60
FINISHED
DFS312
SUPPLY THE CORRECT IMAGE COPY
DFS325
UPDATES MISSING
RE-EXECUTE CHANGE ACCUMULATION
DFS313
CHANGE CTL CARD OR SUPPLY THE CORRECT IMAGE COPY
DFS326
CORRECT C.A. USED
FINISHED
DFS314
SUPPLY AN IMAGE COPY OR REMOVE CHG. ACCUM.
LOCATE CORRECT CHG. ACCUM.
60
DFS315
NOTIFY PSR & SYSTEMS PROGRAMMING TO APAR
DFS316
SPECIFY THE CORRECT DBD OR RE-EXECUTE DBDGEN
DFS327
CORRECT D.S. USED
LOCATE CORRECT C.A. OR LOG TAPES
FINISHED
60
DFS317
CORRECT THE DD NAME IN COLUMN 13
DFS319
DFS329
CORRECT DBD
CORRECT C.A.
DB REORG
NOTIFY SYSTEMS PROGRAMMING
60A
LOCATE CORRECT DBD
LOCATE CORRECT C.A.
RE-EXECUTE CHG. ACCUM. PURGE OLD RECORDS
60


60A
DFS330
Y
C.A. DATASET
Y
RE-EXECUTE CHANGE ACCUMULATION
60
N
N
SUPPLY ALL LOG VOLUMES IN SEQUENCE
Y
DFS332
Y
LOG VOL MISSING
N
NOTIFY SYSTEMS PROGRAMMING
N
DFS333
Y
SUPPLY CORRECT IC, CA, OR LOG VOLUME
N
DFS337
Y
INCREASE DATA BASE BUFFER POOL IN JCL
N
DFS338
Y
INCREASE DATA SET ALLOCATED SIZE
N
DFS341
Y
NOTIFY THE PSR
N
DFS356
Y
RE-EXECUTE CHANGE ACCUMULATION
N
LOG READ ERROR
Y
PERFORM LOG READ ERROR
N
UNKNOWN ERROR SEE M & C

PROCEDURE 60. DATA SET RECOVERY

This procedure should be entered when it has been determined that a data set must be recovered. In most cases the failure which requires a data set recovery is an I/O error. Recovery is also the method used to restore a data set to a prior point in time. Care must be exercised when doing this to ensure that all data sets of a data base are recovered. Secondly, all related data bases (Indexes, Secondary Indexes, logically related) must be recovered to the same point in time. Because of the possibility for errors, recovery to a prior point in time is not generally recommended.

The successful execution of the RECOVERY utility is no guarantee of data base integrity. Omission of Change Accumulation volumes or Log volumes may execute successfully, however, the data set is incorrect. If failures occur when executing these things must be checked.

1) DFS339 FUNCTION RV HAS COMPLETED (NORMALLY) (ABNORMALLY) RC=xx

   If the completion is normal and the return code is 00 this procedure is finished.

   If the completion is abnormal the return code is 16. Further checking of messages will reveal the reason for the failure.

2) DFS301 thru DFS307

   These messages indicate errors in or ommisions of JCL. Correct the problem and re-execute the utility.

3) DFS312 DDNAME ddname DOES NOT APPLY TO DATABASE dbdname

   The DBD name beginning in position 4 of the control card does not match the DBD name in the Image Copy header record. The wrong Image Copy was supplied or the control card DBD name specification is incorrect.

5) DFS314 CHANGE ACCUMULATION SUPPLIED BUT WITHOUT IMAGE COPY

   Recovery requires an Image Copy when Change Accumulation is supplied. If no Image Copy is available remove the Change Accumulation and supply the log tapes which were input to Change Accumulation.

6) DFS315 DEVTYPE MACRO FAILED ON DDNAME ddname

   This should not happen. Notify Systems Programming and the Program Support Representative to resolve the problem and to submit an APAR if necessary.

7) DFS316 DBD dbdname CONTAINS AN UNKOWN ORGANIZATION CODE

The DBD being used is in error. Check the DBD library specified, check the DBD name, and/or recompile the DBD.

8) DFS317 DDNAME ddname NCT FOUND ON DDNAME ddname HEADER RECORD

The control card DD name does not match the Image Copy header record DD name. Supply the correct Image Copy or change the DD name specified.

9) DFS319 DDNAME ddname HAD A PERMANENT I/O ERROR

An unrecoverable I/O error occured on an output data set. The recovery must be re-executed.

10) DFS322 FUNCTION RV WAS NOT SUPPLIED AN IMAGE COPY INPUT
DFS323 FUNCTION RV WAS NOT SUPPLIED A CHG ACCUM INPUT
DFS324 FUNCTION RV WAS NOT SUPPLIED AN INPUT LOG FILE

These messages are information only. If the recovery should have included the named file, the data base is not recovered correctly.

11) DFS325 PURGE DATE ON DDNAME ddname IS LATER THAN DATE ON DDNAME ddname

The purge date specified when the Change Accumulation data set was created is later than the creation date of the Image Copy. Any data base updates occuring between the two dates are not included on the Change Accumulation data set thus recovery will not be complete.

It is difficult to determine whether or not data base updates have occured between the two dates. If any doubt exists, re-execute the Change Accumulation and specify the date/time of the Image Copy.

Presence of this message indicates that Change Accumulation procedures should be reviewed and corrected.

12) DFS326 DDNAME ddname INPUT IGNORED DUE TO PURGE DATE ON DDNAME ddname

The creation date of the Change Accumulation data set is earlier than the creation date of the Image Copy. Check the Change Accumulation data set. It is possible that using an incorrect data set caused the message.

If the correct data set was used the Change Accumulation procedures should be reviewed and corrected.

13) DFS327 NO RECORDS ON DDNAME ddname FOR RECOVERED DATA SET

No log records existed on the log data set or no Change Accumulation records existed for the data set being recovered.

Verify that the correct data sets were supplied.

14) DFS329 CHANGE ACCUMULATION HEADER ON DDNAME ddname IS NOT CONSISTENT WITH DBD dbdname

The DBD contains a data set organization code which does not match the Change Accumulation header record.

Specification of the wrong DBD or using the wrong Change Accumulation data set could cause this.

If the DBD has been changed and a reorganization has been done to change the data base organization, an Image Copy should have been immediately taken. If this is the case, re-execute Change Accumulation and purge the records prior to the Image Copy date/time.

If no Image Copy was performed at the time of the reorganization, the data base must be recovered up to the point of the reorganization. The reorganization must then be re-executed and the data base Image Copied. A Change Accumulation must be executed specifying a purge date/time equal to the re-organization and a recovery executed to recover the data base to the current point in time.

