

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего профессионального образования
«Пермский национальный исследовательский
политехнический университет»

А.С. Балахонов, А.Н. Лыков

КОМПЬЮТЕРНЫЕ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Практикум

Утверждено

Редакционно-издательским советом университета

Издательство
Пермского национального исследовательского
политехнического университета
2013

ББК 81.2Англ.я81

Б20

Рецензенты:

канд. пед. наук, доцент Пермского национального
исследовательского политехнического университета

И.В. Перлова;

канд. филол. наук, доцент Российского государственного
торгово-экономического университета

(Пермский филиал)

Я.Н. Ронжина

Балахонов, А.С.

Б20

Компьютерные и информационные технологии: практи-
кум / А.С. Балахонов, А.Н. Лыков. – Пермь : Изд-во Перм. нац.
исслед. политехн. ун-та, 2013. – 122 с.

ISBN 978-5-398-01072-5

Даны тексты на английском языке, задания к этим текстам и
словарь новых использованных терминов.

Предназначено для формирования умений информативного
чтения на английском языке и для расширения знаний в области
высоких технологий студентам любых специальностей, изучающим
дисциплину «Информатика».

CONTENTS

Предисловие	6
1. Hardware.....	7
1.1. Hardware.....	7
1.2. Microelectronics	8
1.3. Control unit.....	9
1.4. Arithmetic logic unit (ALU)	10
1.5. Memory	11
1.6. Cache	12
1.7. Flash memory	14
1.8. Input/ Output (I/O)	17
1.9. Input hardware.....	18
1.10. Output devices	24
1.11. Hardware connections	26
Использованные термины.....	27
Задания к текстам	28
2. Software.....	30
2.1. Software.....	30
2.2. Machine code.....	32
2.3. Van Neumann architecture	33
2.4. Operating systems	34
2.5. Linux.....	36
2.6. Application software.....	37
2.7. Database systems	41
Использованные термины.....	42
Задания к текстам	43
3. Programming	44
3.1. Computer programming	44
3.2. Object-oriented programming	48
3.3. Programming languages	50

3.4. Drupal as a CMS	53
Использованные термины	55
Задания к текстам	56
4. Networks	57
4.1. Networks	57
4.2. The Internet	57
4.3. Internet domain	59
4.4. Fast Ethernet.....	61
4.5. TCP/IP	62
4.6. The World Wide Web	63
4.7. Wireless networking.....	65
4.8. Bluetooth	66
4.9. Network Communications.....	69
4.10. Storage area network (SAN)	70
4.11. Profinet	72
4.12. Broadband communications	75
4.13. Router	77
Использованные термины	80
Задания к текстам	81
5. Data mining.....	82
5.1. Data mining	82
5.2. Research on the Internet	83
5.3. Google	84
Использованные термины	86
Задания к текстам	86
6. Data transfer	87
6.1. Data transfer	87
6.2. E-mail	89
6.3. E-mail protocols	91
Использованные термины	93
Задания к текстам	93

7. Graphics / design	95
7.1. Graphics and design	95
7.2. Word processing.....	96
7.3. Desktop Publishing.....	97
7.4. Computer-aided design.....	98
7.5. Product data management.....	102
7.6. Product lifecycle management.....	104
Использованные термины.....	109
Задания к текстам	109
8. Future	110
8.1. Future developments.....	110
8.2. Architecture of future information systems.....	112
8.3. The future of information technology	114
Использованные термины.....	117
Задания к текстам	117
Frequently Used Words	118

ПРЕДИСЛОВИЕ

Издание предназначено для обучения иноязычному информативному чтению (с целью получения информации) студентов I–II курсов, изучающих дисциплину «Информатика», и представляет собой сборник оригинальных текстов, которые содержат информацию в области высоких технологий, необходимую специалисту любого профиля.

Расположение текстов в сборнике повторяет последовательность изучения компьютерных и информационных технологий в рамках специальности «Информатика». Иноязычные тексты тематически дублируют материал лекций и лабораторных работ по дисциплине «Информатика» и предъявляются с 1–2-недельным запаздыванием от аналогичного материала по специальности. Таким образом, преподаватель английского языка может быть уверен, что студент поймёт содержание читаемого текста.

Например, в начале сентября преподаватель по «Информатике» начинает читать лекции по теме «Аппаратное обеспечение». Следовательно, преподаватель английского языка начинает работать с данным пособием в середине сентября (гл. 1). В октябре читают лекции «Программное обеспечение», а иноязычные тексты в рамках этой тематики (гл. 2) предъявляются в ноябре.

Задания к текстам нацелены на углубление понимания прочитанного текста и на присвоение содержащейся в нём профессиональной информации. Они предназначены для работы как в аудитории, так и дома. Вопросы к конкретному предложению или абзацу текста составляют сами студенты и задают их во время ответа товарища или при парной работе, как предусмотрено в заданиях. Задания к текстам также содержат «реальные» вопросы, согласованные с преподавателями по «Информатике», задаваемые на зачёте и экзамене по этой специальности в середине и конце учебного года. Поэтому целесообразно рекомендовать студентам готовить ответы на такие задания в письменной форме для того, чтобы они могли их использовать при подготовке и ответе на зачёте или экзамене по «Информатике».

1. HARDWARE

1.1. Hardware

Hardware is the equipment involved in the function of a computer. Computer hardware consists of the components that can be physically handled. A general purpose computer has four main components: the arithmetic logic unit (ALU), the control unit, the memory, and the input and output devices (collectively termed I/O). These parts are interconnected by busses, often made of groups of wires.

Inside each of these parts are thousands to trillions of small electrical circuits which can be turned off or on by means of an electronic switch. Each circuit represents a bit (binary digit) of information so that when the circuit is on it represents a "1", and when off it represents a "0" (in positive logic representation). The circuits are arranged in logic gates so that one or more of the circuits may control the state of one or more of the other circuits.

The control unit, ALU, registers, and basic I/O (and often other hardware closely linked with these) are collectively known as a central processing unit (CPU). Early CPUs were composed of many separate components but now CPUs are typically constructed on a single integrated circuit called a microprocessor. Nearly all modern processors are microprocessors, and the two terms are often used interchangeably.

There are six basic types of access that a processor can perform with external chips. The processor can write data to memory or write data to an I/O device, read data from memory or read data from an I/O device, read instructions from memory, and perform internal manipulation of data within the processor.

In many systems, writing data to memory is functionally identical to writing data to an I/O device. Similarly, reading data from memory constitutes the same external operation as reading data from an I/O device, or reading an instruction from memory. In other words, the processor makes no distinction between memory and I/O.

1.2. Microelectronics

The potential of integrated circuits is so wide that in addition to replacing similar discrete component circuits they are responsible for creating a completely new technology of circuit design.

There are two basic approaches to modern microelectronics – monolithic integrated circuits and film circuits.

In monolithic ICs all circuit elements, active and passive, are simultaneously formed in a single small wafer of silicon. The elements are interconnected by metallic stripes deposited onto the oxidized surface of the silicon wafer.

Active elements (transistors and diodes) and passive elements (resistors and capacitors) are formed in the silicon slice by diffusing impurities into selected regions to modify electrical characteristics, and where necessary to form p-n junctions. The various elements are designed so that all can be formed simultaneously by the same sequence of diffusions.

Film circuits are made by forming the passive electronic component and metallic interconnections on the surface of an insulation substrate. Then the active semiconductor devices are added, usually in discrete wafer form. There are two types of film circuits, thin film and thick film.

In thin film circuits the passive components and interconnection wiring are formed on glass or ceramic substrates, using evaporation techniques. The active components (transistors and diodes) are fabricated as separate semiconductor wafers and assembled into the circuit.

Thick film circuits are prepared in a similar manner except that the passive components and wiring are formed by silk-screen techniques on ceramic substrates.

Microelectronics embraces the entire body of the electronic art which is connected with, or applied to, the realization of electronic circuits, subsystems, or the entire systems from extremely small electronic devices. The terms «microelectronics» and «integrated circuits» are sometimes used interchangeably, but this is not correct.

Microelectronics is a name for extremely small electronic components and circuit assemblies, made by thin-film, thick-film or semiconductor techniques.

An integrated circuit (IC) is a special kind of microelectronics. It is a circuit that has been fabricated as an inseparable assembly of electronic elements in a single structure. It cannot be divided without destroying its intended electronic function. Thus, ICs come under the general category of microelectronics, but all microelectronic units are not necessarily ICs.

1.3. Control unit

The control unit (often called a control system or central controller) manages the computer's various components; it reads and interprets (decodes) the program instructions, transforming them into a series of control signals which activate other parts of the computer. Control systems in advanced computers may change the order of some instructions so as to improve performance.

A microcontroller is a processor, memory, and some I/O devices contained within a single integrated circuit. The buses that interconnect the processor with its I/O exist within the same integrated circuit. The range of available microcontrollers is very broad.

Microcontrollers are very similar to System-on-Chip (SoC) processors, intended for use in conventional computers such as PCs and workstations. Soc processors have a different suite of I/O, reflecting their intended application, and are designed to be interfaced to large banks of external memory. Microcontrollers usually have all their memory on-chip and may provide only limited support for external memory devices.

The control system's function is as follows (some of these steps may be performed concurrently or in a different order depending on the type of CPU):

- Read the code for the next instruction from the cell indicated by the program counter.
- Decode the numerical code for the instruction into a set of commands or signals for each of the other systems.
- Increment the program counter so it points to the next instruction.

- Read whatever data the instruction requires from cells in memory (or perhaps from an input device). The location of this required data is typically stored within the instruction code.
- Provide the necessary data to an ALU or register.
- If the instruction requires an ALU or specialized hardware to complete, instruct the hardware to perform the requested operation.
- Write the result from the ALU back to a memory location or to a register or perhaps an output device.
- Jump back to step (1).

1.4. Arithmetic logic unit (ALU)

The ALU is capable of performing two classes of operations: arithmetic and logic.

The set of arithmetic operations that a particular ALU supports may be limited to addition and subtraction, or might include multiplication, division, trigonometry functions such as sine, cosine, etc., and square roots. Some can only operate on whole numbers (integers) whilst others use floating point to represent real numbers, albeit with limited precision. However, any computer that is capable of performing just the simplest operations can be programmed to break down the more complex operations into simple steps that it can perform. Therefore, any computer can be programmed to perform any arithmetic operation – although it will take more time to do so if its ALU does not directly support the operation. An ALU may also compare numbers and return Boolean truth values (true or false) depending on whether one is equal to, greater than or less than the other (“is 64 greater than 65?”).

Whenever instructed by the processor, the ALU performs an operation (typically one of addition, subtraction, NOT, AND, OR, XOR< shift left/right, or rotate left/right) on one or more values. These values, called operands, are typically obtained from two registers, or from one register and a memory location. The result of the operation is then placed back into a given destination register or memory location. The status outputs

indicate any special attributes about the operation, such as whether the result was zero, negative, or if an overflow or carry occurred. Some processors have separate units for multiplication and division, and for bit shifting, providing faster operation and increased throughput.

1.5. Memory

The memory of the computer system contains both the instructions that the processor will execute and the data it will manipulate. The memory of a computer system is never empty. It always contains something, whether it is instructions, meaningful data, or just the random garbage that appeared in the memory when the system is powered up.

A computer's memory can be viewed as a list of cells into which numbers can be placed or read. Each cell has a numbered "address" and can store a single number. The computer can be instructed to "put the number 123 into the cell numbered 1357" or to "add the number that is in cell 1357 to the number that is in cell 2468 and put the answer into cell 1595". The information stored in memory may represent practically anything. Letters, numbers, even computer instructions can be placed into memory with equal ease. Since the CPU does not differentiate between different types of information, it is the software's responsibility to give significance to what the memory sees as nothing but a series of numbers.

In almost all modern computers, each memory cell is set up to store binary numbers in groups of eight bits (called a byte). Each byte is able to represent 256 different numbers ($2^8 = 256$); either from 0 to 255 or -128 to +127. To store larger numbers, several consecutive bytes may be used (typically, two, four or eight).

A computer can store any kind of information in memory if it can be represented numerically. Modern computers have billions or even trillions of bytes of memory.

The CPU contains a special set of memory cells called registers that can be read and written to much more rapidly than the main memory area. There are typically between two and one hundred registers depending on the type of CPU. Registers are used for the most frequently needed data items to avoid having to access main memory every time data is needed. As data is

constantly being worked on, reducing the need to access main memory (which is often slow compared to the AI U and control units) greatly increases the computer's speed. A register keeps track of which location in memory the next instruction is to be read from.

Computer main memory comes in two principal varieties: random-access memory or RAM and read-only memory or ROM. RAM can be read and written to anytime the CPU commands it, but ROM is pre-loaded with data and software that never changes, therefore the CPU can only read from it. ROM is typically used to store the computer's initial start-up instructions. In general, the contents of RAM are erased when the power to the computer is turned off, but ROM retains its data indefinitely. In a PC, the ROM contains a specialized program called the BIOS that orchestrates loading the computer's operating system from the hard disk drive into RAM whenever the computer is turned on or reset. In embedded computers, which frequently do not have disk drives, all of the required software may be stored in ROM. Software stored in ROM is often called firmware, because it is notionally more like hardware than software. Flash memory blurs the distinction between ROM and RAM, as it retains its data when turned off but is also rewritable. It is typically much slower than conventional ROM and RAM however, so its use is restricted to applications where high speed is unnecessary.

The internal data storage of the processor is known as its registers. The processor has a limited number of registers, and these are used to hold the current data/operands that the processor is manipulating.

1.6. Cache

Most PCs are held back not by the speed of their main processor, but by the time it takes to move data in and out of memory. One of the most important techniques for getting around this bottleneck is the memory cache.

The idea is to use a small number of very fast memory chips as a buffer or cache between main memory and the processor. Whenever the processor needs to read data it looks in this cache area first. If it finds the data in the cache then this counts as 'a cache hit' and the processor need not

go through the more laborious process of reading data from the main memory. Only if the data is not in the cache does it need to access main memory, but in the process it copies whatever it finds into the cache so that it is there ready for the next time it is needed. The whole process is controlled by a group of logic circuits called the cache controller.

One of the cache controller's main jobs is to look after 'cache coherency' which means ensuring that any changes written to main memory are reflected within the cache and vice versa. There are several techniques for achieving this, the most obvious being for the processor to write directly to both the cache and main memory at the same time. This is known as a 'write-through' cache and is the safest solution, but also the slowest.

The main alternative is the 'write-back' cache which allows the processor to write changes only to the cache and not to main memory. Cache entries that have changed are flagged as 'dirty', telling the cache controller to write their contents back to main memory before using the space to cache new data. A write back cache speeds up the write process, but does require a more intelligent cache controller.

Most cache controllers move a 'line' of data rather than just a single item each time they need to transfer data between main memory and the cache. This tends to improve the chance of a cache hit as most programs spend their time stepping through instructions stored sequentially in memory, rather than jumping about from one area to another. The amount of data transferred each time is known as the 'line size'.

If the requested information isn't in the cache, the data is read from the disk along with a large chunk of adjacent information. The cache then makes room for the new data by replacing old. Depending on the algorithm that is being applied, this may be the information that has been in the cache the longest, or the information that is the least recently used. The CPU's request can then be met, and the cache already has the adjacent data loaded in anticipation of that information being requested next.

In more sophisticated computers there may be one or more RAM cache memories, which are slower than registers but faster than main memory. Generally computers with this sort of cache are designed to move

frequently needed data into the cache automatically, often without the need for any intervention on the programmer's part.

1.7. Flash memory

Flash memory is an electronic (i.e. no moving parts) non-volatile computer storage device that can be electrically erased and reprogrammed.

