Phone List - A Sample Information System

Here are some telephone numbers :

Einstein, Albert	06921-5794
Schneider, Anna	06123-45678
Lee, Kim	069-876543
Lewis, Susie	03123-48576

Imagine that this list continues, containing a total of **40 million entries**. It is not in alphabetical order, nor are the phone numbers grouped by city. It would be very difficult to do anything useful with this list of numbers. Imagine that this represents all of the telephone numbers in Germany. There are several different ways to **organize** the **data** to make it more useful:

Last name - Alphabetical order by last name
First name - Alphabetical order by first name
One City - Only display a specific city (determined by the first part of the number)
Phone order - Sort the entire list by telephone number (thus 03123..., then 06123..., then 069, then 06921...)

Different organizational schemes make the list useful for different purposes.

- To find a specific person, you probably use the last-name list (or perhaps the one-city list, if the names are sorted)
- To find a person if you don't remember the last name, you use the first-name list
- To find a doctor in your city, use the one-city list
- To find a doctor in your neighborhood (as close as possible) use the phone-order list and look for the same prefix as your phone number

The original list is simply **data -** it is not organized. As soon as the data is organized in some fashion, it becomes more useful. Part of the organization might involve **throwing away** some of the data - for example, throwing out all the numbers which are not in the correct city. Another part of the organization might involve a better **presentation**, which makes the list easier to read - for example, putting the last name in **bold** print, or double-spacing the list.

Forty million phone numbers is a lot of data. It seems like it should be useful because it is complete. However, there are lots of problems associated with this large amount of data:

- If printed on paper, with 200 entries per page, it would require 20,000 pages, which is **too big** and heavy to carry
- Copying 20,000 pages would also be expensive and take a long time
- Phone numbers are constantly changing as people move, so keeping the list **up-to-date** will be very difficult
- If one phone number changes, the person must be placed in a different place in the sorted lists

Large quantities of data can be handled more efficiently by **electronic systems** (computers) than with conventional **hard-copy** (paper) systems.

If you visit a German post-office, you may see a collection of 100 telephone books covering the entire country. These books are printed and replaced once a year - thus there will be lots of **out-of-date** (incorrect) entries in the books in the middle of the year, and they will not be corrected for several months.

The actual **permanent storage** of the phone numbers is not in the telephone books in the post office - it is in a **computer database**. Once the data is stored electronically, its use can be expanded. For example, modern telephone systems permit a **call-forwarding** service - if an employee is out of the office in a meeting all day long, they can have their phone calls **transferred automatically** by leaving an electronic message in the telephone system. In the past, this same service would be accomplished by a secretary answering the phone and transferring the call, but now a computerized system **automatically** transfers the call.

Computers also enable **bad uses** of data. In the United States, computers make phone calls to try and sell products - this is only possible because the telephone lists are stored in the computer. In the past, telephone marketing was done by people reading lists of phone numbers and making phone calls. This was more expensive (the phone-callers had to be paid) and thus less of it happened. Thanks to automation, computers can annoy lots more people and more often.

This sample information system is referred to below as the **phone-list** system.

Other Sample Information Systems

Multimedia Encyclopedia (e.g. Encarta)

Old-fashioned encyclopedias were printed on paper, in many volumes. They contain a few pictures and lots of text. The articles are stored in alphabetical order - for example, the **A** book contains "Africa" and "ant" and "army".
Each article ends with a list of references.

A **multimedia encyclopedia** contains all of the same information, plus animations, video clips and sound clips - for example, an animation of an automobile engine running, a speech by Martin Luther king, a recording of a Beethoven symphony. As well as a list of references, the multimedia encyclopedia may contain **hyper-links** allowing the user to click and move automatically to a related article. Some encyclopedias link to **web-sites.**

A multimedia encyclopedia is typically stored on a **CD-ROM** or **DVD**, and requires a **fast computer** with support for **sound and graphics** (multimedia).

Medical Records

Doctors must keep records of their patients, including the name, birthdate, address, phone number, insurance ID number, list of diseases, list of treatments, beginning and ending dates of a hospital stay, dietary requirements, allergies, etc. A **Data-Base-Management-System (DBMS)** is usually the best choice for this storage. In a private practice, a doctor may use a single **PC** to store the database. A hospital would use a **Local Area Network** with a **server** containing all the data, and a **work-station** on each doctor's and nurse's desk. The data in this system is generally **text-based**, so very simple PCs are adequate as **work-stations**. The server certainly does not require any multimedia capabilities (it might not even have a monitor attached) but must be **very fast,** to manage multiple connections and contain **lots of hard-disk storage** to store data for lots of patient records.

Travel Agent

Travelers often purchase airline tickets and make hotel reservations through a travel agent. The travel agency uses **paper catalogues** and **on-line services** to collect and present information to customers. They use **telephones, paper mail, e-mail and on-line services** to make reservations and purchase tickets. They collect large sums of money from customers, so they usually have **financial software** to keep track of the money.

Air Traffic Control

Air traffic controllers depend on a **real-time** computer system to track planes and display the tracking information on monitors. The system does not need to store very much information – archives of previous weeks are not very useful – but must respond immediately to input. Everything must be done in **real-time** – that means as fast as it happens in the real world.

Google

Google is a huge collection of **servers** (about 1 million servers) containing a massive database of information about billions of web-sites. It also provides for thousands (or millions?) of simultaneous connections. Google’s system of servers forms the largest single computer system on planet earth.

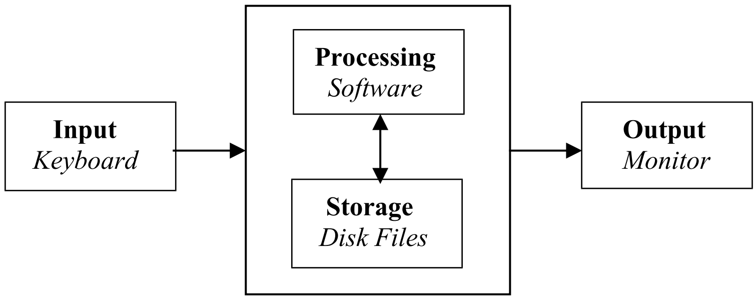
Google doesn’t just retrieve saved data – it also **collects** and **processes** data. Automated “spider” programs exam millions of web-pages each day. A very clever (artificial intelligence = AI) program **parses** the words in the web-page and updates **indexes** that keep track of where words occur in the web. The AI is quite clever and “understands” when two words mean the same thing, like “car” and “automobile”.

Since it deals with vast quantities of information and communication, Google’s software must be extraordinarily efficient. Google decided to use **Linux** as the OS for their servers because it is simple and efficient, as well as being an **open source** product that they could **modify** to meet their needs more perfectly. And since it has so much hardware, Google had to optimize the cost and reliability of the hardware, as well as the operating costs and energy consumption. They created **server farms** in huge warehouses containing thousands of machines to optimize operations.

Computer Systems

Information storage and processing are moving **away from paper** toward **digital media** - audio recordings, video recordings, and other computerized storage. Especially in computer systems, there is a difference between **data** and **information**. A **data-file** may contain thousands of numbers representing stock prices. People don't find it easy or informative to read the list of numbers. When the numbers are **processed** (sorted or collected) in some fashion, the result is more useful than the original data – then we call it **information** rather than data. Most large information systems are implemented in **computer systems.**

A **Computer(ized) System** is a very general concept, not necessarily associated with **computers** – it might use cell-phones or media players or other digital devices. Every computer system contains **four component concepts:**



(Q1) For each sample information system above (page 1-2), identify the principle sources of input.
Remember that the **system** contains more than just computers - **people** are likely to be a source of input.

(Q2) For each sample system above, give an example of output that should be produced on paper.

Computer Science studies the problems associated with **automating** and **digitizing** information systems. Thus, Comp Sci doesn't deal with paper magazines as information systems, but might compare electronic magazines to their paper counterparts or consider the problem of converting from paper to digital format.

Standard Hardware

In a typical **PC** (Personal Computer) system, **standard hardware** includes:

Input	***Storage***	***Processing and Communication***	***Output***
Keyboard Mouse Microphone Scanner Digital camera Video camera Touch screen	Hard-disk Floppy diskette CD-ROM Streamer tape RAM (temporary) Flash Ram (permanent) Video Tape	Graphics adapter Sound card Modem Video-capture card	Monitor (VDU, CRT) Printer Speaker Data-file

Software

Hardware is pretty much "dead" without **software** to control it. Software **controls** all parts of the system.

Input - Data Collection (Data-capture)

During input, software **validates** and **converts** the data before storing it. Some examples:

Validation - Users make errors when typing data into a database. To prevent errors, software can:

- Compare names to a **standard list** to make sure the name is in the list
  "Germanie" is not a valid name for a country.

- Compare numbers to a **valid range.** For ages, the valid range is between 0 and 120.
  211 is probably not a valid age - the user probably pressed the 1 key twice by accident.

Digitizing - Pictures are chopped up into **pixels.** The color of each pixel is **converted** to a number and stored.
Music is digitized by **sampling** sound waves and recording thousands of measurements per second.
The results are stored in data files consisting of millions of **Bytes** (numbers).

Compression - Sound and video take up too much space, and would quickly fill up a hard-disk.
Compression removes unnecessary data, storing a much smaller amount.
A typical conversation between two people contains lots of silent space, where neither one is talking. If the **sampling rate** is 40000 numbers per second, then 5 seconds of silence contains about 200 KiloBytes of data. This can simply be removed and replaced by a code consisting of two numbers - 0 5000 - meaning no sound for 5000 milliseconds.

Processing

Large collections of data - e.g. 40 million phone numbers - are appealing at first glance. "Wow! You mean this CD contains **all** the phone numbers in Germany?" But vast collections of data are actually **useless** without further **processing.**

Common Data-Processing Tasks

Searching - Retrieve part of the data, matching certain **search criteria.**
In a school data-base, find all the 11th graders, or all the students with **GPA > 3.25 .**

Sorting - Putting the data in a specific order - e.g. alphabetical list of names, or in prices in order from largest to smallest.

Analysing -
Relationships - e.g. noticing that stock prices increase on the same days when the temperature goes up
Statistical analysis - e.g. finding the average age of the employees in a company
Trends - e.g. finding a stock whose price is steadily increasing

Clean-up - Removing duplicate names, finding values which are probably incorrect (such as blank fields)

Output

After input and processing, data can be **reported (displayed** or **printed)** in a variety of ways. This may require **reorganization**, such as organizing the data into **lists** and **tables**, or **representing** the data in a different way, such as a **graph** or a **statistical analysis.**

Data, Information, Knowledge, and Wisdom

Data by itself is not particularly useful - just a long list of names or numbers. Data which has been processed and reorganised may become useful - at this point it can be called **information.** When information is consumed by human beings who understand its significance, the result is called **knowledge.** Human beings slowly collect information, knowledge and experience. After many years of collection, the resulting ability to act sensibly is called **wisdom.**

Lots of numbers and text, collected ---> Data

Lots of Data + Lots of Processing (IT) ---> Information

Information + Human Understanding ---> Knowledge

Information + Knowledge + Experience ---> Wisdom

An example - Medical Research

In the 1800's, hospitals were quite dirty and it was common for patients to develop infections due to the filthy conditions. It seems obvious to us now that people develop infections easily in a dirty environment. However, this was not obvious to doctors 200 years ago. Modern medical researchers collect lots of data, analyse it and search for connections between diseases and other factors such living conditions, genetic background or dietary habits.

The process of data collection and analysis is easier, faster, more reliable and more effective when computers are used. Nevertheless, relationships between sets of data are only useful if a human can **understand** and **interpret** the meaning of the relationship. For example, data collected from heart-attack victims might show a relationship between poor eating habits and a high rate of heart attacks. That relationship (information) can be detected and reported by a computer analysing the data. An ambitious author might seize this information and write a book about how to prevent heart attacks by eating better. But a doctor with more **understanding** might realize that the bad eating habits and the heart attacks are **both caused** by a third factor - people with very busy, high stress jobs. The addition of **understanding** to the **information** results in **knowledge** - knowledge does not come from the information alone.
A doctor with **more experience** and **more knowledge** is more capable of using this information **wisely -** giving better advice than the poorly informed author of the diet book.

Depending on Computers

Modern industrial societies **depend heavily** on computers. Air-traffic control systems, stock market trading, cash registers, and eMail are just a few examples of systems which are highly automated - if the computers are turned off, the corresponding tasks will simply fail.

Advantages of Machines

Computers are **fast and reliable** - reliability means that they will perform the same task over and over again, always the same. Machines don't get tired and make more mistakes at the end of the day than in the morning. Robots are enormously **strong** and can work in **dangerous environments** where people might be injured or die - in very cold or very hot rooms, or in factories where dangerous chemicals are used.

Disadvantages of Machines
Machines are **not clever or flexible or adaptable.** People can respond to **new situations** with ingenuity and flexibility. Machines would simply continue to do what they were programmed to do, even if new conditions made this senseless. People can **make plans** and carry them out. People have a variety of **senses** - seeing, hearing, touching - which are very useful. These senses are only barely beginning to develop in machines.

Machines + People = Powerful Teamwork

Many computerized systems depend on **cooperative work** being performed by machines and people together. The Airbus uses computers to control the movements of the plane. The pilots don't control engines and other parts of the machine directly. Instead, the pilots **make decisions** and **give commands** to the flight computer, which then moves the control surfaces and controls the engines to match the pilot's request. In a business, a **bookkeeper** uses a computer to **calculate** income and expenses, and to coordinate these calculations to prepare a **budget.** This can be dozens of pages long. A human being doing these calculations by hand would make a lot of errors, and would require a lot more time. The computer would not know **what data** should be typed in, or how the different parts of the budget should be **organized.** With the human being planning and giving commands to the computer, and the computer doing the calculations quickly and accurately, the result is a powerful combination which can prepare a budget quickly and efficiently. Especially where **vast quantities of data** are involved (e.g. the German phone-list) computers are indispensable.

