2023

Concepts of Software

INTERNATIONAL UNIVERSITY OF EAST AFRICA

'Learning to Succeed'

OPERATING SYSTEMS

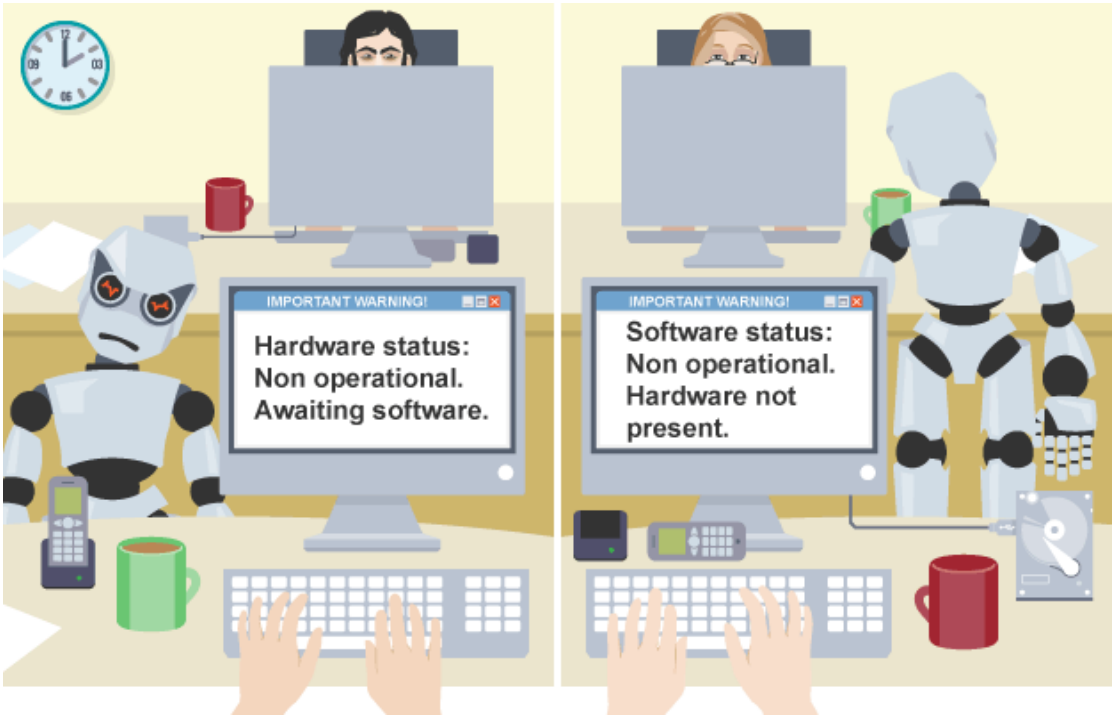
BIT1212

Table of Contents

1.0 What is software?

Hardware is the physical parts of the computer and software is the programs that run on a computer.

There is a close relationship between hardware and software. Without software, hardware is very limited and without hardware, software would not be able to run. They need each other.



Software makes hardware useful. It gives it the instructions it needs to operate. When hardware runs software, it loads the software into its RAM.

There are two main types of software:

- Systems software
- Applications software

1.1 Application software

Applications software is used to carry out tasks on a computer, such as writing an email, making a poster, doing homework and messaging friends. Some of the applications we might use to do this include a word processor, web browser and graphics software.

Some software, such as word processors, spreadsheets and desktop publishers are called general purpose software because it is possible to carry out lots of different tasks using that application.

Some software is called special purpose software because it performs one specific task. This might include a flight simulator, payroll software or an application for revising maths.

The word '**app'** is short for application software. This is a piece of software that allows us to do an everyday task. Today's smartphones and tablets allow us to download apps from online stores, allowing us to:

- See the latest news headlines
- Listen to music
- Message and communicate on social media
- Study, and receive and submit homework
- Play games
- Check the weather
- Book tickets for the cinema and theatre
- Order food takeaways
- Shop online
- Check locations and navigate

Types of Application Software

1.1.1 Word processing

Word processors can be used to create any type of text-based document, from a letter to a novel. You can change a document's appearance using a number of formatting options such as:

- change the font and font size
- bold, italicise and underline words
- colour the text and the background
- highlight words of importance

You can also add tables, images, clip art and shapes to a document. Use formatting sensibly and sparingly.

1.1.2 Mail merge

Mail merge lets you create a template and use it to personalise a document that you're going to send to lots of people. For example, you write a party invitation that starts 'Dear James' and you use James' name throughout. To turn this invitation into a template, replace all instances of James with a name placeholder. The name placeholder is linked to records in a database that contains all of your guests' names. Mail merge will use your newly created template to make personalised invitations for all of your guests.

1.1.3 Desktop publishing

Desktop publishing software (DTP) is used to create documents like leaflets, brochures and newsletters.

Modern word processors have the basic features of DTP software but features such as templates and frames make DTP software better for complex page layouts.

- Templates – examples to base your own document on. Templates provide an easy way of making documents look professional. If you want something that stands out, be aware that the same template can be used by many people.
- Frames - boxes that can contain text or graphics. These can be moved around freely and resized to create the layout that you want.

