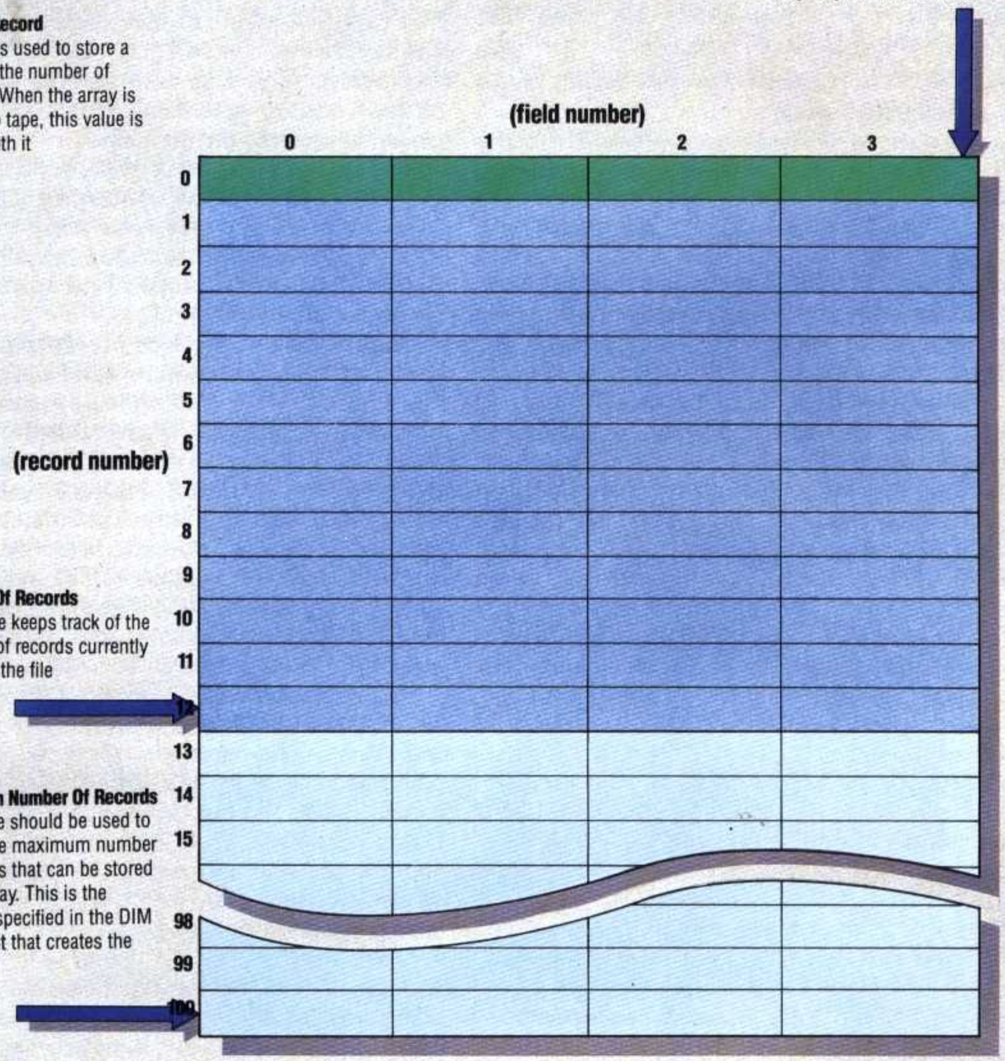
Storing Records And Fields In A Basic Array

A two-dimensional string array can be used to simulate a random access file. The first subscript is used to identify a particular record and the second identifies fields within a record

Number Of Fields
It's handy to have this number defined in a variable at the start of the program. This makes it easier to change to using larger records later, because it would mean altering only one instruction

Header Record
A$(0,0) is used to store a count of the number of records. When the array is SAVEd to tape, this value is stored with it



Number Of Records
A variable keeps track of the number of records currently in use in the file

Maximum Number Of Records
A variable should be used to record the maximum number of records that can be stored in the array. This is the number specified in the DIM statement that creates the array

Deleting A Record From The Array
This program segment removes record number N from the array using the method we have already detailed. All the records below N are moved up one space, overwriting the data currently stored in position N

```
100  LET R = R + 1
110  IF R > M THEN  PRINT "ARRAY FULL": RETURN
120  FOR I = R TO N + 1 STEP  - 1
130  FOR J = 0 TO F
140  LET A$(I,J) = A$(I - 1,J)
150  NEXT J: NEXT I
170  LET A$(N,0) = N$: LET A$(N,1) = C$
180  LET A$(N,2) = D$: LET A$(N,3) = E$
190  RETURN
```

Inserting A Record Into The Array
This program segment inserts a new record into the array at position N in the file. All records past the point of insertion are moved down to create a gap for the new record stored in N$, C$, D$ and E$

```
200  FOR I = N TO R - 1
210  FOR J = 0 TO F
220  LET A$(I,J) = A$(I + 1,J)
230  NEXT J: NEXT I
240  LET R = R - 1
250  RETURN
```

KEVIN JONES

Loading And Saving A Record Array

Variables SAVEd with Program

The Spectrum saves all its variables along with any saved programs (although it can also save arrays on their own). The sample program fills an array with data and the SAVE instruction will store the array on tape with the program. When the program is loaded again it will automatically start at line 700 because the SAVE instruction refers it to LINE 700. The RUN instruction must not be used because it clears all variables

Spectrum

```
100 REM***SPECTRUM DEMO***
200 DIM A$(100,20)
300 FOR K=1 TO 100
400 LET A$(K)="RecNo."+STR$(K)
500 NEXT K
600 STOP
700 PRINT "ARRAY CONTAINS..."
800 FOR K=1 TO 300
900 PRINT A$(K),
1000 NEXT K
1100 STOP
SAVE "DEMOPROG" LINE 700
LOAD "DEMOPROG"
```

SAVEing Named Arrays

The Oric Atmos provides the commands STORE and RECALL to save and load particular arrays from tape. This makes it easy to store a file held in an array

Oric Atmos

```
100 REM Save array to tape
110 A$(0,0)=STR$(R)
120 PRINT "Please position tape,
press PLAY and  RECORD then hit RETURN"
130 A$=KEY$:IF A$="" THEN 130
140 STORE A$,"AFILE",S
150 PRINT "FINISHED"
160 RETURN

200 REM Load array from tape
210 PRINT "Please position tape
 and press PLAY then   RETURN"
220 A$=KEY$:IF A$="" THEN 220
230 RECALL A$,"AFILE",S
240 R=VAL(A$(0,0))
250 PRINT "FINISHED"
260 RETURN
```

Using Serial Files

The BBC Micro is one of several home computers to support true serial files on cassettes. These two subroutines store and reload the array by creating a serial file. The first field is the record count

BBC Micro

```
100 REM Save array to tape
105 PRINT "Please position tape"
110 X=OPENOUT("AFILE")
120 PRINT£X,R
130 FOR I= 1 TO R
140 FOR J= 1 TO F
150 PRINT£X,A$(I,J)
160 NEXT J: NEXT I
170 CLOSE£X
180 PRINT "FINISHED"
190 RETURN

200 REM Load array from tape
205 PRINT "Please position tape
and press PLAY then RETURN"
210 IF INKEY(0)=-1 THEN 210
215 X=OPENIN("AFILE")
220 INPUT£X,R
230 FOR I= 1 TO R
240 FOR J= 1 TO F
250 INPUT£X,A$(I,J)
260 NEXT J: NEXT I
270 CLOSE£X
280 PRINT "FINISHED"
290 RETURN
```