Dangers

Too many people **believe anything produced by a computer system.** This may be a sensible attitude under many circumstances, but **not always.** The **high reliability** of computer systems is **no guarantee** that the system is producing **correct information.** There are many **reasons to mistrust machines:**

Human Error - Data is often **typed** into computer systems by people. The people make mistakes. The resulting **bad data** (garbage) is often stored by the computer without question. For example, 40 million telephone numbers are unlikely to be typed in with no errors. Names will be misspelled, phone numbers will contain **transposed digits** (backward). If a secretary types the wrong thing into a hospital data-base, a patient could receive the wrong medication and end up dead.

> ***GIGO*** **= Garbage In, Garbage Out**

Machine Stupidity - If a computer makes an error, it won't notice. In fact, it will almost certainly repeat it - many, many times. The Year 2000 problem (Y2K) results from computers which store the year as two digits. Hospitals have already experienced problems with birthdates which were before 1900, so that a patient born in 1897 has a calculated age of 2 years in 1999, instead of 102. A famous **computer-criminal** reprogrammed a bank computer system to take 10 cents away from each customer's account, and place the accumulated sum (thousands of dollars) into one account for the criminal. The computer continued doing this once each month for well over a year, and the criminal escaped with several hundred thousand dollars.

> ***To err is human - but to really foul things up, you need a computer.***

Unethical People - Computers are a **powerful tool** - they can perform **lots of good work**. They can equally perform lots of **bad deeds.** The amazing speed of information transfer through the Internet can allow you to send and receive eMail, but can equally allow an unseen person to read your eMail. You can buy books from Amazon.com, but bad people can use the Internet to sell your credit card number to a criminal.

An Information Economy

In the 1700s, Europe had an **agricultural economy** - most people worked on farms, growing crops and livestock.
In the 1800s, the **industrial revolution** changed the economy to an **industrial economy**. Machines enabled farmers to grow a lot more food with less labor, and many workers began working in factories, producing machines, clothing, etc.
Increasing **automation** allows more and more production to be done with less labor. In the 1900s, many workers have left the factories and now work in offices. Earlier they were called **paper pushers** because they moved papers from one side of their desk (the **in box**) to the other side (the **out box**). The office might be part of an automobile company, but the workers don't build or move around cars - they are moving **information** around. The actual **goods and services** still need to be produced - cars need to be manufactured and doctors need to do operations. However, more and more goods and services are now produced by fewer and fewer workers - the rest of the work force has turned into "business" people, busy controlling the distribution of these products. Their direct concern is **information** - prices, stock levels, phone numbers and addresses of customers, etc.

An automobile manufacturer might produce great cars - for example, Rolls Royce. But they still need to be able to **sell** the cars in order to make a profit. **Marketing** (selling) requires information to sell products. What is the right price to charge? Who are the likely customers? How can we inform the customers about our products?

Buying Information - Information as a Commodity

In the **Information Economy,** information can be collected and transmitted, purchased and sold as if it were a real product. Stock-brokers sell their advice about which stock prices are likely to go up and down. These predictions (advice) are information that is sold to customers who use it to decide which stocks to buy. The stock-brokers base their advice on information about past stock prices, as well as business plans for companies and political events (e.g. news). The stock-brokers pay money to news agencies who collect and provide this information. The news agency is in turn paying money to researchers and reporters who collect the information. They might be paying the researchers for their work, but what they are actually getting for their money is information. Many people subscribe to a daily newspaper. They might use the old newspapers to wrap their garbage, but that isn't the reason they buy the newspaper - they buy it to get the information it contains. This includes the weather report, political events, classified ads, and advertisements. People are buying advertisements? Yes, they are! Advertisements are valuable information - they tell you what products are available, where, and for what price. A street map, or a book containing street maps for the entire country, is a very simple example of useful information being purchased by normal people.

A Digital World

Everything seems to be "going digital" these days. Digital television is going to replace normal television. Digital recordings (CDs) replaced vinyl phonograph records about 20 years ago. Digital thermometers are quite popular now as a replacement for old-fashion mercury thermometers. Digital cameras have taken over from old 35 mm film cameras. Digital watches were a big hit in the 1970's and 1980's, but old-fashioned watches with "hands" (Swatches) have made a comeback. Did you know that your Swatch is actually a digital watch inside?

Analog data is recorded as **continuous measurements**. Old-fashioned radios had tuning dials which moved smoothly across the when a knob was turned, with no specific stopping points. **Digital data** is limited to **specific values (discrete) -** counting rather than measuring. A digital radio dial counts 89.4 , 89.7 , 90.0 , ... when a button is pushed. It cannot stop "between" two stations, say at 89.653.

In the **real world**, sound can have any frequency. Middle A is 440 Hz. A slightly higher tone is 441 Hz . But a violin can also be tuned to 440.137 Hz - there are no specific limits. When a sound is **digitized**, it will be converted to numbers using a **fixed counter**. A violin sound at 400.137 Hz might be digitized to 400.1 (e.g. rounded off).

Although analog data is more "realistic" than digital data, a digital recording can be **copied perfectly** because it contains **exact numbers**. Cassette tapes use analog recording techniques. If a cassette tape is copied, and the copy is recopied, and that goes on through 10 different tapes, the final recording will have very poor quality. The original frequencies are not copied exactly. But digital data can be copied absolutely perfectly, so a CD can be copied and recopied hundreds of times and the final copy will be **exactly** as good as the original - e.g. "perfect". Thus one advantage of digital data is **error-free (perfect) reproduction.**

Another advantage is that digital data can be easily **processed and changed**. Editing and changing analog data usually only succeeds on large pieces of data - for example, cutting out 5 seconds of a recording. Digital editing can be done at a very small scale - cutting out a few milliseconds. Digital data can also be processed in a large variety of ways.
The computerized "special effects" that you see in action films and which you can perform with programs such as Paint Shop

Pro (e.g. negatives, changing one color to a different color, morphing) can be performed with relatively inexpensive equipment on digital data. The same processes are difficult or expensive to perform on analog data.

The most obvious advantage of digital data is that it can be **stored and transmitted in computer systems** - e.g. transmitted through the Internet. The wide availability of digital computing equipment and communication devices makes digital data a very popular choice.

What is a **disadvantage** of digital data? If you have ever tried to use a computer "paint" program, you probably realized quickly that it is easier to draw with a pencil and paper. Artists complain that they cannot really produce "art" with a computer paint application. The mouse is a problem, but the bigger problem is that you cannot smear colors together the way you do with real paint. Real paint and brushes are more flexible than computerized tools. Similarly, a traditional photograph is a much more realistic reproduction than a digital photograph from a digital camera. And you can magnify (blow up) a real photograph without much loss in quality, but when you "zoom" into a digital picture, the **pixels** become quite large and ugly, and ruin the quality of the picture. **Analog data is more realistic** than digital data.

Making the Connection

A **network** consists of computers connected by **cables** (wires) or other **communication devices** (satellites, radios, etc). Networks contain three different types of computers -

Workstation (Client) - The PC where a user sits and types in or retrieves information.

Server - A computer which **contains** information that can be retrieved (accessed) by a workstation. Servers don't have a user sitting in front of them - they function automatically, without supervision, and are generally active constantly, around the clock.

Router - Receives data from one computer and transmits it to another computer. A router is not a server - it does not contain any information. And it does not have a user sitting at it. Routers are very highly automated, and might have all their software "hard-wired" into chips - no hard-disk storage here.

The Internet

When you type a **web-site-address** into a browser, this identifies a server which you are trying to "visit". Your computer creates a **packet** containing the server's IP address (123.234.13.1), the address of your workstation, and a message requesting an HTML page to be sent. It sends the packet through the telephone lines to the nearest router. The router examines the server address, and selects the best router to "forward" your packet to. The packet is passed from router to router to router until it reaches a router which can give the packet to the server. The web-server responds by taking the requested HTML page, chopping it up into small packets of several KiloBytes each, and transmitting these back to your workstation. The packets don't necessarily follow the same path back to your computer. Indeed, each packet may follow a different path, depending on how busy each router is.

The Internet is annoyingly slow at times, but in many cases faster than alternative information systems. An eMail usually gets all the way around the world in less than a minute - compare that to several days for paper mail. The World-Wide-Web provides a connection from your workstation to millions of servers. Businesses can transmit large data files quickly from one building to another. For example, a graphic artist who is preparing an advertisement for a company can send a graphics file to the company, where it can be displayed on a computer screen, and the recipient can quickly return comments via eMail. Businesses can also transfer data-bases (lists of phone numbers and addresses) easily from a computer in one city to another office in a different city - or simply share access to a common server.

Local Area Networks

The World-Wide-Web is a **WAN (Wide Area Network).** Most businesses have a **LAN (Local Area Network)** connecting all the computers in one building. This allows secretaries to share documents. A single data-base can be stored on a server and accessed by all the employees - thus only one copy of a customer's address and phone number need to be stored, rather than every employee having their own paper copy (or digital copy) of all the customers' addresses.

Easy access to personal data can also be a bad thing. Some people don't want their address and phone number to be available to all the employees in a company. Also, universal access to a company’s data can also provide access for the **wrong people**. A secretary could accidentally destroy important data. A criminal could purposely steal or destroy data. The network makes this easier than it would be otherwise, especially if the criminal can gain access to the server without even entering the building.

Components

Computer systems are broken down into **components,** in four categories:

Computer System = Hardware + Software + People + Organization

It may seem strange to consider people to be part of a computer system. However, the **SYSTEM** is more than just the computer. **PCs** (personal computers) rarely do anything alone without a user pressing keys. It is actually **possible** for PCs to accomplish work unattended, but the **system** usually involves a user entering data, giving commands, supervising the computer's activities, and reading the results. You should be aware that **99% of all errors** in computer systems are **caused directly by people** entering incorrect data or issuing incorrect commands. Highly trained, competent users make fewer mistakes and the computer system is more productive than one with a poorly trained, incompetent user. Thus, the quality of the user can significantly influence the functioning of the computer, and thus the user is a very important component in the system.

Following are outlines of the most important hardware, software, people and organization components.

Hardware

Input keyboard, mouse, microphone, scanner, digital camera, video camera, graphic pad, sensors (thermometer, light sensor, magnetic), OCR (optical character recognition), touch screen + stylus

Storage hard-disk, floppy diskette, streamer tape, CD, DVD, USB stick

CPU (Central Processing Unit)
Pentium processor, memory (RAM), graphics card, sound card, network card, parallel port, serial port

Output

Hard-Copy (paper) printer, plotter

Display (screen) CRT (cathode ray tube), LCD (liquid crystal display), VR goggles (virtual reality)

Sound sound card + speakers

Machine Readable radio signals, network signals, data files on disk/tape, punched cards, magnetic ink

Communication
modem, telephone, network cables, infra-red, radio-link, cellular telephone

Software

System Software

OS (Operating System)
Windows, UNIX (Linux), DOS (Disk Operating System)

Utilities
file-manager, encryption, virus-scanner, disk-repair, back-up

Applications

Office Suite
Word-processor
Data-base
Spreadsheet
Graphics / presentations
PIM (Personal Info Management) = appointments, phone numbers, addresses, eMail

Training / Education
tutorials, simulations, training, wizards
computer-aided-learning, on-line testing, distance-learning

AI (Artificial Intelligence)
expert-system, knowledge-base, fuzzy-logic, neural-networks, pattern recognition,
computer-generated art, monitoring, performance analysis, error analysis

Multimedia / Entertainment
hypertext, multimedia, presentations, story-board, virtual-reality, games

Communication
browser, search-engine, eMail, streaming-media

People

Production
engineers, assemblers, programmers, salesmen, installers

Users
employees, customers, home users, hobbyists

Maintenance
repairmen, support (technicians and trainers), system-operators (servers)

Organization

Business Rules (for the company)

Operational Processes (for the business)

IT Rules, Regulations and Guidelines (for software and hardware)

Seurity Policies (about user access)

Hardware

Hardware refers to the physical components of a computer system. You can touch them. They are usually specific-purpose, not flexible. Hardware components can **wear out** or **break**, in which case they must be replaced.

Peripheral devices are extra pieces of hardware which you attach to a computer - printer, mouse, joystick, an extra disk drive, modem.

Ports are the connectors where you attach peripheral devices. Modern PCs have a variety of ports for different purposes. The **plugs** for various ports are different shapes and sizes, to prevent inexperienced users from plugging a cable into the wrong port.

Serial port - Transmits a single **bit** at a time. The modem and mouse connect to the serial port. Some printers could be connected to the serial port, but this is no longer common.

Parallel port - Transmits an entire **byte** (8 bits) at a time. It is thus 8 times faster than the serial port. Printers are usually attached to the parallel port. External disk drives can also be attached here.

SCSI adapter - A less common, more expensive, faster connection for disk drives and other devices requiring very high speeds. Most PCs don't have a SCSI adapter, but this can be added.

USB (Universal Serial Bus) - A new standard allows **any and all** devices to be connected. In this past 10 years, it has pretty well replaced serial and parallel ports.

Platform

There are many different standard **platforms** (basic computer configuration). The two most important issues are the **CPU** (Central Processing Unit = microprocessor) and the **OS** (Operating System). Other issues include the primary input device (not always a keyboard), the primary output device (not always a monitor), and the primary storage device (not always a hard-disk).