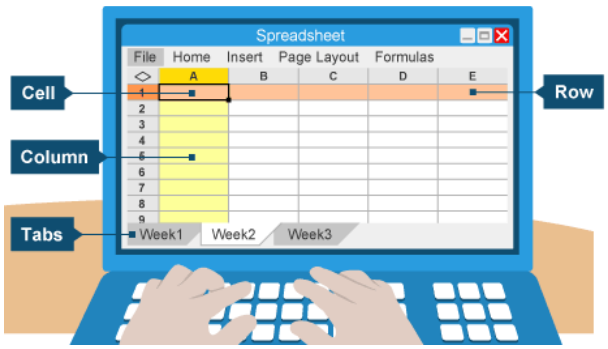
WYSIWYG (What You See Is What You Get)

With WYSIWYG, what is shown on screen is exactly how it will look when printed. A number of office applications work in this way, including DTP software.

1.1.4 Spreadsheets

Spreadsheets are used for calculations, simple databases and modelling.

A spreadsheet is made up of rows, columns and cells. Columns are labelled alphabetically, starting at A, and rows are labelled numerically starting at 1. Each cell has a unique cell reference. The first cell in a spreadsheet is A1, A2 is below A1, and B1 is to the right of A1. A cell can contain data, labels and formulae.



Formatting

The formatting options in word processors are present in spreadsheet applications. You can change the font type or font size of the text or make it bold, italicised and underlined in a cell.

Cells can be formatted too. You can change the background colour and add borders to them.

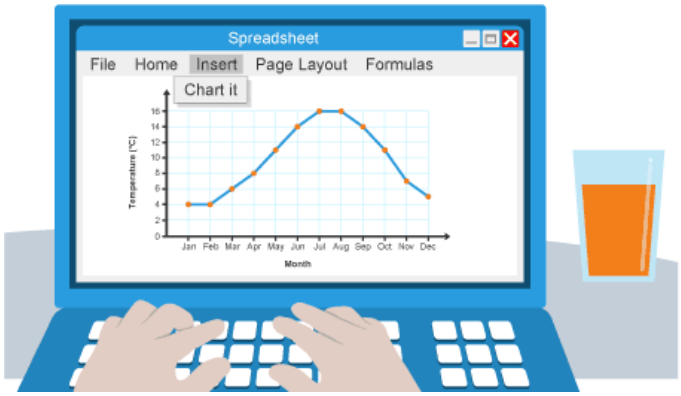
Formulas and functions

Spreadsheets are perfect for performing calculations with data. To do this you need to write a formula. All formulas start with an equals sign (=). You could use a formula to calculate a total. If one of the values that makes up the total changes, the total updates automatically.

More advanced formulas are called functions. These are complex formulas created for you. There are many to choose from and also specialist ones designed for particular jobs or areas of expertise.

Presenting information

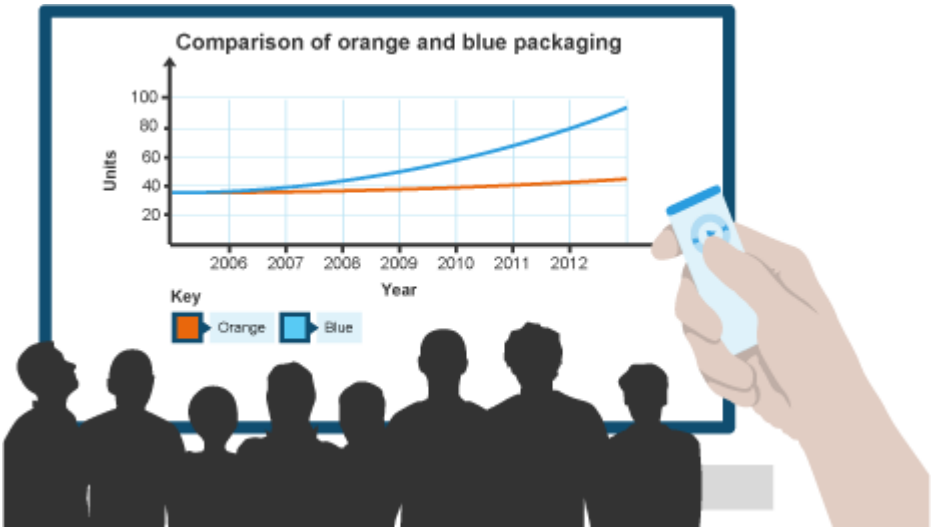
Spreadsheet applications can automatically create graphs and charts to give a visual representation of your data.



1.1.5 Presentation software

Presentation software is used to create slideshows. A slideshow is a series of slides. An individual slide can contain text, images, animations, sound and video. Slides can show automatically one after another or the presenter can choose when to show the next slide manually

A slideshow can act as a visual prompt to a person giving a talk or presentation, or it can give more information about the current topic.



The features of presentation software include:

- Slide master - controls the presentation's appearance and keeps all the slides in the same style.
- Animations – special effects to keep your audience interested.
- Transitions – different ways to change from one slide to the next, eg the current slide dissolves into the next one.
- Speaker notes – the presenter can add notes to remind them what to say when each slide is shown. The notes appear on the presenter's screen but not on the audience's.