15) DFS330 DDNAME ddname IS OUT OF SEQUENCE FOR REASON x

The Change Accumulation record key or RBN is lower than the Image Copy, or a log record date/time is lower than a previously read log record.

If the reason code is 1 or 2, it is probably because the wrong Change Accumulation data set was input (possibly a Change Accumulation of a test data base with the same name) or the Change Accumulation was out of sequence. Supply the correct Change Accumulation data set or recreate the data set paying particular attention to the sort steps.

If the reason code is 3 the log tapes have been supplied out of sequence. Check for the presence of all log volumes and supply them in sequence.

16) DFS332 OSAM RECORD FROM DDNAME ddname HAS RBN nnnnn AND IS BEYOND THE CURRENT END OF THE DATA SET

While executing, Recovery keeps track of the current EOF. This current EOF is the last block written. Each Image Copy, Change Accumulation, or log record is matched to

the EOF. The input record can never exceed the EOF plus one block unless blocks are being skipped which is not allowed.

Check for missing log volumes. This is the most common situation causing this message. Supply any missing volumes to a Change Accumulation or log only input execution.

A second situation causing this message is performing a HISAM unload but not a reload, and using the unload file as input to Recovery. It this is the case supply a previous unload or Image Copy and all log volumes produced since that time.

A third situation can cause this message. If the data set has an embedded EOF, Image Copy will stop when the first EOF is encountered. Records could exist on the log tape for data base records residing between the two EOFs. Most likely this is caused by executing two IMS/VS Control Regions simultaneously against the same data base. Failure to scratch and reallocate the data sets during a recovery or reorganization could also cause an embedded EOF. There is no recommended solution to this problem. Notify Systems Programming for a resolution.

17) DFS333 KEY ID ON DDNAME ddname DOES NOT EXIST IN DDNAME ddname DATA SET

When performing a TRACK RECOVERY, an Image Copy or Change Accumulation record contained a key not found in the data set, or under all recovery conditions a log record contained a key not found in the data set.

Check the suspected data sets for wrong or omitted volumes. Supply the correct volumes.

18) DFS337 BUFFER POOL TOO SMALL

Increase the buffer pool size specification in the EXEC or DFSVSAMP JCL cards.

19) DFS338 DATA SET DDNAME ddname IS TOO SMALL

Increase the allocation of the data set.

20) DFS341 CATASTROPHIC ERROR OCCURED IN DL/I

This is probably an internal IMS/VS failure. Report the condition to the Program Support Representative.

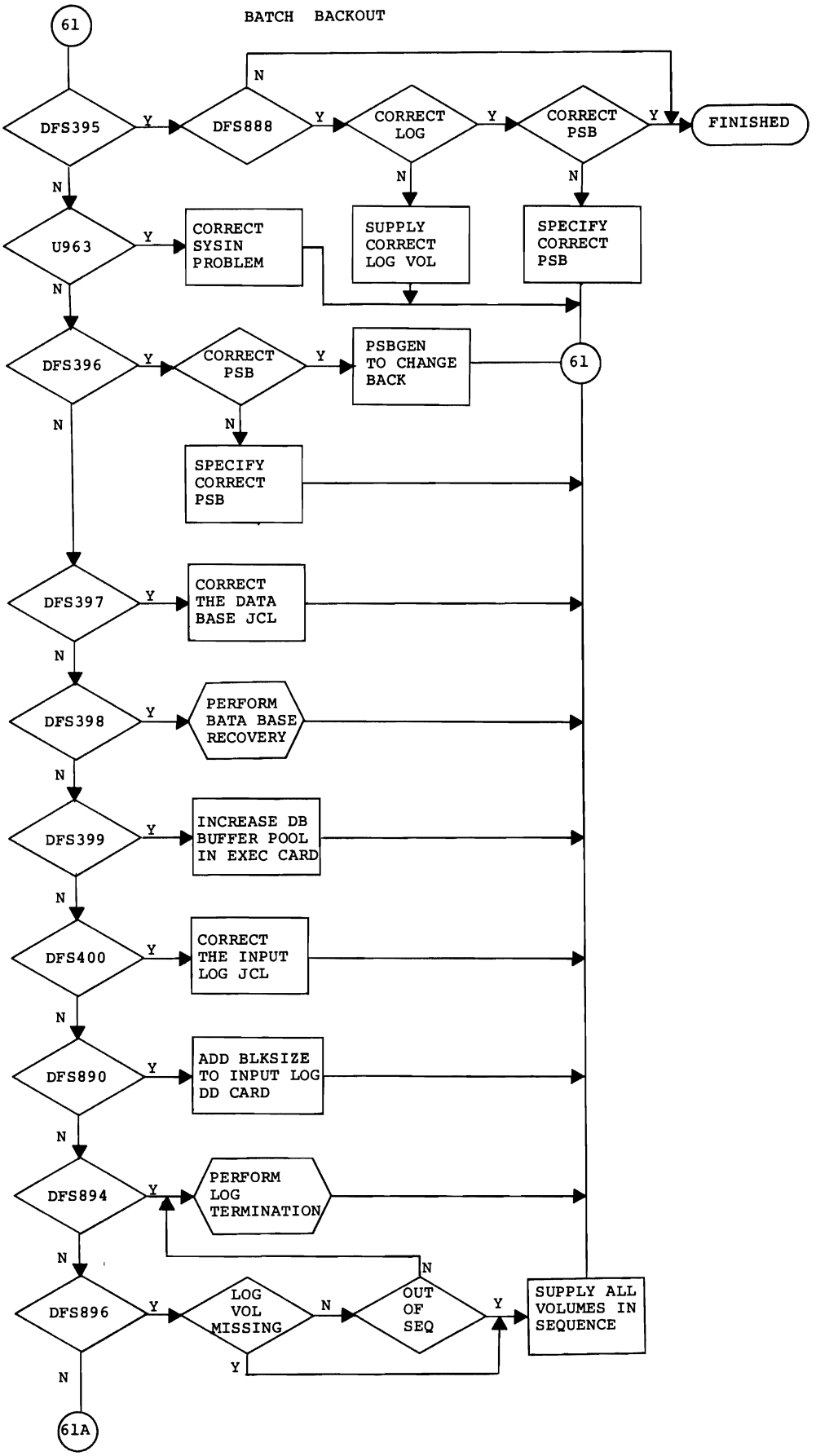
21) DFS356 HEADER RECORD NOT FOUND ON DDNAME ddname

