## 3.3.1Operating Systems

- □ Programs that (usually without the user’s knowledge) handle the background tasks required on a computer for its effective functioning, as opposed to application software, which is installed and run by the user for a specific purpose, it co-ordinates the interactions between the hardware, other software and the user.
- □ **System** software = **operating system** + **utility** software.
- □ **Utility** software provides tools for system maintenance and repair and are not necessarily part of the OS, e.g. anti-virus software.
- □ Roles of the OS: communicating with peripherals (monitors, printers, disk drives, etc., most need programs called **drivers** to allow communication, sometimes these drivers need to be installed into the OS separately), co-ordinating concurrent processing of jobs (a single processor apparently **multi-tasks** by switching rapidly between different jobs, the OS co-ordinates this by taking advantage of pauses in one process e.g. waiting for input to perform another), memory management (with several tasks running concurrently, the OS avoids errors by protecting each task’s memory area and allocating space as required, which may involve the use of virtual memory on the hard drive), resource monitoring and accounting (keeps a log of available resources connected to the system, on some systems users are charged for the time and resources they use e.g. for printing), security (most OS’s require a **log on** with a **user ID** and **password**, allowing recording of user activity and restriction of resources to only those with relevant **permissions**), program and data management (it acts as a librarian, accessing and loading files requested by the user and by other programs, requires the use of an organised file system, with a way of representing a folder structure), network management (data within the computer must be made compatible and transmitted according to standard network **protocols**, the OS must be able to recognise and filter data according to their purpose and intended destination) and providing an appropriate **user interface** (the way that the computer presents itself to the user, **command-line**, **menu-driven** or **Graphical User Interface**, **GUI**).

## Users and Tasking

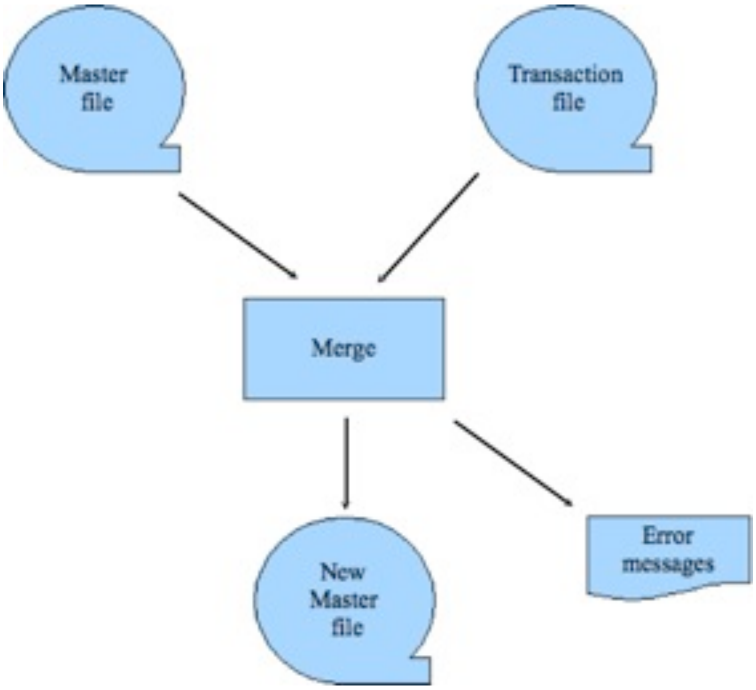
- □ Computer types include: personal (PC), portable, mainframe, server and client,supercomputer.
- □ Most portable and personal computers are **single-user** (i.e. one person uses them at a time), **multi-user** machines e.g. servers can handle a large number of simultaneously logged on users, either through their own PCs or through **dumb terminals** that do not have their own processors, multi-user systems use a single shared CPU.
- □ **Multi-tasking** computers (single or multi-user) allow each user to execute more than one program at a time (possible on a PC to print a document, work on a spreadsheet and listen to a music stream concurrently), the processor is actually skipping between tasks very rapidly.
- □ **Single access** computers only allow one person ever to use them e.g. most smart phones, **multi-access** computer can be used by several people at different times e.g. an ATM (this is different from single/multi-user).

## 3.3.3 Types of Processing

- □ **Batch processing**: jobs to be processed are stored until later and then processed all in one go, can be cheaper since the jobs get done all together and can be done when the system is quiet (e.g. at night), examples are processing and printing of pay slips (payroll), printing school reports.
- □ **Online processing**: some instantaneous reference to an online database is required, often some of the processing can be batched later, also called **interactive processing**, examples: booking a hotel on the internet, withdrawing cash from an ATM.
- □ **Realtime processing**: processing occurs instantaneously - any delay would stop the whole operation, most control systems require this, examples: an autopilot on a plane, an intensive care unit monitoring system, most computer game play.

## 3.3.4 Master and Transaction Files

- □ Organisations like banks need to keep data in one central **database** (copies would get out of sync), access to that database may be restricted or transactions may take place at different times in different places, somehow the everyday data must be merged with the main data.
- □ The **master** file contains all the data of the database, **transaction** files contain just the batch of data which will be used to update the database e.g. just today’s business on an ATM, if data are added, edited or deleted, the transaction file must be used to update the master.
- □ At an appropriate time (when the master file is not being updated by other transaction files), the two will be compared and any difference will cause that record in the master file to be updated, once the process is complete, the transaction file may be deleted (depending on the backup regime).



## 3.3.5 Reliability

- □ The importance of a failure in a computer system will depend on the nature of the task being performed, some processes are **safety critical**.
- □ The reliability of any system is only as good as the data that is entered, accuracy and reliability of data is known as **integrity** (loss of data integrity can be accidental), **security** means protection against deliberate access, loss or change.
- □ Threats to data integrity include: hackers, theft of hardware, hardware failure, fire, flood etc.
- □ The risks to personal or company data are greater now because data may be copied easily, giving no evidence that a copy was made and data may be accessed over networks remotely.
- □ Data can be checked upon entry in two ways: **verification** (is it accurate?, double data entry or proof-reading) and **validation** (is it reasonable?, routines written to validate data type, value list, range check, check digit correctness, etc.).
- □ For better data security, use **physical** security (locked server rooms, CCTV, guards), **user ID** and **password**, **permissions** e.g. EPOS operator in a supermarket cannot refund without permission, **encryption** (for storage and transmission).
- □ Passwords should be of reasonable length, not be a dictionary word or name, be changed regularly and contain letters, characters and symbols.
- □ Duplication or even triplication of hardware (**redundancy**) allows a system to fail without consequence (said to be **fail safe**), safety critical systems are usually **mirrored** perhaps several times to allow for a failure.
- □ Data itself should be **backed up** on and off site and in different formats, previous master files can be recreated if transaction files are kept, often 3 generations of transaction file are kept (called the **grandfather** method), an organisation should have a published **backup strategy**.
- □ **Backup utilities** perform backups automatically to eliminate human forgetfulness, a backup may be: **incremental** (only recent changes saved) or **full** (a complete new copy each time).
- □ An **archive** removes old files from the working system.