1.1.6 Graphics software

There are three main types of graphics software:

- photo-editing
- painting
- drawing

Some of the basic changes they let you make to an image include:

- contrast and brightness adjustment
- rotation
- cropping (choosing part of the image and getting rid of the rest)
- resizing

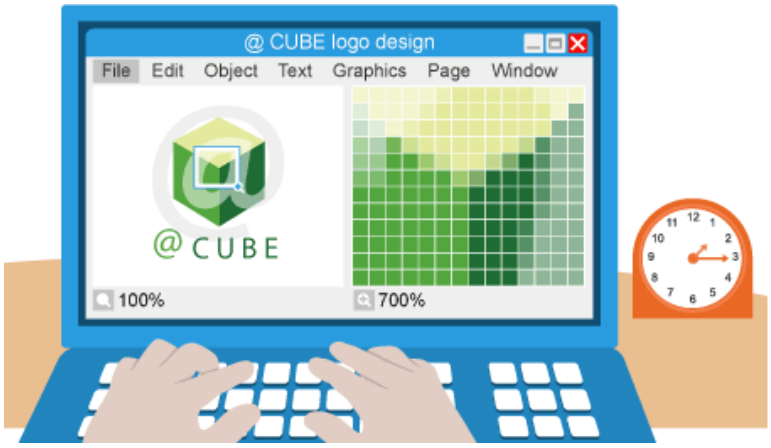
Photo editing

Photo-editing software is used to edit photos taken with a digital camera or scanned and saved to your computer.

Airbrushing is a technique that features in photo-editing software. It is used by newspapers and magazines, sometimes to cover up skin imperfections, to change eye colour and to make people look slimmer.

Painting

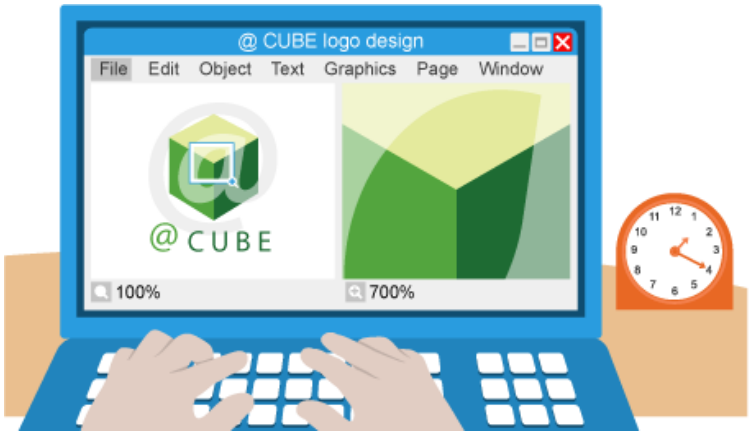
Painting software can create and edit bitmap images. A bitmap is an image made up of individual dots called pixels. Bitmap images lose quality when their size is increased. If you increase the size enough, the image looks as if it is made up of little squares. These are pixels.



Files with the extensions .bmp, .jpg and .gif are bitmap files.

Drawing

Drawing software creates and edits vector images. Vector images are made up of lines, curves and shapes based on coordinates. An enlarged vector image will not lose quality.



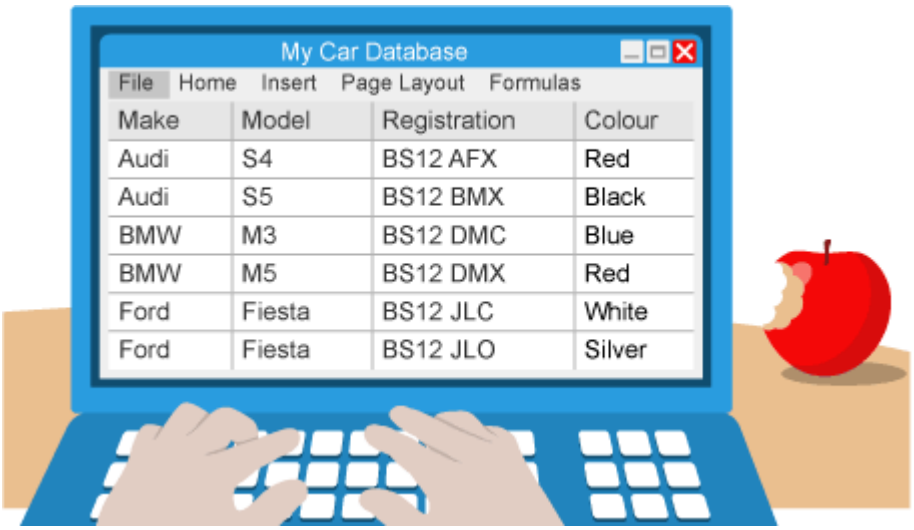
1.1.7 Database software

Database software stores information in an organised way. Individual bits of data are stored in fields. Each field has a unique name, e.g., Address.

A record is a collection of fields that all relate to one another, together they are information. A record about a car would include its:

- make
- model
- registration or number plate
- colour

A table is a collection of records. In the example given, a table would contain individual records of cars.



Key fields exist as a unique identifier. In a car salesman's database 'Registration' would be a suitable key field. None of his cars have the same number plate, but some of them will be made by the same manufacturer or share the same colour. The key field allows the computer to separate each car's record. Key fields are sometimes referred to as **primary keys**.

1.1.8 Online apps

With broadband internet, we also have access to online applications. These online apps are not installed on our laptop, smartphone or tablet but are instead accessed through a web browser. Online versions of apps are often simpler in nature than installed apps but have the advantage that they can be used by anywhere through an internet connected device. Examples include Google Docs and Microsoft 365.

1.2 System Software

Programs that are designed to maintain or operate the computer system are known as **system software**.

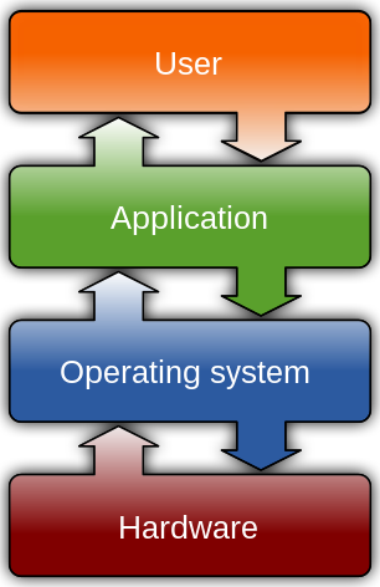
System software:

- controls the hardware, including any peripherals
- allows other programs (applications) to run
- provides an interface for the user to interact with the computer
- maintains the system

It includes the operating system, drivers and utility software.