Flash memory was developed from EEPROM (electrically erasable programmable read-only memory). There are two main types of flash memory, which are named after the NAND and NOR logic gates. The internal characteristics of the individual flash memory cells exhibit characteristics similar to those of the corresponding gates.

Whereas EEPROMs had to be completely erased before being rewritten, NAND type flash memory may be written and read in blocks (or pages) which are generally much smaller than the entire device. The NOR type allows a single machine word (byte) to be written or read independently.

The NAND type is primarily used in main memory, memory cards, USB flash drives, solid-state drives, and similar products, for general storage and transfer of data. The NOR type, which allows true random access and therefore direct code execution, is used as a replacement for the older EPROM and as an alternative to certain kinds of RQM applications. However, NOR flash memory may emulate ROM primarily at the machine code level: many digital designs need ROM structures for other uses, often at significantly higher speeds than flash memory may achieve. NAND or NOR flash memory is also often used to store configuration data in numerous digital products, a task previously made possible by EEPROMs or battery-powered static RAM.

Example applications of both types of flash memory include personal computers, PDAs, digital audio players, digital cameras, mobile phones, synthesizers, video games, scientific instrumentation, industrial robotics, medical electronics, and so on. In addition to being non-volatile, flash memory offers fast read access times, as fast as dynamic RAM, although not as fast as static RAM or ROM. Its mechanical shock resistance helps explain its popularity over hard disks in portable devices; as does its high

durability, being able to withstand high pressure, temperature, immersion in water, etc.

Although flash memory is technically a type of EEPROM, the term "EEPROM" is generally used to refer specifically to non-flash EEPROM which is erasable in small blocks, typically bytes. Because erase cycles are slow, the large block sizes used in flash memory erasing give it a significant speed advantage over old-style EEPROM when writing large amounts of data. Flash memory now costs far less than byte-programmable EEPROM and has become the dominant memory type wherever a significant amount of non-volatile, solid state storage is needed.

Principles of operation

Flash memory stores information in an array of memory cells made from floating-gate transistors. In traditional single-level cell (SLC) devices, each cell stores only one bit of information. Some newer flash memory, known as multi-level cell (MLC) devices, including triple-level cell (TLC) devices, can store more than one bit per cell by choosing between multiple levels of electrical charge to apply to the floating gates of its cells.

The floating gate may be conductive (typically polysilicon in most kinds of flash memory) or non-conductive (as in SONOS flash memory).

NOR flash

In NOR gate flash, each cell has one end connected directly to ground, and the other end connected directly to a bit line. This arrangement is called «NOR flash» because it acts like a NOR gate: when one of the word lines (connected to the cell's CG) is brought high, the corresponding storage transistor acts to pull the output bit line low. NOR Flash continues to be the technology of choice for embedded applications requiring a discrete non-volatile memory device. The low read latencies characteristic of NOR devices allow for both direct code execution and data storage in a single memory product.

A single-level NOR flash cell in its default state is logically equivalent to a binary «0» value, because current will flow through the channel under application of an appropriate voltage to the control gate, so

that the bitline voltage is pulled down. A NOR flash cell can be programmed, or set to a binary «1» value, by the following procedure:

- an elevated on-voltage (typically >5 V) is applied to the CG
- the channel is now turned on, so electrons can flow from the source to the drain (assuming an NMOS transistor)
- the source-drain current is sufficiently high to cause some high energy electrons to jump through the insulating layer onto the FG, via a process called hot-electron injection

Erasing

To erase a NOR flash cell (resetting it to the «0» state), a large voltage of the opposite polarity is applied between the CG and source terminal, pulling the electrons off the FG through quantum tunneling. Modern NOR flash memory chips are divided into erase segments (often called blocks or sectors). The erase operation can only be performed on a block-wise basis; all the cells in an erase segment must be erased together. Programming of NOR cells, however, can generally be performed one byte or word at a time.

NAND flash

NAND flash also uses floating-gate transistors, but they are connected in a way that resembles a NAND gate: several transistors are connected in series, and only if all word lines are pulled high is the bit line pulled low. These groups are then connected via some additional transistors to a NOR-style bit line array in the same way that single transistors are linked in NOR flash.

Compared to NOR flash, replacing single transistors with serial-linked groups adds an extra level of addressing. Whereas NOR flash might address memory by page then word, NAND flash might address it by page, word and bit. Bit-level addressing suits bit-serial applications (such as hard disk emulation), which access only 1 bit at a time. Execute-In-Place applications, on the other hand, require every bit in a word to be accessed simultaneously. This requires word-level addressing. In any case, both bit and word addressing modes are possible with either NOR or NAND flash.

To read, first the desired group is selected (in the same way that a single transistor is selected from a NOR array). Next, most of the word lines are pulled up above the VT of a programmed bit, while one of them is pulled up to just over the VT of an erased bit. The series group will conduct (and pull the bit line low) if the selected bit has not been programmed.

Despite the additional transistors, the reduction in ground wires and bit lines allows a denser layout and greater storage capacity per chip. (The ground wires and bit lines are actually much wider than the lines in the diagrams.) In addition, NAND flash is typically permitted to contain a certain number of faults (NOR flash, as is used for a BIOS ROM, is expected to be fault-free). Manufacturers try to maximize the amount of usable storage by shrinking the size of the transistor below the size where they can be made reliably, to the size where further reductions would increase the number of faults faster than it would increase the total storage available.

1.8. Input/Output (I/O)

I/O is the means by which a computer exchanges information with the outside world. Devices that provide input or output to the computer are called peripherals. On a typical personal computer, peripherals include input devices like the keyboard and mouse, and output devices such as the display and printer. Hard disk drives, floppy disk drives and optical disc drives serve as both input and output devices. Computer networking is another form of I/O.

I/O devices are often complex computers in their own right, with their own CPU and memory. A graphics processing unit might contain fifty or more tiny computers that perform the calculations necessary to display 3D graphics. Modern desktop computers contain many smaller computers that assist the main CPU in performing I/O.

I/O devices exist within the same linear space as memory devices, and the same instructions are used to access each. This is known as memory-mapped I/O. Memory-mapped I/O is certainly the most common form.

1.9. Input hardware

Input hardware consists of external devices – that is, components outside of the computer's CPU – that provide information and instructions to the computer.

There are five main kinds of hand-operated units: (1) a keyboard, (2) a mouse, (3) a trackball, (4) a light pen, and (5) a touch screen. There are three principal types of automatic units: (1) a modem, (2) a scanner, and (3) a microphone. In addition, all storage devices can function as input devices.

A keyboard is the main input device. Modern computer keyboards are different from typewriter keyboards. For example, along the top of the keyboard of a desktop PC is a row of function keys, which are designated F1, F2, F3, and so on. Function keys perform special tasks, such as removing a passage of text from one part of a document and inserting it in another.

The keyboard itself is an electronic device. When the user types a character, electronic circuits in the keyboard translate the character into bit charges. The charges travel through wire to a buffer, a temporary storage location in the computer. As soon as the charges appear in the buffer, the microprocessor moves them to RAM. At the same time, the microprocessor instructs the monitor to put the character on the screen.

A keyboard allows the user to type text and commands to the computer. Some keyboards have special function keys or integrated pointing devices, such as a trackball or touch-sensitive regions that let the user's finger motions move an on-screen cursor.

A standard PC keyboard has various groups of keys:

- Alphanumeric keys – these represent letters and numbers, arranged as on a typewriter.
- A numeric keypad appears to the right of the main keyboard and contains numeric and editing keys; the Num Lock key is used to switch from numbers to editing functions.
- Function keys appear at the top of the keyboard and can be programmed to do special jobs.

- Cursor keys include ‘arrow keys’ which move the insertion point, and keys such as Home, End, Page Up, and Page Down, which let you move around documents.
- Dedicated keys are used to issue commands or produce alternative characters. For example: Ctrl changes the functions of other keys (e.g. Ctrl + X cuts the selected text).
- Caps Lock sets the keyboard in ‘CAPITALS’ mode; it only affects letters.
- Enter (or Return) is pressed to select options from a menu or to start a new paragraph.
- Backspace deletes the character to the left of your current position.

A mouse is a hand-held device that lets you move a pointer (or cursor) and select items on the screen. It has one or more buttons to communicate with the PC. A scroll wheel lets you move through your documents or web pages. The pointer looks like an I-bar, an arrow or a pointing hand.

An optical mouse has an optical sensor instead of a ball underneath. A cordless (wireless) mouse has no cable; it sends data via infrared signals or radio waves.

Mouse actions:

- to click, press and release the left button.
- to double-click, press and release the left button twice.
- to drag, hold down the button, move the pointer to a new place and then release the button.
- to right-click, press and release the right button; this action displays a list of commands.

A joystick is a pointing device composed of a lever that moves in multiple directions to navigate a cursor or other graphical object on a computer screen.

A light pen enables a computer user to write words or draw pictures directly into a computer. The user touches the point of the pen on the screen of a special pad, then makes writing or drawing motions. Electronic devices in the pen and under the surface of the pad interact to translate the motions into computer code. The microprocessor then uses the code to put an image

corresponding to the motions on the screen. Some artists and designers use an electronic pen and pad to produce illustrations.

A modem is an electronic device that communicates with other computers over telephone lines. Thus, a modem functions as both an input device and an output device. In many of the latest computers, the modem is built-in.

A modem translates bit charges into tones, then sends the tones over telephone lines to modems of other computers. The other modems reverse the translation process. The speed at which modems transmit and receive data is measured in bits per second (bps). Typical modems built in the mid-1990's have speeds ranging from 14,400 to 28,800 bps. Data can travel at much higher speeds over fiber-optic phone lines or television cables.

A scanner is a peripheral unit that uses light and devices that sense light to digitize photographs and other illustrations, that is to translate them into numerical form for processing by a computer. A scanner equipped with optical character recognition (OCR) software translates the scanned image of a page of text into individually coded characters. Publishers are major users of scanners, but scanners are also becoming popular with home computer users.

There are different types

- A **flatbed** is built like a photocopier and is for use on a desktop; can capture text, colour images and even small 3D objects.
- A **film scanner** is used to scan film negatives or 35 mm **slides** – pictures on photographic film, mounted in a frame.
- A **hand-held** scanner is small and T-shaped, ideal to capture small pictures and logos.
- A **pen scanner** looks like a pen; you can scan text, figures, barcodes and handwritten numbers.

Barcode scanners read barcodes on the products sold in shops and send the price to the computer in the cash register. **Barcodes** consist of a series of black and white stripes used to give products a unique identification number.

The **resolution** of a scanner is measured in **dpi** or dots per inch. For example, a 1,200 dpi scanner gives clearer, more detailed images than a 300 dpi scanner.

Most scanners come with **Optical Character Recognition** software. OCR allows you to scan pages of text and save them into your word processor; they can then be edited.

An optical scanner uses light-sensing equipment to convert images such as a picture or text into electronic signals that can be manipulated by a computer. For example, a photograph can be scanned into a computer and then included in a text document created on that computer.

A microphone is a device for converting sound into signals that can then be stored, manipulated, and played back by the computer. A voice recognition module is a device that converts spoken words into information that the computer can recognize and process.

Digital cameras

A digital camera doesn't use film. Photos are stored as digital data (bits made up of 1s and 0s), usually on a tiny storage device known as a flash memory card. You can connect the camera or memory card to a PC and then alter the images using a program like Adobe Photoshop, or you can view the images on a TV set. Many printers have a special socket so that you can print images directly from a memory card or camera.

Webcams (short for Web cameras) let you send and receive live video pictures through the Internet. They're primarily used for video conferences – video calls – but they can be used to record photos and video onto your hard disk.

The resolution of webcams is expressed in megapixels (million pixels). Webcams connect to the PC via a USB (universal serial bus) or FireWire port; they display video at 24 to 30 frames (pictures) per second. Some include a headset with a microphone and earpiece.

Voice input

Today you can also interact with your computer by voice with a voice-recognition system that converts voice into text, so you can dictate text directly onto your word processor or email program. You can also control your PC with voice commands; this means you can launch

programs, open, save or print files. Some systems let you search the Web or chat using your voice instead of the keyboard.

Storage Hardware

Storage hardware provides permanent storage of information and programs for retrieval by the computer.

The two main types of storage devices are disk drives and memory. There are several types of disk drives: hard, floppy, magneto-optical, and compact. Hard disk drives store information in magnetic particles embedded in a disk. Usually a permanent part of the computer, hard disk drives can store large amounts of information and retrieve that information very quickly. Floppy disk drives also store information in magnetic particles embedded in removable disks that may be floppy or rigid. Floppy disks store less information than a hard disk drive and retrieve the information at a much slower rate. Magneto-optical disc drives store information on removable discs that are sensitive to both laser light and magnetic fields. They can typically store as much information as hard disks but have slightly slower retrieval speeds. Compact disc drives (CD-ROM) store information on pits burned into the surface of a disc of reflective material. The information stored on CD-ROMs cannot be erased or overwritten with new information. They can store about as much information as a hard drive but have a slower rate of information retrieval.

Hard disks

A hard disk consists of one or more circular platters, of which either or both surfaces are coated with a magnetic substance used for recording the data. For each surface, there is a read- write head that examines or alters the recorded data. The platters rotate on a common axis; a typical rotation speed is 3600 rotations per minute, although high-performance hard disks have higher speeds. The heads move along the radius of the platters; this movement combined with the rotation of the platters allows the head to access all parts of the surfaces.

The processor (CPU) and the actual disk communicate through a disk controller. This relieves the rest of the computer from knowing how to use

the drive, since the controllers for different types of disks can be made to use the same interface towards the rest of the computer. Therefore, the computer can say just «hey disk, give me what I want», instead of a long and complex series of electric signals to move the head of the proper location and waiting for the correct position to come under the head and doing all the other unpleasant stuff necessary. (In reality, the interface to the controller is still complex but much less than it would otherwise be.) The controller can also do some other stuff, such as caching, or automatic bad sector replacement.

The surfaces are usually divided into concentric rings, called tracks, and these in turn are divided into sectors. This division is used to specify locations on the hard disk and to allocate disk space to files. To find a given place on the hard disk, one might say: «Surface 3, track 5, sector 7». Usually the number of sectors is the same for all tracks, but some hard disks put more sectors in outer tracks (all sectors are of the same physical size, so more of them fit in the longer outer tracks). Typically, a sector will hold 512 bytes of data. The disk itself can't handle smaller amounts of data than one sector.

Each surface is divided into tracks (and sectors) in the same way. This means that when the head for one surface is on a track, the heads for the other surfaces are also on the corresponding tracks. All the corresponding tracks taken together are called a cylinder. It takes time to move the heads from one track (cylinder) to another, so by placing the data that is often accessed together (say, a file) so that it is within one cylinder, it is not necessary to move the heads to read all of it. This improves performance. It is not always possible to place files like this; files stored in several places on the disk are called fragmented.

The number of surfaces (or heads, which is the same thing), cylinders, and sectors vary a lot; the specification of the number of each is called the geometry of a hard disk. The geometry is usually stored in a special, battery-powered memory location called the CMOS RAM, from where the operating system can fetch it during boot-up or driver initialization.

1.10. Output devices

Output hardware consists of external devices that transfer information from computer's CPU to the computer user.

Display screens

The screen of a computer is often known as the monitor, or VDU (visual display unit). Inside the computer, there is a video card which processes images and sends signals to the monitor.

When choosing a monitor, you have to take into account a few basics:

Type of display – the choice is between a CRT or an LCD screen.