Desktop (PC)	***Laptop (Notebook)***	***Palmtop (Pocket)***	***Thin Client (network)***
4 GigaHz CPU GUI 500 GB Hard Disk Full Keyboard Mouse 2-4 GB RAM LCD 1280x1024 General purpose, office applications Cheap and powerful	2 GHz CPU GUI 250 GB Hard Disk Compact Keyboard Mouse pad 1-2 GB RAM LCD 1920x1024 General purpose, office applications Powerful and portable	100 MHz CPU Text-based No Hard Disk Tiny Keyboard (or None) Touch screen, Stylus 250 MB RAM LCD 640x320 Store addresses, phone, appointments Small, portable, long battery life	333 MHz CPU GUI No Hard Disk Full Keyboard Mouse 64 MB RAM Touch-screen (?) Internet, data-base access, eMail Very cheap, safe

The four platforms described above meet various needs of businesses, in a variety of situations.

The **desktop** PC has a large keyboard for efficient word processing, a large monitor to display multiple windows, and a large amount of RAM to run many applications at once. The disk drive can store large amounts of data and many applications.

The **notebook** provides almost the same power as the desktop, but in a single, portable box. The notebook is a lot smaller, so it cannot provide quite as much power. It also needs to run from batteries, so some of the components are made smaller and less powerful (slower CPU) to save energy and increase battery life. Even at that, it generally only runs for about 2 hours from the battery. This is just enough time to do a couple hours work in an airplane during a business trip. When the businessman arrives in the hotel, he/she can plug the notebook into the wall, use a telephone to connect to their office network, and do the same work they would do at the office. The mouse is replaced by a small stick or pad on the keyboard, which can be used with the computer sitting on your lap (hence the name laptop).

Palmtops are not used for word-processing. Many don't even have a keyboard - if they do, the keyboard is very tiny and could not be used by a secretary to type quickly. There is no hard-disk, so applications like Microsoft Word cannot run. The software is permanently stored in ROM (Read Only Memory). Data is stored in RAM, which is constantly on. The tiny screen can display a few lines of text. The main purpose is to carry around a small amount of information - addresses, phone numbers, appointment calendar, and possibly short eMails. It fits in a pocket, can be held in one hand, and thus can be used anywhere. It uses very little electricity, so the batteries will last several days or weeks. The GUI is sacrificed as it would require a much more powerful CPU and more memory, and is not required for the simple tasks of retrieve small amounts of text.

The **Thin Client** attempts to provide only the most necessary facilities for office work at the cheapest possible price. The main difference from a desktop PC is the missing hard-disk - doesn't save that much money - perhaps 500 DM. However, it saves on **maintenance** costs. The Thin Client is always used in a network. All the data and all the applications are stored on a server. In a company with several thousand PCs, it is very expensive to hire lots of technicians to run around installing software and fixing all the problems which occur on individual PCs. By keeping the software installations on a server, all the installation and configuration can be done one time in one place.

Compatibility

Computers need to exchange data with other computers. A palmtop can download addresses and phone numbers from a desktop computer over a connecting cable, but only if the two machines can "talk" to each other. This is called **data compatibility** - they can exchange data. **Software compatibility** means that they can run the same programs. Notebook computers run the same programs as desktop machines, but palmtops require special software which runs in a much smaller memory space, without a hard-disk, and without the same GUI. **Hardware compatibility** means that two machines can use the same peripheral devices. A palmtop might have a printer interface to use the same printer as a desktop machine, but it probably won't have a connector to use the same mouse. The network computer will probably be hardware compatible to the desktop computer, but without support for a hard-disk. Notebook computers a pretty much hardware compatible with desktop machines as far as external peripherals are concerned, but it is not possible to insert the same adapter cards (video card, network card) into the notebook, because the cards are too big. Notebook computers have a special expansion slot called a PCMCIA slot. This allows very small expansion cards to be plugged in, providing a network card, a modem connection, or connection for an external disk drive.

Specification

Every software package specifies the **minimum configuration** required for the program to function properly. Multimedia programs and games may well require a sound card or a very high-resolution monitor. A spreadsheet or other program doing lots of mathematical calculations may need a very fast CPU or a lot of RAM. If printing is an important feature of the application, it may state that a printer is required. In most cases the hard-disk storage requirements of the application will be stated.

Ergonomic Issues

Heavy computer users (several hours per day) can develop **repetitive strain injuries** in their hands, wrists, and other joints. **Carpal tunnel syndrome** is an inflammation of the nerve at the base of the palm - this is caused by doing a lot of typing or mouse work with the wrists held in an awkward position. Too many hours staring at a monitor under poor lighting conditions can cause head-aches, eye strain, and vision problems. Too much sitting can cause a variety of physical problems.

Ergonomics is the study of techniques to produce equipment which minimizes the risk of health problems. Proper chairs promote good posture and reduce the probability of back problems. Wrist supports, proper lighting, better monitors, specially designed keyboards are all examples at ergonomic design.

Other health issues are less clear. Monitors create radiation, but there is little evidence that this actually harms people. Working with computers may cause anxiety and other psychological problems, but there is little clear evidence of this. In any case, many companies are guilty of saving money on hardware while sacrificing their employees comfort and health.

Ethical Issues - Businesses invest increasing amounts of money into computer equipment in the hope of improving productivity and thus increasing profits. Some employees may lose their jobs as their work is taken over by computer systems. Other employees get assigned to new jobs, in many cases using new computer equipment. The employees need and want **training** to succeed in their new job. But again many businesses try to save money by not providing sufficient training. The can produce **job dissatisfaction**, **incompetence, psychological problems,** and a variety of **social problems** among the employees. In some extreme cases untrained employees have accidentally sent eMail to the wrong person, causing embarrassment for several employees.

Environmental Issues - The rapid improvement in speed and storage capacity means that old computer equipment is essentially useless and cannot be resold, even at very low prices. There is no useful way to reuse or recycle this equipment, so it is thrown away. The rapid expansion of the sale and use of computer hardware has not been matched by facilities for disposing of old hardware. Chips, circuit boards and monitors cannot simply be dumped into land-fills - they are a dangerous waste product. Several thousand different chemicals are used in the manufacturing process, and trace elements of these chemicals remain in the equipment. Many factories are located in underdeveloped countries where labor and land are cheap, and where little attention is given to proper disposal of waste products. Considerable environmental damage has been done by hardware factories in some countries. In recent years some companies have begun recycling printer toner cartridges by refilling them and reusing them.

Two remarkable environmental problems caused by computer equipment are paper and electricity. Computer systems were supposed to create "paperless offices" by storing all information electronically rather than on paper. There is some evidence that exactly the opposite has occurred - rather than decreasing the use of paper, computers make it so easy to produce paper that its use in offices has actually increased. And this paper is generally only used for a brief time, and then throwing away, creating a large waste-disposal problem.

It won't surprise anyone that computers use electricity, and that using more computers increases electricity consumption. The surprising fact is that in many cases computers are **never turned off**. Servers certainly cannot be shut down. But many people leave their PCs on all the time because restarting them is time-consuming, and because some work can actually be done by the computer at night, unattended. The increase in energy consumption is thus considerable greater than expected. Fortunately, newer computers provide "stand-by" modes which leave the computer on, but not actually doing any work and using only a tiny fraction of the energy normally required.

Input Devices

Keyboards are good for typing. **Voice recognition** has the potential to replace the keyboard with a microphone, but still does not work very well, requires very powerful computers to work at all, requires a significant amount of training and practice before it can be used efficiently, and does not work well in noisy, crowded offices. But voice recognition does allow data input in situations where the hands are busy doing something else, or the user cannot be seated at a keyboard.

Digital cameras are easy to use and can transfer directly into the computer. The pictures can then be printed and **scanned** into the computer using a scanner. This can actually produce a better quality picture than a digital camera, but the overall process takes longer (including developing the film). A scanner is certainly the preferred method for inputting a pre-existing picture (e.g. from a magazine or book). A digital camera could be used, but the resulting quality would be extremely poor. Scanners have a much higher resolution (more dots per inch) than digital cameras.

Scanners can be used to input documents as well as pictures. **OCR** (Optical Character Recognition) software can transform a scanned document into a normal word-processing document that can be saved and edited. Scanners are not generally used for **handwriting recognition.**

Touch-sensitive screens are used in palm-top computers. This saves space as no keyboard is required. It is not a very efficient way to input text, but is adquate for limited data input.

Output Devices

Printers output information onto paper. In the past printers existed which could only print text, not graphics. These are no longer common. **Laser** printers are very fast (10 pages per minute or more) and print very high quality images, both text and graphics. Most printers still only print black ink. Color images are transformed to **grey scale** images, where different shades of gray represent the colors. Color laser printers are very expensive, but **color ink-jet** printers are quite cheap and produce acceptable quality, although quite slowly (less than one page per minute).

Plotters are used by engineers to produce extremely accurate drawings. A plotter pushes a real pen around on the page, drawing smooth curves with no "jaggies". Plotters are only sensible for producing line-drawings and graphs, not for text or photographs.

CRT were the most common output device for personal computers until about 2005. They provide resolutions up to 100 dots per inch in 16 million colors. **LCD** (Liquid Crystal Display) screens are used in notebook computers because they use a lot less power than a CRT and are a lot smaller. LCDs have lots of advantages over CRTs: use less power, perfectly flat image, no distortions, no flickering, virtually no radiation. LCDs have become so inexpensive that they have pretty well replaced CRTs.

VR (Virtual Reality) goggles are glasses which have two different images in the two lenses. The images are just slightly different, to fool your eyes into believing they see a true 3-dimensional image. This illusion cannot be provided by a CRT or an LCD. But VR goggles are very expensive and provide only a very poor quality image. They are only sensible for games or simulations requiring a 3-D illusion.

Special Needs

Some people have difficulty using computers due to physical handicaps. Indeed, left handed people find the mouse and the keyboard more difficult to use than right handed people. It is possible to reprogram the mouse to swap the left and right mouse keys, but many mouses are curved and only fit properly in the right hand. Poor vision is also a very common problem - people with severe sight problems can be helped by using large fonts and reducing the screen resolution. Special programs are available which allow the user to magnify part of the screen. People who cannot use their hands at all can use speech-recognition systems for input. Blind users can install special software which reads documents to them, producing a synthetic voice through speakers. The same 6-key keyboards used by blind people to type Braille can be used for input instead of a normal keyboard. Fully paralyzed users have been provided with computer systems which can be used with a mouth wand touching a touch-sensitive screen. Recent research has produced sensors which can respond to eye-movements, brainwaves, and tiny muscle movements.

Storage Devices

Hard-Disks are the most common storage device for PCs. They provide 100 - 1000 GigaBytes of storage at a very low price and with very fast access (10-100 Megabytes per second). **Floppy diskettes** are no longer commonly used. They store only 1.44 MegaBytes (0.00144 GigaBytes) - this is enough to store lots of simple text documents, but only a few graphics files. Floppy diskettes can be used to move data from one computer to another. This is not so useful if a **network** is available. It is now possible to **eMail** files from one computer to another which is not even in the same network. And floppy diskettes are no longer useful for **back-ups**, as you would need 700 diskettes (and about 10 hours) to backup 1 GigaByte from a hard disk.

USB sticks have replaced floppy diskettes. They provide 1 GB or more of storage, and don't require a special drive to be installed in the computer – just a USB port.

CD-ROMs contain 700 MegaBytes of data. They are very cheap to mass-produce, costing only about 0.20 EU each. CD writers can be used to copy 700 MB from a hard-disk to a CD. This is now a popular method for transferring data between computers and making backups. A blank CD is very cheap, so this is a much better method than using floppy diskettes, even if only 100 MB is being transferred.

DVDs store 4 GB – 8 GB each, so considerably more than a CD. They cost about twice as much, but store 5-10 times as much data.

Tape drives can store very large quantities of data cheaply - 50 GB on a tape costing 20 EU. Tapes are a popular back-up device for network servers, as the process is cheap and very fast - much faster than writing a CD.

Cloud Storage - on web-servers

Software

System software is programs that control the hardware directly, running in the **background** without the user being involved - for example the **Operating System**, printer **drivers**, and network **utilities**.

Application software refers to programs which the user runs to perform specific tasks - word-processors, drawing programs, browsers, games, etc. Software is not a physical object, like hardware. Software can be **downloaded** and **copied**. It can be **configured** (changed), **patched** (fixed), and **upgraded** (changed to a newer version).

Licensing

A new PC usually has an operating system already installed on the hard-disk. That doesn't mean the OS is **freeware.** The PC manufacturer actually pays for an **OEM license (OEM = Original Equipment Manufacturer)**, which allows them to copy the OS onto every machine they sell. The **end-user** can also buy the OS directly. If you don't like Windows 7, you can **upgrade** to Windows 8 by purchasing the software, including an **End-User-License.**

Some software is **freeware** - no license is required to copy it or use it, so it costs nothing. **Linux** is an OS which is distributed as freeware. This began as a project by a Finnish university student (Linus Torwald), who distributed his **kernel** (central program) so that other **developers** could add new **modules**. The resulting OS does not belong to anyone, so it is available to everyone - it is **open source.** Some freeware is still under **copyright** to a specific programmer or company - many software companies give away freeware as a marketing tool to attract customers to their **commercial products.** Copyrighted freeware cannot be sold by some other company, or combined with another company's software.