1.2.1 Operating System

An operating system manages the hardware within a computer system. When a computer is turned on, and after the Basic Input/ Output System (BOIS) has loaded, the operating system is the first piece of software that will load. It sits between hardware and application software and manages any communication between.



An operating system manages hardware by carrying out tasks such as:

- controlling hardware components
- providing a platform for software to run on
- providing a user interface

- multitasking facilities
- managing the computer's memory
- managing peripherals
- managing files
- managing users

Example operating systems include:

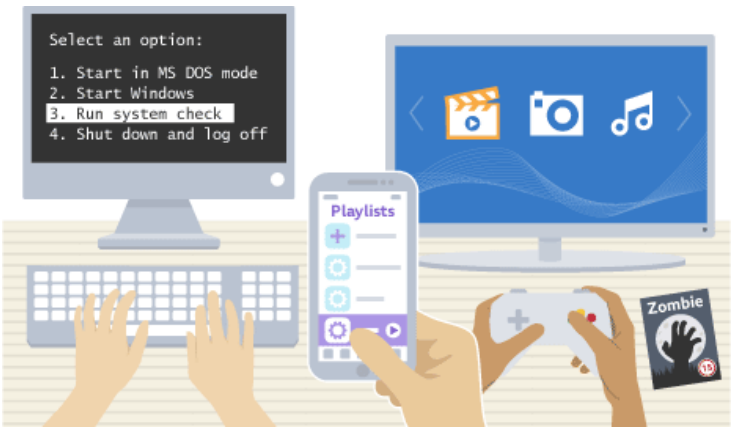
- Microsoft Windows
- Apple OS X
- Linux
- Android
- IOS

User interfaces on operating systems

A user interface is a program, or suite of programs that allows a user to interact with a computer. There are three types of interfaces to be considered:

- graphical user interface (GUI) - sometimes known as WIMP (Windows, Icons, Menus, Pointers) interface
- mobile user interface (Mobile UI)
- command line interface (CLI)

A graphical user interface is familiar to most users of PCs and laptops. GUIs feature a desktop where everything is displayed. Applications run in Windows, and all objects (apps, hardware and files) are represented by icons. Application features are accessible through the use of menus. Users interact with the interface by using a mouse and on-screen pointer.



GUIs are powerful and easy to use, but require a lot of processing power.

Mobile UIs are similar in many ways to GUIs, except that they respond to touch. Fingers are used to open programs and interact with them. Gestures such as swiping are used to scroll within documents. Pinching and stretching are used to re-size images.

Mobile UIs are found on smartphones and tablets.

Command line interfaces are text-based. Users control the computer by typing in commands.

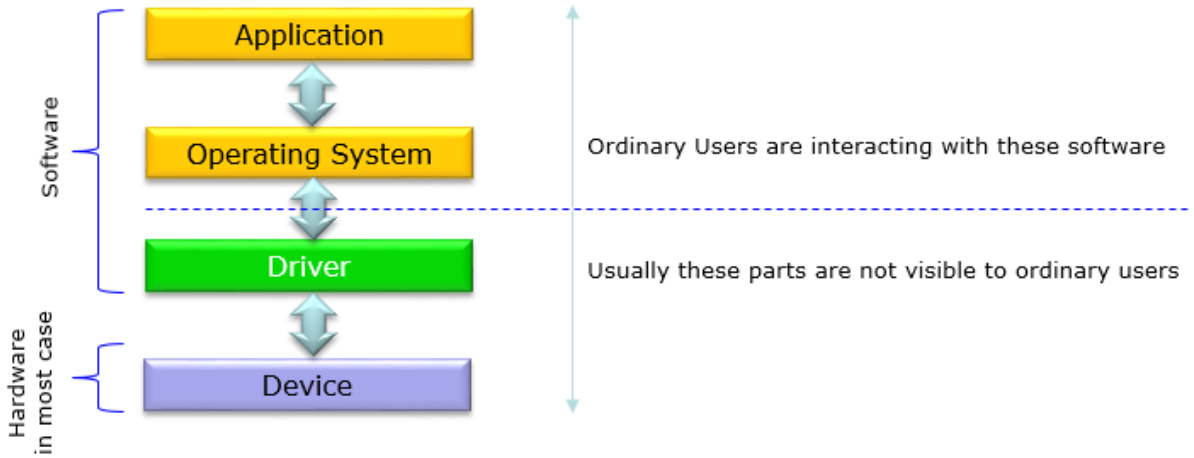
CLIs require little processing power and are extremely powerful, but are difficult to use. Originally most interfaces were CLIs, and they still exist within modern operating systems, for example the command prompt app in Windows, and Terminal in OS X.

1.2.2 Device Drivers

A **device driver** is a small piece of software that tells the operating system and other software how to communicate with a piece of hardware.

For example, printer drivers tell the operating system, and by extension whatever program you have the thing you want to print open in, exactly how to print information on the page.

Device Drivers are essential for a computer system to work properly because without a device driver the particular hardware fails to work accordingly, which means it fails in doing the function/action it was created to do. Most use the term **Driver,** but some may say **Hardware Driver**, which also refers to the **Device Driver.**



1.2.3 Translators

A translator translates a program written in a high-level programming language into machine code that a computer can understand.