The Cathode Ray Tube of a monitor is similar to a traditional TV set. It has three electron guns (one for each primary colour: red, green and blue) that strike the inside of the screen, which is coated with substances called phosphors, which glow and create colours. CRTs are cheap, but they are heavy, can flicker and emit radiation.

A **Liquid Crystal Display** is made from flat plates with a liquid crystal solution between them. The crystals block the light in different quantities to create the image. Active-matrix LCDs use TFT (thin film transistor) technology, in which each pixel has its own transistor switch. They offer better quality and take up less space, so they are replacing CRTs.

Screen size – the viewing area is measured diagonally; in other words, a 17" screen measures 17 inches from the top left corner to the bottom right.

Resolution – the clarity of the image depends on the number of pixels (short for picture elements) contained on a display, horizontally and vertically. A typical resolution is 1,024 x 768. The sharpness of images is affected by dot pitch, the distance between the pixels on the screen, so a dot pitch of 0.28 mm or less will produce a sharp image.

Brightness – the luminance of images is measured in cd/m² (candela per square metre).

Colour depth – the number of colours a monitor can display. For example, a VGA monitor produces 256 colours, enough for home use; a

SuperVGA can produce up to 16.7 million colours, so is ideal for photographic work and video games.

Refresh rate – the number of times that the image is drawn each second. If a monitor has a refresh rate of 75 Hertz (Hz), it means that the screen is scanned 75 times per second. If this rate is low, you will notice a flicker, which can cause eye fatigue.

Printers

Printers take text and image from a computer and print them on paper.

A **dot-matrix printer** uses a group, or matrix, of pins to create precise dots. A print head containing tiny pins strikes an inked ribbon to make letters and graphics. This **impact printing** technology allows shops, for example, to print multi-part forms such as receipts and invoices, so it's useful when self-copying paper is needed. It has two important disadvantages: noise and a relatively low resolution (from 72 to 180 dpi).

An **ink-jet** (also called bubble-jet) **printer** generates an image by spraying tiny, precise drops of ink onto the paper. The resolution ranges from 300 to 1,200 dpi, suitable for small quantities or home use.

A standard ink-jet has a three-colour **cartridge**, plus a black cartridge. Professional ink-jets have five-colour cartridges, plus black; some can print in wide format, ranging from 60 cm up to 5 metres (e.g. for printing advertising graphics).

Some ink-jet based printers can perform more than one task. They are called **multi-function printers** because they can work as a scanner, a fax and a photocopier as well as a printer. Some units accept memory cards and print photos directly from a camera.

A **laser printer** uses a laser beam to fix the ink to the paper. A laser works like a photocopier; a powder called toner is attracted to paper by an electrostatic charge and then fused on by a hot roller.

Laser printers are fast and produce a high resolution of 1,200 to 2,400 dpi, so they are ideal for businesses and for proofing professional graphics work.

Lasers use a **page description language** or PDL which describes how to print the text and draw the images on the page. The best-known languages are Adobe PostScript and HP Printer Control Language.

A professional **imagesetter** is a typesetting printer that generates very high-resolution output (over 3,540 dpi) on paper or microfilm. It's used for high-quality publications.

A **plotter** is a special type of printer which uses ink and fine pens held in a carriage to draw detailed designs on paper. It's used in computer-aided design, maps, 3-D technical illustrations, etc.

Modern computers are designed so that a technician can change their capabilities by adding or removing components. In a typical PC, for example, many components are mounted on thin, rigid boards called circuit boards. The primary microprocessor and main memory are on a circuit board called the motherboard. Other components, such as sound and graphics coprocessors, come on circuit boards called cards. These cards can be plugged into sockets called expansion slots inside the computer. Peripheral devices, such as printers and monitors, connect by wire or cable to sockets called ports

1.11. Hardware connections

To function, hardware requires physical connections that allow components to communicate and interact. A bus provides a common interconnected system composed of a group of wires or circuitry that coordinates and moves information between the internal parts of a computer. A bus is characterized by two features: how much information it can manipulate at one time, called the bus width, and how quickly it can transfer these data.

A serial connection is a wire or set of wires used to transfer information from the CPU to an external device such as a mouse, keyboard, modem, scanner, and some types of printers. This type of connection transfers only one piece of data at a time, and is therefore slow. The advantage to using a serial connection is that it provides effective connections over long distances.

A parallel connection uses multiple sets of wires to transfer blocks of information simultaneously. Most scanners and printers use this type of connection. A parallel connection is much faster than a serial connection, but

it is limited to distances of less than 3 m between the CPU and the external device.

A bus is a physical group of signal lines that have a related function. Buses allow for the transfer of electrical signals between different parts of the computer system and thereby transfer information from one device to another. For example, the data bus is the group of signal lines that carry data between the processor and the various subsystems that comprise the computer. The “width” of a bus is the number of signal lines dedicated to transferring information. For example, an 8-bit-wide bus transfers 8 bits of data in parallel.

Microprocessors use the three- bus system architecture. The three buses are *address bus*, *the data bus*, and *the control bus*.

The data bus is bidirectional, the direction of transfer being determined by the processor. The address bus carries the address, which points to the location in memory that the processor is attempting to access. It is the job of external circuitry to determine which external device a given memory location exists in and to activate that device. This is known as address decoding. The control bus carries information from the processor about the state of the current access, such as whether it is a write or a read operation. Different processors have different control lines, but there are some control lines that are common among many processors. The control bus may consist of output signals such as read, write, valid address, etc. A processor usually has several input control lines too, such as reset, one or more interrupt lines, and a clock input.

Использованные термины

boolean value – булевское значение

bus – шина

cache hit – кэш-попадание

cell – ячейка

central processing unit – ЦПУ

circuit – цепь

erase – стирать

execute – выполнять

garbage – «мусор»
impurity – примесь
input – ввод; вводить
hardware – аппаратное обеспечение
junction – переход
keyboard – клавиатура
launch – запускать
line – строка
load – загружать
memory – память
output – вывод; выводить
performance – производительность
portable – портативный
RAM – ОЗУ – оперативное запоминающее устройство
reset – возврат в исходное положение
resolution – разрешающая способность
ROM – ПЗУ – постоянное запоминающее устройство
scroll – прокрутка
serial connection – последовательное соединение
software – программное обеспечение
storage capacity – ёмкость памяти
store – хранить
switch – переключатель
turn off – выключать
turn on – включать

Задания к текстам

1. Выпишите новые слова и, пользуясь их транскрипцией, произносите их вслух.
2. Прочитайте вслух абзацы 1 и 2.
3. Озаглавьте каждый абзац текста.
4. Ответьте на вопросы:
– интересен ли для вас прочитанный текст? Если да, то почему?

– какую незнакомую информацию вы получили при чтении этого текста?

– какие аппаратные средства вы используете на лабораторных занятиях и дома?

5. Подготовьте 2–3 вопроса по содержанию текста и задайте их во время ответа товарищей по группе.

6. Используя прочитанные тексты и информацию занятий по «Информатике», составьте сообщение «Аппаратное обеспечение» и выступите с ним на занятиях по английскому языку.

7. В Интернете найдите дополнительный материал по теме прочитанных текстов.

8. На основе составленного сообщения и дополнительного материала из Интернета подготовьте реферат «Аппаратные средства» и выступите с ним на олимпиаде по английскому языку.

9. Сформулируйте пакет данных для ответа на вопрос «Современные аппаратные средства» (вопрос реальный) на зачете по «Информатике» в конце семестра.

2. SOFTWARE

2.1. Software

System software controls the basic functions of a computer, e.g. operating systems, programming software and utility programs.

Software, computer programs – instructions that make the hardware (the machines) do work. Software as a whole can be divided into a number of categories based on the types of work done by programs. The two primary software categories are operating systems (system software), which control the workings of the computer, and application software, which addresses the multitude of tasks for which people use computers. System software thus handles such essential, but often invisible, chores as maintaining disk files and managing the screen, whereas application software performs word processing, database management, and the like.

Instructions in a computer are numbers, just like data. Different numbers, when read and executed by a processor cause different things to happen. A good analogy is the mechanism of a music box. A music box has a rotating drum with little bumps, and a row of prongs. As the drum rotates, different prongs in turn are activated by the bumps, and music is produced.

These programs are usually stored and transferred via the computer's hardware to and from the CPU. Software also governs how the hardware is utilized; for example, how information is retrieved from a storage device. The interaction between the input and output hardware is controlled by software called the Basic Input Output System software (BIOS).

Two additional categories that are neither system nor application software, although they contain elements of both, are network software, which enables groups of computers to communicate, and language software, which provides programmers with the tools they need to write programs.

In practical terms, a computer program may be just a few instructions or many millions of instructions, as do the programs for word processors and web browsers for example. A typical modern computer can execute billions of instructions per second (gigaflops) and rarely makes a mistake

over many years of operation. Large computer programs consisting of several million instructions may take teams of programmers years to write, and due to the complexity of the task almost certainly contain errors.

In most cases, computer instructions are simple: add one number to another, move some data from one location to another, send a message to some external device, etc. These instructions are read from the computer's memory and are generally carried out (executed) in the order they were given. However, there are usually specialized instructions to tell the computer to jump ahead or backwards to some other place in the program and to carry on executing from there. These are called "jump" instructions (or branches). Many computers directly support subroutines by providing a type of jump that "remembers" the location it jumped from and another instruction to return to the instruction following that jump instruction. Program execution might be likened to reading a book. While a person will normally read each word and line in sequence, they may at times jump back to an earlier place in the text or skip sections that are not of interest. Similarly, a computer may sometimes go back and repeat the instructions in some section of the program over and over again until some internal condition is met. This is called the flow control within the program and it is what allows the computer to perform tasks repeatedly without human intervention.

Comparatively, a person using a pocket calculator can perform a basic arithmetic operation such as adding two numbers with just a few button presses. But to add together all of the numbers from 1 to 1,000 would take thousands of button presses and a lot of time, with a near certainty of making a mistake. On the other hand, a computer may be programmed to do this with just a few simple instructions.

Program design of small programs is relatively simple and involves the analysis of the problem, collection of inputs, using the programming constructs within languages, devising or using established procedures and algorithms, providing data for output devices and solutions to the problem as applicable. As problems become larger and more complex, features such as subprograms, modules, formal documentation, and new paradigms such as object-oriented programming are encountered. Large programs involving

thousands of line of code and more require formal software methodologies. The task of developing large software systems presents a significant intellectual challenge.

2.2. Machine code

A sequence of instructions is a machine-code program. Each type of processor has a different instruction set, meaning that the functionality of the instructions (and the bit patterns that activate them) varies. Processor instructions are often quite simple, such as “add two numbers” or “call this function”. In some processors, however, they can be as complex and sophisticated as “if the result of the last operation was zero, then use this particular number to reference another number in memory, and then increment the first number”.

In most computers, individual instructions are stored as machine code with each instruction being given a unique number (its operation code or opcode for short). The command to add two numbers together would have one opcode, the command to multiply them would have a different opcode and so on. The simplest computers are able to perform any of a handful of different instructions; the more complex computers have several hundred to choose from, each with a unique numerical code. Since the computer’s memory is able to store numbers, it can also store the instruction codes. This leads to the important fact that entire programs (which are just lists of these instructions) can be represented as lists of numbers and can themselves be manipulated inside the computer in the same way as numeric data. The fundamental concept of storing programs in the computer’s memory alongside the data they operate on is the crux of the von Neumann, or stored program, architecture. In some cases, a computer might store some or all of its program in memory that is kept separate from the data it operates on. This is called the Harvard architecture after the Harvard Mark I computer. Modern von Neumann computers display some traits of the Harvard architecture in their designs, such as in CPU caches.

While it is possible to write computer programs as long lists of numbers (machine language) and while this technique was used with many early computers, it is extremely tedious and potentially error-prone to do so

in practice, especially for complicated programs. Instead, each basic instruction can be given a short name that is indicative of its function and easy to remember – a mnemonic such as ADD, SUB, MULT or JUMP. These mnemonics are collectively known as a computer's assembly language. Converting programs written in assembly language into something the computer can actually understand (machine language) is usually done by a computer program called an assembler.

2.3. Von Neumann architecture

Instructions are read (fetched) from memory, while data is both read from and written to memory.

This form of computer architecture is known as a Von Neumann machine, named after John Von Neumann, one of the originators of the concept. With very few exceptions, nearly all modern computers flow this form. Von Neumann computers are governed by the sequential control of a program. In other words, the computer follows a step-by-step program that governs its operation.

There are some interesting non-Von Neumann architectures, such as the massively parallel Connection Machine and the nascent efforts at building biological and quantum computers, or neural networks.

A classical Von Neumann machine has several distinguishing characteristics:

A processor can be directed to begin execution at a given point in memory, and it has no way of knowing whether the sequence of numbers beginning at that point is data or instructions. The instruction 0x4143 may also be data (the number 0x4143, or the ASCII characters "A" and "C").

The processor has no way of distinguishing between data and instruction, it will blindly execute anything that it is given, whether it is a meaningful sequence of instructions or not.

There is nothing to distinguish between a number that represents a dot of colour in an image and a number that represents a character in a text document. Meaning comes from how these numbers are treated under the execution of a program.

Data and instructions share the same memory. This means that sequences of instructions in a program may be treated as data by another program. A compiler creates a program binary by generating a sequence of numbers (instructions) in memory. To the compiler, the compiled program is just data, and it is treated as such. It is a program only when the processor begins execution. Similarly, an operating system loading an application program from disc does so by treating the sequence of instructions of that program as data. The program is loaded to memory just as an image or text file would be, and this is possible due to the shared memory space.

The processor's memory space may contain the operating system, various programs, and their associated data, all within the same linear space.

Each location in the memory space has a unique, sequential address. The address of a memory location is used to specify (and select) that location. The memory space is also known as the address space, and how that address space is partitioned between different memory and I/O devices is known as the memory map. The address space is the array of all addressable memory locations.

Modern computers based on the von Neumann architecture often have machine code in the form of an imperative programming language.

2.4. Operating systems

When a brand new computer comes off the factory assembly line, it can do nothing. The hardware needs software to make it work. Are we talking about applications software such as wordprocessing or spreadsheet software? Partly. But an applications software package does not communicate directly with the hardware. Between the applications software and the hardware is a software interface – an operating system. An operating system is a set of programs that lies between applications software and the computer hardware.

The most important program in the operating system, the program that manages the operating system, is the supervisor program, most of which remains in memory and is thus referred to as resident. The supervisor controls the entire operating system and loads into memory other operating system programs (called non-resident) from disk storage only as needed.

An operating system has three main functions: (1) manage the computer's resources, such as the central processing unit, memory, disk drives, and printers, (2) establish a user interface, and (3) execute and provide services for applications software. Keep in mind, however, that much of the work of an operating system is hidden from the user. In particular, the first listed function, managing the computer's resources, is taken care of without the user being aware of the details. Furthermore, all input and output operations, although invoked by an applications program, are actually carried out by the operating system.

The operating system typically provides a set of software tools for application programs, providing a mechanism by which they too can access the screen, disk drives, and so on. Not all embedded systems use or even need an operating system. Often, an embedded system will simply run code dedicated to its task, and the presence of an operating system is overkill. In other instances, such as network routers, an operating system provides necessary software integration and greatly simplifies the development process.

An operating system is a set of programs that control the hardware and allow people and applications to communicate with the hardware. Typical functions of the OS are handling input/output operations, running programs and organizing files on disks. The OS also gives access to networks and allows multitasking: a user can run several programs (and do various tasks) at a time. Examples are:

- the **Windows** family – designed by Microsoft and used on most PCs. When you run a program, your PC opens a window that lets you work with different tools and menus.
- **Mac OS** – created by Apple and used on Macintosh computers
- **Unix** – found on mainframes and workstations in corporate installations, as it supports multi-users
- **Linux** – developed under the GNU General Public License; anyone can copy its source code, modify and redistribute it. It is used on PCs and in appliances and small devices.