Many **freeware** programs are the result of **custom software** that was written for a specific company or individuals to use, but the programmer was not interested in **marketing** it further, so they **released it into the public domain**. This means they do not retain a copyright on the software, so anyone may do anything they wish with the program. However, no one else can "claim" a copyright - nobody can "own" the software.

Commercial software is the most commonly used type of software - products like Windows, Microsoft Office, and games. Commercial software is distributed as **executable** files. This means the end user cannot change the programs, except by changing a few settings.

Open Source software is freeware that allows anyone to see – and CHANGE – the original source code. Open Office is an open-source alternative to Microsoft Office. It's possible for any programmer to take the source code for Open Office and change it to do something else or to do it differently. Of course, most end-users are not programmers, so they wouldn't bother trying to understand or improve the millions of lines of source code.

Hardware and software products have 3 basic cost components: **R&D (research and development), production (manufacturing),** and **distribution (shipping, advertising, and sales).** The big difference between hardware and software is that hardware has a fairly high production cost - this may well be 50% or more of the end cost to the user. Software has virtually no production cost at all - a CD costs about 2 DM to manufacture, and the box and the manual only add a few marks to the cost. Distribution costs for software are less than hardware - software doesn't weigh very much and cannot be damaged in shipping. Thus most of the price you pay for software pays for development (programmers' salaries) and profits for the company. No wonder that Microsoft is so profitable.

Commercial software packages contain a **license agreement**, which usually says the user can only use the software on one computer - it is not legal to copy one piece of software onto many computers. Some software contains **copy protection** code to prevent illegal copying, but this often causes installation and compatibility problems for users, so it is no longer common. **Pirates** can copy several very expensive programs onto a single CD for a couple marks and sell it illegally, making an easy profit from someone else's work. The legal penalties for software piracy include high fines and several years in jail.

Installation and Compatibility

Software packages specify the **minimal hardware requirements** for the software to run successfully. A speech-recognition program may require a very fast CPU (300 MHz), a lot of RAM (64 MB), and a specific sound card. Most commercial packages require a CD-ROM drive for installation, as the software is distributed on a CD.

Installation instructions should be included in the software package. Users are supposed to read and follow the installation instructions, but many don't. Why? Either they are too lazy (often the case) or the instructions are poorly written, incomplete, unclear, incorrect, or difficult to understand (very often the case). To solve this problem, sophisticated **installation wizards** have been added to many programs to help the users with the installation and prevent problems. If the hard-disk does not have enough free space to complete the installation, a good wizard won't allow the installation to start. Without a wizard, the user is expected to check for enough free space before starting the installation. If they don't check, they can end up with a full hard-disk and a non-functional installation of the software. A full hard-disk might prevent Windows from restarting, the user could end up with a totally dead computer! A wizard may also detect the presence of **incompatible software** in the system, which could prevent the software from functioning correctly, and alert the user to this problem.

An increasingly important aspect of software is **user support** - e.g. **telephone help-lines** where the user can call and get help from a technician. Commercial software packages contain a **registration card** which the user must send to the company - this entitles the user to call the help-line, or send eMail and get answers about problems. It also entitles the user to **cheap (or free) updates.** Most software is outdated and unusable after a few years. Software companies are continually improving their products, releasing **new versions** every couple years. Registered users get cheaper upgrades than other people, who must pay the full price for the new version.

Integrated software packages (e.g. Microsoft Office) become more and more popular with both users and developers, because one installation process installs several components, so installation and configuration take up less time. Furthermore, these applications have been **designed for compatibility** with each other, which means the users have fewer problems and require less support.

User Interface

The basic user interface is determined by the Operating System. Windows is a **GUI (Graphical User Interface).** This is a **WIMP system** (Windows, Icons, Menues, Pointer). DOS was a simpler system - **text-based command-line interface (CLI)**. In DOS, there was no mouse and no pictures. There was only text, and you typed commands, using words which were not very **user-friendly** .

Task	***Windows (GUI)***	***DOS (text-based command-line interface)***
Start a program	Double-click on the icon	**cd C:\MSOFFICE\WINWORD** **WINWORD.EXE**
Find a file	Open My Computer Double click on C: Double click on a folder, ...	**C:** **DIR** **CD MyDocs** **DIR** **CD c:\OldDocs** **DIR** **......**
Copy a file to diskette	Drag and Drop from one folder into another	**COPY C:\MyDocs\Essay.Doc A:**

Historical Development

Command-line interfaces like MS-DOS were popular from 1970 through 1985. The hardware at that time was not sophisticated enough to support a GUI. Before 1980, monitors only displayed text. The mouse was invented in the late 1970's, but was not incorporated into an operating system until the early 1980's. GUI operating systems also require a lot of memory – every window is displayed as a graphics image, taking up a lot more memory than storing text. Command-line interfaces are simple, only require text-mode, and can run on simple CPUs with little RAM. The first commercial GUI was created for the Apple Macintosh around 1985. Microsoft Windows came a few years later.

Word-processors and spreadsheets became quite popular in the early 1980s, before GUIs were available. The availability of GUIs on personal computers enabled word-processors to support **DTP** (Desk Top Publishing) features, such as fonts and pictures. Spreadsheets originally became popular for doing bookkeeping and planning budgets - these tasks only require numbers to be typed in text-mode and calculations to be performed. The extension of spreadsheets to create graphs was enabled by GUI operating systems. Graphic design and multimedia presentations were possible under command-line operating systems with graphics monitors, but only became popular with the advent of GUIs.

On-line help (the Help menu) has become standard, and been augmented by **wizards** and **agents**, which help you with tasks. Many programs don't bother with paper manuals anymore - they are heavy to ship, expensive to produce, and readers don't use them anyway. **Hypertext** help utilities which include links to relevant topics and searches are much faster and cheaper to produce. **Context-sensitive** help responds automatically with appropriate information related to the task you are currently doing. For example, if you are running the grammar check you will get a different help-screen than if you are doing a mail-merge.

Extensions and Future Developments

Some programs extend the user-interface for special purposes. A good graphics program may support a **graphics tablet** to allow the user to draw more efficiently. Games programs may support **virtual reality** devices such as joysticks, 3-D goggles, and gloves.

Speech recognition is becoming quite popular and will probably become a standard part of future Operating Systems. Now, it is an **add-on**, which only functions with some of the application software. For example, you probably cannot talk to your favorite game program.

Windows does not require users to learn the strange **syntax** of the DOS command-line interface. However, users do have to worry about **folder-names**, **paths, and file-types.** In the future, **natural language** interfaces will not only allow you to talk to the computer, but they may **understand** more flexible use of the language. Rather than saving your document on **drive C:** in the **MyDocuments Folder** as **Essay2.doc**, you may be able to tell the computer:
"This is the second copy of my essay. Save it on the hard-disk in the usual place." Try typing that command into a computer!

Most of the graphics you see on current PCs are poor quality photographs or cartoon animations. We can expect **real-time video** displays to become more common in the future. They are possible now, but they require too much storage space - several GigaBytes to store a movie - and can only be displayed by very fast CPUs. DVD (Digital Versatile Disk) drives can store 5 GB on a single disk, which makes true multimedia a more realistic possibility. Imagine a future program where the little wizard in the corner (clippy) is a **real motion picture** of a **real person** who is **talking** to you and **listening** to your questions! That may become just as common-place in 10 years as Windows is now. Twenty years ago, people would have been amazed by a Windows 98 machine running a web browser. Imagine 10 years from now where video-conferencing is as common and cheap and easy as eMail is now - hence replacing eMail!

The goal in all this development, from 20 years ago through 20 years into the future, is to make the **computer-human interface** as **easy** and **natural** as possible - e.g. **user-friendly.** The interface is always limited by the capabilities of the hardware, and everyone wants the hardware to be as cheap as possible. So the interface is not always designed to fit the **human expectations.**

Human beings are very flexible and adaptable - machines are not. So human beings tend to adapt their expectations and wishes to the clumsy capabilities of the machine. But the interface improves - becomes more **human oriented** and **user friendly** as hardware becomes more powerful and cheaper. If speech recognition really works, nobody will need to learn to use a keyboard. Why draw with a mouse? You should be using something that looks, feels, and works like a pencil. And why push the mouse around on the table? Why not draw directly on the screen, like you do with a piece of paper? All this is possible now, but the hardware to do it is still quite expensive.

How fast does hardware become better and cheaper? **Moore's Law** was proposed by the founder of Intel 50 years ago. He predicted that **the density of transistors in chips would double every 18-24 months,** hence microprocessors would double in power every 18 months. Amazingly his prediction has been very close to true for 40 years. At the same time, the price of a specific piece of hardware tends to be cut in half every 18 months. Thus a 2 TeraByte hard-disk which costs 100 EU now will cost only 50 EU 18 months from now. Consider a 2,000 EU PC which uses speech recognition for all tasks (no typing), allows the user to point and draw directly on the screen using a **stylus,** has all the appropriate software installed including Office, Web Browser, high quality drawing program, has a BluRay drive and can show movies, etc. Few people would be willing to pay 2,000 EU for this machine, even though it is very **powerful** and **user-friendly**. However, 3 years from now the same machine will have its price cut in half and cut in half again, and will only cost 500 EU - a very typical price for a new PC. Thus we could expect this kind of PC to be the "standard" desktop configuration in 3 years.

Operating Systems and Utilities

Important **functions** of an operating system:

Managing a User Interface
Loading and Running Programs
Controlling Hardware
Coordinating Multi-tasking
Managing Storage Devices and Files
Various Utilities

User Interface - WIMP (Windows, Icons, Menues, Pointers)

Loading Programs - When you double-click an icon to start a program, Windows must find the program on the hard-disk, copy the program into the memory (RAM), allocate any extra memory the program might need, and start the program running at the correct place.

Controlling Hardware - Individual applications do not actually communicate directly with the hardware. Microsoft-Word does not actually transmit information to the printer - instead it tells the OS to send the document to the printer and leaves the rest of the work to the OS. A graphics program that is inputting a picture from the scanner is actually not **controlling** the scanner directly - the OS controls the scanner, and passes the data along to the graphics program.

Multi-tasking - Windows is a **multi-tasking** operating system. This means several programs can be running at the same time. You can start a virus-scanner running and then start downloading music files from the Internet. While the download and the virus-scan are running in the **background,** you can open a document in a word-processor, copy part of it into your eMail system and send it to a friend, check your appointment calendar, and play a game while you are waiting for a visitor to arrive. The OS must coordinate the functions of all these programs and prevent them from interfering with each other. If the virus-scanner is currently checking the same document which you have open in the word-processor, the OS will not allow you to save the document until the virus scanner has finished checking it. If two different applications attempt to use the printer at the same time, the OS must store one of the print-jobs temporarily in a **buffer** on the hard-disk, and send it to the printer when the other job is finished.

Virtual Memory - If you start too many applications, the memory (RAM) will be full. For example, Internet Explorer 4 needs about 8 MB of RAM when it is running, Word also needs about 8 MB. So if you start both of these, and your computer only has 16 MB of RAM, then the memory is full. To allow other programs to run, Windows can **swap** part of one program temporarily onto the hard-disk - perhaps the spell-check module if not currently in use. This frees up some RAM so another program can start running. By using a large **swap-file** on the hard-disk, more programs can run than actually fit into the RAM. Thus is seems like your computer has more memory - this is called **virtual memory.**

File-Management - The hard-disk is the major storage device, containing both applications and data-files. The hard-disk is divided up into **sectors** - each sector contains 512 bytes (half a KiloByte). Files must be stored in specific sectors, and the operating system much keep track of which file is stored in which sector. Each disk has a **file allocation table** which to keep track of this. The user doesn't have to worry about this - they just give the file a name and let the OS find some empty sectors to put it in. The user sees **folders** and **file-names**, but this structure does not tell whether your files are stored near the middle of the disk or near the outer edge - only the OS needs to worry about this.
The disk also has a **directory** area where the names of files, their size and date are stored.

Utilities - four common ones are **defragmentation, encryption, backup, and virus-scanner.**

Defragmentation - Most files require lots of sectors for storage. A typical document might be 100 KB, so it needs 200 sectors for storage. The OS will try to select 200 sectors directly next to one another (contiguous) to store the file in. But if the user is constantly installing new programs and erasing them and installing other programs, the disk drive can become **fragmented.** This means that the empty sectors are scattered all over the disk-drive, rather than being together in one area. When a big file is saved, it must be stored in lots of disconnected pieces. Then when the file is loaded, the read-write head must jump around on the disk. This is much slower than loading the file from a continuous line of sectors. After a few months or so of use, a hard-disk usually becomes fragmented and seems to be **slower**. It is not physically slower, but it takes more time to load the fragmented files. A **defragmentation utility** can move files around, swapping sectors between files to end up with all the files in contiguous (continuous) storage locations, and all the free space in one large block. Running the defragmentation program frequently has two advantages: (1) it keeps the hard-disk functioning efficiently and thus saves time; (2) defragmentation goes much faster if the disk is only slightly fragmented, so the defrag utility also runs faster.

Backups

Another utility program is the **backup** utility. This makes extra copies of important files on a tape, a second hard-disk, or a removable drive such as a CD-writer or a floppy diskette. A **complete backup** is a copy of the entire hard-disk. This usually can only be made on a tape, as other **removable media** do not hold enough data.

A **partial backup** only backs up some of the files. This might be your data files, or only one single application, or only the files which have been changed since the last backup. If the last backup was made on 15 March, then a week later a backup would only copy the files with dates after 15 March. A good backup utility can be **configured** to run automatically each day or each week, and to select the files which need to be backed up.