1. Compilers

A compiler is a translator which creates a file containing the machine code known as an executable file because it can be executed by the processor. It can also be referred to as the object file. The original high-level programming language file is known as the source file.

When a program is compiled, the whole source code is translated into the executable file at once and can then be distributed to resellers, customers and individual computers. As it is in an executable format, it can only run on operating systems for which the compiler has translated it. For example, programs that have been compiled for Windows© will not work on Linux© unless they are compiled again for Linux. The same situation exists with mobile phone and tablet operating systems.

2. Interpreters

Interpreters also translate a program written in a high-level programming language into machine code, but use a different method. Instead of translating the whole source code at once, it is translated one line at a time.

This can be less efficient than a compiler because it takes time to do the translating as the program is being executed and also because statements within the programming loops (e.g., FOR, REPEAT, WHILE) have to be analysed each time around.

An interpreter translates each line of the source code into an intermediate stage and then execute the line of code. It reports on errors as each line of source code is translated.

Interpreters are often used to translates macros or application-based script languages (for example Visual Basic for Applications), which can be particularly useful when a document needs to be opened on a variety of operating systems. Interpreters are also used when testing programs so that parts of the program can be executed without having to compile the whole program.

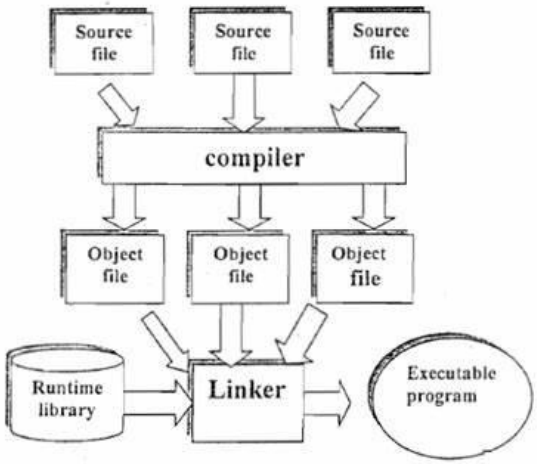
Java uses both a compiler and an interpreter. The original Java source code (files with a .java extension) is compiled into Java bytecode (files with a .class extension) which is an intermediate code. In order that the Java program can work on any operating system, a Java Virtual Machine installed on the computer is used to interpret the bytecode at the time of execution.

1.2.4 Linkers

Computer programs often consist of several modules (parts) of programming code. Each module carries out a specified task within the program. Each module will have

been compiled into a separate object file. The function of a linker (also known as a link editor) is to combine the object files together to form a single executable file.

In addition to the modules used, the program may make reference to a common library. A common library contains code for common tasks that can be used in more than one program; such as mathematical functions, memory management, open and save dialogues, progress bars and input / output. The linker can link modules from a library file into the executable file, too.



1.2.5 Utility Software

Utility software is system software that performs some sort of maintenance on the computer system. Utility software does not include the operating system, but an operating system may come pre-installed with some utility software.

1. Antivirus software

Antivirus software is sometimes referred to as antimalware software, as it deals with other threats such as adware and spyware as well as viruses. It has two main functions. The first is an anti-virus monitor that is continually monitoring the system for viruses and malware. If the anti-virus monitor detects any unusual behaviour or tell-tale signs of viruses or malware then it will prevent them from being executed so they cannot cause damage to files or programs.

The second function is to check for viruses or malware that may already be on a system. This is usually known as scanning the system. If anything is found then the user will usually be given the option to disinfect the affected area, put it into quarantine or ignore it. Ignoring it is very dangerous because it means the virus or malware will be executed and may have unexpected results. Disinfecting is the safest option as it completely removes the threat from the system, but it does mean that any data or program that had been affected would be deleted. The compromise is to put the affected area into quarantine. This is a safe area where the virus or malware

cannot be executed, but the data or program remains isolated until it can be checked more thoroughly.

2. Backup software

Data can be lost accidentally, or deliberately. A user may accidentally delete or overwrite a file, or a hard disk may fail, preventing access to any files stored on it. A hacker or malicious user may deliberately delete or overwrite data.

To prevent data loss, regular copies of the data should be made. A copy of data is known as a backup. Backups can contain a copy of all files on a computer, or just ones specified by a user.

Network managers make regular backups of all files on a network using backup software. The software automatically makes a backup at a scheduled time of day, usually during the evening when the network is quiet. Backups are usually made to a high-capacity secondary storage device, or to the cloud.

If data loss occurs, data can be retrieved from the backup. The software allows all backed-up files to be retrieved or just specified files.

Two types of backups are possible:

- full backup
- incremental backup

A full backup involves making a copy of every file on the computer or network. They can require a lot of storage space and can be time consuming to make.

Incremental backups take a copy of any new files created since the last backup, or of any files that have been edited, such as user documents.

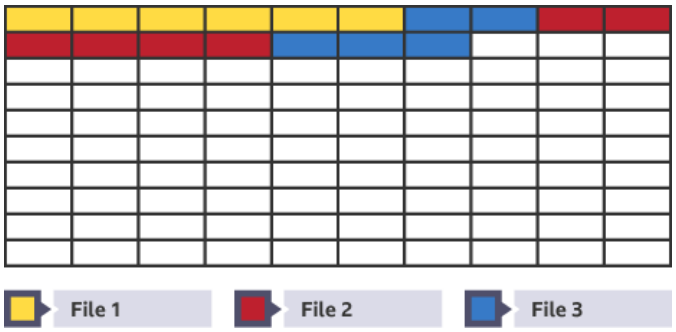
Most network managers make an initial full backup and then switch to daily incremental backups. This way all data is backed up and daily backups are small and less time-consuming.

