LINES OF ENQUIRY

In the last instalment of our business series we looked at how certain packages handle stock control. An efficient program needs to monitor all stock movement, from supplier to the shop shelves. The packages achieve this by a system of coding structures. We now look more closely at the design requirements of these structures.

We have discussed in some detail the different ways stock lines can be identified in the stock data file. In addition to simply allocating them a code number, however, the system should allow the user to record information on each stock line.

This information will need to be categorised into relatively constant data and the amount of data will be determined by the storage capacity of your computer. The design of such a system will therefore have to be very economical. The amount of information must be sufficient to meet basic management needs, but too much information will make impossible demands on the computer's processing and storage power.

Stock Answers

The Stock Recording System is one of the many Dragon 64 applications programs written to run under OS9. This is a multi-programming and multi-tasking operating system developed by Dragon from the UNIX operating system

IAN McKINNELL

Dragon Data's Stock Recording System for the Dragon 64 has seven *fields* in which to record information for each stock record. These consist of an item number, description, re-order level, cost price, sale price, and the unit of measure.

The fields form an important structure in a stock system. The difference between the cost price and sale price will provide a measure of gross profit. Items can also be grouped, which helps both to analyse and summarise in reports. The unit of measure field is essential because commodities are packaged in varying ways. On one stock line the owner will want to count each individual item, on others, such as nails and screws, the count will be by the box.

Perhaps the most fascinating aspect of designing a stock control system is that much of the information required consists of *dynamic* rather than *static* data. We have already seen that standard ledger records consist of relatively constant information about the customer or account (known as the header part of the record) and information about the various transactions taking place on that account.

With stock control, however, the firm boundaries that exist between the static part of the record and the dynamic or transaction-based part of the record become much less clearly defined. The stock item and group descriptions and coding are the equivalent of the static information on the customer or supplier records in sales or purchase ledger systems. But with a stock control package, static data has to be supplemented by a great deal of information derived from stock transactions.

For example, customers on a sales ledger master file change their address or telephone number relatively infrequently. Therefore, the program will have a master file maintenance option that will allow you to amend particular client records. The information or amendment cannot be calculated by the program.

If we take the sales price and cost price fields on the stock item record as an example, these can change each time the business replenishes its stock. The logical solution here is for the program to take this information (together with the date of the price change and the stock volumes affected) directly from a transaction input routine recording goods received, and not from a file maintenance program. There will still be, of course, a need for a file maintenance program to amend stock descriptions, or add or delete a stock line from the stock file, but a lot of data can be generated from basic transaction information.

In order to understand how a stock control program operates, we need to look at the input