Some **servers** use an extra hard-disk which is always identical to the main hard-disk - every file is saved on both drives. This is called **mirroring.** Some servers are **mission critical** for an organization - business cannot continue if the server breaks down. With a mirrored hard-disk, the backup drive is always immediately ready for use, so that there is very little **down-time** if the main hard-drive fails. Restoring an entire hard-disk in small pieces from a bunch of different tapes could take quite a long time - restoring a server this way could well take an entire day.

Backups protect against lost data - files that are accidentally erased or have been changed incorrectly and contain incorrect data - and against equipment failure such as a broken hard-disk which can no longer be used. The major **cost** of lost data or broken equipment is the time required to **restore** the missing data (copy it back onto the hard-disk) or to get replacement equipment up and running. If a business stops for an hour due to a broken server, it might waste hundreds of man-hours and cost the company thousands or even millions of dollars in lost productivity or lost business. If the data **cannot be restored**, then the time required is infinite, which could mean the cost is incredibly high. Some large businesses have declared bankruptcy as a result of lost data files.

Tapes are a popular backup **media**, costing about 40 EU to store 160 GB, or 0.25 EU per GB, although the tape drives are quite expensive - on the order of 500-1000 EU. Tapes are also reasonably fast. A good tape drive can backup 1 GB in about 1.5 minutes. Normally a **sequence** of backup tapes are used - one for Monday, one for Tuesday, etc. This way if a problem occurs on Monday but is first discovered on Friday, the data from Monday is still available on a backup tape. Disk drives are faster than tapes and cost about the same, but cannot be easily exchanged. CD writers and DVD writers are now a popular backup option, especially for personal use, as they are quite cheap - 0.20 EU for 700 MB, so about .30 EU per GB. However, DVD don't hold nearly as much data as a tape, so they are only useful for relatively small jobs.

Malicious Software

Virus - a **program** which can make copies of itself and store them inside other programs or data files. The virus makes these copies automatically and secretly, so the user doesn't know it is happening. When the newly "infected" software runs, the virus runs automatically and makes more copies of itself. In a network, a virus spreads by copying itself into files on a server or onto other workstations. If no network is available, viruses spread because users copy files onto diskettes and carry them to other computers. Some viruses have actually been distributed on original diskettes and CDs containing commercial programs - this means the software producer had an infection in their network.

Trojan Horse - slightly different - it does not copy itself automatically or infect other files. It relies on the user to spread it around. A typical Trojan Horse program is a game containing hidden commands that do damage or annoy the user. The user happily downloads the "free" game onto his/her PC, runs it, enjoys it, and only notices a week later that some of his important files have been erased (damage), or that the computer makes funny noises (annoyance).

Logic Bomb - pretty much the same as a Trojan Horse, but this refers to malicious code which is inserted into software with no intention of it spreading to other systems. An unhappy employee might program a logic bomb into a company's database to get even with the company for not promoting him/her.

Worm - a malicious program specifically designed to copy itself over a network. A famous Internet worm was released in 1988 by a University student - he claimed it was an experiment which got out of control. It used eMail to send itself to lots of computers. In each computer it made thousands of copies of itself, slowing down the system until it no longer functioned properly. This caused 6000 Internet servers to be shut down. He was the first person in the United States to be convicted under the newly created Computer Fraud and Abuse Act.

Payload - this is the bad part of the program that does damage or causes annoyance. Some famous examples:

Yankee Doodle Virus - a virus that plays the song "Yankee Doodle" every day at 17:00
Stoned Virus - slows the computer down and then prints "This computer is Stoned"
Anti-CMOS - a very dangerous which erases the BIOS setup, after which the computer will no longer start up

If a virus does too much damage too fast, it will not succeed in spreading very well, as it will destroy the very files and hardware which were supposed to spread it further.

Executable File Virus - the virus infects executable files (programs). It becomes active when an infected program is loaded and executed.

Boot Viruses - the virus resides in the **boot sector** of a diskette. Infection occurs when the computer is booted (started) with a diskette in the disk drive. It may then infect the boot sector of the hard-disk, so that the virus is activated every time the computer starts.

Macro Virus - now that Microsoft Word documents can contain **macros**, new viruses have been developed which run as macro code, and thus spread around in documents rather than in program files. Previously, **virus scanners** only needed to check program files. Now they must also check documents. Even worse, documents are copied and transmitted (eMail) more commonly than programs, so macro viruses can spread more easily than program viruses.
Infection occurs when the document is opened, at which time the macro executes automatically.

Protection

Backups are not the best protection - they are the ONLY protection!

A **virus scanner** might be able to detect viruses and destroy them, but it will not restore data after it has been destroyed. The best protection against data-loss is regular backups. Nevertheless, it is still wise to use a **virus scanner** regularly, to detect viruses **before** they do any damage. Good scanners are active all the time, and actually **prevent infection** by detecting infected files as they are being copied onto the computer, before they are executed. A **disinfection** utility can remove the virus from infected files and boot sectors, thus removing the virus without destroying the infected files.

How big is the threat? Viruses really do exist - there are many thousands of different viruses in existence. Most of them are harmless jokes, which make funny noises or display silly messages. But some viruses are very destructive.

Direct costs - Virus infection is commonly costly in three ways: (1) destroyed data and malfunctioning computers must be repaired or replaced, and this costs money (or time); (2) before the infection is discovered, malfunctions waste users' time; (3) preventative measures such as anti-virus software can be both expensive and time-consuming to use.

Indirect costs - Consider the PC's in a network in a large bank. Possible consequences of a virus infection include:

- If a virus erases data, that data might represent money for the bank's customers, who might actually lose money as a result.

- A bank employee might not complete an important contract on time due to a malfunctioning computer. The bank loses a large sum of money, and the employee loses his/her job.

- If a virus infection comes from a consultant, the bank may decide not to do business with them any more. Thus, a virus can damage someone's reputation. According to journalist Steve Bass, "I had my system infected by a nasty virus on a CD-ROM distributed by a public relations firm. (The group's now representing Russian bankers peddling the ruble.)"

The cost of virus-related damages is difficult to estimate. Many companies willingly spend many thousands of dollars on anti-virus software and precautions, as insurance against unpredictable, multi-million dollar losses from virus infections.

There are **many other threats** to computers and data files, and the other threats are much more likely to cause damage. What is the biggest threat? **People!**

Security

Threats

Accidents and Mistakes - Clumsy, untrained, incompetent, careless users accidentally erase files or make incorrect changes to data. People install the wrong software, or install it incorrectly - it functions incorrectly or interferes with other software.

Unauthorized Access - People read information which they are not supposed to read - employees reading other employees eMail and using the information to get a promotion or steal customers, one company stealing information from another company, reading private information and using it to embarrass or threaten someone.

Sabotage - Malicious employees or outsiders purposely destroy data, for example customer information or financial records, to damage the reputation or productivity of the company.

Theft - Stealing money or equipment by changing computer records, sending money to the wrong account, delivering equipment to the wrong place, erasing records of payment due or marking items as paid when they were not.

Protection

Backups guarantee that destroyed data can be restored if necessary.

Encryption software scrambles files using a secret code and password. Only someone who knows the **encryption key** (password) can **decrypt** the file. This prevents people from reading files and accessing forbidden information.

Limiting Physical Access - locked doors, security guards, alarm systems requiring access codes to enter a door, video surveillance cameras, ID cards which must be worn and "swiped" through readers.

Passwords prevent unauthorized access. In most businesses, only a few people are permitted unlimited access to all the computer systems and data. Secretaries don't need to access the financial records. Bookkeepers don't need to access the customer addresses used by the salesmen. The salesmen don't need to read the president's eMail.

Passwords Security - Passwords are easy to implement in computer systems, but sometimes annoying and difficult to use. Users forget passwords, so some people write the password down on a piece of paper and leave it in the desk drawer, or even worse post it on a sticky label on the computer. Passwords should never be written down anywhere. They should never be "shared" with another employee, even someone you trust, because they might do something silly like writing it down or repeating it to someone else or leaving the computer switched on and logged in to your account.

Good Passwords - A good password is :

Easy to remember (related to something you won't forget)
Hard to guess
Avoid personal information - not your wife's name or your birthday or the name of a hobby
Changed often - someone might already know your password and you are unaware of that.
For safety, passwords should be changed often, like once a week.
Never shared - every user needs their own password - group passwords are a disaster waiting to happen.
Not a single word
Not too short - certainly more than 3 letters **-** a brute force attack can easily crack a three letter code
Avoid commonly used passwords - the most popular password in the world is "love".
Another popular one is "password".

An intruder could use a **brute-force** attack to guess a password - this is a computer program which tries every word in the dictionary, or every combination of 1, 2, and 3 letters (there are only 18,000 combinations of 3 letters). A computer to do this reasonably quickly. This only works for single-word passwords. Also, many computer systems will **lock** an account if too many incorrect passwords are typed in - for example, after the third unsuccessful try the account is locked, and cannot be used even with a password. This is not a very popular thing with users - if they change their password often, they may type the old password by mistake, then misspell the new password, and then they only have one more chance before the account is locked. This is also not popular with **system administrators** who get lots of phone calls from angry employees who need their account unlocked.

Some companies forbid employees from changing their own passwords - they prefer to assign passwords, to prevent the employees from choosing a common single word or a very short password. Unfortunately this usually results in passwords which are difficult to remember, which increases the probability of the employee writing it down or forgetting it and ending up with a locked account.

Creating "Good" Passwords - some common methods:

Mixing Words - Choose two words and mix the letters together, like this:

white dove ----> wdhoivtee

This makes an unguessable password which is easy to remember, but difficult to type.

Acronym - Write a sentence, and take the first letters of the words.

I Love Baseball And Volleyball ---> ILBAV

This a little more difficult to remember than mixed words, but easier to type.

Follow a "secret rule"

Take a word and swap the first and last letters: Doggy ---> yoggD
Follow each letter by a digit, in order: D1o2g3g4y5

This sort of thing works better if each person has their own secret rule.

Password Lists - A server must contain a list of all the passwords - this is a favorite target for hackers. To prevent an intruder from getting the entire list of all the passwords for an entire company, the password lists are saved in **encrypted** (secret code) format. However, many operating systems use "standard" encryption algorithms, which have "standard" methods for decrypting them. Password lists are a difficult security problem.

Security Levels - Many operating systems accord different security levels to different users, and only a very few users have **administrator** status - access to all the files in the system. This helps to protect critical files such as password files.

Bugs

A bug is an **unexpected error** in a computer program or system, which causes the program to malfunction Programs are quite complex, and most programs are so large that no person actually understands the entire program in detail. There is always some little part of the program that was written incorrectly or poorly designed and has escaped the attention of the developers. Bugs are an unavoidable consequence of powerful, sophisticated, complex software.

When a bug is discovered, the developers may do one of three things:

Fix it - if they have time
Ignore it - because it is not very important or dangerous
Document it - write it down and warn the users of the problem

Many bugs are handled by the last method, as they exist in software which is already installed and being used, so it is basically impossible fix it. Unfortunately, documenting the bug means that **hackers** also know it exists. Hackers exploit known bugs to gain unauthorized access to servers. One example is the NT screen-saver bug. Whenever the screen saver starts, it first runs in **supervisor** mode, meaning that it has total access to the entire system, regardless of the user's security level. It immediately interrogates the operating system to find out the correct security level. However, hackers have written screen savers which don't properly ask for the user's security level, and thus remain in supervisor mode. Once the screen-saver starts, the hacker has access to the entire system, including password lists and critical software.

Spell-Check

Most Word-Processors now contain **spell-check** features, and some even contain **grammar checkers** which attempt to identify grammar errors.

Spell checkers are use a very simple **algorithm** (process) - each word is checked against a **dictionary** (list of words). If the word is in the dictionary, the spell-checker assumes it is okay. If the word is not there, it signals an error, and produces a list of similar words which might be the actual correct spelling. The spell check will not identify any errors in the sentence below - the words are incorrect, but they all exist in the dictionary:

A spell Czech mod jewel buy its elf can miss sum air ores.

A **grammar checker** will detect some of the errors in the sentence above, but not all of them.

A spell checker may also identify a **correct word** as being **incorrect**. This happens with proper names - Word's spell-check suggests that Mulkey should be changed to Milky.

Some common errors, which a spell-check **does detect** are:

Double letters - the user accidentally pressed a key tooo manny tiimes

Missing letters - the usr has mised out a few leters

Spelling mistakes - some peeple just dont spel veri whell.

Spell checkers are not perfect, but they are useful even if they only detect 80% of the errors. Many secretaries don't like them, because they are not perfect and they seem slow. They are especially annoying when used on documents containing lots of names, or lots of numbers, or foreign words. The spell check keeps asking about words which are correct, and this is annoying. However, it is still faster and more effective to use the spell check than to rely on proof-reading alone.

Thesaurus - A thesaurus is a list of **synonyms** (e.g. "fast" and "speedy" have similar meanings) - this can help the use select a **better** word than the one which immediately comes to mind. It can also help the user to **vary** their language, so the text doesn't get boring.

Ethical Issues - A **student** can use sophisticated spell-check and grammar-check tools to **improve** their essays. Is this **fair and ethical**? Some teachers don't believe so - they want to know what the actual quality of the student's English is, without help from the computer. However, these tools are here to stay, and no one seriously proposes that using these tools constitutes cheating. Students should not **trust** all the suggestions made by a spell-check or grammar-check. A computer's command of English is definitely inferior to most people's - even small children speak better English than a computer program.

