Inter-Office Memorandum

To	File Management Working Group	Date	3 May 1977
From	Paul McJones	Location	Palo Alto
Subject	An approach to B-file implementation	Organization	SDD/SD

XEROX

Filed on: <McJones>BFileIssues2.memo

Disks are formatted in fixed-length sectors each capable of holding a page of information. What is the glue which holds pages/sectors together into files, arbitrary-length containers usable by the next level of software? Labeling each sector with the file identifier and ordinal page number of the page it contains leads to a simple and robust design: there is no centralized structure, damage to which can affect all files.

Of course an index structure is needed to support fast random probes, but this structure can be totally reconstructed after a crash by scanning the labels of all sectors. (This should take ≤ 30 seconds for a floppy and even less time for an LCDM.) Note that the occurrence of a crash can be detected by maintaining a Boolean variable "Running" on the disk, which is set to True when the system starts up and set to False when the system has finished normal shutdown. By checking this variable, the system startup procedure can safely decide whether to rebuild the index or use the one written on the disk at the last shutdown.

Not all operations provided by this file system would be atomic (with respect to crashes), but all would have the property that if repeated (by a client program) after a crash they would have the same effect as if no crash had intervened. For example a multipage write operation proceeds page-by-page; if a program driven by an intentions list repeats the write after a crash, no harm will come from rewriting the pages which had already been written. The operations which affect the structure of files are a little trickier. Thus there is no append operation: the operation to change the length of a file takes the new length as a parameter and *eu* does nothing if the file is already that long.

An advantage of index structures not enjoyed here is the ability to alter the order and identity of pages in a file simply by rewriting the (relatively small) index. The problem is that rewriting the index isn't enough; the page labels themselves are the truth and must be rewritten to effect a change. Thus transaction implementations running on top of a page-label file implementation will use copying techniques to update files. (The choice between undo lists and intention lists is still open.)