

TO: Distribution
FROM: M. L. Goudy
DATE: 30 September 1973
SUBJECT: Operation of the Multics Salvager

This operation applies to Multics Operation on the Multics Series Model 6180.

Discussion

The Multics Salvager is designed to be run after a normal Multics system shutdown or after a system crash and attempted emergency shutdown. It is run by bootloading a special Multics System Tape containing the Salvager System. The Multics Salvager will scan through the Multics directory hierarchy, noting and correcting all error conditions. The salvager has four modes of operation; FAST, ACTIVE, REGULAR and LONG. The mode of operation is a function of the successfulness of the previous shutdown (emergency or normal sequence.) The modes are described below.

- A. FAST- If the shutdown or emergency shutdown was successful the fast mode of operation is used. No directories are touched. Certain system variables are modified.
- B. ACTIVE- If emergency shutdown part 1 was successful then the active mode of operation is used. Only these directories that were active at the time of the crash will be scanned and their errors corrected.

- C. REGULAR - If the shutdown sequence failed (e.g., locking errors in wired_shutdown) or was not run, then this mode of operation is used. In the regular mode of operation, all directories are scanned and errors are corrected, but directories are not reformatted unless a serious error condition exists.
- D. LONG - If it is desired to reformat all the directories in the Multics hierarchy this mode of operation is used.

Most error messages from the Salvager are printed on the on-line printer. Serious errors which are detrimental to the Multics directory hierarchy are printed on the operator's typewriter. Should one of these occur, a programming staff member should be notified. When the Salvager has completed its operation, the Multics directory hierarchy should be "clean", and, if no serious errors have occurred, the Multics System may be re-booted.

Processor Switches

The processor data switches located in the processor maintenance panel control the mode of operation of the Salvager. They must be set before bootloading the Salvager. Switch 0 controls the printing of segment names in the Multics directory hierarchy. When Switch 0 is up, the names of all directories scanned will be printed. When it is down, only those segments in which the Salvager finds error conditions will be identified. Normally, the Salvager is to be run with Switch 0 in the down position except when operating in the long mode. Switch 1 controls the regular and long modes of operation. It also overrides the software indicators which control the ACTIVE and FAST modes of operation. When

switch 1 is on at the start of a salvage the long mode of operation will be started. Should it then be turned off, the regular mode of operation will be entered. Normally, the Salvager is to be run with Switch 1 in the down position. Switch 3 controls whether the salvager output will be directed to the on-line printer. If switch 3 is up, printer output will be suppressed. This feature is useful if the line printer is not operational.

Operation

1. Mount the current Salvager MST (as found in the current system log) on drive 2.
2. With the system in BOS waiting for a command, type "SALV". The Salvager MST should start to read in. Enter the time in the Multics log as "SALV".
3. After about one minute, the operator's typewriter should type "BEGIN" (Mode name) SALVAGING". The Salvager is now in operation.
4. When the Salvater finishes operation it will type "END (mode name) SALVAGING" on the operator's typewriter. If the Salvager loops or returns to BOS with or without an error message on the operator's typewriter, follow the instructions located under "Errors Fatal to the Salvager".
5. Examine the operator's console output for any error listed under "serious errors" below. If one or more such errors were noted during the ACTIVE mode of operation, contact the programming staff, and rerun the salvager under their direction.
6. File the Salvager output in the designated output area.
7. Bring up the Multics System.

Note: Any and all errors reported for the segment ">pdđ" or ">process_dir_dir" and for segments contained in this directory are to be considered non-serious errors and are not to be reported to the programming staff.

Errors Fatal to the Salvager

Should the salvager loop or return BOS without completing its BEGIN-END cycle the following should be done:

1. A complete dump should be taken.
2. The ESD command should be issued in BOS to shut the salvager down.
3. The salvager should be restarted.

Should the failure occur again, the above procedures should be repeated and a programming staff member should be contacted.

Salvager Messages

Output from the Salvager is divided into two parts: that appearing on the operator's console and that appearing on the printer, or in the segment ">online_salvager_output". The messages in this document are presented in the following format. On the first line is a number which is the number of the salvager message; this is followed by the message exactly as it appears on the operator's console. The second line gives the severity (either 0, 1, or 2) and the segment of the salvager which causes the message to be issued. The meaning of severity is as follows:

<u>severity</u>	<u>significance</u>
Ø	No action by operator.
1	Error will crash the salvager.
2	Possible omissions that will be noticed by the user when he logs in.