Optical Character Recognition - OCR

You can use a **scanner** to scan a page of **text**. The page is save as a **picture(graphic image)**. Then an **OCR program** examines the picture. It must **recognize** individual letters. This can be difficult if the printing is in a strange font, or smeared, or too light, or on a colored background, or if the page is not perfectly straight. Current OCR programs are quite good and can compensate for some problems. Nevertheless, even the best are only **99% accurate** - that means that one letter out of a hundred will not be recognized correctly - there will be one mistake in every second line of text. Good OCR programs run a **spell-check** afterward, to try to find the errors. The resulting file is saved as a normal WP document, and can then be edited further. It is **not normal** for a word-processing program to contain an OCR utility.
The OCR software is a separate program.

Graphics

Computers display pictures using **pixels -** little dots. Simple VGA monitors (Video Graphics Array) have a **grid** of pixels 800 across, 600 high - a **resolution** of 800x600. Higher resolutions are available - this means that the pixels are smaller and closer together, displaying finer details.

In a **graphics image** (picture), each pixel has a **color** that is represented by a **number** in the computer's memory.
The **color depth (palette)** is the assortment of colors that are available. Simpler pictures only need 16 colors, more complex pictures use 256 colors, and very realistic pictures use 16 million colors **(true-color).**

Typical **Resolution**:	640 x 480	800 x 600	1024 x 768
Typical **Color Depth**:	16 (4-bit)	256 (8-bit)	16 million (24-bit)

One **byte** stores **8 bits**. So 8-bit color requires exactly one byte to store each pixel. A single byte can store 2^8 = 256 different numbers (0 to 255). Each number represents a different color - this can be any collection of colors. If the picture is a human face, there will be many different shades of pink and flesh tones. If the picture is in the forest, there will be many shades of green, blue, and brown - red might not be included at all.

A **scanner** must divide a picture into pixels. If a 256-color palette is used, the scanner must choose a color (number) for each pixel - it uses the palette and chooses the color which is the **closest match** to the color in the picture. To get better results, a bigger palette is used, so there is a larger set of color to choose from, with many more shades of each color available. With 16 million colors, there is a very large range of choices, so the resulting picture is much more realistic. 16-million colors is called true-color because the human eye cannot actually distinguish more colors, so recording more colors is pointless.

Using more colors (16 million) and a high-resolution (1024x768) produces more realistic pictures than lower resolutions and a smaller palette. Why not always use high resolution and a large palette? Two reasons:
(1) **compatibility -** some computers have cheaper monitors and can only display lower resolutions and smaller palettes; (2) **storage size (memory)** - higher resolutions and larger palettes mean it takes more space to store the picture.
Especially in the Internet, both compatibility and storage size are very important. The pictures should display properly on many different kinds of machines (some with cheaper graphics equipment) and the pictures must be transmitted - bigger pictures take longer to transmit.

Storage Example - Compare a 256-color 640x480 image to a 16-million-color 1024x768 image.

256 colors (8-bit) = 1 Byte per pixel
640x480 = 307,200 pixels = 300 KiloBytes

16-million (24-bit) color = 3 Bytes per pixel
1024x768 = 786,432 pixels
* 3 bytes/pixel = 2,359,296 = 2,300 KiloBytes
= 2.3 MegaBytes

So the high-quality image requires 8 times as much storage space as the lower quality image - in the Internet, this will require 8 times longer to transmit.

Graphics File Formats
The standard file-format for storing a **bit-mapped** image is .BMP . The sizes calculated above are for .BMP files.
Even the low-quality image (300 KB) is too big for a Web-page. With a fast modem, this requires 40 seconds to load.
Web-sites use **compression** software make the graphics files software, transmit the smaller files, then the PC decompresses the file and displays it. Four common compressed formats are:

.GIF = Graphics Interchange Format, invented and copyrighted by CompuServe
.JPEG = Joint Photographic Experts Group, a public-domain format developed for the Internet
.PCX = Invented by the Paint-Shop-Pro developers
.TIF = Tagged Image Format - developed by a group of software manufacturers

Robotics, Artificial Intelligence (AI), Expert Systems

Asimov's Laws of Robotics
Isaac Asimov wrote a series of science fiction stories about "intelligent" robots. To prevent the robots from being dangerous, the manufacturer installed 3 laws in their brains:

1. A robot may not injure a human being or, through inaction, allow a human being to come to harm.

2. A robot must obey the orders given it by human beings except where that would conflict with the First Law.

3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.

Although these laws exist only in science fiction, they are a good description of proper robot **ethics**.

Turing Test
Alan Turing was a mathematician who directed the Colossus project in England - a computer that was built to decipher the German army's Enigma code during World War II. The success of this project was instrumental to the Allied victory. Turing went on to do more work with computers. In 1950, Turing studied the possibility of computers "thinking" the way that human beings think. At that time most computer scientists debated whether a computer could be designed to function the same way that the human brain functions. Turing proposed that it did not matter whether the computer functioned like a brain - he said it only needed to accomplish the **same results** as a brain. He proposed the following test, now called the **Turing Test:**

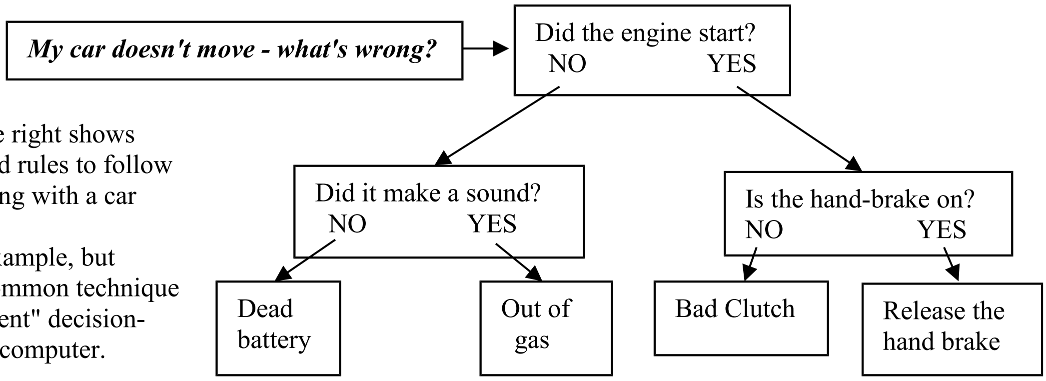
> Imagine a party game. There are two computer terminals (keyboard and monitor) in front of two doors. You are told that there is a human being behind one door, and a computer behind the other door. You may type questions into both keyboards and you will receive replies. You are to ask questions and study the answers and attempt to guess which terminal is connected to the human and which is connected to the computer.

In modern terms, we can imagine sending eMail to an account and receiving answers from a person, but from another account receiving answers from a computer. What questions would you ask in order to trap the computer into revealing itself?

Turing claimed that if you cannot tell the difference - if you guess wrong about which door is hiding the person - then the computer has demonstrated intelligent behavior, because it is indistinguishable from a human being.

In 1966, Joseph Weizenbaum wrote the **Elisa** program that pretended to be a psychiatrist, asking the patient questions and encouraging the patient to talk about themselves and their problems. This program might fool someone for a few minutes, but not for long. It apparently did fool a colleague who actually believed that Weizenbaum was sitting at another terminal answering the questions. This was the first program that was a serious attempt to pass the Turing test, but it did not succeed.

Expert System
An expert system is a large collection of **data (experience)** and **rules (understanding)**, collected from an expert in a specific field. The computer uses the rules to ask questions, then analyzes the answers by comparing them to its database of experience. **Mycin** is a famous expert system, created around 1970, which could diagnose blood diseases by inputting test results and symptoms and comparing them to a large database of known diseases and symptoms. During the 1970's expert systems appeared to be a promising path toward AI. But it turned out that it was far too difficult to specify all the rules and input all the data necessary for an expert system to perform well, even in a very limited area.



Decision Tree
The decision tree at the right shows the questions to ask and rules to follow to find out what is wrong with a car that won't move.
This is a very trivial example, but demonstrates a very common technique for specifying "intelligent" decision-making behavior for a computer.

Boolean Logic - the decision tree above is an example of **Boolean Logic.** The laws of Boolean logic only allow two answers - True or False, Yes or No. **Digital computers** are built from **transistors** and **switches** that are On or Off - no in-between states. Thus Boolean Logic is the basic thinking process of a computer, so a simple decision tree is easy to program.

Fuzzy Logic - an attempt to use a more realistic concept of true and false. In the real world, statements are **probably true** or **perhaps false** or **usually on** or **frequently off.** Very few things in the real world actually behave according to Boolean Logic. Fuzzy logic attempts to use **probabilities** and **experiential learning** to allow a computer to "think" in a more complex, human-like fashion. **Learning** is a crucial part of fuzzy logic. The computer is supposed to input data, draw conclusions, check the accuracy of the solutions, then revise its own internal rules to improve the accuracy of its decisions. The rules for the decisions change as more data (experience) is collected. This is called **dynamic learning.**

Current AI research is succeeding in providing useable speech-recognition software. This doesn't make the computer seem intelligent, but it does make it a lot easier to communicate with. Speech recognition is a very difficult task for a computer, because people don't all talk the same, don't use a standard vocabulary, and use idioms and refer to experiences which the computer cannot understand. "Wow! Did you see that hot car parked out back? Really cool."
Poor computer - it probably thinks the car is both hot and cold, it doesn't know where "out back" is, and it probably cannot "see" anything. Nevertheless, it can probably produce a reasonably accurate printed copy of the sentence.

Speech recognition is a good area for fuzzy logic techniques. The words "high" and "hi" sound identical. If the computer hears this sound in the middle of a sentence, it is more likely to be "high". Nevertheless, it is not certain.
A person might say "You know, I always say hi when I see a friend in the hall." This is the unusual situation. The word "Hi" usually appears at the beginning of a sentence. When a computer listens to a sentence, there will be many uncertainties, which it must clear up and make a best guess at what all the words were. "I read a red sign that caught my eye due to the sine wave at the top. " - that is a tough sentence for the computer. Speech recognition systems require **training** to **learn** how to understand your voice - a real example of dynamic learning.

Neural Networks - an attempt to create computer systems which develop connections (networks) in a similar fashion to the growth of neurons and synapses in the human brain. Once again a promising idea which has not really produced very effective results.

How far is AI now?
Alan Turing guessed that by the year 2000, a computer would pass the Turing Test. Although some computer systems have achieved some success in some very limited errors (playing chess is one such area), people don't really get fooled into thinking they are dealing with a human being when it is actually a computer. No computer has come close to actually passing the Turing Test in a general sense. Why not? Human beings interact with each other and carry on conversations based on personal experiences and common experience. They talk about emotions and sensations (pain, heat, cold, joy, anger). Computers don't experience these things and don't understand them. A computer may be able to answer a question like "How are you feeling?" by giving a stored reply - "Fine, thanks." But it cannot engage in a discussion about "Describe the differences and similarities between Renaissance painting and the advertisement I am holding in my hand."

A lot of stock market trading is being aided by AI programs that analyze the behavior of stock prices and try to predict future price changes. In computer games, the opponents attack and fight in using apparently intelligent tactics. Computer software can store standard phrases and even analyze plot and character development to help a writer with a play or a novel. But all of these behaviors involve a human user, with the computer as an advisor.

About 10 years ago, the New York stock market crashed, largely due to unsupervised computer systems issuing commands to sell stocks. Other programs responded by selling stocks, and it all got out of hand and a lot of people lost a lot of money. After that, severe restrictions were put in place to control wild cycles of computer generated activity. The computer systems really are not "intelligent" - just very fast!

Other examples of AI applications:

- Image recognition - the military uses AI systems to analyze reconnaissance photographs and recognize vehicles
- Games - in 1997 world chess champion Kasparov lost two chess games to IBM's computer program Deep Blue
- Software agents - programs which search networks or interact with computer systems, replacing the actions of a normal human user
- Transportation Control - air-traffic control, planning a route for an automobile trip

Converging Information Technologies (Communication)

Broadcast Media - one transmitter sends a program to **many recipients** at a **specific time.** This is **synchronous** transmission - you must be watching/listening at the same time that the transmission occurs. This is traditionally a **one-way** transmission - the receiver cannot transmit back to the sender. **Radio and TV** are examples.

Point-to-Point - **one transmitter** communicates with **one recipient.** This is **synchronous** transmission - you must be receiving at the same time that the transmission occurs. **Telephone, FAX, and Short-wave radio** are examples. These are traditionally **two-way** communications.

Channel - a single **wire** or **radio frequency band** used for a transmission. In Germany the TVs have 40 channels. A telephone is a single channel.

Bandwidth - the total amount of **data** which can be transmitted per second in a **transmission channel** (wire or radio frequency band). The bandwidth of the very fastest modems is **56 KiloBits** per second. This is **7 KiloBytes.** At this rate, it takes 200 seconds (over 3 minutes) to transmit an entire floppy diskette (1.44 MegaBytes).

Multiplexing - using one channel to carry many transmissions. One method is **time-slicing** - transmit one message for 1 millisecond, then switch to a different transmission for 1 millisecond, then back to the first transmission for a millisecond, then back to the second. This requires sophisticated equipment (e.g. computers) at each end to manage the rotation of the connections. If the channel is carrying a **synchronous signal** (e.g. someone's voice) each signal must be reconstituted at the receiver. Multiplexing one very fast channel to carry several slow transmissions simultaneously.
If you have music playing from your stereo during a phone call, and the other person hears both your voice and the stereo, then you are performing multiplexing.

Packet Switching - networks divide transmissions into **small pieces** that are passed from one computer to the next. Each packet contains a **destination address**, **origin address**, and some **data. Routers** read the addresses and send the packet along to another router that is closer to the destination. The router also **remembers** the packet temporarily, in case another packet with the same addresses comes along. It tries to use the same **route** for the next packet, but if that route is busy, it will try **alternate routes**.