3. Data compression

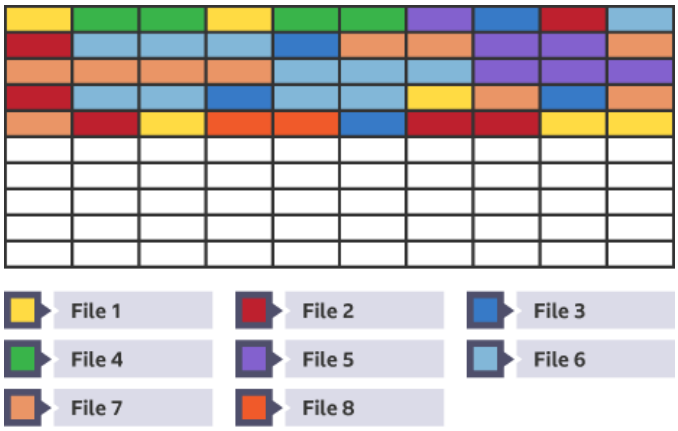
Data compression utilities will reduce the original size of fi les so that they use up less storage space. This can be achieved on a fi le-by-fi le basis or for a set of files or even a set of folders. It will be slower to open the compressed fi le than the uncompressed fi le, but as it is smaller it will use up less storage and can be transferred from one location to another more quickly.

4. Disc defragmentation

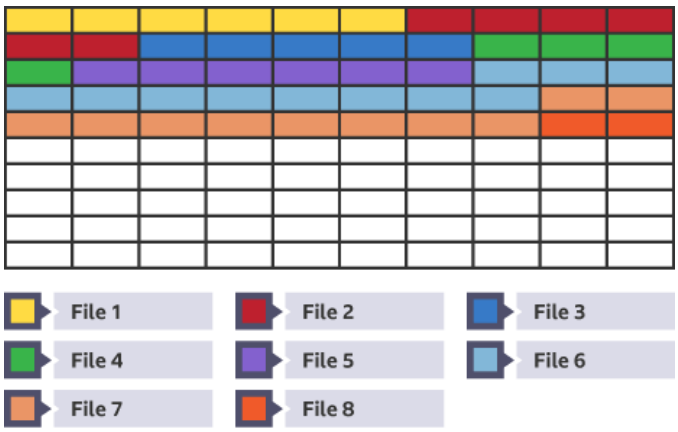
When a file is stored on a hard disk it is actually stored not as a whole file, but as a series of segments. Sometimes the segments run together in sequence (see File 1) and sometimes the segments are split up over a disk (see File 3). This is known as fragmentation.



Over time, more and more files become fragmented, as do individual files. A fragmented disk takes longer to read from and write to, making a computer run slower.



Defragmentation software takes the fragmented files and rearranges the segments so that they run contiguously. This decreases read/write time, thereby speeding up computer performance.



5. Encryption software

Encryption software disguises the contents of files so they can only be understood by authorised users. The software uses a complex algorithm to scramble the content so that it appears to be gibberish. Only authorised users can descramble the content.

The software can encrypt specified files, or the whole of the hard disk on which the files are stored.

6. Formatting software

When a disc is prepared for first time use, it needs to be formatted. Formatting is the process of organising the tracks on the disc into sectors. Each sector is where data will be stored. A used disc can also be formatted, in which case all data will be lost and the tracks prepared again as if the disc was being used for the first time.

7. File management software

There are several functions that can be used to manage files. These include the following.

- Creating a new file or folder, and storing the file data there.
- Naming/renaming files or folders. Two files cannot have the same name in a single folder, but they can if they are in different folders.
- Saving a file, to update an existing version or to save a different version.
- Copying a file to a different folder or to an external device. The original version is retained.
- Moving a file to a different folder.
- Deleting a file that is no longer required. On a PC or laptop, it is usually possible to retrieve the file if it has been deleted in error.

Repairing files

Files can be corrupted due to:

- the computer crashing as a file is saved
- physical problems on a storage device
- a malware or virus attack

A corrupt file can sometimes be repaired, and most operating systems have this utility. Another utility can detect and, if possible, recover from physical errors on the disk. It

scans disk surfaces for defects and marks those sections as unavailable to prevent the rewriting of data and data loss.

It is sensible to always make a backup of files in case they cannot be recovered.

Converting files

A file conversion utility lets users save a file as a different file type. This is useful, for example, when:

- translating a music file to run using a different audio application
- compressing the size of an image file
- changing a file to PDF format so that the contents cannot be altered

1.3 Custom-written software and Off-the-shelf software

1.3.1 Custom-written software

When a client requires a software solution that will carry out a specific purpose that is unique to their organisation, they will ask for the software to be written for them. This is known as custom-written software because it is customised to the needs of the client and will meet the requirements of the client.

1.3.2 Why Choose Custom Software?

Easier Integration - In today's age of specialisation and big data, a bespoke software product might be better positioned to integrate with the other digital tools you may be using. If you have a pre-existing IT ecosystem, investing in a custom software program could make integration easier.

Upgrades on Demand - Any company that develops and sells an off-the-shelf software solution will constantly upgrade it. However, they may not necessarily update the program very frequently. If you want quicker updates, you will find most bespoke software development companies will be happy to add any features you want, when you want them.