Operating systems like Linux or Windows are written in C and C++.

2.5. Linux

Linux has its roots in a student project. In 1992, an undergraduate called Linus Torvalds was studying computer science in Helsinki, Finland. Like most computer-science courses, a big component of it was taught on (and about) Unix, Unix was the wonder operating system of the 1970s and 1980s: both a textbook example of the principles of operating system design, and sufficiently robust to be the standard OS in engineering and scientific computing. But Unix was a commercial product (licensed by AT&T to a number of resellers), and cost more than a student could pay.

Annoyed by the shortcomings of Minix (a compact Unix clone written as a teaching aid by Professor Andy Tannenbaum) Linus set out to write his own ‘kernel’ – the core of an operating system that handles memory allocation, talks to hardware devices, and makes sure everything keeps running. He used the GNU programming tools-developed by Richard Stallman’s Free Software Foundation, an organisation of volunteer dedicated to fulfilling Stallman’s ideal of making good software that anyone could use without paying, When he’d written a basic kernel, he released the source code to the Linux kernel on the Internet.

Source code is important. It’s the original from which compiled programs are generated. If you don’t have the source code to a program, you can’t modify it to fix bugs or add new features. Most software companies won’t sell you their source code, because they believe that if they make it available it will destroy their revenue stream.

What happened next was astounding, from the conventional, commercial software industry point of view – and utterly predictable to anyone who knew about the Free Software Foundation. Programmers (mostly academics and students) began using Linux. They found that it didn’t do things they wanted it to do – so they fixed it. And where they improved it, they sent the improvements to Linus, who rolled them into the kernel. And Linux began to grow.

There’s a term for this model of software development; it’s called Open Source. Anyone can have the source code – it’s free. Anyone can contribute to it. If you use it heavily you may want to extend or develop or

fix bugs in it – and it is so easy to give your fixes back to the community that most people do so.

An operating system kernel on its own isn't a lot of use; but Linux was purposefully designed as a near-clone of Unix, and there is a lot of software out there that is free and was designed to compile on Linux. By about 1992, the first 'distributions' appeared.

A distribution is the Linux-user term for a complete operating system kit, complete with the utilities and applications you need to make it do useful things – command interpreters, programming tools, text editors, typesetting tools, and graphical user interfaces based on the X windowing system. X is a standard in academic and scientific computing, but not hitherto common on PCs; it's a complex distributed windowing system on which people implement graphical interfaces like KDE and Gnome.

As more and more people got to know about Linux, some of them began to port the Linux kernel to run on non-standard computers. Because it's free, Linux is now the most widely- ported operating system

2.6. Application software

In information technology, an application is a computer program designed to help people perform an activity. An application thus differs from an operating system (which runs a computer), a utility (which performs maintenance or general-purpose chores), and programming tools (with which computer programs are created). Depending on the activity for which it was designed, an application can manipulate text, numbers, graphics, or a combination of these elements. Some application packages offer considerable computing power by focusing on a single task, such as word processing; others, called integrated software, offer somewhat less power but include several applications. User-written software tailors systems to meet the user's specific needs. User-written software include spreadsheet templates, word processor macros, scientific simulations, graphics and animation scripts. Even email filters are a kind of user software. Users create this software themselves and often overlook how important it is.

The delineation between system software such as operating systems and application software is not exact, however, and is occasionally the object of controversy. For example, one of the key questions in the United States v. Microsoft antitrust trial was whether Microsoft's Internet Explorer web browser was part of its Windows operating system or a separable piece of application software. As another example, the GNU/Linux naming controversy is, in part, due to disagreement about the relationship between the Linux kernel and the operating systems built over this kernel. In some types of embedded systems, the application software and the operating system software may be indistinguishable to the user, as in the case of software used to control a VCR, DVD player or microwave oven. The above definitions may exclude some applications that may exist on some computers in large organizations.

In recent years, the term «app» has been used to exclusively refer to applications for mobile devices such as smartphones and tablets. referring to their smaller scope in relation to applications used by PCs.

Application software, also known as an **application** or an **app**, is computer software designed to help the user to perform specific tasks. Examples include enterprise software, accounting software, office suites, graphics software and media players Many application programs deal principally with documents. Applications may be bundled with the computer and its system software, or may be published separately. In recent years, the abbreviation «app» has specifically come to mean application software written for mobile devices, with the abbreviation in particular representing both the smaller size and smaller scope of the software (i.e. an app whose sole purpose is to display an image representation of the current weather).

Application software is contrasted with system software and middleware, which manage and integrate a computer's capabilities, but typically do not directly apply in the performance of tasks that benefit the user. The system software serves the application, which in turn serves the user.

Similar relationships apply in other fields. For example, a shopping mall does not provide the merchandise a shopper is seeking, but provides

space and services for retailers that serve the shopper. A bridge may similarly support rail tracks which support trains, allowing the trains to transport passengers.

Application software applies the power of a particular computing platform or system software to a particular purpose. Some applications are available in versions for several different platforms.

Application software classification

Application software falls into two general categories; horizontal applications and vertical applications. Horizontal applications are the most popular and widespread in departments or companies. Vertical applications are niche products, designed for a particular type of business or division in a company.

There are many types of application software:

- An **application suite** consists of multiple applications bundled together. They usually have related functions, features and user interfaces, and may be able to interact with each other, e.g. open each other's files. Business applications often come in suites, e.g. Microsoft Office, OpenOffice.org, which bundle together a word processor, a spreadsheet, etc.; but suites exist for other purposes, e.g. graphics or music.
- **Enterprise software** addresses the needs of organization processes and data flow, often in a large distributed environment. (Examples include financial systems, customer relationship management (CRM) systems and supply chain management software). Departmental Software is a sub-type of enterprise software with a focus on smaller organizations or groups within a large organization. (Examples include travel expense management and IT Helpdesk.)
- Enterprise infrastructure software provides common capabilities needed to support enterprise **software systems**. (Examples include databases, email servers, and systems for managing networks and security.)

- **Information worker** software lets users create and manage information, often for individual projects within a department, in contrast to enterprise management. Examples include time management, resource management, documentation tools, analytical, and collaborative. Word processors, spreadsheets, email and blog clients, personal information system, and individual media editors may aid in multiple information worker tasks.
- Content access software is used primarily to access content without editing, but may include software that allows for content editing. Such software addresses the needs of individuals and groups to consume digital entertainment and published digital content. (Examples include **media players**, **web browsers**, and help browsers.)
- **Educational software** is related to content access software, but has the content and/or features adapted for use in by educators or students. For example, it may deliver evaluations (tests), track progress through material, or include collaborative capabilities.
- **Simulation software** simulates physical or abstract systems for either research, training or entertainment purposes.
- **Media development** software generates print and electronic media for others to consume, most often in a commercial or educational setting. This includes graphic-art software, desktop publishing software, multimedia development software, HTML editors, digital-animation editors, digital audio and video composition, and many others.
- Product engineering software is used in developing hardware and software products. This includes computer-aided design (CAD), computer-aided engineering (CAE), computer language editing and compiling tools, integrated development environments, and application programmer interfaces.

Applications can also be classified by computing platform such as a particular operating system, delivery network such as in cloud computing and Web 2.0 applications, or delivery devices such as mobile apps for mobile devices.

The operating system itself can be considered application software when performing simple calculating, measuring, rendering, and word processing tasks not used to control hardware via command-line interface or graphical user interface. This does not include application software bundled within operating systems such as a software calculator or text editor.

2.7. Database systems

Database systems were born and have evolved as an application technology due to the necessity for managing the large amount of data produced in the real world. However, it was soon recognized that the emergence of the technology is one of the most significant features of transition in computer application from data processing to information processing and further to knowledge processing. The problem so far has been involving various topics: data models, database languages and query processing, database design, database system design, file organization, database system evaluation, integrity, database machine, distributed database system, high level database applications and so on.

Database systems were the means by which computer technology began to make effective and systematic use of a permanent store, which has been an important feature of information processing capability belonging only to human beings. In this sense, the emergence of database technology is probably a revolutionary development in the world of information processing by computers. It made computers more like human beings than ever and offered us a chance to reconsider the information processing by computers in comparison with that of the human beings. It is expected that analyzing the problem solving process and creative activity by man will serve us in designing future information processing systems.

Knowledge representation has also become a crucial issue in the field of artificial intelligence. In fact, whichever system we consider, how to represent knowledge and then utilize it on a computer is a key problem for the realization of advanced information system such as natural language processing, image or speech understanding, machine vision, intelligent information retrieval, and intelligent man-machine communication.

Использованные термины

Appliance – бытовое устройство
application suite – комплект приложений
assembler – язык Ассемблер
branch – переход
character – буква; знак
computer-aided engineering (CAE) – автоматизированная обработка
computer-science – информатика
embedded system – встроенная система
graphical user interface – графический пользовательский интерфейс
imperative language – императивный язык
«jump» instruction – команда перехода
middleware – промежуточное программное обеспечение
open Source – открытый код
query processing – обработка запроса
rotating drum – вращающийся барабан
spreadsheet – табличная программа
step-by-step program – пошаговая программа
subroutine – подпрограмма
tablet – графический планшет
teaching aid – учебное пособие
Van Neumann architecture – фоннеймановская архитектура
VCR – видеомаягнитофон

Задания к текстам

1. Выпишите новые слова и, пользуясь их транскрипцией, произнесите их вслух.
2. Прочитайте вслух абзацы 1 и 2.
3. Озаглавьте каждый абзац текста.
4. Ответьте на вопросы:
 - интересен ли для вас прочитанный текст? Если да, то почему?
 - какую незнакомую информацию вы получили при чтении этого текста?
 - какие аппаратные средства вы используете на лабораторных занятиях и дома?
5. Подготовьте 2–3 вопроса по содержанию текста и задайте их во время ответа товарищей по группе.
6. Используя прочитанные тексты и информацию занятий по «Информатике», составьте сообщение «Аппаратное обеспечение» и выступите с ним на занятиях по английскому языку.
7. В Интернете найдите дополнительный материал по теме прочитанных текстов.
8. На основе составленного сообщения и дополнительного материала из Интернета подготовьте реферат «Аппаратные средства» и выступите с ним на олимпиаде по английскому языку.
9. Сформулируйте пакет данных для ответа на вопрос «Современные аппаратные средства» (вопрос реальный) на зачете по «Информатике» в конце семестра.

3. PROGRAMMING

3.1. Computer programming

Computer programming (often shortened to programming, scripting, or coding) is the process of designing, writing, testing, debugging, and maintaining the source code of computer programs. This source code is written in one or more programming languages (such as Java, C++, C#, Python, etc.). The purpose of programming is to create a set of instructions that computers use to perform specific operations or to exhibit desired behaviours. The process of writing source code often requires expertise in many different subjects, including knowledge of the application domain, specialized algorithms and formal logic.

From the moment you turn on your computer, it is running programs, carrying out instructions, testing your RAM (Random-access memory), resetting all attached devices and loading the operating system from hard disk or CD-ROM.

Each and every operation that your computer performs has instructions that someone had to write in a programming language. These had to be created, compiled and tested- a long and complex task. An operating system like Microsoft's Windows Vista took millions of man hours to write and test the software.

There have been many attempts to automate this process, and have computers write computer programs but the complexity is such that for now, humans still write the best computer programs.

To write a program, software developers usually follow these steps:

- First they try to understand the problem and define the purpose of the program.
- They design a **flowchart**, a diagram which shows the successive logical steps of the program.
- Next they write the instructions in a high-level language (Pascal, C, etc.). This is called **coding**. The program is then **compiled**.

- When the program is written, they **test** it: they run the program to see if it works and use special tools to detect **bugs**, or errors. Any errors are corrected until it runs smoothly. This is called **debugging**, or bug fixing.
- Finally, software companies write a detailed description of how the program works, called **program documentation**. They also have a **maintenance** program. They get reports from users about any errors found in the program. After it has been improved, it is published as an updated version.

Within software engineering, programming (the implementation) is regarded as one phase in a software development process.

As time has progressed, computers have made giant leaps in the area of processing power. This has brought about newer programming languages. Popular programming languages of the modern era include C++, C#, Visual Basic, SQL, HTML with PHP, Perl, Java, JavaScript, Python and dozens more. Although these high-level languages usually incur greater overhead, the increase in speed of modern computers has made the use of these languages much more practical than in the past. These increasingly abstracted languages typically are easier to learn and allow the programmer to develop applications much more efficiently and with less source code. However, high-level languages are still impractical for a few programs, such as those where low-level hardware control is necessary or where maximum processing speed is vital. Computer programming has become a popular career in the developed world, particularly in the United States, Europe, Scandinavia, and Japan. Due to the high labor cost of programmers in these countries, some forms of programming have been increasingly subject to offshore outsourcing (importing software and services from other countries, usually at a lower wage), making programming career decisions in developed countries more complicated, while increasing economic opportunities for programmers in less developed areas, particularly China and India.

Quality requirements

Whatever the approach to software development may be, the final program must satisfy some fundamental properties. The following properties are among the most relevant:

- Reliability: how often the results of a program are correct.
- Robustness: how well a program anticipates problems not due to programmer error.
- Usability: the ergonomics of a program: the ease with which a person can use the program for its intended purpose, or in some cases even unanticipated purposes.
- Portability: the range of computer hardware and operating system platforms on which the source code of a program can be compiled/interpreted and run.
- Maintainability: the ease with which a program can be modified by its present or future developers in order to make improvements or customizations, fix bugs and security holes, or adapt it to new environments.
- Efficiency/performance: the amount of system resources a program consumes (processor time, memory space, slow devices such as disks, network bandwidth and to some extent even user interaction): the less, the better.

Readability of source code

In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code. It affects the aspects of quality above, including portability, usability and most importantly maintainability.

Readability is important because programmers spend the majority of their time reading, trying to understand and modifying existing source code, rather than writing new source code. Unreadable code often leads to bugs, inefficiencies, and duplicated code.

Methodologies

The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). There exist a lot of differing approaches for each of those tasks. One approach popular for requirements analysis is Use Case analysis. Nowadays many programmers use forms of Agile software development where the various stages of formal software development are more integrated together into short cycles that take a few weeks rather than years. There are many approaches to the Software development process.

Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.

A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).

Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.

Debugging

Debugging is a very important task in the software development process, because an incorrect program can have significant consequences for its users. Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages. Use of a static code analysis tool can help detect some possible problems.

Debugging is often done with IDEs like Eclipse, Kdevelop, NetBeans, Code::Blocks, and Visual Studio. Standalone debuggers like gdb are also used, and these often provide less of a visual environment, usually using a command line.

3.2. Object-oriented programming

One of the principal motivations for using OOP is to handle multimedia applications in which such diverse data types as sound and video can be packaged together into executable modules. Another is writing program code that's more intuitive and reusable; in other words, code that shortens program-development time.

Perhaps the key feature of OOP is encapsulation – bundling data and program instructions into modules called ‘objects’. Here’s an example of how objects work. An icon on a display screen might be called ‘Triangles’. When the user selects the Triangles Icon – which is an object composed of the properties of triangles and other data and Instructions – a menu might appear on the screen offering several choices. The choices may be (1) create a new triangle and (2) fetch a triangle already in storage. The menu, too, is an object, as are the choices on it. Each time a user selects an object, instructions inside the object are executed with whatever properties or data the object holds, to get to the next step. For instance, when the user wants to create a triangle, the application might execute a set of instructions that displays several types of triangles – right, equilateral, isosceles, and so on.