Network Communications - networks can **simulate** both broadcast and point-to-point transmission.
The functional units are **cables**, **routers, servers, clients, protocols, and a NOS (Network Operating System).**
Most networks use **packet-switching** to achieve **multiplexing** behavior across the network. The packet-switching strategy is actually **asynchronous**, but is used to simulate **synchronous** communication - for example, **video-conferencing.**

LAN (Local Area Network) - consists of equipment that is all under the control of a **single organization.** Our school has a LAN, with **servers** containing data about the students, faculty, and purchase orders. Each teacher has a **work-station (client)** in their classroom. We use both **Novell** and **Windows NT** as operating systems.

WAN (wide-area-network) - consists of equipment which is **not** under the control of a single organization. The **Internet** is the obvious example, and it spans the entire globe. There is not real **NOS**, as there are so many different types of machines connected. Instead, the **TCP/IP (Transmission Control Protocol/Internet Protocol)** is used throughout the network to **coordinate** the efforts of millions of separate computers.

Convergence

Watching TV on Your Computer Screen
There are three different ways to do this - (1) buy a **TV Adapter** on an expansion card and plug it into your computer - this is a TV receiver which produces a digital output and displays it on the computer monitor; (2) watch **videos** which are transmitted from web-site; (3) watch a **real-time feed** (live program) from someone else's computer (**video conferencing)** or from a commercial sender which is compresses their signal and **broadcasting it** across the Internet.

Sending Computer Signals To a TV
Teletext is a computer signal (like a text-file) which is **embedded** in a TV transmission (multiplexing). If the TV contains the proper circuitry, it can **decode** and **display** the Teletext signal. Teletext commonly transmits news, weather, and traffic information.

Smartphones - you can do (almost) "everything" with a Smartphone

Video-on-Demand
This will replace trips to the video store. You make a phone-call, or transmit a request across the Internet, and someone in the central office (?) and a computer begins transmitting the movie which you requested, and it plays on your TV.
This has been tried in experiments in the US, but was not very popular. For very high quality video, this requires a lot of **band-width -** much higher than provided by a typical Internet/modem link. It also requires quite a **fast transmitter**, and probably requires a lot of transmitters for lots of customers. But considering that a trip to the video store probably requires a 40,000 DM automobile, video-on-demand is still not a dead concept, just not quite good enough yet.

Push vs Pull
The Internet is a **pull** system - you **ask** for data from a server, and then it sends it. **Internet Explorer 4** introduced the concept of **channels**, which would **push** data to your computer without a direct request. This is actually achieved by your computer **automatically pulling** the data in the traditional way - you just don't notice. **Push technology** was quite the rage about around 1997, but has not turned out to be popular - it slows down your computer substantially, and you mostly end up with information you didn't want anyway. In the paper world, this is like **junk mail**. But you pay for a 2000 DM computer and an Internet connection for the privilege of receiving it.

Content vs Technology
The **information** which is transmitted is called **content.** Far too many "computer people" are only interested in the technology - **how** do we do it. The construct brilliant systems (e.g. **push technology**) without enough useful content. They leave that part up to other people. They build the network and maintain it, but that is the easy part. Providing 24 hours a day of useful content on 40 television channels, 100 radio channels, and thousands of Internet channels requires an awful lot of people being busy creating content constantly. As a result, **free content** is very popular - the millions "personal" Web-sites which people create for free and allow other people to look at. Or watch **GIGA** - they must provide 5 hours of content per day. The result is not 3 new feature-length films per day - that would be impossible. Instead, we have the **talking heads** syndrome - a human being sitting in front of the TV camera and talking. This content is easy to produce, and comprises a large fraction of much of the content of TV and radio stations.

Cellular Phones (Handy's)
Cellular phones are tiny computerized devices that can send and receive **digital radio** signals. The advantage of digital rather than traditional analog transmission is that digital transmissions are easy to multiplex, and the quality can be maintained because it is easier to filter out noise. Cellular phone systems require an **antenna** nearby. The city or country is divided into **cells** (small circles covering several kilometers). The call is picked up by the nearest antenna, relayed to the antenna nearest the receiver, and then transmitted to the receiver. Some newer cellular phones include extra circuitry to display Internet pages, receive and display eMail, or to perform the functions of a PIM (personal information manager).

Pager (Beeper) - Like a cellular phone, but it can only **receive** a signal, and that is a very simple signal - a beep or a few words of text. It cannot send messages back.

Other Phone Features
FAX machines capture pictures, transmit them across a traditional telephone line directly to another FAX machine (point-to-point) where the image is printed. The technology is relatively old (commercially available for over 40 years) and is now simple to use. It is rather expensive for long-distance transmission because one must pay the cost of the phone call.

Answering machines were formerly cassette-tape-recorders that were attached to the telephone. Now **digital answering machines** don't require tapes. The advantage is that the machine can be "smarter", providing **voice-generation** that can interact with the caller.

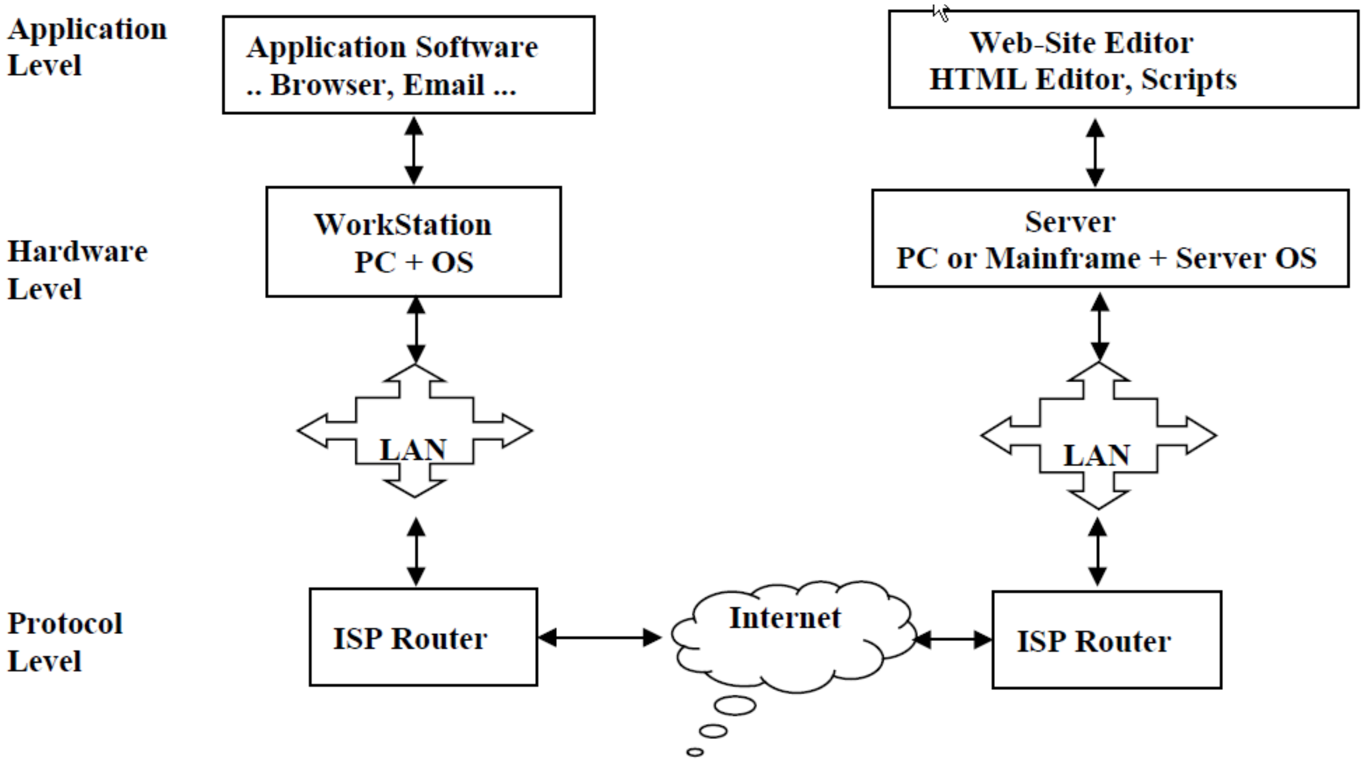
Automated Dialing - many phones will store a list of phone numbers and dial them with a single keypress. **Auto-redial** allows the phone to dial the same number over and over again - useful for an important call to a busy phone.

Call Forwarding - a telephone can be programmed to automatically transfer an incoming call to another telephone - useful for a businessman who is in a meeting in another office.

Internet

The Internet was created as a **public system** used to connect lots of **private networks** to each other. It was then expanded to allow **individual computers** to connect through **Internet Service Providers (ISP)**.

The basic connection works like this:



ISO (nternational Standards Organization) - specifies the basic concepts for networking.
These include the three levels shown above - **Applications, Hardware, Protocols** - and also describes three standard **Network Topologies:**

Bus - all the clients share one single cable connected to the server

Star - each client has its own cable connecting it to the server

Ring - like a bus, but actually makes a complete circle - then if one cable is broken or disconnected, the network still functions.

IP Address - 4 numbers below 256, like this: 176.234.111.15
The numbers have no particular meaning, but the address is unique - like a telephone number.

URL - **Universal Resource Locator -** each server and file in the Internet has a unique address, like this:

www.youtube.com/a38vu893yfrhj

WWW is the name of a server program. YouTube.com is the domain name. .COM is the **top-level-domain.**
The rest identifies a **resource** (file) located on the server.

FTP - File Transfer Protocol - a simple system for allowing **file servers** to transfer files to clients through the Internet. This system existed before the World Wide Web began (before 1990).

World Wide Web (WWW) - the WWW was invented around 1990 to make the Internet easier to use. The concept included HTTP, HTML, and web-addresses (DNS).

HTTP - Hyper Text Transfer Protocol - the protocol (communication system) used in the World Wide Web. Users don't actually see this, but it allows **browsers** to function simply.

HTML - Hyper Text Markup Language - a set of codes which make it possible to produce attractive web-pages. In the early 1990's, HTML pages were created using **text-editors** and the developers wrote the HTML tags **by hand.** Later (after 1995) **WYSIWYG** systems allowed web-pages to be constructed with the same ease as using a Word-Processor.

Web Address (Domain Name) - a set of words separated by dots and slashes, like this:

www.cnn.com\topnews.html

A **Domain Name Server (DNS)** must read this address and look up the matching IP address (e.g. 192.204.123.10). The IP is the actual address used to locate the server. If you have a home connection, your ISP takes care of the DNS translations. In an Intranet, the organization may have their own DNS server.

INTERNIC is the organization that keeps track of which web addresses correspond to which URLs. If you want to set up a web-server, you must **register** your **IP address** and **Domain Name** with INTERNIC - this is quite cheap, costing only 70 dollars for 2 years.

There are no actual "rules" about domain names, but there are some general groups and some guidelines:

Suffix	**Group**	**Guidelines**
.COM	commercial	businesses including those using advertising
.EDU	education	schools and universities, no advertising or selling
.UK	United Kingdom	Countries - these are also .COM addresses, but the
.DE	Deutschland	country suffix keeps things organized and
.AU	Australia	allows more addresses to be created
.RU	Russia	Each country has a 2 letter code
.GOV	Government	MIL, GOV, ORG were the original groups for the Internet
.MIL	Military	
.ORG	Organization(usually non-profit)	

Search Engine - e.g. AltaVista, Yahoo, etc. These servers keep huge databases of web-pages, indexed by topic and content.

Boolean Operations - **AND (+), OR, NOT(-)** can be used to limit a search engine so it doesn't return so many useless results.

Encryption (Cryptography) - secret code software which ensures that your message can only be read by the intended recipient, not by "hackers".

Etiquette (netiquette) - rules and guidelines for behavior on the Internet. For example, **SPAM** (junk eMail) is a bad thing - mailing eMail to lots of people who are not actually interested. **Advertising** is not really wanted by most people. USING ALL CAPITAL LETTERS IN EMAIL IS LIKE SHOUTING - don't do it. Pornography and bad language are also discouraged.

LISTSERV - a mailing list where one person can post an eMail and it is automatically copied to all the members of the listserv. This is useful for groups of professionals who wish to carry on discussions via eMail.

CHAT - like eMail, but it occurs in "real-time" (synchronous) - the participants are all actually sitting at their computers at the same time.

ISDN - a digital telephone connection - this is more reliable and faster (128 KiloBits per Second) than a traditional analog connection (max 56 KiloBits per second).

Future Developments

Faster Internet

Cache (Temporary Storage) - browsers store a copy of every web-page on the local hard-disk. This copy is kept for several days, and then thrown away. The storage area is called the **cache.** It includes all the pictures for the page as well. If you visit the same web-site again 5 minutes later (say by pressing the **back** button), the page and all its pictures are loaded from the cache - this is very fast, and avoids the usually delay of contacting the web-site and waiting for the data. Future development will increase the **cleverness** and **size** of local cache, decreasing the actual data-traffic load on your Internet connection.

ISDN (Integrated Services Digital Network) - supplies a **digital** telephone line with **2 channels** at **64 KiloBits per second.** This can carry a computer signal on one channel and a simultaneous phone call on the other. This is only a little bit faster than normal modems that can carry up to 56 KiloBits per second. The real advantage is that normal telephone lines contain lots of **noise** that causes modems to retransmit or to **step down** to lower speeds -
typically 28 KiloBits. The digital line is pretty much error-free, and runs continuously at 64 KiloBits per second.