Scalability Requirements - Scalability is a particularly important metric for growing businesses, whether already quite big or just starting up. Whilst your needs may be limited right now, you will need to consider requirements for when your business does start growing. If scalability is a requirement for your software development needs, it may be more efficient for you to invest in a custom software product.

Unique Services - In order to stand apart from your competitors, you need unique services or functionality that they can't, or don't, offer. If this exclusivity is important to you, then you may not be satisfied with the generic solutions offered by some off-the-shelf products. In such a case, you should invest in custom software solutions.

On-going Maintenance - Off-the-shelf software products don't come with the option of on-going maintenance. If you have a system bug, you may need to report it and wait for the fix in their next upgrade. If you have a bespoke software product developed, your software development company will give you quicker bug fixes and on-going maintenance.

Training - With custom software, you might need to pay the software development company to train your staff. However, the training can be focused on the features each team member would be using.

Also, since your employees aren't expected to learn how to use the product or its features themselves, the training is quicker and their questions are answered immediately.

Efficient Size - An off-the-shelf product is designed for a mass market. This is why it could have features and functions that you might not need, but will have to pay for and install. A custom software program, on the other hand, is less bloated and will only have the features that you specifically want. In fact, at One Beyond, we work with the Minimum Viable Product (MVP) so you get an efficient product that gives you the functionality you need without any extraneous bytes clogging up your IT system.

Features Flexibility - With an off-the-shelf product, you have no choice but to accept the features offered, even if they aren't absolutely ideal for your needs. If you want a different set of features, you will need to look around for another program that offers them. Custom software development, especially when using the Agile software development methodologies, allows you much more flexibility. You can choose the specifications you want, or change and edit the features you require (to some extent) during the build.

Number of Users - Most commercial off-the-shelf products charge you a fee for the number of employees using the program. If your company grows, you will have to pay more to accommodate the increase in the staff. A custom software program, on the other hand, belongs to you. You can have as many users as you want, without having to pay a user fee for them.

Product Quality - As mentioned earlier, off-the-shelf software programs can be bloated with unnecessary features that you don't need. Additionally, these products may not offer the level of quality you desire.

If you want a high-quality program that has been through intensive software quality testing to give you optimal performance, custom software development will give you the results you need.

1.3.3 Off-the-shelf software

When software already exists and is purchased online or from a shop, it is known as off-the-shelf software. The software will have been written for a general purpose that is likely to be useful to a large market. Anybody can purchase the software for a specified price.

1.3.4 Why Choose Off-the-Shelf Software?

Lower Cost - Off-the-shelf products can be cheaper than custom software as the cost of development is divided among the end users instead of a single client.

Quicker Implementation Time - Off-the-shelf programs can be implemented very quickly as all you need to do is buy and install.

Company Needs - Not all companies have the need for a custom software solution, so if you don't have highly specialised requirements, you would find it more efficient to buy a generic off-the-shelf product.

Unsure of Requirements - If your company is brand-new, you may still be 'getting a feel' of your requirements. In which case an off-the-shelf program will allow you to explore your requirements before you commit to a bespoke software build later down the line.

Community Help - Since an off-the-shelf product is being used by millions of other people, you will usually find an active community helping and supporting each other. You can get instant support or help from community member if you use an off-the-shelf software product.

1.4 Software Categories

1.4.1 Proprietary Software

Proprietary software, also called **closed source**, is a type of software owned by a single entity and licensed to other parties who wish to use it. This means this type of software doesn't allow users to access the source code. It typically restricts how users can use and modify it, differentiating it from open-source software.

Once you purchase proprietary software, you receive a license to use it. This license limits how you can use it and restricts its further distribution. For instance, the owner

may not allow you to make copies of the program or modify it in any way. This means the source code remains closed, and users can only access the compiled version of it.

In addition, proprietary software typically has a copy protection system that makes it hard to copy and distribute without permission. Proprietary software is usually licensed per user or on a subscription basis, which provides software owners with a steady and reliable source of income.

Proprietary software companies typically hold all rights and ownership over the software. The owner can, therefore, take legal action against anyone who violates the terms of their license. Companies and users are, therefore, advised to read and adhere to the license requirements of any proprietary software they purchase.

Examples of Proprietary Software

Various industries use proprietary software for different applications. This section will look at some of the most popular proprietary software categorized by use.

1. **Proprietary Antivirus Software**

With the rise of online threats, antivirus software has become a necessity for computers and other devices. Proprietary antivirus software is primarily designed to protect users from viruses, malware, and other cyber threats. Examples of proprietary antivirus software include:

- Norton Antivirus
- McAfee Antivirus
- G DATA Software
- Kaspersky Anti-Virus

2. **Proprietary Operating Systems**

These underlying software structures enable computers and other devices to function. Proprietary operating systems have been around for a long time, with most of them created by major computer manufacturers like Microsoft and Apple. Examples of proprietary database software include:

- Windows 10, 11
- iOS
- ChromeOS
- MS-DOS

3. Proprietary Database Software

Database software is used to store, organize, and analyse data. Proprietary database software allows users to manipulate data and build reports without having access to the source code. Examples of proprietary database software include:

- Oracle Database
- IBM DB2
- SQL Anywhere
- SQL Server Express

Advantages of Proprietary Software

Proprietary software has various advantages for its users. These include:

Bug Free

Many proprietary software companies invest heavily in quality assurance and testing. This ensures the software is bug free and provides users with reliable solutions for their needs. Proprietary software companies also provide free patches to fix any bugs that may arise. This makes this type of software more stable and reliable than its open-source counterpart.