Many industry observers feel that the encapsulation feature of OOP is the natural tool for complex applications in which speech and moving images are integrated with text and graphics. With moving images and voice built into the objects themselves, program developers avoid the sticky problem of deciding how each separate type of data is to be integrated and synchronized into a working whole.

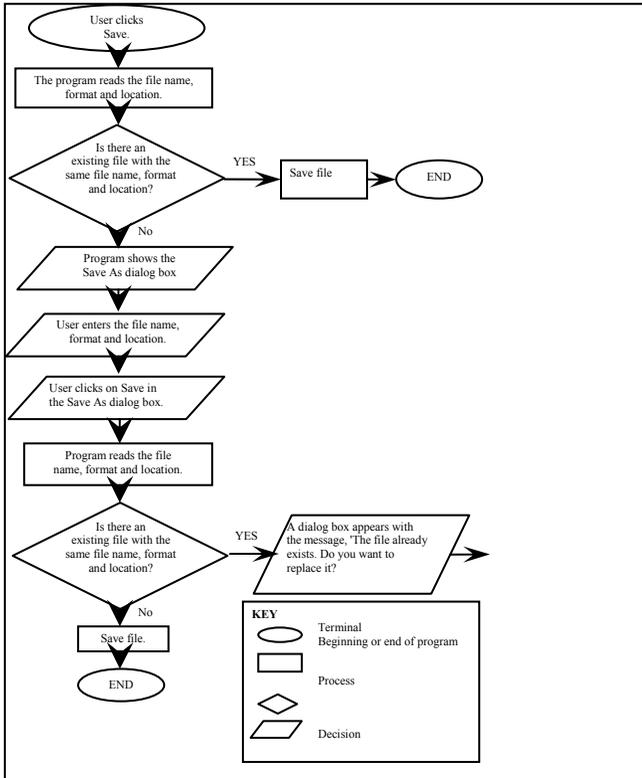
A second key feature of OOP is inheritance. This allows OOP developers to define one class of objects, say ‘Rectangles’, and a specific instance of this class, say ‘Squares’ (a rectangle with equal sides). Thus, all properties of rectangles – ‘Has 4 sides’ and ‘Contains 4 right angles’ are the two shown here – are automatically inherited by Squares. Inheritance is a useful property in rapidly processing business data. For instance, consider a business that has a class called ‘Employees at the Dearborn Plant’ and a specific instance of this class, ‘Welders’. If employees at the Dearborn plant are eligible for a specific benefits package, welders automatically qualify for the package. If a welder named John Smith is later relocated from

Dearborn to Birmingham, Alabama, where a different benefits package is available, revision is simple. An icon representing John Smith – such as John Smith’s face – can be selected on the screen and dragged with a mouse to an icon representing the Birmingham plant. He then automatically ‘inherits’ the Birmingham benefit package.

A third principle behind OOP is polymorphism. This means that different objects can receive the same instructions but deal with them in different ways. For instance, consider again the triangles example. If the user right clicks the mouse on ‘Right triangle’, a voice clip might explain the properties of right triangles. However, if the mouse is right clicked on ‘Equilateral triangle’ the voice instead explains properties of equilateral triangles.

The combination of encapsulation, inheritance and polymorphism leads to code reusability. ‘Reusable code’ means that new programs can easily be copied and pasted together from old programs. All one has to do is access a library of objects and stitch them into a working whole. This eliminates the need to write code from scratch and then debug it. Code reusability makes both program development and program maintenance faster.

The diagram shows part of a simple algorithmic flow chart for the Save command in a computer program. An algorithm is a set of logical rules that we use to solve a problem. Computer programmers often use algorithms to plan their programs, but the only language a computer understands without translation is machine language. This uses the binary system of 1 and 0, which matches the electrical positions ‘on’ and ‘off’. We can also show these numbers in English by Yes/No or True/False.



3.3. Programming languages

Programming languages provide various ways of specifying programs for computers to run. Unlike natural languages, programming languages are designed to permit no ambiguity and to be concise. They are purely written languages and are often difficult to read aloud. They are generally either translated into machine code by a compiler or an assembler before being run, or translated directly at run time by an interpreter. Sometimes programs are executed by a hybrid method of the two techniques.

A programming language is used to write computer programs such as:

- Applications
- Utilities

- Servers
- Systems Programs

A program is written as a series of human understandable computer instructions that can be read by a compiler and linker, and translated into machine code so that a computer can understand and run it.

Different programming languages support different styles of programming (called programming paradigms). The choice of language used is subject to many considerations, such as company policy, suitability to task, availability of third-party packages, or individual preference. Ideally, the programming language best suited for the task at hand will be selected. Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute. Languages form an approximate spectrum from “low-level” to “high-level”; “low-level” languages are typically more machine-oriented and faster to execute, whereas “high-level” languages are more abstract and easier to use but execute less quickly. It is usually easier to code in “high-level” languages than in “low-level” ones.

Low-level languages

Machine languages and the assembly languages that represent them (collectively termed low-level programming languages) tend to be unique to a particular type of computer. For instance, an ARM architecture computer (such as may be found in a PDA or a hand-held videogame) cannot understand the machine language of an Intel Pentium.

The only language a PC can directly execute is machine code, which consists of 1s and 0s. This language is difficult to write, so we use symbolic languages that are easier to understand. For example, **assembly languages** use abbreviations such as ADD, SUB, MPY to represent instructions. The program is then translated into machine code by software called an **assembler**.

Machine code and assembly languages are called low-level languages because they are closer to the hardware.

Higher-level languages

Though considerably easier than in machine language, writing long programs in assembly language is often difficult and is also error prone. Therefore, most practical programs are written in more abstract high-level programming languages that are able to express the needs of the programmer more conveniently (and thereby help reduce programmer error). High level languages are usually "compiled" into machine language (or sometimes into assembly language and then into machine language) using another computer program called a compiler. High level languages are less related to the workings of the target computer than assembly language, and more related to the language and structure of the problem(s) to be solved by the final program. It is therefore often possible to use different compilers to translate the same high level language program into the machine language of many different types of computer. This is part of the means by which software like video games may be made available for different computer architectures such as personal computers and various video game consoles.

High-level languages, however, are closer to human languages; they use forms resembling English, which makes programming easier. The program is translated into machine code by software called a **compiler**. Some examples are:

- FORTRAN – used for scientific and mathematical applications
- COBOL – popular for business applications
- BASIC – used as a teaching language; Visual BASIC is now used to create Windows applications
- C – used to write system software, graphics and commercial programs
- Java – designed to run on the Web; Java applets are small programs that run automatically on web pages and let you watch animated characters, and play music and games.

The languages used to create Web documents are called **markup languages**; they use instructions (markups) to format and link text files. Examples are:

- HTML – the code used to create Web pages
- VoiceXML – it makes Internet content accessible via speech recognition and phone. Instead of using a web browser on a PC, you use a telephone to access voice-equipped websites. You just dial the phone number of the website and then give spoken instructions, commands, and get the required information.

More advanced techniques, for instance using Objects, Generics, Multi-threading mean that the modern programming languages are far more powerful.

As computers get faster, have more RAM, applications will get more complex, it is likely that more development will shift from C++ to the higher level languages such as Java and C#.

Microsoft have put a lot of faith in C# as their answer to Java and have the financial leverage to continue plugging it for a very long time. We expect both Java and C# to become the two dominant programming languages.

3.4. Drupal as a CMS

Ever since its advent in the web arena, Drupal CMS has evolved constantly to emerge as one of the most complex content management system. It's acceptability and popularity can be mapped by the fact that several top notch sites are using the CMS as their support system- The Economist, The White House, MIT, Harvard, Popular Science and Sony Music are to name a few. As a matter of fact, today 538,813 people are using Drupal in 228 countries across the world in 182 available languages.

For those who are new to Drupal but have been using Wordpress for a while will find some striking similarities between the two content management systems. Though, because of its interactive and sophisticated programming interface, drupal is considered more than a simple CMS tool- it is viewed as an end-to-end development solution. The system in drupal showcases advanced admin options that are supported by in-depth report generating tools. The basic use of the platform doesn't require any technical whereabouts, but it is put more into use as you hire Drupal developers and advanced level enterprises.

While Drupal has emerged as one of the most user-friendly CMS platforms, it can be too much to handle for some users. This post explores the various advantages and disadvantages that Drupal poses as a content management system.

Advantages:

- Drupal incorporates all basic features that you may need to manage content effectively. The Core version of Drupal includes basic utility features such as RSS- feeds, user registration, page layout customization, menu management and system administration.
- Drupal offers you great modules that can be used as the building blocks of your website, enabling you create and manage content and rules quickly. Using the in- built themes and templates offered by Drupal, you need not start building a site from scratch, irrespective of how intricate your development venture may be.
- Drupal facilitates easy organization of content. It lets you create custom lists, organize content through path URLs, create defaults and associate content. This makes organizing, searching, resusing and managing content simplified.
- The 7000+ plugins of Drupal allow you to add new capabilities to your development forte. The plugins are free to download and use, which means you, can use as many plugins as you feel like to enhance the functionalities of your Drupal portal.

Disadvantages:

- Drupal might offer great power and functionalities but it is not as user-friendly as other CMS available in the web arena, because of its high learning curve. CMS like Wordpress and Joomla may offer better usability but they lack the power of Drupal.
- In terms of performance, Drupal lags behind Wordpress in two aspects- scalability and loading. The expanded breadth of Drupal functionalities is to be blamed for the slower speed of the platform.

Moreover, the significantly large learning curve is quite time consuming at times.

- The backward compatibility feature is lacking in the Drupal CMS. This means, if you are used to some other content system or program, then Drupal might not be the right CMS choice.
- Considering both the advantages and disadvantages of the Drupal CMS, it can be said that the CMS is a perfect choice for those looking for powerful features, to enhance the functioning and usability of your next content-driven website.

Использованные термины

ambiguity – двусмысленность

availability – доступность

backward compatibility – совместимость сверху-вниз

content management system (CMS) – система управления контентом

debugger – программа отладки

efficiency – эффективность

Entity – Relationship Modeling – моделирование отношений между логическими объектами

flow – chart – графическая схема программы

IDE – интегрированная среда разработки

image – изображение

linker – компоновщик

maintainability – удобство эксплуатации

offshore out-sourcing – передача заказов за границу

overhead – непроизводительные затраты

plugin – программа-вставка

portability – мобильность

processing power – вычислительная мощность

readability – читаемость

reliability – надёжность

robustness – прочность

target computer – целевой компьютер
triangle – треугольник
software engineering – разработка программного обеспечения
Unified Modeling Language (UML) – унифицированный язык моделирования
usability – применимость

Задания к текстам

1. Найдите в тексте интернациональные слова и переведите их, не пользуясь словарем.
2. Выпишите новые слова в свой словарь и, пользуясь их транскрипцией, произносите их вслух.
3. Внимательно прочитайте текст и определите его основную информацию; сделайте об этом сообщение на занятии.
4. Ответьте на вопросы:
 - какую новую информацию в области программирования вы получили, прочитав этот текст?
 - какие средства программирования, упомянутые в тексте, вы используете на лабораторных занятиях ?
5. Подготовьте 2–3 вопроса по содержанию текста и задайте их во время ответа товарищей по группе.
6. Выйдите в Интернет; найдите дополнительную информацию по программированию; сделайте о ней сообщение на занятии.
7. Используя информацию из прочитанных текстов и интернет-информацию, а также материалы лекций по «Информатике», подготовьте ответ на вопрос «Современные программные средства» (вопрос реальный) на зачете/экзамене по «Информатике» в конце семестра.
8. Используя всю полученную информацию, напишите реферат «Языки программирования» для участия в научно-технической конференции на иностранных языках в конце семестра (при получении призового места у вас будет «автомат» на экзамене по английскому языку).

4. NETWORKS

4.1. Networks

A bridge is a hardware and software combination used to connect the same type of networks. Bridges can also partition a large network into two smaller ones and connect two LANs that are nearby each other.

A router is a special computer that directs communicating messages when several networks are connected together. High-speed routers can serve as part of the Internet backbone.

A gateway is an interface that enables dissimilar networks to communicate such as two LANs based on different topologies or network operating systems.

A backbone is the main transmission path, handling the major data traffic, connecting different LANs together.

A LAN is a network contained within a small area, for example a company department.

A modem is a device for converting digital signals to analogue signals and vice versa to enable a computer to transmit and receive data using an ordinary telephone line.

4.2. The Internet

The Internet, a global computer network which embraces millions of users all over the world, began in the United States in 1969 as a military experiment. It was designed to survive a nuclear war. Information sent over the Internet takes the shortest path available from one computer to another. Because of this, any two computers on the Internet will be able to stay in touch with each other as long as there is a single route between them. This technology is called packet switching. Owing to this technology, if some computers on the network are knocked out (by a nuclear explosion, for example), information will just route around them. One such packet-switching network already survived a war. It was the Iraqi computer network which was not knocked out during the Gulf War.

The most popular Internet service is e-mail. Most of the people who have access to the Internet, use the network only for sending and receiving e-mail messages. However, other popular services are available on the Internet: reading USENET News, using the World-Wide Web, telnet, FTP, and Gopher.

In many developing countries the Internet may provide businessmen with a reliable alternative to the expensive and unreliable telecommunications systems of these countries.

To use the Internet you basically need a computer, the right connection software and a modem to connect your computer to a telephone line and then access your ISP (Internet Service Provider).

The modem (modulator-demodulator) converts the digital signals stored in the computer into analogue signals that can be transmitted over telephone lines. There are two basic types: external with a cable that is plugged into the computer via a USB port, and internal, an expansion card inside the computer. A PC card modem is a different, more versatile option for laptops and mobile phones.

At first most computers used a dial-up telephone connection that worked through the standard telephone line. Now a broadband connection, a high data transmission rate Internet connection, has become more popular: either ADSL (Asymmetric Digital Subscriber Line), which allows you to use the same telephone line for voice and fast access to the Internet, or cable, offered by most TV cable providers.

The basic equipment has changed drastically in the last few years. You no longer need a computer to use the Internet. Web TV provides email and access to the Web via a normal TV set plus a high-speed modem. More recently, 3Generation mobile phones and PDAs, personal digital assistants, also allow you to go online with wireless connections, without cables.

Telephone lines are not essential either. Satellites orbiting the earth enable your computer to send and receive Internet files. Finally, the power-line Internet, still under development, provides access via a power plug.

The Internet consists of many systems that offer different facilities to users:

- WWW, the World Wide Web, a collection of files or pages containing links to other documents on the Net. It's by far the most popular system. Most Internet services are now integrated on the Web.
- Email, or electronic mail, for the exchange of messages and attached files.
- Mailing lists (or listservs) based on programs that send messages on a certain topic to all the computers whose users have subscribed to the list.
- Chat and instant messaging, for real-time conversations; you type your messages on the keyboard.
- Internet telephone, a system that lets people make voice calls via the Internet. Video conference, a system that allows the transmission of video and audio signals in real time so the participants can exchange data, talk and see one another on the screen.
- File Transfer Protocol (FTP), used to transfer files between computers.
- Newsgroups, where people send, read and respond to public bulletin board messages stored on a central computer.
- TELNET, a program that enables a computer to function as a terminal working from a remote computer and so use online databases or library catalogues.

