

Presentation Information produced by ICT systems is usually better presented than that produced by other means. The layout is likely to be well designed – perhaps using an organisational template so that it not only looks good but conforms to the business's 'house style'.

Appropriate Use

There are some limitations to what ICT systems can do. They are really only good at processing data that can be stored numerically. There are obvious difficulties over providing value judgement information such as which film will I enjoy most, which pop group is the best, does God exist.

Computers are also limited by the accuracy of the data that they process. Hence the expression GIGO – garbage in, garbage out. As a consequence a great deal of effort is made to ensure that the data that the computers process is accurate.

A computer can only work with the data structure that has been pre-determined for the application. For example it will not be able to produce information about the ethnic profile of the workforce if there is no data field corresponding to ethnic origin.

Computer systems can only process the data according to the pre-determined rules that have been programmed into them. They cannot make independent judgements outside the scope of those rules although they may support human decision making. This means that they are also unable to behave creatively or intuitively.

There are also situation where, while a computer system may provide a possible solution to a problem, its use will not be appropriate. Once example of this is where human interaction is important such as in a doctor's surgery In theory it is possible for an ICT system to be constructed to diagnose minor illnesses. However, when a patient goes to the surgery they often want the human interaction and reassurance that can only be provided by human to human contact. An impersonal diagnosis from a 'machine' would not satisfy them.

Types of Processing.

There are various different ways that processing can be done in an ICT system. The actual processing method depends mainly on the software being used and it is possible for a single ICT system to perform different types of processing for different tasks.

The idea of a transaction is useful in describing the different methods of processing. A transaction can be thought of as a collection of data that represents something happening as a single event in the real world. All of the data must be processed. If it is not possible to process all the data for the transaction then none of it should be processed. As an example, consider an Internet purchase where the item selected, customer address and customer card details are all entered as data. All the items in this group of data is needed to

complete the transaction. If any of it cannot be processed then the entire transaction must be abandoned and an error message or request for an entry to be corrected displayed.

Batch Processing

In batch processing, transactions are collected together to form a batch. The batch of collected transactions is then processed in one go.

Electricity billing is a good example of an application where batch processing is suitable. Meter readers collect a large amount of data during the day. The data, which consists of account numbers and meter readings, is collected in a transaction file. The transaction file is then sorted in order of account number and merged with the customer file to produce an updated customer file and a set of bills that are posted to customers the next day.

Batch processing systems typically require sequential file access files, that is files where the records are accessed in order one-by-one from start to end of the file.

Transactions that are to be processed are stored in a special file called transaction file. The transaction file records will include an identifier that uniquely gives the matching record in the main file, in other words the master file key field. In the case of electricity billing this would be the account number.

When all the transactions have been collected, the transaction file is sorted in order of the master file key field. The transaction file and the master file are then merged to create a new updated master file together with any other output that the system is designed to produce.

Batch processing is suitable for situations where there is a lot of data to be processed and where the data can be collected together and processed all at once. It is not suitable for situations where transactions have to be processed as they occur or where the data files have to be up-to-date at all times. Since the user must wait some time (hours, days or even weeks) to receive the output that has produced from the input, batch systems are not interactive.

As already mentioned, electricity or any utility billing system is suitable for batch processing. Another suitable application is payroll. Here data about hours worked for each employee is collected at the end of the week. The data is collected together for input and processed as a batch to produce payslips.

In both of these applications, every record in the file will be processed in turn since each customer will get a bill and each employee will be paid.

Transaction Processing

As previously mentioned a transaction can be thought of as a unit of processing that must be completed before another is processed. In a batch system all the data is accepted by the system and then the transactions processed one by one.

In a transaction processing system once the data for a transaction has been input no more transactions will be accepted until the first transaction has been dealt with. This is sometimes achieved by locking access to the data from the moment a transaction is started and until it is completed.

Transaction processing is used where response time is important such as booking systems, enquiry systems, book issue in libraries, stock control systems and so on. In these situations, batch processing would be inappropriate.

A short delay between the transaction occurring and output from the system is acceptable and the length of the delay will depend on the situation. In a holiday booking system the customer will be prepared to wait for several minutes perhaps if told that the data needed is being accessed by another terminal but will be available shortly. Nor would a delay of a second or two create problems when issuing library books.

If a transaction means several items of data, perhaps in different files, will be changed then any changes made must be undone if the transaction fails. For example suppose transaction related to the movement of money from one bank account to another and the first account had been checked to make sure that it contained sufficient funds and then the amount was debited from that account. If the transaction failed at the second stage, perhaps because the number for the destination account was invalid, then the transaction would not complete. The entire transaction must be undone and so the removal of the money from the first account must be cancelled.

One problem that can arise is that if incorrect data is entered into a transaction system then its effects may result in the incorrect processing of later transactions. A particular holiday may be shown as full and later customers may be turned away if an earlier booking was entered with the wrong code. By the time the error is spotted the holiday that the original customer requested might be fully booked.

Interactive Processing

Interactive processing is when there is two way communication between the user and the system during a transaction. Typically the user supplies data to the system and the system then produces output. The user then produces further input, based on the output that the system provided.

Most operating systems are interactive. Batch processing systems can never be interactive but many transaction processing systems are. Internet shopping is highly interactive where the person uses a browser and search engine to find the lowest price for an item, selects the appropriate store, views a list of items, selects the one they want, orders and pays for it. At each stage of this process the user is supplying data (sometimes in the form of mouse clicks) and deciding on their next input on the basis of the information presented on the screen.

Internet banking, games software and teaching software are other examples of highly interactive processing. In all of these situations the delay between data entry and response must be short otherwise the user may become impatient with the system.