To the Editor
The Computer Journal

Sir

On an evolutionary memory concept

On the basis of certain assumptions (the future validity of which seems assured by current trends in microprocessing and mass storage), we should like to postulate the evolution of a concept christened *Lateral Memory*. We should like to indicate also possible construction and management techniques.

We make the following assumptions:

(*a*) A natural and technologically-feasible outcome of computer data-storage development, via the path blazed for example by Virtual Memory, mass storage and microminiaturisation is *National Store* or, more simply, STORE: Subscriber TOTAL Recall Equipment, perhaps (see d'Agapeyeff, 1975, for example);

(*b*) Developments in data coding will some day again follow a trend away from binary representation, through decimal, to *n*-state (or, more strictly, endless-state) representation;

(*c*) There is a method, as yet possibly undiscovered, by which individual microminiature program processors 'softwired' (Castell, 1973) by individual users for individual applications, can be rigorously and reliably associated with that individual's data, *National Data* or anything else allowed to that user in STORE.

A picturesque summary of these assumptions might be that they make program processing relate to STORE as breathing relates to fresh air.

The implementation of STORE leads to an evolutionary machine memory concept which we now christen *Lateral Memory*, by analogy with De Bono's (1967) well known concept of Lateral Thinking. In the same way that lateral thinking optimises (human) processing power by refusing to allow the processor resource to persist in endless algorithmic loops, so lateral memory optimises data storage by refusing to allow STORE to persist in endless constant data representations.

A further understanding of the lateral memory concept may perhaps be obtained by calling it—again, rather picturesquely—'cocktail party memory'. This is by analogy with the human data storage processing carried out by someone at a cocktail party, listening-in in 'time-sharing' fashion to all the conversations going on within earshot. By an inbuilt faculty of the processing executed by the human brain, the conversations are pieced together as meaningful wholes—the human data storage method (whatever that may be) does not allow itself to persist in recording data endlessly but maximises use of its scarce resource. However, little or no data is 'lost'.

Thus, as with lateral thinking, lateral memory is seen to be a processing technique available quite readily to the human computer (some human computers, of course, being better at it than others). Again, it seems likely that, in human terms, it is just that—a processing technique, rather than a property of the data storage itself.

In machine terms, however, although we might feel more ready to describe lateral memory as a 'local store management technique', it ought to be viewed as an evolutionary memory concept, in view of the assumptions stated above.

Actually, this is not entirely convincing without one further assumption: the distinction between 'algorithm' and 'data' will become increasingly blurred, so that it will no longer be appropriate to think of 'units of processing operating on distinct units of data (bits)'. We hesitate before introducing another new term but, nevertheless, present a further characteristic of lateral memory: it will be constructed from mixed 'processing/data units' or *probits*. (Recall that assumption (*c*) ensured the integrity of association of a user's application microprocessor(s)—microprobitter(s)?—with his own data—prodata?—and it will be seen that probit is a particularly apt term (Oxford Dictionary, 1973)).

Finally, we indicate a technique by which *National Supervisory Software* could carry out its global (lateral memory) STORE management—that is, we suggest an example for the method mentioned in assumption (*c*):

Assumption (*b*) implies that each probit may take one of *n* states. More strictly, each probit may be considered as being in one of an endless number of states, in fact, in a continuum of states. By suitable definition of the functions we arrive at *information density*, ρ, *processing pressure* p and *state-speed*, u, of a probit, all functions of the state, i.e. position in the continuum x and, of course, the time, t (which to avoid even further conceptual complexities we assume is a global independent variable—a 'universal property'). Domains of p, ρ, u and bounds on x, t need not concern us here but it will come as no surprise that these, too, could cause conceptual difficulties. (The 'suitable definition' begs a lot of questions but should lead to the conclusion that three such parameters form a complete set for retrieval of a probit.)

The suggested technique then proceeds as follows: choose a lateral memory design so that p, ρ, u obey gasdynamic mass and momentum conservation equations in (x, t), viz.,

$$\frac{\partial \rho}{\partial t} + u\frac{\partial \rho}{\partial x} = 0$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0$$

and an energy-conserving state-equation

$$p = p(\rho) \ .$$

Then any particular lateral memory state must, by design, be a solution of these equations and national supervisory software need only refer to the user's solution to retrieve his p, ρ, $u(x, t)$ with integrity. Furthermore, there are a battery of transformation techniques for finding, keeping and interrogating such solutions (Bateman, 1943; Rogers, 1969; Castell and Rogers, 1974).

(The somewhat arbitrary choice—on the basis, however, of the need for simple non-linearity with non-trivial solutions—of the gasdynamic governing equations is not out of place with the earlier intuitive comment about the 'fresh air' nature of lateral memory.)

Yours faithfully,

S. P. CASTELL

Bremar Holdings Ltd
Bremar House
Sale Place
London W2 1PT
25 June 1976

References

D'AGAPEYEFF, A. (1975). The impending revolution in business data processing, *Computer Bulletin*, Vol. 2, No. 6, pp. 8-10. See also: The Trillion Bit Memory, *Computer Weekly*, p. 24, 18-25/12/75;

CASTELL, S. P. (1973). A New Word, Letter to *Computer Digest*, p. 10, 11th July.

BONO, E. DE (1967). *The Use of Lateral Thinking*, Cape.

Probity: uprightness, integrity, incorruptibility, *Pocket Oxford Dictionary*, 1973.

BATEMAN, H. (1943/44). The Transformation of Partial Differential Equations, *Quart. Appl. Math.*, Vol. 1, pp. 281-295.

ROGERS, C. (1969). Invariant Transformations in Non-Steady Gasdynamics and Magnetogasdynamics, *ZAMP*, Vol. 20, pp. 370-382.

CASTELL, S. P. and ROGERS, C. (1974). Applications of Invariant Transformations in One Dimensional Non-Steady Gasdynamics, *Quart. Appl. Math.*, October, pp. 241-251.

To the Editor
The Computer Journal

Sir

Professor Baecker's article 'Areas and record classes' (this *Journal*, Vol. 18, p. 223) contains some statements about PL/I which, in my opinion, confuse the definition of the language with the behaviour of an implementation of the language.

In particular, he states that 'garbage collection, or any system action to relocate BASED variables, is prohibited, as is the use of secondary storage except in a virtual memory system.' This is not so. There is nothing in the definition either of IBM's version of PL/I (IBM, 1965) or in the proposed Standard PL/I (ANSI, 1975) that would prevent the choice of an implementation that would permit compaction of BASED storage.

There are many implementation techniques that might be used: for example, adding an additional level of indirectness so that pointers do not contain the actual address of the data, but contain a fixed offset into an auxiliary pointer table that contains the true