4.3. Internet domain

The Internet is a network of thousands of independent networks, each network consisting of several million "host" computers. If communication between two computers is required, they must be able to identify one another. Therefore, Internet users desiring to establish a connection to another computer connected to the Internet require a fixed address in order to establish the connection with such computer and to access the services it provides. For the purpose of clear and unequivocal identification on a global

basis on the Internet, each and every computer connected to the Internet has a numerical IP (Internet Protocol) address, such as the numbers 129.18.10.25. It rapidly became evident that, while the use of numerical addresses is an appropriate means of communication for computers, people prefer names to numbers. Thus a system was developed that assigns names to the computers connected to the Internet, in addition to IP addresses. The so-called “domain name system” (DNS) evolved, a global unequivocal and logical naming system assigning to each and every computer connected to the Internet a hierarchically structured name.

Each individual domain name consists of several levels (domain levels), which are separated by full stops. For instance, in the name Mercedes.com, Daimler-Benz is the second-level domain which can be chosen freely by the user. The abbreviation com designates the top-level domain, which is an administrative unit with the purpose of dividing the name space into different categories and delegating it to different registration locations.

Whereas trademark law permits the use of the same mark by different companies, provided they do not operate in competing business fields and/or in the same geographical areas, due to its function as an address on the Internet a domain name can only be assigned once under one and the same TLD.

As a result, numerous companies have been prevented from using their trademark or business designation as a domain name if the mark was already registered in the .com or country code TLD. Therefore, in order to enable more companies to register their own marks as domain names, in future additional TLDs shall be created.

The newly defined gTLDs read as follows:

- *firm* for business or farms;
- *store* for business offering goods;
- *web* for entities concentrating on activities related to the World Wide web;
- *arts* for entities emphasizing cultural and entertainment activities;
- *Rec* for entities emphasizing recreation/entertainment activities.

4.4. Fast Ethernet

In computer networking, Fast Ethernet is a collective term for a number of Ethernet standards that carry traffic at the nominal rate of 100 Mbit/s, against the original Ethernet speed of 10 Mbit/s. Of the Fast Ethernet standards 100BASE-TX is by far the most common and is supported by the vast majority of Ethernet hardware currently produced. Fast Ethernet was introduced in 1995 and remained the fastest version of Ethernet for three years before being superseded by gigabit Ethernet.

Fast Ethernet is an extension of the existing Ethernet standard. It runs on UTP data or optical fiber cable and uses CSMA/CD in a star wired bus topology, similar to 10BASE-T where all cables are attached to a hub. And, it provides compatibility with existing 10BASE-T systems and thus enables plug-and-play upgrades from 10BASE-T. Fast Ethernet is sometimes referred to as 100BASE-X where X is a placeholder for the FX and TX variants.

The 100 in the media type designation refers to the transmission speed of 100 Mbit/s. The «BASE» refers to baseband signaling. The TX, FX and T4 refer to the physical medium that carries the signal.

A Fast Ethernet adapter can be logically divided into a Media Access Controller (MAC) which deals with the higher level issues of medium availability and a Physical Layer Interface (PHY). The MAC may be linked to the PHY by a 4 bit 25 MHz synchronous parallel interface known as a Media Independent Interface (MII) or a 2 bit 50 MHz variant Reduced Media Independent Interface (RMII). Repeaters (hubs) are also allowed and connect to multiple PHYs for their different interfaces.

The MII may (rarely) be an external connection but is usually a connection between ICs in a network adapter or even within a single IC. The specs are written based on the assumption that the interface between MAC and PHY will be a MII but they do not require it.

The MII fixes the theoretical maximum data bit rate for all versions of Fast Ethernet to 100 Mbit/s. The data signaling rate actually observed on real networks is less than the theoretical maximum, due to the necessary header and trailer (addressing and error-detection bits) on every frame, the

occasional «lost frame» due to noise, and time waiting after each sent frame for other devices on the network to finish transmitting.

4.5. TCP/IP

The Internet is an International computer Network made up of thousands of networks linked together. All these computers communicate with one another; they share data, resources, transfer information, etc. To do it they need to use the same language or protocol: TCP / IP (Transmission Control Protocol / Internet Protocol) and every computer is given an address or IP number. This number is a way to identify the computer on the Internet.

At the heart of the Internet Protocol (IP) portion of TCP/IP is a concept called the Internet address. This 32-bit coding system assigns a number to every node on the network. There are various types of addresses designed for networks of different sizes, but you can write every address with a series of numbers that identify the major network and the sub-networks to which a node is attached. Besides identifying a node, the address provides a path that gateways can use to route information from one machine to another.

Although data-delivery systems like Ethernet or X.25 bring their packets to any machine electrically attached to the cable, the IP modules must know each other's Internet addresses if they are to communicate. A machine acting as a gateway connecting different TCP/IP networks will have a different Internet address on each network. Internal look-up tables and software based on another standard – called Resolution Protocol – are used to route the data through a gateway between networks.

Another piece of software works with the IP-layer programs to move information to the right application on the receiving system. This software follows a standard called the User Datagram Protocol (UDP). You can think of the UDP software as creating a data address in the TCP/IP message that states exactly what application the data block is supposed to contact at the address the IP software has described. The UDP software provides the final routing for the data within the receiving system.

The Transmission Control Protocol (TCP) part of TCP/IP comes into operation once the packet is delivered to the correct Internet address and

application port. Software packages that follow the TCP standard run on each machine, establish a connection to each other, and manage the communication exchanges. A data-delivery system like Ethernet doesn't promise to deliver a packet successfully. Neither IP nor UDP knows anything about recovering packets that aren't successfully delivered, but TCP structures and buffers the data flow, looks for responses and takes action to replace missing data blocks. This concept of data management is called reliable stream service.

After TCP brings the data packet into a computer, other high-level programs handle it. Some are enshrined in official US government standards, like the File Transfer Protocol (FTP) and the Simple Mail Transfer Protocol (SMTP). If you use these standard protocols on different kinds of computers, you will at least have ways of easily transferring files and other kinds of data.

Conceptually, software that supports the TCP protocol stands alone. It can work with data received through a serial port, over a packet-switched network, or from a network system like Ethernet. TCP software doesn't need to use IP or UDP, it doesn't even have to know they exist. But in practice TCP is an integral part of the TCP/IP picture, and it is most frequently used with those two protocols.

4.6. The World Wide Web

Tim Berners-Lee invented the World Wide Web (WWW) when he discovered a way to jump to different files on his computer using the random, or unplanned, links between them. He then wrote a simple coding system, called HTML (Hyper Text Markup Language), to create links to files on any computer connected to the network. This was possible because each file had an individual address, or URL (Uniform Resource Locator). He then used a set of transfer rules, called HTTP (Hyper Text Transfer Protocol), to link Web files together across the Internet. Berners-Lee also invented the world's first browser. This lets you locate and view Web pages and also navigate from one link to another.

The WWW became available to everyone in 1991 and the number of Internet users grew from 600,000 to 40 million in five years. Today, that

number is much larger and there are now many browsers that provide Web pages, information and other services. You can also do research, download music files, play interactive games, shop, talk in chat rooms and send and receive e- mail on the WWW.

What the Web is

The World Wide Web, Web or WWW is a network of documents that works in a hypertext environment, i.e. using text that contains links, *hyperlinks* to other documents.

The files, *web pages*, are stored in computers, which act as servers. Your computer, the client, uses a *web browser*, a special program to access and download them. The *web pages* are organized in *websites*, groups of pages located on the Web, maintained by a *webmaster*, the manager of a website.

The Web enables you to post and access all sorts of interactive multimedia information and has become a real *information highway*.

How to surf the Web

To surf or navigate the Web, access and retrieve web pages or websites, you need a computer with an Internet connection and a web browser. After you have launched it, you must type the website address or **URL (Uniform Resource Locator)**, which may look like this:

<http://www.cup.org/education/sample.htm>

<http://> indicates the type of **protocol** that the server and browser will use to communicate. Here it is Hypertext Transfer Protocol.

[www.](http://www.cup.org/education/sample.htm) shows that it is a resource on the World Wide Web.

[cup.org](http://www.cup.org/education/sample.htm) is the **domain name** of the web server that hosts the website.

[education](http://www.cup.org/education/sample.htm) is the **path**, the place where a web page is located.

[sample.htm](http://www.cup.org/education/sample.htm) is the **filename** or name of a single web page.

The different parts are separated by full stops (.) and forward slashes (/). When we say a URL, we say dot (.) and slash (/).

To find interesting sites you can use **search engines**, where the website information is compiled by **spiders**, computer-robot programs that collect information from sites by using keywords, or through **web indexes**, subject directories that are selected by people and organized into hierarchical subject categories. Some **web portals** – websites that offer all

types of services, e.g. email, forums, search engines, etc. – are also good starting points.

The most relevant website addresses can be stored in your computer using the **bookmarks** or **favourites** function in your **browser**.

Websites usually have a beginning page or home page. From this starting point you can navigate by clicking your mouse on hyperlinks in texts or images.

What you can do on the Web

The Web is an open door to a universe of multimedia resources that people use in many different ways. Here are just a few:

“In my **weblog**, an electronic journal I maintain on the Web, you can read and post opinions in chronological order. In my role as **blogger**, the manager of a **blog**. I can promote this new type of discussion.”

“**E-learning**, education via the Web, is a great opportunity for people like me who haven’t got time to attend classes.”

“Online shopping, i.e. cybershopping or e-commerce, saves you time and gives you the comfort of buying from your personal computer. The goods are then sent to you, so it’s very easy.”

4.7. Wireless networking

Wireless (WiFi) networks are just like fixed LANs but instead of using cables, devices are linked by radio waves.

Each computer in a wireless network requires a wireless network interface card (NIC). These can be built in or you can use plug-in adaptors. These allow each component in the network communicate with a wireless access point (AP) to create a wireless local area network (WLAN). The AP operates like a router in a fixed LAN. It also provides a bridge which plugs into the hub of a fixed LAN allowing both fixed and wireless users to talk to each other. If your LAN is connected to the Internet, the WLAN can also use it. If not, you can connect the WLAN to the Internet via an ADSL or cable modem.

What are the advantages of a wireless network? You don’t need cabling. In older buildings, it can be expensive to install cables and access points. With WiFi, one access point can cover an entire floor or even a

building. You can work anywhere within range of the access point. On a sunny day, you could work outside. You can make any room in the house your study. There are now WiFi hotspots in hotels, libraries and airports so you can link to a network away from home or your office.

There are disadvantages. Fixed LANs can run at 1000 Mbps. Wireless networks are much slower and the further you are from an access point, the slower the rate. Although there are savings on the cost of cabling, wireless NICs are more expensive than the wired versions. Then there is the problem of interference, if a neighbour is using the same channel, and security. Other users may be able to intercept your data. Encryption programs like Wired Equivalent Privacy (WEP) can help.

4.8. Bluetooth

When you use computers, entertainment systems or telephones, the various pieces and parts of the systems make up a community of electronic devices. These devices communicate with each other using a variety of wires, cables, radio signals and infrared light beams, and an even greater variety of connectors, plugs and protocols.

The art of connecting things is becoming more and more complex every day. In this article, we will look at a completely different way to form the connections, called Bluetooth. Bluetooth is wireless and automatic, and has a number of interesting features that can simplify our daily lives.

Bluetooth is a low-power, short range wireless networking standard designed for local area voice and data communications. Mobile computers, mobile phones and headsets, PDAs and PCs, will all exchange information using the specification agreed to by the over 2 400 companies in the Bluetooth Special Interest Group (SIG). The SIG companies are working together to ensure interoperability between products and include some of the top brands in wireless; names like 3Com, Ericsson, IBM, Intel, Lucent, Microsoft, Nokia, Toshiba, and Motorola.

Bluetooth is a global de facto standard for wireless connectivity. Based on a low-cost, short-range radio link, Bluetooth cuts the cords that used to tie up digital devices.

When two Bluetooth equipped devices come within 10 meters range of each other, they can establish a connection together. And because Bluetooth utilizes a radio-based link, it doesn't require a line-of-sight connection in order to communicate. Your laptop could send information to a printer in the next room, or your microwave could send a message to your mobile phone telling you that your meal is ready.

Bluetooth technology is a cutting-edge (advanced, progressive) open specification that enables short-range wireless connections. Bluetooth wireless networking technology is built into a variety of devices, such as desktop computers, printers, smart phones, headsets, digital cameras and mobile phones. It uses low-power radio frequency band (2,4GHz) to transfer information wirelessly.

Voice Applications

A native feature of the Bluetooth specification is synchronous voice channels. Bluetooth has the ability to reserve bandwidth for carrying digital voice data. Bluetooth can support as many as three simultaneous, full duplex voice conversations within a piconet.

However nowadays there are a couple of ways to get around using wires. One is to carry information between components via beams of light in the infrared spectrum. Infrared is used in most television remote control systems, and with a standard called IrDA (Infrared Data Association). It's used to connect some computers with peripheral devices. Infrared communications are fairly reliable and don't cost very much to build into a device, but there are a couple of drawbacks. First, infrared is a "line of sight" technology. For example, you have to point the remote control at the television or DVD player to make things happen. The second drawback is that infrared is almost always a "one to one" technology. You can send data between your desktop computer and your laptop computer, but not your laptop computer and your PDA at the same time.

The second alternative to wires, **cable synchronizing**, is a little more troublesome than infrared. In synchronizing, you attach the PDA to your computer (usually with a cable) and make sure that the data on the PDA and the data on the computer match. However this procedure can be really irritating.

Bluetooth is intended to get around the problems that come with both infrared and cable synchronizing systems. From the user's point of view, there are three important features to Bluetooth:

- It's wireless. When you travel, you don't have to worry about keeping track of a briefcase full of cables to attach all of your components, and you can design your office without wondering where all the wires will go.

- It's inexpensive.

- You don't have to think about it. Bluetooth doesn't require you to do anything special to make it work. The devices find one another and strike up a conversation without any user input at all.

One of the ways Bluetooth devices avoid interfering with other systems is by sending out very weak signals of 1 milliwatt. By comparison, the most powerful cell phones can transmit a signal of 3 watts. The low power limits the range of a Bluetooth device to about 10 meters, cutting the chances of interference between your computer system and your portable telephone or television. Even with the low power, the walls in your house won't stop a Bluetooth signal, making the standard useful for controlling several devices in different rooms.

It is unlikely that several devices will be on the same frequency at the same time, because Bluetooth uses a technique called spread-spectrum frequency hopping. In this technique, a device will use 79 individual, randomly chosen frequencies within a designated range, changing from one to another on a regular basis. In the case of Bluetooth, the transmitters change frequencies 1,600 times every second, meaning that more devices can make full use of a limited slice of the radio spectrum.

In the future, Bluetooth is likely to be standard in tens of millions of mobile phones, PCs, laptops and a whole range of other electronic devices. As a result, the market is going to demand new innovative applications, value-added services, end-to-end solutions and much more. The possibilities opened up really are limitless, and because the radio frequency used is globally available, Bluetooth can offer fast and secure access to wireless connectivity all over the world. With potential like that, it's no wonder that Bluetooth is said to become the fastest adopted technology in history.

4.9. Network Communications

1. The application layer is the only part of a communications process that a user sees, and even then, the user doesn't see most of the work that the application does to prepare a message for sending over a network. The layer converts a message's data from human-readable form into bits and attaches a header identifying the sending and receiving computers.

2. The presentation layer ensures that the message is transmitted in a language that the receiving computer can interpret (often ASCII). This layer translates the language, if necessary and then compresses and perhaps encrypts the data. It adds another header specifying the language as well as the compression and encryption schemes.

3. The session layer opens communications and has the job of keeping straight the communications among all nodes on the network. It sets boundaries (called bracketing) for the beginning and end of the message, and establishes whether the messages will be sent half-duplex, with each computer taking turns sending and receiving, or full-duplex, with both computers sending and receiving at the same time. The details of these decisions are placed into a session header.