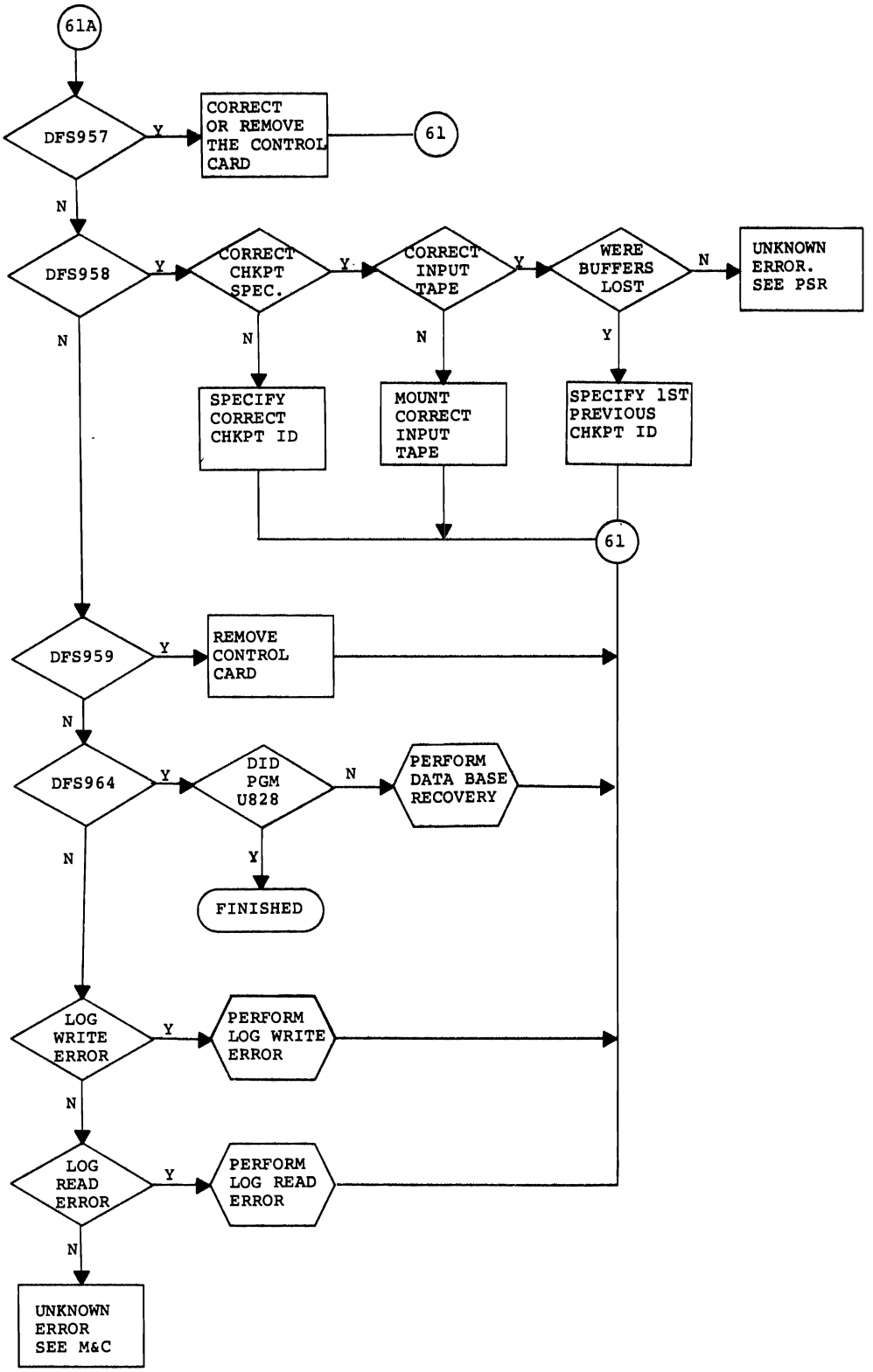
The Change Accumulation header record DBD name does not match the DBD name in a following change accumulation record.

Insure that all volumes were submitted in the correct sequence. If this is the case, re-create the Change Accumulation data set.

22) UNKNOWN ERROR

Refer to the Messages and Codes manual. Perform any actions required.


BATCH BACKOUT
61
DFS395
DFS888
CORRECT LOG
CORRECT PSB
FINISHED
U963
CORRECT SYSIN PROBLEM
SUPPLY CORRECT LOG VOL
SPECIFY CORRECT PSB
DFS396
CORRECT PSB
PSBGEN TO CHANGE BACK
61
SPECIFY CORRECT PSB
DFS397
CORRECT THE DATA BASE JCL
DFS398
PERFORM BATA BASE RECOVERY
DFS399
INCREASE DB BUFFER POOL IN EXEC CARD
DFS400
CORRECT THE INPUT LOG JCL
DFS890
ADD BLKSIZE TO INPUT LOG DD CARD
DFS894
PERFORM LOG TERMINATION
DFS896
LOG VOL MISSING
OUT OF SEQ
SUPPLY ALL VOLUMES IN SEQUENCE
61A
Y
N


61A
DFS957
Y
CORRECT OR REMOVE THE CONTROL CARD
61
N
DFS958
Y
CORRECT CHKPT SPEC.
Y
CORRECT INPUT TAPE
Y
WERE BUFFERS LOST
N
UNKNOWN ERROR. SEE PSR
N
N
N
Y
SPECIFY CORRECT CHKPT ID
MOUNT CORRECT INPUT TAPE
SPECIFY 1ST PREVIOUS CHKPT ID
61
DFS959
Y
REMOVE CONTROL CARD
N
DFS964
Y
DID PGM U828
N
PERFORM DATA BASE RECOVERY
N
Y
FINISHED
LOG WRITE ERROR
Y
PERFORM LOG WRITE ERROR
N
LOG READ ERROR
Y
PERFORM LOG READ ERROR
N
UNKNOWN ERROR SEE M&C

PROCEDURE 61. BATCH BACKOUT

Data base backout capability is provided with the IMS/VS system to allow the user to backout changes made during the execution of a DLIBATCH region. Secondly, it should be used to backout data base changes made by online regions which were active at the time of system failure when Emergency restart fails.

If Backout fails to successfully execute, the user may correct the problem and re-execute Backout. The same input tape (or a copy of) must be used for both executions and all output tapes must be saved for future Recovery operations.

The output log tape produced by Backout cannot be successfully used as input to a subsequent Backout.

