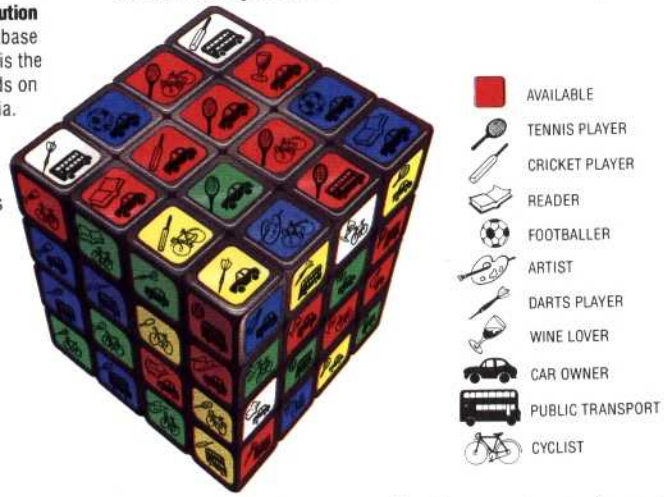
DAISY WHEEL

Until recently, microcomputer owners requiring letter-quality printing were restricted to one variety of printer — the *daisy wheel*. The name derives from the printing mechanism: individual letters are mounted on plastic spokes or 'petals' arranged in the shape of a wheel about three inches (7.5 cm) in diameter. The wheel spins until the desired character is at the top, and then a hammer strikes the 'petal' against the ribbon, making an impression of the character on the paper. Electronic typewriters also use this technique. In some models the letters are mounted in a 'thimble' shape.

Daisy wheel printers are considerably slower than their dot matrix counterparts; a £400 model can print only 10 to 15 characters per second (cps), as opposed to typical dot matrix speeds of 100-160 cps. The fastest daisy wheel models can print at 60 cps, but these are priced at around £2,000. The main restriction on speed is the fact that the printer motor must constantly accelerate and decelerate while turning the wheel. Close examination of a daisy wheel may give the impression that letters are arranged at random, but the layout is designed to reduce the average amount that the wheel must turn when printing standard English text.

DATABASE

A *database* is a general-purpose software package that is designed for the easy storage and retrieval of data. All the information about one person (in a mailing list) or object (in cataloguing or stock control applications) is contained in a *record*, which corresponds to a card in a card index file. Each record consists of *fields* of information, each with its own field name. For example, the field named 'Price' in a stock control program may contain the data '£34.25' in one particular record. A collection of records sharing the same layout and field names is called a *file*.



Searching For The Solution
The essence of a database management system is the ability to select records on multiple search criteria. The Rubik Cube represents the database, and its facets are the records

Simple databases restrict the user to work on a single file at a time, but multi-file packages allow data to be transferred between different types of file. In an invoicing application, for instance, information from a customer file that contains the customer's address and credit limit may be merged with data from a product file that gives details of prices, product codes and descriptions. The most sophisticated microcomputer databases are programmable: the package can 'learn' sequences of commands that are used frequently and will execute them at a single keystroke. Such packages are also referred to as *application generators*.

DATA COMPRESSION

The rule of thumb method for evaluating data storage requirements is to allow for one eight-bit byte per character, as well as making some allowance for operating overheads such as file header information and control characters (see page 348). *Data compression* involves trying to improve on this rate for two purposes: increasing the effective capacity of a disk drive, and reducing the time needed to transmit data over a distance.

At first this may seem very difficult to achieve, especially as all characters and some commands are ASCII coded. Consider, however, messages that use just upper case text and a maximum of six other punctuation symbols — just five bits would then be needed to represent the 32 (2^5) possibilities. Characters will no longer correspond one-to-one with the byte locations, and a special program must be employed to compress the data before recording on disk, and then decompress it again for use by an application program. Such utilities (also called *compactors*) can be purchased to work on standard CP/M files. An additional benefit is some measure of security — compressed files can't easily be read on another system.

Compression rates greater than eight-to-five rely on frequency analysis — some sequences of characters occur more frequently than others. In normal text, a standard list of the 100 most common words (and, the, etc.) will account for half the text. If these words are replaced by a single byte, with a flag-bit to differentiate them from normal characters, the saving will be considerable. This is called *tokenising*.

DATA CORRUPTION

Data corruption is most likely to happen when writing onto a magnetic surface (because of imperfections in the media), or when transmitting data over a distance (because of extraneous noise). It can happen to data residing in memory, if an individual RAM chip should overheat and behave unreliably, or there is strong electrical interference from nearby.

Magnetic media should always be handled carefully, but the only way to guard against corruption generally is to employ the principle of *redundancy* — the opposite of data compression. This means taking more space or time to represent a single piece of data than is strictly necessary. Parity and Hamming codes are the commonest means of safeguarding against data corruption.