Serious Errors on the operator's console.

Programming staff should be contacted if any of the following errors appears.

s1. ENTRIES MAY BE LOST.

severity: Ø source: salvage_directory

Entry thread has been repaired, but number of entries found is not equal to number that directory had before repair.

s2. FATAL CHANGE TO DIR FILE MAP, DO A DUMP W, ESD, AND SALVAGE OVER.

severity: 1 (crash salvager) source: salv_rebuild_directory

Unexpected interaction with page control, a bug exists. The directory will be deleted by the next salvage.

s3. FATAL DIRECTORY ERROR.

severity: Ø source: salvage_directory

Message on printer will indicate reason, directory will be deleted.

s4. FATAL DIRECTORY GROWTH, DO A DUMP W, ESD, AND SALVAGE OVER

severity: 1 (crash salvager) source: salv_rebuild_directory

Instead of shrinking a directory, the Salvager has increased its size, a bug. Directory will be deleted by next salvage.

s5. FATAL DIRECTORY MAP MODIFICATION, DO A DUMP W, ESD, AND SALVAGE OVER

severity: 1 (crash salvager) source: salv_activator

Unexpected interaction with page control, a bug exists. Directory will be deleted by next salvage.

s6. FATAL ENTRY ERROR

severity: Ø source: salvage_entry

Message on printer will indicate reason and entry name. The entry will be deleted when the containing directory is rebuilt.

s7. FATAL ENTRY ERROR.

severity: \emptyset source: salvage_entry

Message on printer will indicate name of containing directory and entry's offset. The entry will be deleted when the containing directory is rebuilt.

s8. INIT_SALV_FSDCTS: NEED A a1 FOR THE a2 PARTITION

severity: 1 (crash salvager) source: init_salv_fsdcts

a1 is the name of a device and a2 is the name of the partition. Configuration cards may be wrong.

s9. INIT_SALV_FSDCTS: a1 PARTITION NOT FOUND

severity: 1 (crash salvager) source: init_salv_fsdcts

a1 is the name of the partition. Configuration cards may be wrong.

s10. ON_LINE SALVAGER: ATTEMPTED RESALVAGE OF p

severity: 1 (crash Multics) source: on_line_salvager

p is the address of a directory which is causing a system loop.

s11. ON_LINE_SALVAGER: ERROR FROM SUM ON p

severity: 1 (crash Multics) source: on_line_salvager

p is the address of a directory whose branch cannot be found.

s12. ON_LINE_SALVAGER: ERROR ON PATHNAME o

severity: 1 (crash Multics) source: on_line_salvager

o is the octal error returned because file system was unable to obtain pathname of segment when given its segnumber.

s13. ON_LINE_SALVAGER: LOCK ERROR

severity: 1 (crash Multics) source: on_line_salvager

Unable to lock directory to be salvaged. Lock arrays
may have become garbaged.

s14. ON_LINE_SALVAGER: SALVAGING OF A LEVEL 1 DIRECTORY NOT DONE

severity: 1 (crash Multics) source: on_line_salvager

Cannot salvage the root on line because it has no branch.

s15. SALV_PRINT: USABLE TO INITIATE OUTPUT SEGMENT o.

severity: 1 (crash Multics) source: salv_print

o is the octal error returned when attempting to initiate
the segment "online_salvager_output".

s16. REUSED ADDRESS

severity: \emptyset source: salv_check_map

Salvager has zeroed out a page that was reused and
nulled out the address in this segment's file map.