1) DFS395 BACKOUT COMPLETE FOR PSB name (TO CHKPT id)

   This is the normally expected message indicating a successful execution. If the input control card specified a checkpoint ID, the ID is printed in the message.

2) U963 UNABLE TO OPEN SYSIN

   This is probably a JCL error. Correct it and re-execute Backout.

3) DFS396 SYSTEM ERROR DURING BACKOUT OF DATA BASE name

   The wrong PSB was specified or has been changed. If wrong PSB, specify the correct one and re-execute Backout. If the PSB has been changed since the failure, it must be changed back via a PSBGEN and Backout re-executed.

4) DFS397 BACKOUT UNABLE TO OPEN DATA BASE name

   This is probably a JCL error. Correct it and re-execute Backout.

5) DFS398 I/O ERROR DURING BACKOUT IN DATA BASE name

   Enter the DATA BASE RECOVERY procedure. Input to this recovery must be all log tapes produced since the last Image Copy including the one being created during this execution. After Recovery re-execute Backout.

6) DFS399 BUFFER POOL TOO SMALL FOR BACKOUT OF DATA BASE

   Increase the data base buffer pool specification in the PARM= of the EXEC card and re-execute Backout.

7) DFS400 BACKOUT UNABLE TO OPEN THE INPUT LOG TAPE

This is probably a JCL error. Correct it and re-execute Backout.

8) DFS888 NO DATA BASE RECORDS READ FOR PSB.

The program performed no data base updates or the wrong log tape was used as input. In the latter case, supply the correct tape and re-execute Backout.

9) DFS890 BLKSIZE NOT SUPPLIED FOR IMSLOGR DD CARD

Supply the DCB BLKSIZE parameter for the input log data set and re-execute Backout.

10) DFS894 INVALID RECORD ENCOUNTERED ON INPUT LOG TAPE

This is probably caused by using an input log tape which had not been properly closed. Enter the LOG TERMINATION procedure to close or recover the log tape, and re-execute Backout.

11) DFS896 INCOMPLETE SPANNED RECORD ON INPUT LOG TAPE

A missing log volume, a volume out of sequence, or an unclosed log tape could cause this. Make sure all volumes are accounted for and they were specified in the correct sequence. Correct any errors and re-execute Backout.

12) DFS957 NO CHKPT ID IN SYSIN

The control card is in error. Columns 1-6 do not contain 'CHKPT' or the remainder of the card is blank. Specify the checkpoint correctly or remove the card entirely along with the SYSIN DD statement and re-execute Backout.

13) DFS958 CHKPT NOT FOUND ON LOG

This could be caused by specifying an incorrect checkpoint ID or using an incorrect log tape.

If the log buffers were lost when the log was closed, it is possible that the last checkpoint record was lost. The operator may specify an earlier checkpoint. This checkpoint ID may be obtained from the previous DFS681 message or the log tape may be printed using DFSERA10. Select and print the checkpoint records, then specify the last checkpoint on the log tape. This print will verify that the correct tape was used.

If everything is correct and the checkpoint is on the tape a system error probably exists. It should be reported to the PSR.

14) DFS959 CHKPT NOT WITHIN LAST SCHEDULE OF PGM

This is the result of having two execution of a BMP on one log tape volume. The specified checkpoint ID is located within the first execution and a later schedule record is found. Since it is permissable to backout only to the last checkpoint, remove the control card and SYSIN DD statement. Re-execute Backout.

15) DFS964 INSERT FAILED ON PRIOR UPDATE. RECOVERY REQUIRED

If the program being backed out failed with a U828 this message may be ignored. The U828 occurs when a duplicate Secondary Index insert is attempted and the Secondary Index is defined as having unique entries. The target data base is updated, then a type '52' log record is produced and VSAM is called to insert the Secondary Index. This insert fails due to the duplicate so a U828 is issued. When Backout is executed the '52' with no '50' condition is encountered. Backout of the target data base is done and all other Secondary Index updates are backed out. Since no insert occured, no backout of the insert is required and Recovery is unnecessary.

In all other cases Recovery is required. When the second execution of Backout is performed, the message will again be issued. Ignore the message from the second execution but make sure the ignored one is from a Backout performed after a Recovery.

16) LOG WRITE ERROR

Enter the LOG WRITE ERROR procedure to attempt to recover from the failure. If swapping drives or usage of Dual Log corrects the problem continue with Backout execution.

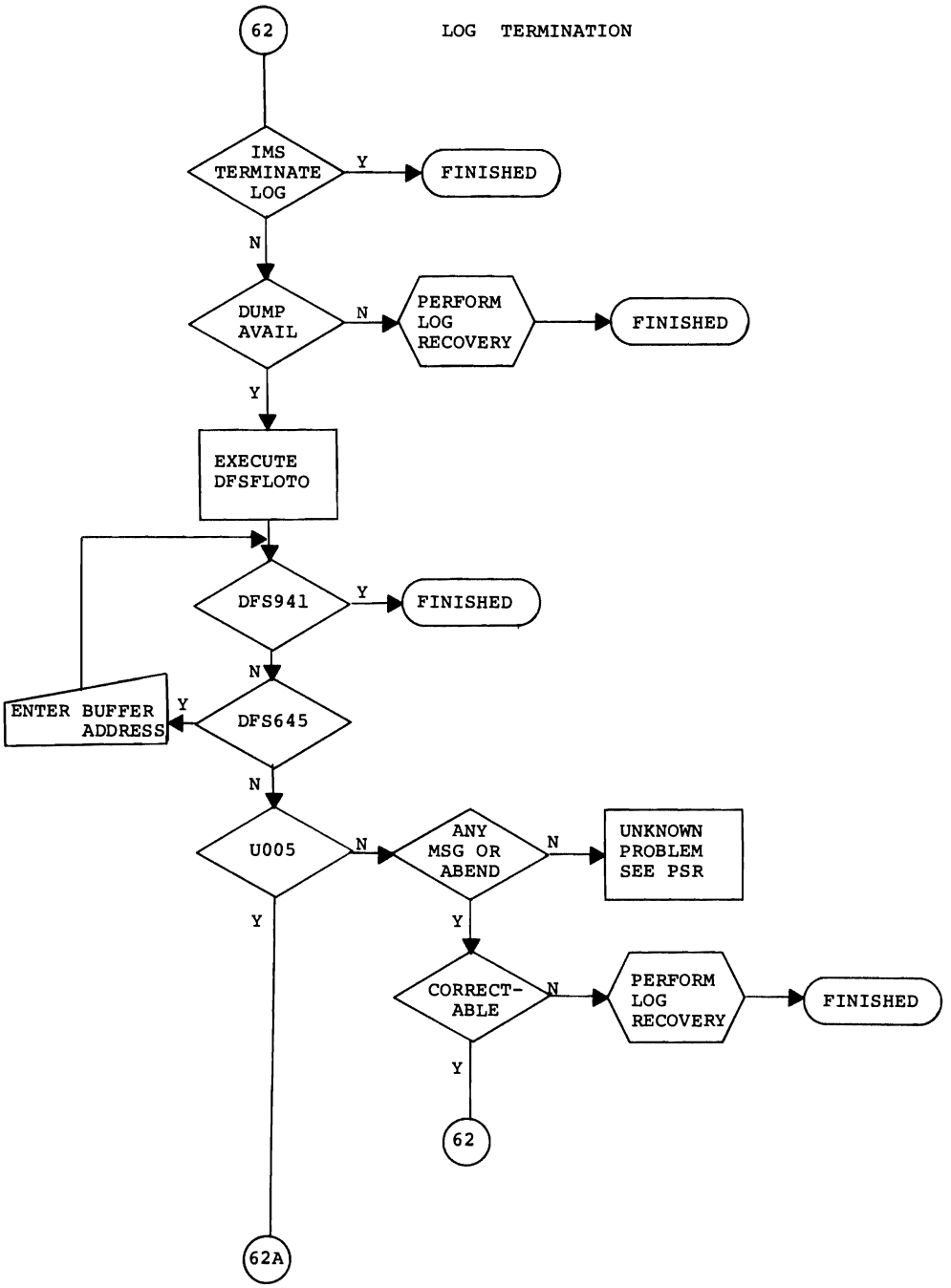
If the problem persists the user must close the output log and re-execute Backout.

