To: Distribution

From: T. H. Van Vleck

Date: November 18, 1975

Subject: Storage System Error Recovery

## INTRODUCTION

This memorandum describes the error recovery strategies which will be used in the new Multics Storage System. The reader is assumed to be familiar with the operation of the current system.

## Causes, Symptoms, and Remedies

When an error occurs, it manifests itself to the supervisor as a symptom, such as a nonzero major status or a return to BOS. A symptom may have many possible causes; for example, a nonzero major status may reflect a piece of grit on a disk surface, or a loose wire in the disk controller, or a software error storing garbage in the IOM mailbox. The supervisor programs do not deal with causes -- they deal with symptoms, and attempt to find remedies for symptoms. Discussion of causes is useful in order to enumerate possible symptoms and to decide on remedies, but it is unnecessary to enumerate all possible errors, since the system only cares about correcting the errors it detects.

MTB-220 describes the design of the salvager, which shares our basic policy of providing repair procedures, integrated into the system, which will be invoked when damage is discovered.

If a process encounters a crawlout while it has a directory locked, and if dir.modify equals the process ID, then the directory is salvaged. If any entries are lost by the salvager, a retrieval request for the directory will be queued automatically; when a copy of the directory has been retrieved, the directory and the copy will be merged to recover the missing entries.

## STRATEGIES

### Disk Errors

When a disk error status is returned, the current system prints an error message via SYSERR, and retries the operation. If the error status is for a device attention, the operation is retried until the status goes away. Otherwise, the operation is retried three times and then page control is informed of a fatal error. For read errors, page control signals page_fault_error to the user process. For write errors, the system abandons the bad address and assigns another address.

The new strategy is more complicated. When disk_control encounters an error, it consults a set of tables for the device type being used. (These tables are currently shared by all supported disk types.) The table entry for the type of error gives the following information:

maximum number of retries
whether to reseek

whether to read detailed status

address is bad
data path is bad
disk or disk drive is bad

The operation is retried without comment until the retry count runs out. If the error persists, the error interpretation specifies whether to assume that the disk address, the data path to the disk unit, or the disk unit itself is unusable. If too many bad address errors occur, the system will assume that the data path to the disk unit is unusable. If all data paths to a disk are unusable, the system will decide that the disk unit is inoperative. When a disk is marked inoperative, a flag is set on in its PVT entry indicating this. Attempts to use a disk unit marked inoperative will cause an error code to be returned. A special interrupt for a device marked inoperative will reset the flag.

When page control is informed of a disk error on a read, the segment will be set "page control out of service" (pcos). Attempts to reference a segment which is pcos will fail: the page fault will be transformed into a segment fault. Attempts to segment fault on a segment which is pcos will also fail: they will be transformed into a signal of the seg_fault_err condition, with a code whose message is "Segment unusable due to I/O error."

The pcos flag will be a regular VTOC attribute, so that if a RWS error damages a segment but nobody references it for several days, we still will not give the user an inconsistent segment without warning.

The user will be able to turn off the pcos switch for his segment by a call to hcs_. It is then his responsibility to verify the segment's contents before using it. The on-line salvager will reset the pcos switch of a directory before touching it.

For write errors, if the address is unusable, page control will ultimately cause a new address to be assigned and retry the write. Automatic alternate track assignment is not used for Multics operation, and Multics disks are formatted without alternate tracks, mostly for performance reasons: two extra revolutions would be required for use of an alternate track. It is cheaper to scrap the whole record (16 sectors) if any sector is in error. Each disk pack has a reserved area for the logging of errors. The details of how errors will be logged and how the volume salvager will know that a bad record is not to be released into the paging pool have not yet been worked out. An interim strategy is to cause the volume salvager to zero and reread all records found protected but not in any file map; if the record is still unusable this operation will detect it and leave it still unassigned.