Normal Messages on operators console. (No action need be taken.)

s17. BEGIN a SALVAGING.

severity: \emptyset source: salvager

a indicates which salvaging mode is being used, i.e.
active, regular, etc.

s18. BEGIN PAGING DEVICE FLUSH.

severity: \emptyset source: salvager

s19. END OF a SALVAGING.

severity: \emptyset source: salvager

a is the mode of salvaging.

s20. END PAGING DEVICE FLUSH

severity: Ø source: salvager

Successfully flushed paging device.

s21. ON_LINE_SALVAGER: BEGIN SALVAGING OF DIRECTORY p1 FOR p2

severity: Ø source: on_line_salvager

Message to operator that an on_line_salvage is attempting to clear up a system inconsistency. p1 is the pathname and p2 is the process id.

s22. ON_LINE_SALVAGER: ERROR o APPENDING s TO DUMP DIRECTORY.

severity: Ø source: on_line_salvager

o is the octal error code, s is the segment name. Will not be able to see the structure of the bad directory before the on_line_salvager corrects it.

s23. ON_LINE_SALVAGER: ERROR o INITIATING s.

severity: Ø source: on_line_salvager

o is the octal error, s is the segment name. See s22.

s24. ON_LINE_SALVAGER: SALVAGING COMPLETED

severity: Ø source: on_line_salvager

Successfully finished on_line_salvaging.

s25. ON_LINE_SALVAGER: UNABLE TO DELETE BAD DIR o

severity: Ø (rings bell) source: on_line_salvager

o is the octal code returned from file system.

s26. ON_LINE_SALVAGER: UNABLE TO DEACTIVATE SUBTREE o
ON_LINE_SALVAGER: NOW SETTING SALVAGED DIRECTORY OUT OF SERVICE

severity: Ø (rings bell) source: on_line_salvager

o indicates octal code preventing the deactivation of the subtree.

s27. ON_LINE_SALVAGER: USER PROCESS p TERMINATED BECAUSE OF
BAD PROCESS DIRECTORY

severity: 2 (process terminated) source: on_line_salvager

p is the name of the process id.

s28. SEGMENT: a

severity: \emptyset source: salv_name

a is the pathname of the segment. This identification
always precedes any error comments.

s29. ZERO FILE MAP WORD.

severity: \emptyset source: salv_check_map

This segment has a file map word that was zeroed out.
A bug may exist somewhere in hardcore.

Messages on Salvager Printouts

The following messages appear only on salvager printouts. Severity will be indicated as either \emptyset , 1 (fatal), or 2 (user should be warned). A \emptyset severity requires no action, a severity 1 is also printed on the operator's console, while a severity 2 indicates possible omissions that a user would notice the next time he logs in.

1. Acl count changed from d1 to d2

severity: \emptyset source: salvage_entry

d1 and d2 are decimal numbers. This segment's acl count has been changed to indicate the current number of acl entries and the containing directory will be rebuilt to reflect this change.

2. Acl threads bad, deleting acl.

severity: 2 (user warning) source: salvage_entry

An unrecoverable error has been found in this entry's acl. For protection purposes this acl will be deleted and the containing directory rebuilt.

3. Address out-of-bounds: addr = o, dev = d1, page # = d2.

severity: \emptyset source: salv_check_map

An address in this segment's file map is out-of-bounds.
Address has to be reset to null.

4. Begin salvaging of directory p1.
For p2.

severity: \emptyset source: on_line_salvager

p1 is the pathname while p2 is the process id. This message appears at the start of an on_line_salvage.

5. Begin a salvaging.

severity: \emptyset source: salvager

a is the mode of salvaging. This message is printed in conjunction with sl7.

6. Begin paging Device Flush.

severity: \emptyset source: salvager

This message is printed in conjunction with sl8.

7. Cacl count changed from d1 to d2.

severity: \emptyset source: salvage_directory

d1 and d2 are decimal numbers. The directory will be rebuilt to reflect this change to the cacl.

8. Cacl thread error, cacl deleted.

severity: 2 source: salvage_directory

An unrecoverable error has been found in the cacl. For protection purposes, the cacl will be deleted and the directory rebuilt.

9. Component: a. Acl.name.person, deleting acl.

severity: 2 source: salv_rebuild_acl

a is the pathname of the segment.
The person part of an access name has an unrecoverable error. For protection purposes, the acl will be deleted and the containing directory rebuilt.

10. Component: a. Acl.name.project, deleting acl.

severity: 2 source: salv_rebuild_acl

a is the pathname of the segment. The project part of an access name has an unrecoverable error. For protection purposes, the acl will be deleted and the containing directory rebuilt.

11. Component: a. Allocation errors, deleting acl.

severity: 2 source: salv_rebuild_acl

a is the pathname of the segment. Unable to allocate enough room to reconstruct entire acl. For protection purposes the act will be deleted and the containing directory rebuilt.