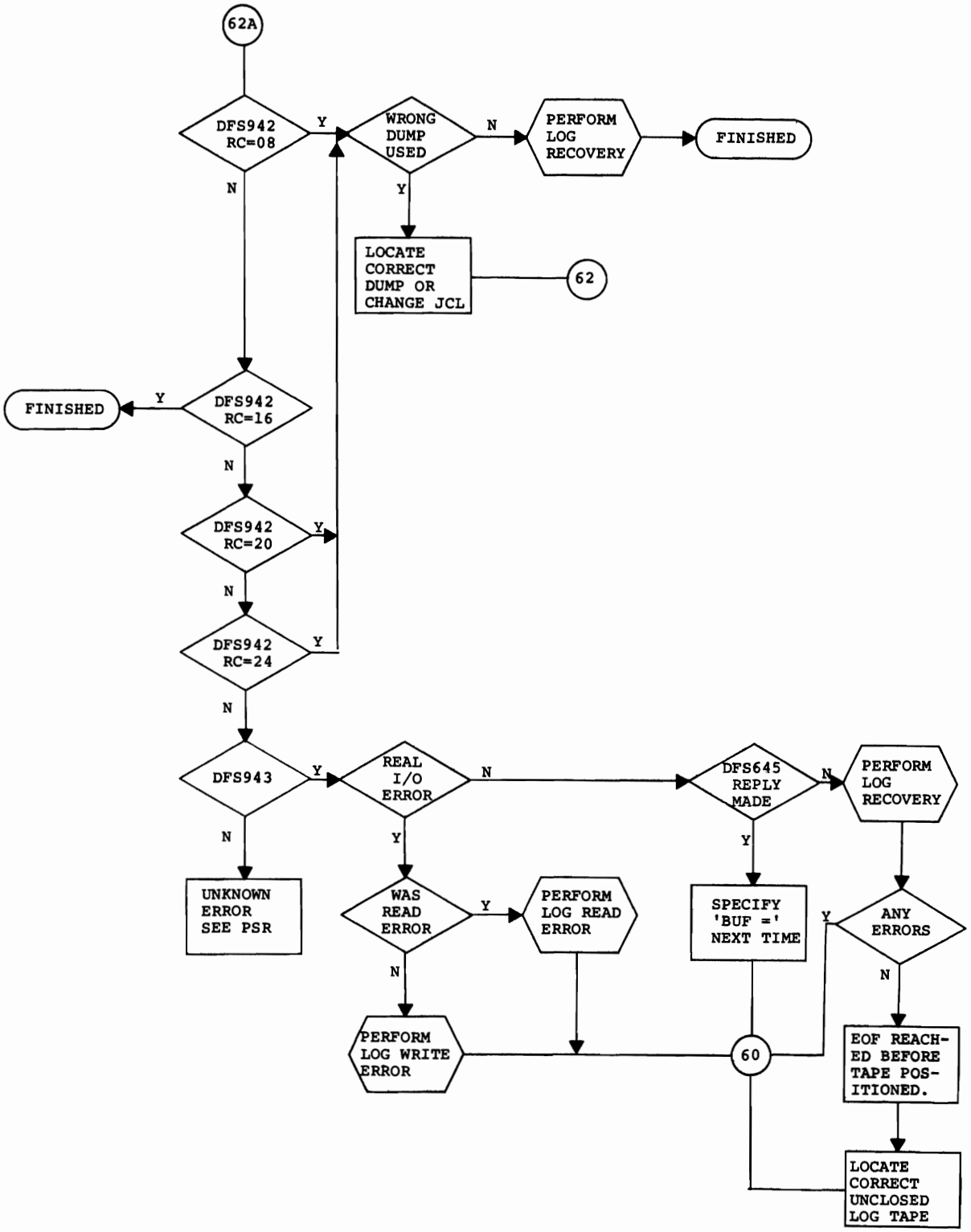
17) LOG READ ERROR

Enter the LOG READ ERROR procedure to attempt to recover from the failure. Upon returning re-execute Backout.

18) UNKNOWN ERRORS.

Contact Systems Programming and the PSR if necessary.


