Programming Staff Note 15

Computation Center
Nov., 1963

FROM: Louis Pouzin

SUBJECT: Compression and Expansion of Information for Disk Files

1 - In order to save storage room on the disk and, hopefully, swapping time, a set of routines has been implemented which compresses the information on writing the disk and expands it to the original form on reading. So far, nothing has been inserted in the CTSS supervisor or the disk routines. This will be done when the CTSS modular system is implemented. Therefore, this paper is simply an attempt to present the basic principles of the compression-expansion, some first experimental results, and the next step proposed for inserting this device in CTSS with as much security as possible. Any suggestions will be welcomed.

2 - Compression

Normal information is made up of a series of machine words, each having 36 bits. Using compression when information is moved onto the disk, every word is compared with the previous one. If different, the word is transmitted; if identical, the word is not transmitted, but dropped. Consequently, a series of identical words is replaced by a single word. This word itself is dropped whenever it happens to be one of a set of very common words, a table of which is kept by the compressor.

Given a series of words:

...., A, C, K, D, D, D, 0, 0, 0, 5, Z, Z, L, V, T, ...

where each word contains information represented by the letters or figures named in the list. We may assume that the word whose content is represented by A is actually a BCD word containing AAAAAA. This assumption is made for simplicity; the only point to keep in mind is that two words whose contents are represented by the same symbol are identical.

After compression, the series yields:

```
PZE  6,0,3    A metaword; take 3 words verbatim, then take
A             the next word and repeat it so that there will
              be a total of 6 words following this metaword.
G

K

D

PZE  3,1,0    Take word 00...00 (represented by code 1) and
              repeat it 3 times.
PZE  3,0,1    Take 1 word, then repeat the next one to make
S             a total of 3 words.

Z

PZE  3,0,3    Take 3 words. Thats all.

L

V

X
```

The general form of the metaword is PZE N,K,P, where N tells the number of words to be taken as they are and P is the total number of words in the series which the metaword stands for. The missing words are filled with the (N+1) word or with a common word whose code is K.

It is easy to see that the metaword keeps track of all the information necessary for the expander to recover the original form. It is not implied that this scheme compresses as much as might be imagined; it merely achieves some compression by a fairly fast process.

3 - Expansion

The expansion process is implied by the form of the compression, and there is no need for a detailed description here. Only one detail need be noted. When a file is read, the first word is taken as a metaword, which initializes the chaining, skipping to the next metaword, etc.; however, all files are not necessarily compressed. Therefore, there must be an indication of whether a particular file is compressed or not; this indication is in the file directory. More precisely, compression is designated by a one in the sign bit of the third word of the file directory entry.

Compression is completed at the end of each track, i.e. the range of a metaword never extends from one track to another. Therefore, a metaword occurs as the first word of every compressed track. The control word of every compressed track will have a one in the sign bit; this allows a mixing of compressed and uncompressed tracks in a single file.

Curious readers may be supplied with more detailed information about boundary problems. (See note1, section 5.)

4 - First Experimental Results

The following figures are listed by type of information making up the files. These refer only to files using more than one track, because nothing is gained by compression of files which are less than a track in length.

	Number of files compressed	Total tracks occupied		Percentage reduction,
		before (B)	after (A)	$100(1 - \frac{A}{B})$
FAP	13	49	24	51
MAD, MADTRN	2	6	4	33
SAVED, BSS	13	170	132	22
Total	28	225	160	29

The user whose file directory is reflected in the above figures had a total of 286 tracks including those files which occupied a single track. Assuming that all his files had been compressed, he would have had a total of 221 tracks for 89 files, giving a percentage reduction of 23 per cent.

So far, no time estimates have been made, and it is impossible to tell whether or not the system saves time too. This is not a trivial problem since many parameters must be considered, depending on the way the files are transferred between disk and core. Among these considerations may be: use of buffers, comparative frequency of large and small files , comparative frequency of symbolic and binary files, etc. A realistic estimate could be made by statistical comparison of typical time-sharing sessions performed with and without compression.

5 - Inserting Expansion in CTSS

The final goal is to make compression invisible from the user's point of view. In other words, the user should not have to request compression, nor should wonder which way he should read a particular file. Whether there is compression or not should be a choice made internally by CTSS. As a consequence, any program reading from the disk must use the expander. Modifications must be made to the disk editor program, and to the disk routine .READ. Furthermore, the active file status needs 3 additional words for keeping track of the compression/expansion status (see details in Note 2). Utility routines for storing and restoring the active status for a specified file and the table storage parameters must be modified accordingly (so are the table storage parameters). The .LOAD entry will not need any modification in its modular system version since it will be converted by the supervisor into an equivalent sequence of .SEEK, .READ, and .ENDRD calls.

6 - Inserting compression in CTSS

The first condition is to have expansion already inserted and checked. Thereafter, it will be possible to implement compression gradually, if wished; there is no need for a generalized compression once expansion is able to select the right way of reading files. Modifications are similar to those required for expansion, i.e., disk editor, disk routines .WRITE. .RELRW. .FILE. and .ENDRD. Modifications to the storage section of the supervisor are the same as for expansion. In the modular version .BUFF and .CLEAR entries will be converted into an appropriate sequence of .ASIGN, .WRITE, and .FILE; therefore they will not need any revision.

7- Some Reliability Considerations

Compression and expansion will be controlled by two switches allowing a means of by-passing either feature, whatever the information to be transmitted. This provides the system with a permanent way to recover information as it is, despite any error in stored information. Furthermore, some commands, like COPY, may take advantage of the reduction in the size of files, when there is no need to work on the content of the file itself.

Metawords will be checked both while compressing and while expanding. This does not provide a complete check, but a reasonable amount of plausibility. For example a foolish expansion will not be allowed to run over one track.

At the time of implementation into the system, compression/expansion will not be generalized abruptly. Another set of switches will inhibit the process at all times, unless the user requests that it be turned on through an entry to the supervisor. In this manner compression checking will be possible in a real time-sharing session, by one user, without interferring withother users.

Note 1.

Compressed information is made up word series led by a metaword. Such a quantum is to be handled as a whole and, for reliability and flexibility, is not allowed to extend from one track to another. This quantum is closed at the end of every track and when a call to .FILE is issued. Consequently, one may encounter the following patterns of metawords:

PZE	P,0,N	N words + 1 word repeated to a total of P
PZE	P,K,N	N words + a standard common word repeated to a total of P
PZE	N,0,N	N words, compression was not possible
PZE	P,0,0	1 word repeated P times
PZE	P,K,0	A frequent word repeated P times
PZE	0,0,0	no word

The last case occurs when there is only one word left in a track on offering a quantum and compression is impossible.

Then the quantum begins with the next track, and the empty metaword does not affect the expansion.

Remark: $N \leq P, N \leq 464$, and $N \neq 0$ implies $P \neq 0$

Note 2.

During compression or expansion, the following information must be kept aside so as to restart the process properly.

last word transmitted	1 word
current metaword	1 word
status of switches	1/2 word
word count in the track for expanded content	1/2 word

The word count is not necessary; but it may turn out to be useful for various file manipulations and does not take any more storage.

CAS