12. Component: a. All names replaced by unique name.

severity: 2 source: salv_rebuild_names

a is the pathname of the segment. All names for this segment have been replaced by a unique name due to unexpected errors while hashing. The containing directory will be rebuilt.

13. Component: a. Author encode error o, replaced with "SysDaemon".

severity: \emptyset source: salv_rebuild_directory

a is the pathname of the segment, o is the octal error code. An error has occurred in the reconstruction of the author. The name "SysDaemon" will be used instead, and the containing directory will be rebuilt.

14. Component: a. Bc_author error o, entry auth used.

severity: \emptyset source: salv_rebuild_directory

a is the pathname of the segment, o is the octal error code. An error has occurred in the reconstruction of the bit-count author. The author of this segment will be used as the bit-count author as well, and the containing directory will be rebuilt.

15. Component: a. deleted in directory rebuild

severity: 2 source: salv_rebuild_directory

a is the pathname of the segment. An unrecoverable error has occurred, forcing the deletion of this segment during the rebuilding of the containing directory.

16. Component: a1. Name a2 removed as it already exists in directory.

severity: Ø source: salv_rebuild_names

a1 is the primary name of the segment, a2 is the name that already exists on some other segment. This name will be removed (from a1) and the directory rebuilt.

17. Component: a1. Name a2 removed due to duplication.

severity: Ø source: salv_rebuild_names

a1 is the primary name of the segment, a2 is the name that is already on this segment.

18. Component: a. Unable to allocate space for name.

severity: Ø source: salv_rebuild_names

a is the pathname of the segment. Unable to add more names to this segment because directory is out of space. If this is the primary name then an attempt will be made to use a unique name instead. The containing directory will be rebuilt.

19. Component: a. Unable to copy whole list due to error o, deleting acl.

severity: 2 source: salv_rebuild_acl

a is the pathname of the segment, o is the octal error code. Because acl cannot be rebuilt completely, the entire acl is being deleted for protection purposes, and the containing directory rebuilt.

20. Component: a1. Unable to get hash index for a2 due to code o.

severity: Ø source: salv_rebuild_names

a1 is the primary name of the segment, a2 is the name which cannot be successfully hashed due to octal error code o.

21. Component a1. Unable to hash in a2 due to code o.

severity: \emptyset source: salv_rebuild_names

a1 is the primary name of the segment, a2 is the name which cannot be successfully hashed due to octal error code o.

22. Current length changed from d1 to d2.

severity: \emptyset source: salv_check_map

d1 and d2 are decimal numbers indicating the amount of change.

23. Deleting entry.

severity: 2 source: salvage_entry

This message accompanies s6. The reason for deletion is given by the preceding message. The containing directory will be rebuilt.

24. Deleting a.

severity: \emptyset source: salvage_entry

a is the name of a directory that it is necessary to delete before rebooting Multics.

25. Did bad, page dropped: addr = o, dev = d1, page # = d2.

severity: \emptyset source: salv_check_map

The device id part of the address indicates a non existent device. This address will be nulled out.

26. Directory count changed from d1 to d2.

severity: \emptyset source: salvage_directory

Number of directory branches found does not agree with count in the containing directory . Containing directory will be rebuilt.

27. Directory deleted due to aste.fimchanged on.

severity: 1 (crash salvager) source: salv_rebuild_directory

An unexpected reference, such as out-of-bounds, has occurred which may indicate serious trouble with the salvager's environment. This message appears in conjunction with s2.

28. Directory deleted due to file map change.

severity: 1 (crash salvager) source: salv_activator

Since there should be no changes in the file map, serious trouble in the salvager's environment exists. This message appears in conjunction with s5.

29. Directory deleted due to unexpected growth.

severity: 1 (crash salvager) source: salv_rebuild_directory

A rebuilt directory has a larger size than the original. A bug exists in the salvager. This message appears in conjunction with s4.

30. Directory has no header.

severity: \emptyset source: salv_delete_dir

Attempt to free pages used by the subtree defined by this directory failed.

31. Entry name(s) replaced by unique name a.

severity: 2 source: salvage_entry

Attempt to reference the names of a segment failed, and a unique name a will be used to replace all names. The containing directory will be rebuilt.