62
IMS TERMINATE LOG
Y
FINISHED
N
DUMP AVAIL
N
PERFORM LOG RECOVERY
FINISHED
Y
EXECUTE DFSFLOTO
DFS941
Y
FINISHED
N
ENTER BUFFER ADDRESS
Y
DFS645
N
U005
N
ANY MSG OR ABEND
N
UNKNOWN PROBLEM SEE PSR
Y
Y
CORRECT-ABLE
N
PERFORM LOG RECOVERY
FINISHED
Y
62
62A


62A
DFS942 RC=08
Y
WRONG DUMP USED
N
PERFORM LOG RECOVERY
FINISHED
N
Y
LOCATE CORRECT DUMP OR CHANGE JCL
62
FINISHED
Y
DFS942 RC=16
N
DFS942 RC=20
Y
N
DFS942 RC=24
Y
N
DFS943
Y
REAL I/O ERROR
N
DFS645 REPLY MADE
N
PERFORM LOG RECOVERY
N
Y
Y
UNKNOWN ERROR SEE PSR
WAS READ ERROR
Y
PERFORM LOG READ ERROR
SPECIFY 'BUF =' NEXT TIME
Y
ANY ERRORS
N
N
PERFORM LOG WRITE ERROR
60
EOF REACHED BEFORE TAPE POSITIONED.
LOCATE CORRECT UNCLOSED LOG TAPE

PROCEDURE 62. LOG TERMINATION

This procedure should be entered whenever a system failure is such that the log tape was not properly closed. If any doubt exists relative to whether or not the log tape was closed, perform this procedure.

1) DID IMS TERMINATE THE LOG

If the operator is certain the log was closed, there is no requirement to perform this procedure.

2) IS A DUMP AVAILABLE

DFSFLOTO requires a SYS1.DUMP or a Standalone Dump data set.

3) DFS941 SYSTEM LOG CLOSED

This is the expected end of job message. It indicates successful execution.

4) DFS645 ENTER BUFFER ADDRESS IN HEX

The operator must enter the log buffer address. This is obtainable from the Operations Master Log or from the DFS810 message issued at the beginning of the last execution of IMS/VS.

5) U005 DFSFLOTO WAS UNABLE TO CLOSE THE LOG

If this abend is not received look for other ABENDS or messages and make corrections accordingly. After correcting the problem DFSFLOTO may be executed again.

If the condition is uncorrectable, enter the LOG RECOVERY PROCEDURE to close the log tape.

6) DFS942 SYSTEM LOG NOT CLOSED - (08,20,24)

These are probably caused by using the wrong Dump data set, a JCL error, or trying to close the wrong log tape.

Correct any errors and re-execute DFSFLOTO or enter the LOG RECOVERY PROCEDURE to close the log tape.

7) DFS942 SYSTEM LOG NOT CLOSED - 16

The log DCB was not open in the Dump data set therefore the log should have been closed. If the log really was not closed perhaps the wrong Dump data set was used.

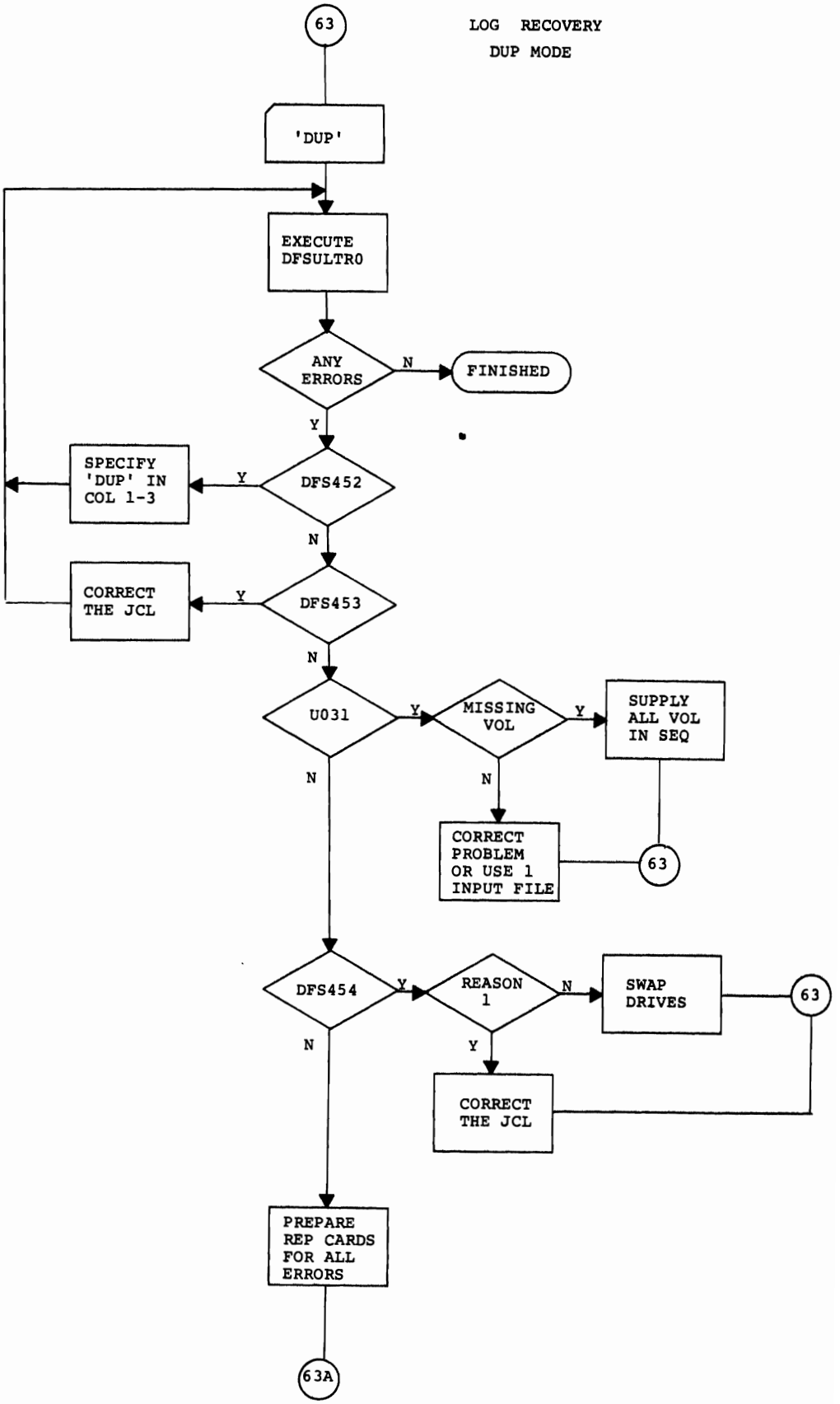
8) DFS943 ERROR ON LOG TAPE (READ,WRITE) ERROR