Easy to Use

Proprietary software companies design their software with the user in mind. These companies invest heavily in user interface design and ensure their software is easy to use, even for those with no technical background.

High Performance

Proprietary software companies typically invest heavily in research and development to ensure that their software offers high performance. Proprietary software is, therefore, often faster and more reliable than open-source alternatives.

Better Support

Proprietary software companies often support users via helpdesks and online resources. This additional layer of security ensures that users can quickly and easily get help if they need it.

Clear Roadmap for Development

Unlike open-source software, proprietary software companies usually provide users with a clear roadmap for development. Proprietary software companies typically have

a team of developers working on the software, which helps them craft software upgrades. This ensures that users know the features to expect and can plan accordingly.

Disadvantages of Proprietary Software

Proprietary software also has some drawbacks, including:

High Cost

Proprietary software companies often charge an initial (and sometimes) ongoing subscription fee for their software. This can be expensive for companies and result in a large upfront cost. Moreover, using proprietary software requires companies to hire proprietary developers to maintain and develop their software. Hiring and training these developers can be expensive and time-consuming.

Fewer Options

Proprietary software companies typically offer limited options and features. This means users may not find the exact solutions they need, and have to settle for something close enough. Proprietary software companies cannot always add custom features or modify the software to fit their specific needs.

Limited Freedom of Use

Proprietary software is often heavily restricted in terms of how users can use it and what changes users can make. Proprietary software companies often require users to sign a license agreement, which specifies how they can use their software. Proprietary software is also typically copy-protected, making it difficult to share or resell.

Restricted Customization

Proprietary software is usually designed for a specific purpose. Therefore, users often cannot customize it to their own needs. Proprietary software companies also don't usually provide users with the ability to modify or tweak their software.

Lack of Transparency

Proprietary software companies often keep their source code private. This means that users may not be able to review the code to ensure that it's secure and reliable.

1.4.2 Open-Source Software

Proprietary software is different from open-source software. Proprietary software is not open to the public, and users cannot access or modify the source code. Open-

source software, however, is publicly distributed and allows users to view, modify, and share its source code.

Open-source software is usually free to use, while proprietary software typically has a license fee. Proprietary software companies can also charge for upgrades and support, while open-source software companies may not. Proprietary software is often more user-friendly since it's designed for a specific purpose. Alternatively, open-source software is customizable.

In some instances, open-source software can become proprietary, known as a proprietary fork, which are versions of open-source software that developers modify and copyright as their own.

1.4.3 Freeware

Freeware is software that users can download and use without restrictions or fees. Independent developers create freeware and typically don't provide support or take responsibility for any issues related to it. Users can share and modify freeware as long as they make the changes available to the public.

1.4.4 Shareware

This type of software is initially free to use. However, users may need to pay a fee or register at some point if they want to continue using it or access additional features. Shareware is usually offered in trial versions, and users can decide whether or not to purchase the full version. Some popular examples of shareware include Adobe Photoshop, WinRAR and Skype.

1.4.5 Abandonware

As the name suggests, this is software that has been abandoned by its developer. It's the perfect solution for users who want to use software without paying a fee since it is no longer supported by its developer. However, due to the inherent security risks, users should steer clear of abandonware as it could contain unpatched vulnerabilities that can put your personal information and data at risk.

1.5 Types of Proprietary Software Licenses

Software licensing is a legal agreement between the company that owns the software and its users. It is a way for companies to protect their intellectual property and ensure that users adhere to the terms of use. When you use proprietary software, you must adhere to its license terms.

A proprietary software license generally includes the following information:

- The proprietary software company's name and contact details.
- What the user can do with the software.
- Ownership of the software and intellectual property rights.
- Prohibited uses of the software.
- License termination terms.
- Proprietary software company's rights and responsibilities.
- Proprietary software company's warranties and liabilities.
- Proprietary software company's customer service policies.
- Proprietary software company's refund policy.
- Proprietary software company's support and maintenance.
- Proprietary software company's liability for damages caused by the software.

Microsoft is a great example of a proprietary software company. The company offers several licenses for its software, such as the Microsoft End User License Agreement (EULA). This license includes the standard proprietary software license terms, such as prohibitions on reverse engineering and redistribution of the software.

The EULA also specifies the rights and responsibilities of users with respect to the software. It also outlines how users can terminate the license in certain circumstances, such as if they violate the terms of use.

Proprietary software licenses typically fall into two categories:

Perpetual Licenses

A perpetual license is a one-time fee for the right to use proprietary software indefinitely. The user does not need to pay ongoing fees and can use the software if they comply with the license terms. While this type of license may appear as a cost-saving option, you may end up paying more for extras, such as support and upgrades.

Subscription Licenses

A subscription license involves a recurring fee for access to the software. The user pays a fee each month or year to use the software and can cancel their subscription when they choose.

1.6 The Future of Proprietary Software

While open-source software has become increasingly popular in recent years, proprietary software still plays an important role in businesses and organizations. Proprietary software companies often have the resources to develop more feature-rich and high-performance software compared to open-source companies. Proprietary software companies can also provide better support and customer service than open-source companies.

In the future, proprietary software will likely remain a key part of business and organizational functioning. As such, proprietary software companies will need to stay competitive by providing high-performance and reliable software. They'll need to offer flexible and cost-effective licensing options and provide better customer service and support. Proprietary software companies will also need to focus on developing innovative solutions to meet the changing needs of customers.

Nevertheless, some industry leaders have predicted that open-source software will eventually replace proprietary software. Proprietary software companies must, therefore, stay abreast of the changing technological landscape and adapt accordingly to remain competitive.