32. End of a salvaging.

severity: \emptyset source: salvager

a indicates the mode of salvaging. This message appears in conjunction with s19.

33. End Paging Device Flush.

severity: \emptyset source: salvager

This message appears in conjunction with s20.

34. End salvaging of directory

severity: \emptyset source: on_line_salvager

This message appears in conjunction with s24.

35. Entry thread error, count has D1, thread has D2.

severity: \emptyset source: salv_delete_dir

Some entry file maps may not be released in sub-tree deletion due to threading error.

36. Entry uid set to \emptyset .

severity: 2 source: salvage_entry

This entry will be deleted when containing directory is rebuilt. Specific reason is given in preceding statement.

37. Entry was out of service.

severity: \emptyset source: salvage_entry

Entry out of service switch will be turned off and the containing directory rebuilt.

38. Error in forward thread, back tracing will be attempted.

severity: \emptyset source: salv_check_thread

Forward thread cannot be followed further. Backward direction will be used to check thread.

39. Fatal directory error.

severity: 2 source: salvage_directory

Previous message indicates why this directory will be deleted when its containing directory is rebuilt. This message appears in conjunction with s3.

40. Fatal error, rel(ep) = 0

severity: 2 source: salvage_entry

This entry will be deleted when the containing directory is rebuilt. 0 indicates the entry relative pointer because the entry's primary name cannot be established. The reason for deletion is given by the preceding comment. This comment appears in conjunction with s6.

41. File map checksum corrected.

severity: 0 source: salv_check_map

The file map checksum was wrong and therefore corrected.

42. File map size d1 exceeded by o in a pointer.

severity: 0 source: salv_check_ptr

d1 indicates the map size, o the octal value of the pointer. a is the pointer name, as given in appendix 1.

43. Found d1 entries, directory count had d2.

severity: 2 source: salvage_directory

Total of all entry counts in directory differs from number found, directory will be rebuilt. This message appears in conjunction with s1.

44. FSDCT changes:

WORD FROM TO

severity: 0 source: init_salv_fsdcts

This is the header for all device map changes made by the salvager.

45. Illegal address o in a pointer, page d missing.

severity: 0 source: salv_check_ptr

o is the octal address, d is the number of the missing page. a is the pointer name, as given in appendix 1.

46. Illegal device I.D: d

severity: \emptyset source: salv_check_map

The device id found in the entry is wrong. It will be replaced by the correct one from the file map if possible.

47. Improper device address: addr = o, dev = d1, page# = d2.

severity: 2 source: salv_check_map

File map's page d2 has a bad address, it will be replaced with a null address.

48. Inconsistent dates detected and reset: a

severity: \emptyset source: salvage_entry

a is any of: dtm, dtem, dtd, and dtu. Date has been reset to current date.

49. Incorrect allocation area pointer

severity: 2 source: salvage_directory

This directory (and its subtree) will be deleted because its allocation area pointer is wrong and therefore no information in the directory is to be trusted.

50. Inferior account count charged from a1 to a2.

severity: \emptyset source: salvage_directory

The amount of change is given by d1 to d2 (decimal).

51. Initial_dir_acl (d) error, iacl deleted.

severity: 2 source: salvage_directory

d indicates the ring number of the initial acl (\emptyset :7). The initial acl is deleted for protection purposes.

52. Initial_seg_acl (d) error, iacl deleted.

severity: 2 source: salvage_directory

d indicates the ring number of the initial acl. The iacl will be deleted when the directory is rebuilt.

53. Initial acl count changed from d1 to d2.

severity: Ø source: salvage_directory

d1 and d2 give the magnitude of the change. d2 represents the total number of all initial acl entries.

54. Link: a being deleted due to error in size of d.

severity: 2 source: salv_rebuild_directory

a is the pathname of the link, d is the erroneous size. The link will be deleted when the containing directory is rebuilt.

55. Link count changed from d1 to d2.

severity: 2 source: salvage_directory

The link count in the directory has been corrected and the directory will be rebuilt.

56. Maximum length d1 changed to current length d2.

severity: Ø source: salvage_entry

Maximum length was found less than current length, and was changed to the current length.