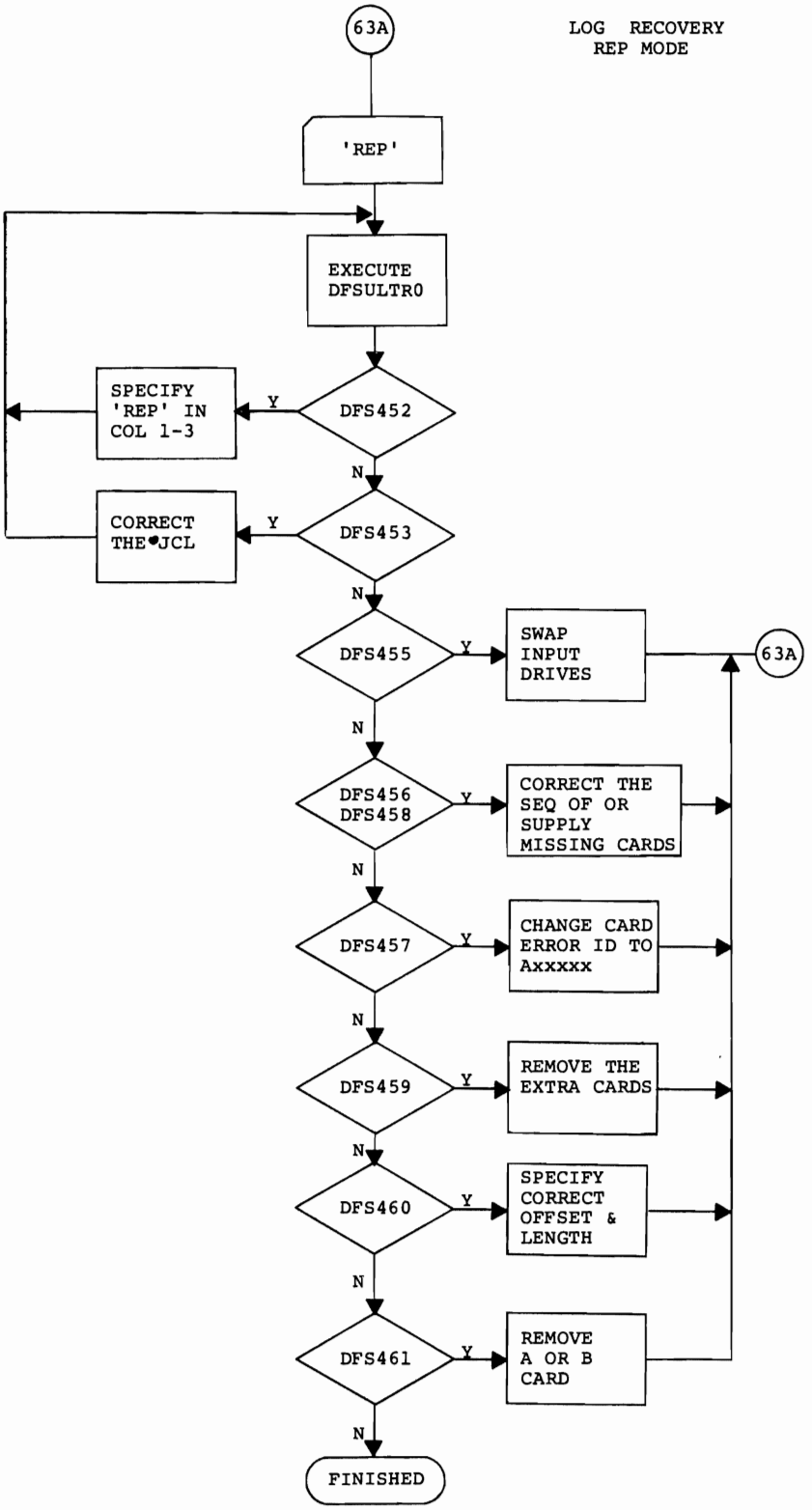
If a real I/O error was encountered the system console should display an OS/VS message. The two procedures, LOG READ ERROR or LOG WRITE ERROR should correct these. DFSFLOTO may then be re-executed.

Another possible reason for this message would be failure to specify BUF= in response to the DFS645 message. If so re-execute DFSFLOTO and reply 'BUF=xxxxx' to the DFS645 message.

A third reason for the message is a blocksize or record length error. The LOG RECOVERY PROCEDURE should find and correct any such errors. Once corrected, re-executed DFSFLOTO.

The fourth possible reason for this message is reaching an EOF on the input log prior to finding the log record which preceeds the log record in the Dumped buffers. About the only thing that could account for this is using the wrong log tape for input.


63
LOG RECOVERY
DUP MODE
'DUP'
EXECUTE DFSULTR0
ANY ERRORS
N
FINISHED
Y
SPECIFY 'DUP' IN COL 1-3
Y
DFS452
N
CORRECT THE JCL
Y
DFS453
N
U031
Y
MISSING VOL
Y
SUPPLY ALL VOL IN SEQ
N
N
CORRECT PROBLEM OR USE 1 INPUT FILE
63
DFS454
Y
REASON 1
N
SWAP DRIVES
63
N
Y
CORRECT THE JCL
PREPARE REP CARDS FOR ALL ERRORS
63A


63A
LOG RECOVERY
REP MODE
'REP'
EXECUTE
DFSULTR0
DFS452
Y
SPECIFY
'REP' IN
COL 1-3
N
DFS453
Y
CORRECT
THE JCL
N
DFS455
Y
SWAP
INPUT
DRIVES
63A
N
DFS456
DFS458
Y
CORRECT THE
SEQ OF OR
SUPPLY
MISSING CARDS
N
DFS457
Y
CHANGE CARD
ERROR ID TO
Axxxxx
N
DFS459
Y
REMOVE THE
EXTRA CARDS
N
DFS460
Y
SPECIFY
CORRECT
OFFSET &
LENGTH
N
DFS461
Y
REMOVE
A OR B
CARD
N
FINISHED

PROCEDURE 63. LOG RECOVERY

This procedure should be entered whenever a log tape cannot be read or when the Log Terminator fails to close the log data set.

The input control card must specify DUP for the first execution, and REP for the second. Two executions are required when any errors are discovered during the DUP operation.

DUP MODE EXECUTION.

If dual log data sets are available, both data sets should be input. Should any errors be encountered on one data set the corresponding record can be found on the alternate. In this manner any possible errors are corrected even if the only purpose of this execution is to close a tape.

1) ANY ERRORS

   If no error messages are received the output log data set may now be used.

2) DFS452 CONTROL CARD ERROR

   A control card was missing or did not contain DUP in columns 1-3. Specify this and re-execute DFSULTRO

3) DFS453 UNABLE TO OPEN ddname

   This is probably a JCL error. Correct it and re-execute DFSULTRO.