4. The transport layer protects the data being sent. It subdivides the data into segments, creates checksum tests – mathematical sums based on the contents of data – that can be used later to determine if the data was scrambled. It can also make backup copies of the data. The transport header identifies each segment's checksum and its position in the message.

5. The network layer selects a route for the message. It forms data into packets, counts them, and adds a header containing the sequence of packets and the address of the receiving computer.

6. The data-link layer supervises the transmission. It confirms the checksum, then addresses and duplicates the packets. This layer keeps a copy of each packet until it receives confirmation from the next point along the route that the packet has arrived undamaged.

7. The physical layer encodes the packets into the medium that will carry them – such as an analogue signal, if the message is going across a telephone line – and sends the packets along that medium.

8. An intermediate node calculates and verifies the checksum for each packet. It may also reroute the message to avoid congestion on the network.

9. At the receiving node, the layered process that sent the message on its way is reversed. The physical layer reconverts the message into bits. The data-link layer recalculates the checksum, confirms arrival, and logs in the packets. The network layer recounts incoming packets for security and billing purposes. The transport layer recalculates the checksum and reassembles the message segments. The session layer holds the parts of the message until the message is complete and sends it to the next layer. The presentation layer expands and decrypts the message. The application layer converts the bits into readable characters, and directs the data to the correct application.

4.10. Storage area network (SAN)

As companies rely more and more on ecommerce, online-transaction processing and databases, the amount of information that needs to be managed and stored on a network can intimidate even the most experienced of network managers. While servers do a good job of storing data, their capacity is limited and they can become a bottleneck if too many users try to access the same information. Instead, most companies rely on peripheral storage devices, such as tape libraries, RAID disks and even optical storage systems. These devices are effective for backing up data online and storing large amounts of information.

But as server farms increase in size and companies rely more heavily on data-intensive applications, such as multimedia, the traditional storage model isn't quite as useful. This is because access to these peripheral devices can be slow, and it might not always be possible for every user to easily and transparently access each storage device.

The most basic way of expanding storage capacity on the network is to hang disk arrays or other storage devices off servers, using the SCSI interface or bus.

While SCSI has been a workhorse over the years for connecting peripherals at a relatively fast speed, distance limitations have kept this

particular bus interface from evolving rapidly. The SCSI standards put a bus length limit of about 6m on devices. While this distance limitation doesn't really affect connecting storage devices directly to a server, it does severely restrict placing RAID and tape libraries at other points on the network.

This is where the concept of Network Attached Storage (NAS) comes in. NAS is simple in concept and execution: disk arrays and other storage devices connect to the network through a traditional LAN interface, such as Ethernet. Storage devices would thus attach to network hubs, much the same as servers and other network devices. However, NAS does have a few drawbacks.

First, network bandwidth places throughput limitations on the storage devices. Another downside to NAS is the lack of cohesion among storage devices. While disk arrays and tape drives are on the LAN, managing the devices can prove challenging, since they are separate entities and not logically tied together. NAS has its place as a viable storage architecture, but large companies need something more.

Large enterprises that want the ability to store and manage large amounts of information in a high-performance environment now have another option: the Storage Area Network (SAN). In a SAN, storage devices such as Digital Linear Tapes (DLTs) and RAID arrays are connected to many kinds of servers via a high-speed interconnection, such as Fibre Channel.

This high-speed link creates a separate, external network, that's connected to the LAN, but acts as an independent entity.

This setup allows for any-to-any communication among all devices on the SAN. It also provides alternative paths from server to storage device. In other words, if a particular server is slow or completely unavailable, another server on the SAN can provide access to the storage device. A SAN also makes it possible to mirror data, making multiple copies available.

SANs offer several advantages. First, they allow for the addition of bandwidth without burdening the main LAN. SANs also make it easier to conduct online backups without users feeling the bandwidth pinch. When more storage is needed, additional drives do not need to be connected to a specific server; rather, they can simply be added to the storage network and accessed from any point.

Another reason for the interest in SANs is that all the devices can be centrally managed. Instead of managing the network on a per-device basis, storage can be managed as a single entity, making it easier to deal with storage networks that could potentially consist of dozens or even hundreds of servers and devices.

You can connect almost any modern server to a SAN, because SAN-support hardware and software spans most PC midrange and mainframe platforms. Ideally, a SAN will lighten your server's workload by offloading many storage-related server tasks to the SAN and by better allocating storage resources to servers.

4.11. Profinet

PROFINET is the open industrial Ethernet standard of PROFIBUS & PROFINET International (PI) for automation. PROFINET uses TCP/IP and IT standards, and is, in effect, real-time Ethernet.

The PROFINET concept features a modular structure so that users can select the cascading functions themselves. They differ essentially because of the type of data exchange to fulfill the partly very high requirements of speed.

In conjunction with PROFINET, the two perspectives PROFINET CBA and PROFINET 10 exist. PROFINET CBA is suitable for the component-based communication via TCP/IP and PROFINET 10 used for the real-time communication with real-time requirements in modular systems engineering. Both communication options can be used in parallel. PROFINET 10 was developed for real time (RT) and isochronous real time (IRT) communication with the decentral periphery. The designations RT and IRT merely describe the real-time properties for the communication within PROFINET 10.

PROFINET CBA and PROFINET 10 can communicate at the same time on the same bus system. They can be operated separately or combined so that a PROFINET 10 subsystem appears as a PROFINET CBA system from a system perspective.

PROFINET component model (PROFINET CBA)

A PROFINET CBA system consists of various automation components. One component covers all mechanical, electrical and IT variables. The component can be generated using the standard programming tools. A component is described using a PROFINET Component Description (PCD) file in XML. A planning tool loads these descriptions and enables the logical interconnections between the individual components to be generated for implementing a plant.

The basic idea of CBA is that an entire automation system can in many cases be divided into autonomously operating subsystems, thereby arranging them very clearly. The design and the functions can actually end up in identical or slightly modified form in several systems. These PROFINET components are usually controlled by manageable number of input signals. Within the component, a control program written by the user executes the required function within the component and passes the corresponding output signals to another controller. The engineering that is associated with it is manufacturer-neutral. The communication of a component-based system is only configured, instead of being programmed. The communication with PROFINET CBA (without real time) is suitable for bus cycle times of approx. 50–100 ms. The parallel running RT channel allows for data cycles similar to PROFINET I0 (a few ms).

PROFINET and the peripherals (PROFINET I0)

Interfacing the peripherals is implemented by PROFINET I0. It defines the communication with field connected peripheral devices. Its basis is a cascading real-time concept. PROFINET I0 defines the entire data exchange between controllers (devices with “master functionality”) and the devices (devices with «slave functionality»), as well as parameter setting and diagnosis. PROFINET I0 is designed for the fast data exchange between Ethernet-base field devices and follows the provider-consumer model. Field devices in a subordinate PROFIBUS line can be integrated in the PROFINET I0 system without any effort and seamlessly via an I0-Proxy (representative of a subordinate bus system). A device developer can implement PROFINET I0 with any commercially available Ethernet controller. It is well-suited for the data exchange with bus cycle times of a few ms. The configuration of an I0-System has been kept nearly identical to

the «look and feel» of PROFIBUS. PROFINET 10 always contains the real-time concept.

PROFINET 10 system consists of the following devices:

- The 10 Controller, which controls the automation task.
- The 10 Device, which is a field device, monitored and controlled by an 10 Controller. An 10 Device may consist of several modules and sub-modules.
- The 10 Supervisor is software typically based on a PC for setting parameters and diagnosing individual 10 Devices.

An Application Relation (AR) is established between an 10 Controller and an 10 Device. These ARs are used to define Communication Relations (CR) with different characteristics for the transfer of parameters, cyclic exchange of data and handling of alarms.

PROFINET 10 addressing

Every module within a PROFINET network has three addresses:

- MAC address
- IP address
- Device name, a logical name for the module within the total configuration

Because PROFINET uses TCP/IP, a MAC and IP address are used. A MAC address changes if the device is replaced. An IP address is a form of dynamic addressing. Because there was a need for a fixed address a device name is used.

PROFINET and real time

Within PROFINET 10, process data and alarms are always transmitted in real time (RT). Real time in PROFINET is based on the definition of IEEE and IEC, which allow for only a limited time for execution of real-time services within a bus cycle. The RT communication represents the basis for the data exchange for PROFINET 10. Real-time data are treated with a higher priority than TCP(UDP)/IP data. RT provides the basis for the real-time communication in the area of distributed periphery and for the PROFINET component model (PROFINET CBA). This type of

data exchange allows bus cycle times in the range of a few hundred microseconds.

The isochronous data exchange with PROFINET is defined in the isochronous real-time (IRT) concept. PROFINET 10 field devices with IRT functionality have switch ports integrated in the field device. They can be based e.g. on the Ethernet controllers ERTEC 400/200. The data exchange cycles are usually in the range of a few hundred microseconds up to a few milliseconds. The difference to real-time communication is essentially the high degree of determinism, so that the start of a bus cycle is maintained with high precision. The start of a bus cycle can deviate up to 1 μ s (jitter). IRT is required, for example, for motion control applications (positioning control processes).

Benefits of PROFINET

Due to the continuous further development of PROFINET, users are provided with a long-term perspective for the implementation of their automation tasks.

The system operator profits from the simple expandability of the system and high degree of availability from autonomously running subsystems.

4.12. Broadband communications

Integrated Services Digital Network (ISDN)

ISDN services can be carried over existing telephone network infrastructure to terminal adapters (TAs) in the client machine. A common ISDN interface standard has a digital communications line consisting of three independent channels: two Bearer (B) channels, each at 64Kbit/s, and one Data (D) channel at 16Kbit/s. The D channel is used to carry signalling and supervisory information to the network, while the B channels carry the data and can be linked to provide a 128Kbit/s data channel.

Wireless connections

The wireless alternatives come in two forms: satellite and cellular. Satellite systems require the use of a modem to maintain the upload. Downstream bandwidth is provided via a dedicated satellite dish, connector hardware and proprietary software.

Cellular systems use assigned radio frequencies and are based around a network of transmitters that are arranged in a cellular network, much like cellular mobile phone systems.

The cable alternative

Cable companies can also offer affordable broadband services over copper coaxial or fibre infrastructure networks. The connection is shared by several customers on a branch, so actual connection rates are variable, unlike ISDN and DSL.

Digital Subscriber Line (DSL)

DSL technology capitalises on the existing network of copper infrastructure, but allows digital signals to be carried rather than analogue. It allows the full bandwidth of the copper twisted-pair telephone cabling to be utilised.

With splitter-based services, the DSL signal is pulled out from the phone line as it enters your premises and is wired separately to a DSL modem. This involves additional hardware and installation by the service provider at the customer site. The shielded option involves no installation, but the telephone company's equipment and some of your equipment might need upgrading.

With Asymmetric Digital Subscriber Line (ADSL), most of the duplex bandwidth is devoted to the downstream direction, with only a small proportion of bandwidth being available for upstream. Much Internet traffic through the client's connection, such as Web browsing, downloads and video streaming, needs high downstream bandwidth, but user requests and responses are less significant and therefore require less on the upstream. In addition, a small proportion of the downstream bandwidth can be devoted to voice rather than data, allowing you to hold phone conversations without requiring a separate line.

DSL-based services are a very low-cost option when compared to other solutions offering similar bandwidth, so they can be made available to the customer at extremely competitive prices.

4.13. Router

A **router** is a device that forwards data packets between computer networks, creating an overlay internetwork. A router is connected to two or more data lines from different networks. When a data packet comes in one of the lines, the router reads the address information in the packet to determine its ultimate destination. Then, using information in its routing table or routing policy, it directs the packet to the next network on its journey. Routers perform the “traffic directing” functions on the Internet. A data packet is typically forwarded from one router to another through the networks that constitute the internetwork until it reaches its destination node.

The most familiar type of routers are home and small office routers that simply pass data, such as web pages and email, between the home computers and the owner's cable or DSL modem, which connects to the Internet through an ISP. More sophisticated routers, such as enterprise routers, connect large business or ISP networks up to the powerful core routers that forward data at high speed along the optical fiber lines of the Internet backbone. Though routers are typically dedicated hardware devices, use of software-based routers has grown increasingly common. When multiple routers are used in interconnected **networks**, the routers exchange information about destination addresses using a dynamic routing protocol. Each router builds up a table listing the preferred routes between any two systems on the interconnected networks. A router has interfaces for different physical types of network connections, (such as copper cables, fiber optic, or wireless transmission). It also contains firmware for different networking Communications protocol standards. Each network interface uses this specialized computer software to enable data packets to be forwarded from one protocol transmission system to another.

Routers may also be used to connect two or more logical groups of computer devices known as subnets, each with a different sub-network address. The subnets addresses recorded in the router do not necessarily map directly to the physical interface connections. A router has two stages of operation called planes.

Routers may provide connectivity within enterprises, between enterprises and the Internet, and between internet service providers (ISPs) networks. The largest routers (such as the Cisco CRS-1 or Juniper T1600) interconnect the various ISPs, or may be used in large enterprise networks. Smaller routers usually provide connectivity for typical home and office networks. Other networking solutions may be provided by a backbone Wireless Distribution System (WPS), which avoids the costs of introducing networking cables into buildings.

All sizes of routers may be found inside enterprises. The most powerful routers are usually found in ISPs, academic and research facilities. Large businesses may also need more powerful routers to cope with ever increasing demands of intranet data traffic. A three-layer model is in common use, not all of which need be present in smaller networks.

Access routers, including 'small office/home office' (SOHO) models, are located at customer sites such as branch offices that do not need hierarchical routing of their own. Typically, they are optimized for low cost. Some SOHO routers are capable of running alternative free Linux-based firmwares like Tomato, OpenWrt or DD-WRT.

Distribution

Distribution routers aggregate traffic from multiple access routers, either at the same site, or to collect the data streams from multiple sites to a major enterprise location. Distribution routers are often responsible for enforcing quality of service across a WAN, so they may have considerable memory installed, multiple WAN interface connections, and substantial onboard data processing routines. They may also provide connectivity to groups of file servers or other external networks.

Forwarding

For pure Internet Protocol (IP) forwarding function, a router is designed to minimize the state information associated with individual packets. The main purpose of a router is to connect multiple networks and forward packets destined either for its own networks or other networks. A router is considered a Layer 3 device because its primary forwarding decision is based on the information in the Layer 3 IP packet, specifically the destination IP address. This process is known as routing. When each

router receives a packet, it searches its routing table to find the best match between the destination IP address of the packet and one of the network addresses in the routing table. Once a match is found, the packet is encapsulated in the Layer 2 data link frame for that outgoing interface. A router does not look into the actual data contents that the packet carries, but only at the layer 3 addresses to make a forwarding decision, plus optionally other information in the header for hints on, for example, QoS. Once a packet is forwarded, the router does not retain any historical information about the packet, but the forwarding action can be collected into the statistical data, if so configured.

Forwarding decisions can involve decisions at layers other than layer 3. A function that forwards based on layer 2 information is properly called a bridge. This function is referred to as layer 2 bridging, as the addresses it uses to forward the traffic are layer 2 addresses (e.g. MAC addresses on Ethernet).

Besides making decision as which interface a packet is forwarded to, which is handled primarily via the routing table, a router also has to manage congestion, when packets arrive at a rate higher than the router can process. Three policies commonly used in the Internet are tail drop, random early detection (RED), and weighted random early detection (WRED). Tail drop is the simplest and most easily implemented; the router simply drops packets once the length of the queue exceeds the size of the buffers in the router. RED drops datagrams early when the queue exceeds a pre-configured portion of the buffer, until a pre-determined max, when it becomes tail drop. WRED requires a weight on the average queue size to act upon when the traffic is about to exceed the pre-configured size, so that short bursts will not trigger random drops.