57. Mod2 nonzero o in a pointer.

severity: Ø source: salv_check_ptr

o indicates the value of the non module 2 pointer, a is the pointer name as given in appendix 1.

58. Name count of a changed from d1 to d2.

severity: Ø source: salv_rebuild_names

a is the name of the segment, d1 and d2 give the value of the change.

59. Name list error has been corrected, but some names may be missing.

severity: 2 source: salvage_entry

Error in name thread corrected, and containing directory will be rebuilt.

60. Non-zero account switch.

severity: Ø source: salvage_directory

Account switch has been turned off and directory will be rebuilt.

61. Non-zero ilock.

severity: Ø source: salvage_directory

The directory was found locked. It will be unlocked and rebuilt.

62. Non-zero mls w.

severity: Ø source: salvage_entry

The mls w switch, which indicates a per process segment will be turned off and the containing directory rebuilt.

63. Non-zero modify switch

severity: Ø source: salvage_directory

The modify switch will be turned off and the directory rebuilt.

64. Non-zero ppml.

severity: Ø source: salvage_directory

The ppml switch, which indicates a per process directory, will be turned off and the directory rebuilt, if possible.

65. Out of bounds file map index, fm_relp: o, size index d.

severity: 2 source: salv_check_map

Cannot access the file map and will delete this entry when containing directory is rebuilt.

66. Out-of-bounds o in a pointer.

severity: \emptyset source: salv_check_ptr

o is the octal value of the pointer, a is the name of the pointer as given in appendix 1.

67. Pathname length at level D1 was D2, reset to \emptyset .

severity: \emptyset source: salv_name

d1 indicates the depth in the subtree at which salvaging. The erroneous length d2 was set to \emptyset . A bug in the salvager exists.

68. Person name thread error.

severity: 2 source: salvage_directory

The person part of an access name, author, or bit-count author may be in error when directory is rebuilt.

69. Project name thread error.

severity: 2 source: salvage_directory

The project part of an access name, author, or bit-count author may be in error when directory is rebuilt.

70. Process terminated because of bad process directory

severity: 2 source: on_line_salvager

Process will be terminated after on_line_salvager finishes.

71. Record count changed from d1 to d2.

severity: \emptyset source: salv_check_map

Record count has been changed to conform with file map.

72. Records used changed from d1 to d2.

severity: \emptyset source: salvage_directory

Number of records used by directory changed. Directory will be rebuilt.

73. Reference o outside of allocation area in a pointer.

severity: \emptyset source: salv_check_ptr

o is the octal value of the pointer, a is the name of the pointer as given in appendix 1.

74. relp= o, dirsw and bs inconsistency.

severity: 2 source: salv_check_thread

The directory switch is set, but the branch switch is not. This entry will be deleted when the containing directory is rebuilt.

75. relp= o, link pathname size d bad.

severity: 2 source: salv_check_thread

The pathname size given by d is bad. This link will be deleted when the containing directory is rebuilt.

76. relp= o, link size d is bad.

severity: 2 source: salv_check_thread

The size of the link d (decimal) is bad. The link at offset o (octal) in the containing directory will be deleted when the containing directory is rebuilt.

77. Reused address: addr= o, dev= d1, page # = d2.

severity: 2 source: salv_check_map

This segment's page d2 will be zeroed out, and then the file map entry for d2 will be nulled out because some other segment also has this page in its file map. This message appears in conjunction with sl6.

78. Segment count changed from d1 to d2.

severity: 2 source: salvage_directory

The number of segments found (d2) is not equal to the number that the directory had indicated previous to this salvage (d1). The directory will be rebuilt.

79. Segment was active.

severity: \emptyset source: salvage_entry

If this is not an active salvage then this message will appear whenever the active switch is on. The switch will be turned off and the containing directory rebuilt.

80. Segment: a.

severity: \emptyset source: salv_name

The pathname a of a segment will be printed preceding any error messages (if it has not already been printed) as well as for all directories whenever switch \emptyset is up. This message appears in conjunction with all s28 messages.

81. Skipping this directory.

severity: \emptyset source: salv_delete_dir

This directory has been found inconsistent. The pages occupied by this subtree cannot be released.

82. Thread_check: d1, d2, d3, d4, d5.

severity: \emptyset source: salv_check_thread

d1 ... d5 give information on an unforeseen threading error. This may uncover a potential bug.