Cable Modems use existing TV cable to deliver Internet data up to 2 MegaBits per second. Cable is basically only a one-way link to your house, so some strategies link a cable modem for **downloads**(data delivery) with a normal telephone modem for **uploads** (requests for data). Your PC sends the **URL** over the telephone, and the web-site returns the web-pages over cable. This is sensible because the web-addresses going over the telephone are only a few bytes, but the web-pages coming back can by hundreds of KiloBytes.

Compression - new compression strategies achieve up to **100 to 1** compression, allowing **video** to be delivered in real time. This requires faster **microprocessors** to decompress the signal in **real-time** - that means it can be displayed at the same speed that you should see it, and it keeps up with the data which is being delivered.

JAVA - a programming language which runs programs as part of a web-page. Rather than downloading animation, a Java program can **generate** the animation locally, by drawing lines, circles, and other basic shapes. The Java program can be a lot smaller than the animation that it generates, so it downloads a lot faster than the corresponding graphics. Java can also produce **interactive** pages without requiring repeated access to the **web-server. JAVA** already exists and is supported by most browsers, but its use can still expand substantially.

More Efficient and Productive Offices

Groupware - links together the separate work of a group of employees. A group can keep all their **appointment calendars** in a central database, and the groupware will help coordinate their activities and plan meetings. If a group is cooperating on a large **project**, groupware can be used to keep track of each employees progress - this is especially useful if one part of the project depends on timely completion of another part.

Video-conferencing - many large companies are now **global** - they have offices in many countries around the world. Employees may fly to another country to attend a **meeting** or **conference** for several days. This is expensive, time-consuming, and stressful. Telephone systems already support **conference calls** which can have several people connected in a large telephone call where everyone hears everyone else. But this only carries sound, not pictures. Video-conferencing allows a group of people to **see and hear** each other, thus participating in a **virtual conference** that can be quite realistic. One participant can **show** a report or a picture to the others. Participants can see each other's faces and read expressions. If this works correctly, it could replace a lot of flights and provide better **communication** for global companies. The technology already exists, but it is very expensive, the connections are not very reliable, and the pictures are delivered at **frame-rates** of 5 frames per second, which is only a very jerky sequence of still frames, not really a video.

Cellular Phones - these already exist, but they are a bit expensive - costing 1 DM or more per minute to use during the business day. Cellular service will quickly become more widely available and cheaper. It will also be expanded to include **eMail** delivery and **database access** to for businessman on the move. This frees people to leave the office rather than sitting around waiting for an important call.

Ethics

Technical issues in **IT** involve what **can** be done, and **how** to do it **efficiently** and **cheaply. Ethical** issues involve what **should** or **should not** be done, the **consequences** of using technology, and methods for **encouraging and enforcing** ethical behavior. **Technical** issues usually involve **hardware and software.** Ethical issues involve **people.**

General Areas of Concern

Automation vs Reliability

Dependence vs Control

Trust vs Abuse

Privacy vs Anonymity

Power vs Corruption

Expensive vs Cheap

Critical Systems - Some computer systems function **automatically,** without direct human control. When you send an eMail, it is **routed** from computer to computer until it reaches its destination. You probably don't worry much about your eMail, as it is not a **critical system**. Critical systems are those which **must function**- otherwise serious damage occurs. Critical systems are often highly automated. Critical systems **must** be **reliable.**

Air-traffic-control is a **critical system** for an airport - it is virtually impossible for planes to land and take off if this system stops functioning. A malfunction could give incorrect data to an airplane's navigational system, causing a crash. If the power goes out or the computers go down (crash), there is no good way to track the incoming planes - they are landing 1 per minute at Frankfurt. The airline companies don't own the air-traffic-control system - they pay money to land at the airport, and the airport maintains the air-traffic-control computers.
What are the alternative methods to this equipment? What was done in the past (30,40, 50 years ago)?
What level of reliability is acceptable? How expensive is that? Who is responsible? Who suffers if it fails?

Hospitals use automated equipment to monitor and support patients in **intensive care units**. Nurses might look in on the patient periodically, but most of the time they rely on the machines to keep pumping blood, providing oxygen, and to sound an alert if the patient's condition worsens. Are the nurses being irresponsible? Are the doctors being irresponsible? If the equipment fails, can the patient's family sue the hospital for negligence?
What are the alternative methods to this equipment? What was done in the past (30,40, 50 years ago)?
What level of reliability is acceptable? How expensive is that? Who is responsible? Who suffers if it fails?

Banks use computer **databases to keep track** of money - small amounts belonging to individuals, large amounts belonging to companies. They use computers to **transfer money** from one bank to another. Is this safe? Could the computer make a mistake and send your money to the wrong bank? Can a **hacker** break in to a bank computer and steal money? If they did, would anyone notice? If they noticed, who pays the damages?
What are the alternative methods to this equipment? What was done in the past (30,40, 50 years ago)?
What level of reliability is acceptable? How expensive is that? Who is responsible? Who suffers if it fails?

Police use computers to **keep track of people -** to collect and store personal information: name, address, phone number, job, income, habits, criminal history. This data can be used for **good** (ethical) purposes - the police can catch criminals more easily by using computer systems. Technology has enabled **DNA** matching. Databases of fingerprints and photographs permit police to have access to much more information about possible suspects. These technologies can also be used to violate people's rights. If a politician wishes to damage the reputation of another politician, he could gain **unauthorized access** to police records and make some of the person's activities public. The police could accidentally arrest the wrong person as the result of incorrect data in a database. What can go wrong?
What are the alternative methods to this equipment? What was done in the past (30,40, 50 years ago)?
What level of reliability is acceptable? How expensive is that? Who is responsible? Who suffers if it fails?

Internet access has spread like wildfire. The number of people connected to the Internet is increasing at a rate of 10% per month - the size of the Internet is more than doubling each year. **Nobody controls the Internet** directly. It spans the entire world, and many different people and many different governments each control small parts of the Internet. The US government is purposely avoiding control, to allow the Internet to grow. Anyone can create a web-site, containing any information they wish. The information might be false. It might be embarrassing or unfair. It might be dangerous. It might be illegal. This is no different from a bulletin board where people hang up paper, or people who pass out leaflets in a city. However, the Internet is different because it has the potential of so many people seeing it, and the normal controls that have evolved for other **media** have not yet evolved for the Internet. Laws exist for very specific violations of music copyrights, book copyrights, plagiarism and slander in print. Such laws have not been written for the Internet, and the existing laws are difficult to apply, especially where people in one country are damaged by a web-site in a different country.

The **reliability** of the information in the Internet is not very high. We have not yet reached a stage of **dependence** on the Internet, but this is coming rapidly. **Most people don't use eMail**, but those who do quickly become dependent, and the number using eMail is **increasing exponentially.**

The Internet gives **power** to people who might not have had it otherwise. On the other hand, it does not give power to everybody, and may take power away from some people who previously had it. Do the people with increased power also have an increased level of responsibility, or does this only increase their ability to **abuse** that power?
In the Internet, you can be **anonymous.** On the other hand, anonymous people can display information about other people, thus violating their **privacy.**

What are the alternative methods to the Internet? What was done in the past (30,40, 50 years ago)?
What level of reliability is acceptable? How expensive is that? Who is responsible? Who suffers if it fails?

Offices have changed substantially over the past 30 years, largely as a result of increased use and new uses of **information technology**. These changes include **telephones, FAX machines, photocopiers, PCs, databases, and computerized design** (CAD for engineers and architects, graphic design software for advertisers). Some employees find the new equipment confusing and difficult to use. Businesses buy computer equipment to increase productivity and save money, and may be reluctant to spend money on **training** for the employees. Indeed, many businesses actually adopt computer systems with the intention of reducing the number of employees. Some companies use computer systems to **spy** on their employees, keeping track of what the employees are doing, how many appointments they have, what eMails they send and receive, how many documents they finish typing.
What are the alternative methods for offices? What was done in the past (30,40, 50 years ago)?
What level of reliability is acceptable? How expensive is that? Who is responsible? Who can suffer if it is abused?

Entertainment, news, and education have not changed as dramatically over the past 30 years as offices, but they have certainly changed as new **information technology** systems are developed. **Satellites** enable world-wide broadcasts of important events (Olympics, wars). **Cable television** has increased the number of TV stations available. **Digital** technology has made radio, TV, and telephone communications cheaper, higher quality, and more reliable. PCs provide **chat lines**, **eMail**, and **multimedia entertainment.** Along with the improved communication come abuses of communication - eavesdropping, false information, information overload, and advertising. Who guarantees that we are not bombarded with advertisements when surfing the Internet? Who can prevent **spam** from clogging our eMail boxes?
What are the alternative methods? What was done in the past (30,40, 50 years ago)?
What level of reliability is acceptable? How expensive is that? Who is responsible? Who can suffer if it is abused?

Computer Crime is not as new as some people might expect - around 1970 a life-insurance company used computers to create phony life-insurance policies for phony people. The policies were then traded (sold) to insurance brokerages. The amount of damages? **A billion dollars!** That was a lot of money back then, as it is now. **Hackers** and **crackers** are not the big computer criminals. The big criminals are nameless executives of big companies who secretly move money between accounts, use computers to steal secret information from competitors, and drive entire companies into bankruptcy. The crimes are not new, but in the **information age** computers make crime easier. **Computer crime** opens an entirely new realm - stealing millions of dollars from a bank in a country half-way around the world, and the bank doesn't even notice the money is gone. **Forgery** was possible in the past, but now a totally new person can be created in a database **electronically**, and the result is indistinguishable from the real thing.

Computers can make you fast, efficient, and powerful
but .... they do the same for criminals.

Global Society

Globalization - many companies operate internationally, around the globe. International and global operations are facilitated by developments in IT. **Teleconferencing**, **eMail**, **improved telephone services,** and **Wide Area Networks** (WANs) make it possible for employees around the world to communicate as easily as in a single large office building.
Corporations in developed countries can build and operate factories in underdeveloped countries - land, materials, and labor are cheaper this way. Combined with cheap shipping (cargo containers) and cheap communications, goods can be offered at a cheaper price this way.

Just In Time (JIT) Manufacturing
Rather than stockpiling inventory in a large city, a company can manufacture "just in time" in direct response to orders. This saves money by avoiding expensive storage facilities, as well as avoiding building up inventory that might become obsolete and never be sold. Profit margins are set lower, with less overhead and fewer losses due to overstocking. JIT is facilitated by information technology - inventories are monitored and purchases made using EDI.

Just In Case (JIC)
Despite the success of JIT, most companies still keep a small inventory on hand, "just in case" a customer needs something immediately.

Electronic Data Interchange (EDI)
World-wide data exchange occurs inside international corporations and between corporations. This requires specific file formats and protocols to be followed, to guarantee that computers can accept data from other computers. EDI documents are used to process **purchase orders, send transcripts,** and **exchange money.**

Incompatibility
Computer systems must **conform to standards** to ensure that one computer can successfully communicate with another. The incredibly rapid expansion in computer systems in over the past two decades has happened without enough attention to standards. Rapid developments in hardware result in **rapid obsolescence -** a 5 year old computer may contain components that cannot be replaced - compatible replacement parts are no longer manufactured.
New versions of software replace older versions, and use new, **incompatible file formats** (witness WinWord 97).

ISO (International Standards Organization) - an organization which develops and documents standards for **data exchange**, hardware **interfaces**, data **file formats**, and software **protocols**. An example standard is a set of two letter codes for countries: AU = Australia, US = United States, DE = Deutschland, UK = United Kingdom, etc.)

UNICODE - this Universal Code will replace 8-bit ASCII (256 characters) with a new 16-bit code (65536 character codes). That is enough to support Roman, Greek, Russian, Chinese, Persian, and many other alphabets. This is part of ISO standard 0646. The introduction of Unicode has been going on for 10 years, but is still not very widely implemented. There are so many data files already saved in 8-bit ASCII, and so much software using that standard, that it is an immense job to change over to a substantially different, incompatible system.

URL (Universal Resource Locator) - a unique code assigned to each computer in the Internet - e.g. 195.102.135.121
Each of the four numbers can go up to 255, providing a total of 4 billion unique codes. When this was invented in the 1970's, there were under a million computers in the entire world. Now there are several hundred million. If that number doubles every two years, as appears to be the case, the 4 billion codes will be used up within 5-10 years.

Information Appliances - in the near future, heaters, toasters, televisions, and many other electrical appliances will be connected to each other (and the Internet!) This will allow you to call your house (not a person, but the house itself) and tell the heater to turn itself on. This proliferation of devices will use up the 4 billion URLs even more quickly.

Internet II - currently committees of engineers are working on new standards for the Internet, to expand the number of URLs, speed up the Internet, and develop new protocols to ensure better service and to regulate Internet activity. This is being called Internet version 2, but there is still no clear standard in sight.

Y2K - Year 2000 "bug" - In the 1960s, many computer systems stored the year as a 2-digit number - e.g. 65 instead of 1965. This made a lot of sense back then, when computers only contained 64 KiloBytes of memory, and the hard-disks were the size of a washing machine and stored only 20 MegaBytes. Every byte saved was money saved. As data files and software accumulated in this format, this turned into a **legacy problem** - new software and data formats continued to use 2-digit years for the sake of compatibility with old software. This continued past 1990. Now we have so many computer systems in the world that nobody actually knows how many machines may stop functioning when their internal clocks change from 99 to 00. The massive **inter-reliance of computer systems** around the world may result in a global crisis on 1st January, 2000, when many computers malfunction simultaneously.