4) U0031 UNABLE TO POSITION ALTERNATE TAPE

   Several problems could cause this (see Messages and Codes). Most likely it's a missing input volume. All volumes must be supplied in sequence.

5) DFS454 UNSUCCESSFUL ALTERNATE TAPE POSITION

   If the reason is 1, it's probably a JCL error. If the reason is 2, swapping input drives will probably correct the problem. Re-execute DFSULTRO.

6) PREPARE REP CARDS

   All error conditions should be noted on the SYSPRINT data set. The user must prepare REP cards to replace error data for these conditions.

REP MODE EXECUTION.

The input to this execution is the interim log tape which was produced during the DUP execution. This log tape contains the copied log records and error ID records preceeding each error block. For that reason this tape should never be input to a DUP execution.

1) DFS452 CONTROL CARD ERROR

   A control card was missing or did not contain 'REP' in columns 1-3. Specify this and re-execute DFSULTRO.

2) DFS453 UNABLE TO OPEN name

   This is probably a JCL error. Correct it and re-execute DFSULTRO.

3) DFS455 I/O ERROR ON INPUT TAPE DURING REP

   If the problem is hardware, swap to a new tape drive. If the error is repeated go back to the DUP execution and recreate the tape.

4) DFS456-DFS458 MISMATCHED CARD AND TAPE ID - SEQ ERROR

   The input tape contains an error ID record for which a matching REP card was not found. Either a card was omitted or the cards are out of sequence. Correct the problem and re-execute DFSULTRO.

5) DFS457 CARD SPECIFIED B RECORD - NO B RECORD ON TAPE

   The error ID sequence number should be changed to Axxxx. Re-execute DFSULTRO.

6) DFS459 EOF ON INPUT TAPE WITH CARDS YET TO PROCESS

   Too many control cards were supplied and are not needed.

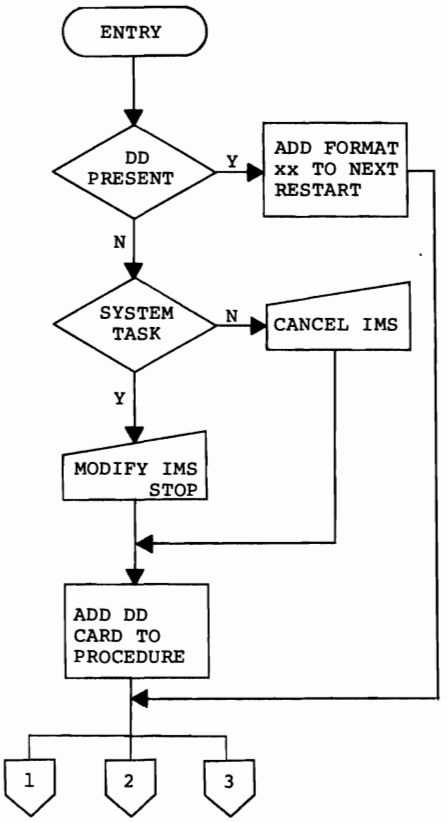
7) DFS460 MOVE WILL NOT FIT IN RECORD

   Either the record offset or the amount of replace data was incorrect. Specify the correct offset or supply less data and re-execute DFSULTRO.

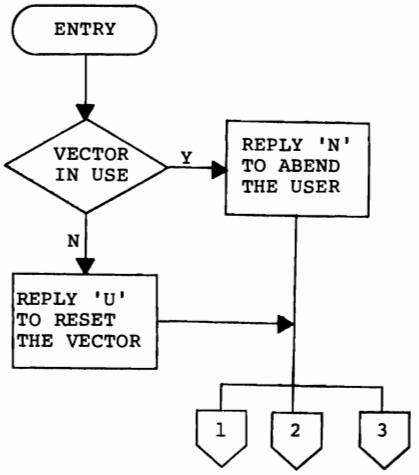
8) DFS461 USER HAS SPECIFIED REP IN BOTH A AND B RECORDS

   Only one of the error records can be corrected. The system will drop the other. Remove one of the cards and re-execute DFSULTRO.

DFS986 ROUTINE



DFS801 ROUTINE



SAMPLE FORMS

The following are sample forms which could be used to record the information necessary to perform a recovery. The OPERATIONS MASTER LOG contains the information necessary to:

1) Determine the type of restart necessary.

2) Determine the type of IMS/VS termination and the checkpoint ID to be used at restart.

3) Account for all log volumes and the creation sequence from initialization to termination.

4) Supply the checkpoint ID when restarting BMPs.

5) Supply the buffer address to the Log Terminator if necessary.

The IMS TAPE CONTROL form contains a record of all Image Copies, all Log volumes, and all Change Accumulations. This information is essential when performing a data set recovery.

These forms or something similar, should be mandatory in every IMS/VS installation. Naturally, as with any form, the value is determined by the consciousness of the people recording the data. The data on these forms can save much time and avoid many operational mistakes during a recovery.

OPERATIONS MASTER LOG

START DATE/TYPE	START TIME	BUFFER ADDRESS	SECURITY SPECIFIED PASS	SECURITY SPECIFIED TERM	STOP DATE	STOP TIME	STOP CKPT/TYPE	LOG SERIAL	BMP RESTART ID
NRE-Ø 4-28-77	Ø8:37	B2CØØØ	Y	Y				LOGØØ1 LOGØØ2	
					4-28-77	12:50	ABEND SØ33		
ERE 4-28-77	13:1Ø	B2BØØØ	N	N				LOGØØ3 LOGØØ4	BMP1ØØ1 BMP2ØØ5
					4-28-77	16:00	FREEZE 77118/ 160122	LOGØØ5	
NRE 4-29-77	Ø8:15	C4BØØØ	N	N				LOGØØ6 LOGØØ7	
					4-29-77	13:ØØ	DUMPQ 77119/ 130244	LOGØØ8	
BATCH 4-29-77	14:00				4-29-77	15:00		LOGØØ9	

IMS TAPE CONTROL

LOG VOLUMES			CHANGE ACCUMULATION		IMAGE COPY		
DATE CREATED	SERIAL NUMBER	REGION TYPE	DATE	SERIAL NUMBER	SERIAL NUMBER	DATE/TIME	DATASET
4-28-77	LOG001	DC			ICDSØ1	4-27-77	DSØ1
	LOG002				ICDSØ2	"	DSØ2
	LOG003				ICDSØ3	"	DSØ3
	LOG004						
	LOG005						
4-29-77	LOG006						
	LOG007						
	LOG008						
	LOG009	BATCH					
			4-29-77	CumØØ1			
				CumØØ2			