83. Thread_check: err in list crosscheck, count changed from d1 to d2.

severity: \emptyset source: salv_check_thread

In double checking, the count of the number of elements found is different from the previous count. This may be an unforeseen threading error or a potential bug.

84. Truncating a tree node.

severity: \emptyset source: salv_delete_dir

a is the pathname of the node. The node will be deleted and all the pages used by segments in this node will be released.

85. Unable to allocate file map.

severity: 2 source: salv_rebuild_directory

An entry cannot be rebuilt because the directory has no more room to allocate a certain size of file map.

86. Unable to deactivate sub tree.
Setting a out of service.

severity: 2 source: on_line_salvager

Unable to insure that no other processes will reference this subtree and therefore set it out of service. This message appears in conjunction with s26.

87. Unable to hash in a primary name.

severity: 2 source: salvage_entry

Unable to replace bad primary name with a unique one. This segment will be deleted when containing directory is rebuilt.

88. Unknown error: addr = o, dev = d1, page # = d2.

severity: \emptyset source: salv_check_map

In checking the file map address with the FSDCT, an unforeseen error has occurred. The address will be nulled out.

89. Unprotected address: addr= , dev = .1, page # = 2.

severity: \emptyset source: salv_check_map

This address wasnot marked as used in the FASCT. It will be marked as used.

90. Zero directory I.D.

severity: 2 source: salvage_directory

The directory unique identifier was found \emptyset . This directory will be truncated and its tree node deleted when the containing directory is rebuilt.

91. Zero entry I.D.

severity: 2 source: salvage_entry

The entry unique identifier was found to be \emptyset . This entry will be deleted when the containing directory is rebuilt.

92. Zero file map index: addr = o, dev = dl, page # = d2.

severity: \emptyset source: salv_check_map

The file map index was found to be \emptyset . It will be set to the null address value. This message appears in conjunction with s29.

93. Zero relative A pointer.

severity: \emptyset source: salv_check_ptr

This pointer has no relative offset. a is the pointer name as described in Appendix 1.

Appendix 1.

The salvager always checks a pointer or relative offset before attempting to use it by calling the procedure salv_check_ptr. The following is a tabulation of the names of the pointer, and the procedure of origin.

B1. acl.name.person

severity: 2 source: salv_rebuild_acl

An error on the person part of an access name has occurred. For protection purposes the acl will be deleted.

Revision 0

B2. acl.name.project

severity: 2 source: salv_rebuild_acl

An error on the project part of an access name has occurred. For protection purposes the acl will be deleted.

B3. Allocation_area

severity: 2 source: salvage_directory

The allocation area of a directory cannot be safely referenced. This directory will be deleted.

B4. author

severity: ∅ source: salv_rebuild_directory

An error has occurred in referencing the author, "SysDaemon.SysDaemon.*" will be used instead when the entry is rebuilt.

B5. Backward thread

severity: ∅ source: salv_check_thread

Traversing thread in backward direction cannot complete due to a bad backward relative pointer.

B6. bc_author

severity: ∅ source: salv_rebuild_directory

An error has occurred in referencing the bit-count author. The author of the entry will be used as the bit-count author when the entry is rebuilt.

B7. directory

severity: 2 source: salvage_directory

The directory header cannot be safely referenced. This directory will be truncated and its branch removed when the containing directory is rebuilt.

B8. Entry thread

severity: ∅ source: salv_check_thread

An error in the threading of entries will be fixed.

Revision 0

B9. file_map

severity: 2

source: salv_check_map

The file map cannot be safely referenced. This entry will be deleted when the containing directory is rebuilt.

B10. Forward thread

severity: Ø

source: salv_check_thread

Traversing thread in forward direction cannot be completed due to a bad forward relative pointer.

B11. hash_table

severity: 2

source: salvage_directory

The hash table cannot be safely referenced. The directory will be truncated and its branch removed when the containing directory is rebuilt.

B12. Link thread

severity: Ø

source: salv_check_thread

Attempt will be made to correct an error in the threading of a link.

B13. primary name

severity: 2

source: salvage_entry

The primary name cannot be safely referenced. This entry will be deleted when the containing directory is rebuilt.

(END)