1978

Why install a data entry system?

Mike Aldrich,
Redifon Computers Ltd

A data entry system is a programmable computer system designed to capture, validate, format, edit and then transmit and receive data to or from another computer for further processing and then return information for local usage. As a piece of hardware it can be found in many shapes and sizes with up to 48 v.d.u. terminals attached, storing up to 250M byte of files and connecting a large array of peripherals. As in all computers, the really important piece is the one you don't see – the software. The data entry system has evolved over the last decade through the phases of key-to-disc and processor controlled keying. The latter terms are as obsolescent as the products they once described. A look at the world of data entry will highlight the reason.

Data entry is a relative newcomer to systems design. For decades the prime method of entering data into automatic processing machines has been the punched card. Invented in the 19th century by Herman Hollerith as a cardboard replica of the old Dollar Bill, it provided a useful and reliable media for transposing what the human had written into what the machine could read. Accuracy, always a problem, was ensured by repeating the punching process, proving that the holes had been put in the same place twice. The amount of data carried on each card was limited to 80 characters. The limitations of the punch card and its cutting equipment constrained the design of input systems and processing systems for 70 years.

Distributed processing

The late 60s saw the first sign of obsolescence for the punched card. The widespread use of commercial computers with voracious appetites for data demanded a better method of input than cutting holes in cardboard. The emphasis, however, was on speeding up the existing method rather than finding a better one. Two solutions appeared. First the better mousetrap approach of the buffered keypunch and then the more innovative approach of replacing mechanics and cards with electronics and magnetics in the shape of key-to-tape soon to be followed by key-to-floppy disc. Productivity down in the electronic punch room improved immediately but the input systems remained unchanged. Most users continued to use punched cards.

A better solution was on the way, probably the first example of distributed processing. The punch room needed computer power to roll-back the frontiers of data preparation and in the early 70s the first minicomputer based key-to-disc systems appeared. With terminals that looked like electronic card punches they brought intelligence to the data preparation task. Although the first systems were little more than punch emulators the ability to have program control of data preparation quickly led to software development that started to find new ways for treating an old problem. In particular the 80 column format was recognised for what it was – an anachronism.

The first generation terminals were soon replaced by v.d.u. terminals that enabled an operator to see the mirror image of the punch document formatted on the screen. With punching, sight verification had not been practical. Software development was providing more and better facilities for intelligent accuracy tests. If the input documents could be displayed on the screen, why not display the original source document? If the source document could be displayed was there any real need for coding documents? Was coding necessary? How smart could the system be made?

Strangely enough, input system design had been neglected while computer system design had progressed by leaps and bounds. Even when the transition had been made to key-to-disc the punch room supervisor had programmed the system in much the same way as the old punches. The wages of punch room staff were piece rate dominated. 16–20000 keystrokes per hour were now possible. The advantages of key-to-disc were substantial. Accuracy and productivity improved dramatically – typically 25%–30% throughput improvement. Punch room noise disappeared as did card dust and cuttings.

But to get from data source to key entry, recording, coding and transcription were required. Documents were subsequently batched, moved and controlled – again and again and again. The system was slow, error-prone and tedious – and that was before the documents even reached the punch room.

Coding and verification were the obvious targets for improvement. Coding had originally been needed to shape data to avoid machine limitations. Computer intelligence now enabled the punch room to use the source document without special coding. The intelligent system could take data from the source document and then expand or abbreviate it, analyse or synthesise it, add or subtract it to provide the appropriate records for later processing.

With the computer checking, inserting, defaulting and formatting the data, why verify? Why do the work twice? Instead of verify why not validate? Validation serves two purposes. It established legality and it established relevance. Whichever approach is taken, the validated data can be processed directly by the mainframe. The edit run is not required.

With the source document being used in the punch room the inevitable question was asked. Do we need centralised data preparation? Clearly, the source document data could be entered directly from the user department. The obvious advantage with direct data entry is that the operator will be familiar with the meaning of the data on the document, reducing the likelihood of errors and making those found easier to correct. Knowledge of the data also re-enforces the validity checks. The only technical pre-requisites are to be able to locate terminals remotely and to provide the remote operator with sufficient system supervisory functions to control the task execution with security control.

The disadvantage of source data entry is generally that the remote operator is not a keypunch professional and keystroke productivity is lost. The point has already been made that keystrokes per hour are not measures of input system efficiency.

To continue the chronology, it was then a small step to consider the possibility of not only remote terminals but also data entry systems remote from their mainframes. Interfaces in the past had been magnetic tapes. Telephone lines were to become the new interfaces. The important requirement was to service terminals and another computer at the same time. Software emulators running IBM 2780/3780, HASP RJE, ICL 7020, ICL 7503, Burroughs TC 3500 and Univac 1004 protocols made the data entry systems function as remote batch terminals.

Towards the future

Development of data entry systems continues at an ever increasing pace. Fundamentally they are highly versatile and sophisticated application programs, systems running on powerful and highly tuned computers. Recent additions of o.c.r. facilities concurrent with terminal and telecommunications handling reveal an ever-increasing range of applications flexibility.

In the UK there are still some 50000 card/paper tape punching machines in use – four times the population of data entry terminals. These old machines perpetuate an unnecessary 19th century approach and an antiquated work environment for those who use them. Today's data entry systems have two useful features. They enable users to cutover from mechanics taking their existing systems with them – true system portability. More importantly they provide the tools for users to develop new generations of input systems to meet the needs of today and tomorrow.