

MEMORANDUM

To: R.Alter and J.Cole

From: B.Cose11

Subject: PDP-1 Incremental Dump

Date: 14 December 1971

For no particularly good reason, the idea of an incremental dump to the PDP-1 cropped up again recently. I've thought about the matter a bit and have succeeded in convincing myself that the job is considerably easier than it first appeared.

Let us consider dumping the entire drum once a week and doing a sort of incremental dump each day. This incremental dump would consist of the changes to the drum since the last full dump and would be written so as to run in time-sharing, therefore not requiring the system to go down. I think a key shift in viewpoint is that dumps of the system are not intended to protect against human error, but rather to protect against catastrophic system failure.

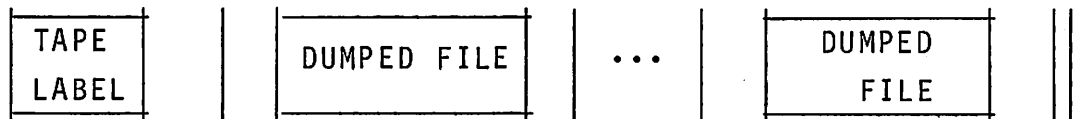
A key question which governs the entire feasibility of an incremental dump scheme is that of exactly which file structures are to be incrementally dumped. The thought of backing up every file structure is untenable.

An overwhelming percentage of the work being done on the PDP-1 is concentrated solely in the programming system, with everything else being modified infrequently, if at all. Therefore, for most of the file structures, a decrease in the dump frequency from once a day to once a week will put no noticeable change in the integrity of the file structures. It is clear then, that the incremental dumper would only have to be concerned with the programming system files.

The size of the programming system is not increasing very rapidly, although there is a large amount of work being done within it. This gives one the picture that there is a block of programs which are constantly being revised with old versions being deleted in their wake and not very many new programs coming into existence. This leads me to think that an incremental dump, which dumped all files in the programming system which had been created since the last complete drum dump, is the proper way to attack the problem. Although there is a great deal of activity as the week progresses, the dump tapes would in fact not get longer because they would not include obsolete and deleted versions. The advantages in reloading from such an incremental dumping system are obvious.

Memorandum
 Page 2
 14 December 1971

I'd like to just say a word on the mag tape format for an incremental dump. I think it should take the space to dump as much information as possible about the files it is saving so that the reloader can be as clever as possible. My first impressions as to the tape format are:



All in ODD PARITY

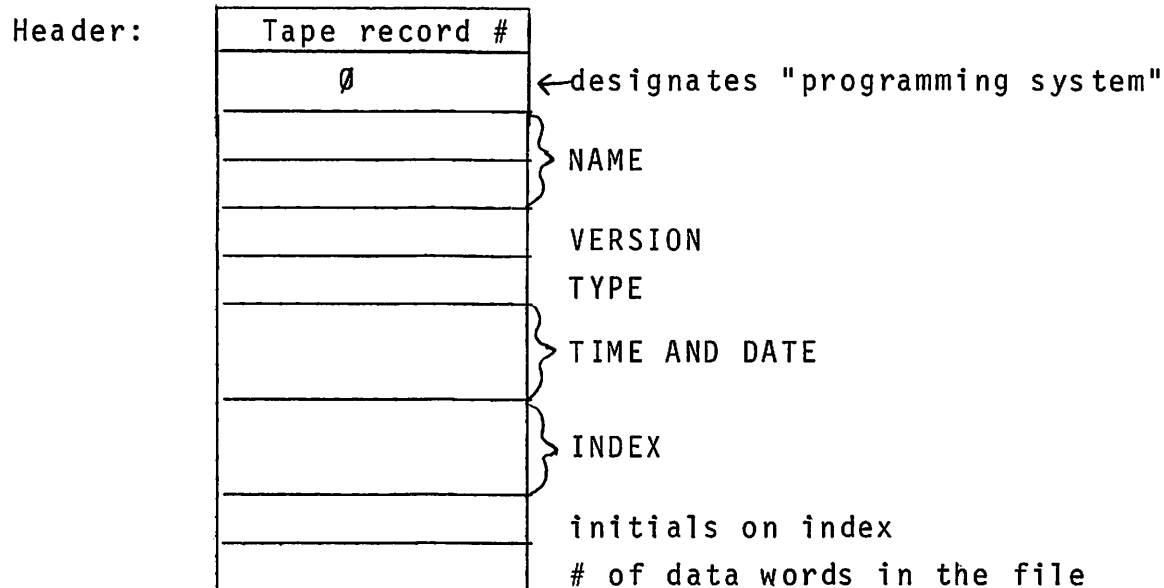
TAPE LABEL CONSISTS OF A CHUNK OF TEXT WHICH WOULD IDENTIFY THE TAPE

DUMPED FILE FORMAT WOULD BE:



The header would identify the file structure of the dumped file, in addition to identifying the file within that structure.

For the programming system:



The data blocks would be 501. words: a tape record # followed by [up to] 500. file data words.

Memorandum
Page 3
14 December 1971

Here is my overview of the programming effort required to implement an incremental dumping scheme.

1. The dump program should be modified to write out the time and date of the dump, as copied from core 15, on two invariant numbers.
2. The incremental dumper, itself. It will scan the entire programming system and write out on mag tape any file which had been created since the time specified by the invariant numbers which the dump program set up.
3. A program to reload files from the incremental dump tape. It can run in two modes: one to reload everything from the incremental dump tape back into the programming system, and the other to recover selected files.
4. A program to list the contents of the incremental dump tape, that is, type out an "index" to the tape.

Here are my first-cut guesses as to the manpower required to implement the system.

1. One man-day.
2. Two man-days.
3. Two man-days.
4. Two man-days.

Given that we all work two man-days per day anyway, I think the entire scheme can be operational in under a week of real work, or in a couple of weeks by "midnight enterprises".

The argument for the incremental dump and the breakdown of the tasks is not intended to be complete or entirely accurate. The arguments can, if necessary, be easily substantiated, but I don't really want to do that if I don't have to. For the work breakdown, I don't believe that is the way work would actually be done, but I do believe that the total amount of work required is pretty close.

Memorandum
Page 4
15 December 1971

An incremental dump would eliminate about 4% of the PDP-1's downtime, is technically possible and would not be an overly lengthy project. Maybe someday somebody will put one together.

BC/jm

cc: J. Levin
P. Wexelblat
F. Webb