## Disk Data Path Failure

Currently we require that any channel be able to reach any disk device. The maximum number of data paths to a given device is four, if we are running in a dual controller, dual channel, environment with 4 PSI links. In the current disk dim, once a request is set up for a channel, it cannot be moved to some other channel, because the queue entry is scrapped. This strategy will be modified in a future disk control, so that when a data path to a disk goes down, the system will attempt to find an alternate path to the device. If all paths to the device are down the device will be marked inoperative.

## Bulk Store Errors

Currently, if the bulk store cannot be written, a message is typed to the operator and the block of bulk store is deconfigured. If bulk cannot be read for a page fault, page_read_error is raised. If bulk cannot be read for an RWS, a message is printed for the operator and the RWS is abandoned,

leaving the user with an old page.

The new system will improve on this strategy slightly. For RWS errors, the segment will be set page control out of service.

## Volume Salvager

The volume salvager will set the pcos switch on for any non-directory which had a reused address correction and so had a page of zeroes inserted into it. Reused addresses involving directories will be handled by salvaging the directory. If the salvage shows no errors in the directory, the page is awarded to the directory. Otherwise, the directory is rebuilt without reference to the reused page (which gets zeroed).

## Emergency Shutdown

The current emergency shutdown attempts to deactivate all segments in order to force all file maps out of core. Since a segment cannot be deactivated if it has active sons, certain counting errors in segment control can prevent emergency shutdown from completing.

Emergency shutdown can be described as a process of flushing the several caches which hold the contents of disk while the system is running. The new system's ESD first attempts to flush the paging device. It then flushes core, the AST, and the VTOC buffers. The new system treats the AST as an array and forces the update of the VTOCE's one by one, without checking the inferior count.

## Paging Device Flushing

Installations which have a paging device know that about the worst crash that can happen is to be unable to flush the paging device. With the new storage system, if a single volume goes down, we may be in the position of having a partially successful flush of the paging device.

The paging device map contains PVT indices and record addresses for the pages on the paging device. If a system crash occurs such that the paging device cannot be flushed, and if packs are then moved to different drives, an attempt to flush the paging device would lead to a major disaster.

When the new system is booted after a crash, initialization examines the root label and discovers that a paging device was active during the last bootload and that it has not been flushed. It therefore knows that the paging device cannot be used for Multics operation until it is completely flushed. A message informing the operator of this fact is typed and the paging device is disabled. The paging device map contains the time of bootload in its header, so that the system can check that it is flushing onto a volume which actually had pages left on the paging device. Whenever a volume is normally accepted, the time of bootload and the PVT index are recorded in the volume label. A normal dismount of a volume clears these fields.

As each volume is added to the storage system's configuration, a check is made to see if it was in use during the bootload which left the unflushed paging device. If a volume has pages on the unflushed paging device, the paging device map is searched for pages which should be moved down to the volume, and these pages are flushed down before the volume is accepted for normal paging. Once all such pages have been removed from the paging device, the PD map is updated to show these records as free and the volume is accepted normally.

Salvaging after a crash which left an unflushed paging device therefore proceeds more or less normally, with the additional changes that the paging device is not used, and that the paging device is flushed incrementally for each volume (after volume salvaging if neccesary) when the volume is accepted. Special operator commands will be available in the ring 1 environment to check whether the paging device is completely flushed, to dump the paging device on tape, to clear a partially flushed paging device, and to start the use of the paging device for normal paging. Pages on the paging device which have not been modified since they were read from disk, and pages which are images of hardcore partition pages, will simply be discarded; a special sweep through the paging device map takes care of this at the outset.

When the operator attempts to exit from the ring 1 environment, the system will check that the paging device has been completely flushed. If not, a message of the form

UNFLUSHED INFORMATION REMAINS ON PAGING DEVICE

will be printed and the system will return to ring 1 command level. The operator will be able to request a list of volumes which have not yet been flushed; he must either mount each volume so that the PD can be flushed, or instruct the system to discard the PD pages for the volume (the latter action might be reasonable if the volume had to be reloaded).