Another function a router performs is to decide which packet should be processed first when multiple queues exist. This is managed through Quality of service (QoS), which is critical when Voice over IP is deployed, so that delays between packets do not exceed 150ms to maintain the quality of voice conversations.

Yet another function a route performs is called policy-based routing where special rules are constructed to override the rules derived from the routing table when a packet forwarding decision is made.

These functions may be performed through the same internal paths that the packets travel inside the router. Some of the functions may be performed through an application-specific integrated circuit (ASIC) to avoid overhead caused by multiple CPU cycles, and others may have to be performed through the CPU as these packets need special attention that cannot be handled by an ASIC.

Использованные термины

ASCII – американский стандартный код для обмена информацией

bookmark – закладка

DNS (domain name system) – система доменных имён

DLT (Digital Linear Tape) – стример с цифровой линейной записью

ISDN (Integrated Services Digital Network) – цифровая сеть интегрального обслуживания

instant messaging – оперативный обмен сообщениями

FTP – обработка данных, устойчивая к отказам

firmware – встроенная программа

favourite – избранное

Gopher – Гофер – интерактивная распределённая информационная система

Media Independent Interface – интерфейс, независимый от среды передачи

Mailing list – список рассылки

MAC (Media Access Control) – управление доступом к среде передачи данных

newsgroup – телеконференция

NAS (Network Attached Storage) – устройства хранения, подключаемые к сети

PDA (Personal Digital Assistant) – персональный цифровой секретарь
RAID (Redundant Array of Independent Disks) – массив независимых дисков с избыточностью
Spider – поисковый агент
SONO (Small office/home office) – малый бизнес/ домашний офис
SCSI (Small Computer System Interface) – интерфейс малых вычислительных систем
SAN (Storage area network) – распределённая сеть хранения
TLD (top – level domain) – домен верхнего уровня
UDP (User Datagram Protocol) – дейтаграммный протокол пользователя
URL (Uniform Resource Locator) – унифицированный указатель ресурса
USB (universal serial bus) – универсальная последовательная шина
WEP (Wired Equivalent Privacy) – алгоритм шифрования для беспроводных сетей

Задания к текстам

1. Выпишите новые слова в свой словарь и, пользуясь их транскрипцией, произносите их вслух.
2. Прочтите вслух первый абзац.
3. Переведите письменно второй абзац.
4. Внимательно прочитайте текст и озаглавьте каждый абзац.
5. Ответьте на вопросы:
 - какими из упомянутых в тексте интернет-систем вы пользуетесь?
 - какие из названных в тексте веб-ресурсов вы используете?
 - какая, по вашему мнению, самая важная характеристика технологии Bluetooth ?
 - какую новую информацию вы получили из прочитанных текстов?

5. DATA MINING

5.1. Data mining

Data mining is simply filtering through large amounts of raw data for useful information that gives businesses a competitive edge. This information is made up of meaningful patterns and trends that are already in the data but were previously unseen.

The most popular tool used when mining is artificial intelligence (AI). AI technologies try to work the way the human brain works, by making intelligent-guess, learning by example, and using deductive reasoning. Some of the more popular AI methods used in data mining include neural networks clustering, and decision trees. Neural networks look at the rules of using data which are based on the connections found or on a sample set of data. As a result, the software continually analyses value and compares it to the other factors, and it compares these factors repeatedly until it finds patterns emerging. These patterns are known as rules. The software then looks for other patterns based on these rules or sends an alarm when a trigger value is hit.

Clustering divides data into groups based on similar features or limited data ranges. Clusters are used when data isn't labelled in a way that is favourable to mining. For instance, an insurance company that wants to find instances of fraud wouldn't have its records labelled as fraudulent or not fraudulent. But after analysing patterns within clusters, the mining software can start to figure out the rules that point to which claims are likely to be false.

Decision trees, like clusters, separate the data into subsets and then analyse the subsets to divide them into further subsets and so on (for a few more levels). The final subsets are then small enough that the mining process can find interesting patterns and relationships within the data.

Once the data to be mined is identified, it should be cleansed. Cleansing data frees it from duplicate information and erroneous data. Next, the data should be stored in a uniform format within relevant categories or fields. Mining tools can work with all types of data storage, from large data

warehouses to smaller desktop databases to flat files. Data warehouses and data marts are storage methods that involve archiving large amounts of data in a way that makes it easy to access when necessary.

When the process is complete, the mining software generates a report. An analyst goes over the report to see if further work needs to be done, such as refining parameters, using other data analysis tools to examine the data, or even scrapping the data if it's unusable. If no further work is required, the report proceeds to the decision makers for appropriate action.

5.2. Research on the Internet

Finding information on the World Wide Web needs an Internet search engine such as Google, AltaVista or Excite. Search engines have a text box where you type in a keyword or words. A search engine is a software program that reads the keywords in the text box and searches the Internet for Web pages, websites and other Internet files that use them. These documents are shown on the computer screen in a results listing.

When carrying out searches, you should usually be specific and brief in your choice of words. If the keyword is too general, or includes too many different meanings, the results listing may not be useful. Different search engines categorize information in different ways, which changes the way they store and retrieve it.

Using upper case letters (capital letters) in a keyword search will only retrieve documents that use upper case. Typing in lower case (no capitals) is usually better because search engines will retrieve documents that use both upper case and lower case letters.

You can narrow a search using logical operators such as AND, OR and NOT. AND retrieves all the words typed in the text box, OR retrieves either of the words and NOT excludes words. Spelling is important when typing in keywords, but a search engine will not usually read punctuation, prepositions and articles.

5.3. Google

10 tips for conducting searches using Google:

Tip 1

Choosing Keywords. Choose your keywords wisely. Try the obvious first and be as specific as possible. Typing in “football” is unlikely to find information on your favourite team. Entering the exact name of the team, for example Arsenal Football Club, should get results.

Tip 2

Refining your search. Google returns pages which contain all your search terms. To make your search more specific, add more terms. The order in which you type the terms will affect the results. For example, to find information about Russian landings on the moon, type “moon landings Russian”.

Tip 3

“–“ ***searches.*** To exclude a particular item from your search, put a minus sign immediately in front of it. For example, “moon landings Russian” -American finds pages that mention Russian but not American landings.

Tip 4

Category searches. Search by category to narrow down the number of relevant pages. The Google Web Directory is a good place to start. For example, Reference > maps > region > Asia > Nepal will produce a map of Nepal.

Tip 5

Definitions. Google will search for definitions online. To find a definition, type: “define”: followed by the words you want to know about. For example to find definition of “dongle”, type “define: dongle”. For technical terms you can try labs.google.com/glossary.

Tip 6

Advanced searches. Use Google Advanced Search page to finetune your search. You can specify language, date occurrences (anywhere on the page, title, URL) and other options. You can also specify words which should not appear on the page.

Tip 7

Numrange searches. You can specify that your result contains numbers within a range you set. The numbers can be quantities of all kinds, prices, years, weights, etc. For example, to search for information on Celtic Football Club between 1870 and 1880, type “Celtic Football Club 1870...1880”.

Tip 8

Phrase searches. Google ignores common words like ‘and’, ‘but’ and ‘to’ To search for complete phrases including such words enclose your search words in double quotes (“”). For example, “I have nothing to offer but blood, toil, tears and sweat.”

Tip 9

“OR” searches. To find pages that include either one keyword or another, insert OR between the words. For example, “alternative energy solar OR wind”.

Tip 10

Image searches. To search for images, select the Images option and make your search as specific as you can. You can limit your search to specific file types, for example GIF or JPEG, like this: lion filetype:jpg.

Использованные термины

data mining – поиск информации

flat file – плоский файл

GIF (Graphics Interchange Format) – формат обмена графическими файлами

keyword – ключевое слово

refining – уточнение

text box – текстовое окно

upper case letter – буква верхнего регистра

warehouse – хранилище метаданных

Задания к текстам

1. Выпишите новые слова в свой словарь и, пользуясь их транскрипцией, произнесите их вслух.
2. Прочитайте первый абзац вслух.
3. Ответьте на вопросы:
 - какие из упомянутых в тексте методов поиска информации вы используете?
 - что нового вы узнали из прочитанного текста?
4. Подготовьте 2–3 вопроса к прочитанному тексту.
5. (Работа в парах). На занятии задайте партнеру подготовленный дома вопрос и выслушайте его ответ; поменяйтесь ролями.

6. DATA TRANSFER

6.1. Data transfer

When you send an e-mail message to somebody, this message can travel through many different networks and computers. The data is constantly being directed towards its destination by special computers called routers. Because of this, it is possible to get into any of computers along the route, intercept and even change the data being sent over the Internet. In spite of the fact that there are many strong encoding programs available, nearly all the information being sent over the

Internet is transmitted without any form of encoding, i.e. «in the clear». But when it becomes necessary to send important information over the network, these encoding programs may be useful. Some American banks and companies even conduct transactions over the Internet.

Secure transactions across the Internet have three goals. First, the two parties engaging in a transaction {say, an email or a business purchase) don't want a third party to be able to read their transmission. Some form of data encryption is necessary to prevent this. Second, the receiver of the message should be able to detect whether someone has tampered with it in transit. This calls for a message-integrity scheme. Finally, both parties must know that they're communicating with each other, not an impostor. This is done with user authentication.

Today's data encryption methods rely on a technique called public-key cryptography. Everyone using a public-key system has a public key and a private key. Messages are encrypted and decrypted with these keys. A message encrypted with your public key can only be decrypted by a system that knows your private key.

For the system to work, two parties engaging in a secure transaction must know each other's public keys. Private keys, however, are closely guarded secrets known only to their owners. When I want to send you an encrypted message, I use your public key to turn my message into gibberish. I know that only you can turn the gibberish back into the original message, because only you know your private key. Public- key cryptography also

works in reverse – that is, only your public key can decipher your private key's encryption.

To make a message tamper-proof (providing message integrity), the sender runs each message through a message-digest function. This function within an application produces a number called a message-authentication code (MAC). The system works because it's almost impossible for an altered message to have the same MAC as another message. Also, you can't take a MAC and turn it back into the original message.

The software being used for a given exchange produces a MAC for a message before it's encrypted. Next, it encrypts the MAC with the sender's private key. It then encrypts both the message and the encrypted MAC with the recipient's public key and sends the message.

When the recipient gets the message and decrypts it, he also gets an encrypted MAC. The software takes the message and runs it through the same message-digest function that the sender used and creates its own MAC. Then it decrypts the sender's MAC. If the two are the same, then the message hasn't been tampered with.

The dynamics of the Web dictate that a user- authentication system must exist. This can be done using digital certificates.

A server authenticates itself to a client by sending an unencrypted ASCII-based digital certificate. A digital certificate contains information about the company operating the server, including the server's public key. The digital certificate is 'signed' by a trusted digital- certificate issuer, which means that the issuer has investigated the company operating the server and believes it to be legitimate. If the client trusts the issuer, then it can trust the server. The Issuer 'signs' the certificate by generating a MAC for it, then encrypts the MAC with the issuer's private key. If the client trusts the issuer, then it already knows the issuer's public key.

The dynamics and standards of secure transactions will change, but the three basic tenets of secure transactions will remain the same. If you understand the basics, then you're already three steps ahead of everyone else.

6.2. E-mail

An **e-mail** is an electronic message sent from one computer to another that can also include **attachments**: documents, pictures, sounds and even computer programs.

Although it's much faster and easier to use than the post, snail mail, the two have many things in common: you send an email to a mail server (an electronic post office) where it is stored in a mailbox, which holds incoming mail until the recipient downloads it. Users are given an email address and a password by an Internet Service Provider (ISP).

A typical email address has three parts:

User name	The @ sign	The domain name or network address:
A person's or name nickname	means 'at'	the mail server where the account is located. The final part of it, the top-level domain, adds information about it, e.g. com = company, .org = non profit institution, co.uk = a company in the UK.

An Internet e-mail address has a user name, the "at" symbol (@), and a domain name. The user name is the name you choose. The domain has two parts separated by a dot (.). The first part is the network that receives the e-mail and the second is the top-level domain (TLD), which shows the type of organization, such as commercial (.com) or educational (.edu). Sometimes the TLD is a country code, such as .cz (Czech Republic).

To use e-mail a client computer needs an e-mail program to connect to a network server. The program can be stand-alone, e.g. Outlook Express, or Web-based, e.g. Yahoo. Stand-alone programs let you compose e-mail offline, but with Web-based programs you must be online.

E-mail uses two kinds of mail servers: an SMTP server, which dials with outgoing e-mail and a POP3 server, which deals with incoming e-mail. SMTP stands for Simple Mail Transfer Protocol; POP stands for Post Office Protocol.

If you send an e-mail to a friend in the same domain as you, your SMTP server simply sends it to the POP3 server in your domain, which adds it to your friend's inbox. If your friend has a different domain name, your SMTP server finds your friend's SMTP server using a Domain Name Service (DNS). When your friend's SMTP server receives the e-mail, it sends your e-mail to its POP3 server, which adds it to your friend's inbox. Emails usually have two main parts.

1. The **header** generally includes these:

TO (name and address of the recipient)

CC (carbon copy sent to another addressee)

BCC (blank / blind carbon copy)

SUBJECT (topic of the message)

2. The **body** (the message itself)

Some email programs also include a **signature**, with added information about the sender, at the end of the message.

You can make your message look more expressive or attractive by using smileys (also called emoticons): little pictures either made with characters from the keyboard such as:-) for happy,:-o surprised,:-(sad, etc. or downloaded images and animations.

Spam

Spam, or junk email, is the name given to unwanted messages, mainly commercial advertising. Some companies, spammers, use it extensively because it's cheaper than other types of advertising: you or your Internet Service Provider pay for it.

Mailing lists and newsgroups

A mailing list is a basic type of discussion group that uses email to communicate. The messages are distributed to all the subscribers, i.e. everyone who belongs to the list.

Newsgroups are similar. The main difference is that the message is not sent to someone's mail server but to a bulletin board where everybody can read and answer the message.

E-POST EXPRESS

You can attach a file while you are online or offline. Open the program and click. Create a new mail message to go to your compose window. Click on the paperclip icon with the word “Attach” below it. An Insert Attachment dialog box appears, which shows your computer directory. Click on the file you want to send and then click the Attach button. The file and an icon appear in the Attach field. Send multiple files by repeating the procedure. The files can be any size but some seniors will not accept files of more than one megabyte. To remove a file, click on the attachment with the right mouse button then click Remove. When you are finished, click Send.

6.3. E-mail protocols

Although the format of a mail message, as transmitted from one machine to another, is rigidly defined, different mail protocols transfer and store messages in slightly different ways. The mail system you’re probably used to employs combination of SMTP and POP3 to send and receive mail respectively. Others may use IMAP4 to retrieve mail, especially where bandwidth is limited or expensive.

Simple Mail Transfer Protocol

SMTP is used to transfer messages between one mail server and another. It’s also used by email programs on PCs to send mail to the server. SMTP is very straightforward, providing only facilities to deliver messages to one or more recipients in batch mode. Once a message has been delivered, it can’t be recalled or cancelled. It’s also deleted from the sending server once it’s been delivered. SMTP uses ‘push’ operation, meaning that the connection is initiated by the sending server rather than the receiver. This makes it unsuitable for delivering messages to desktop PCs, which aren’t guaranteed to be switched on at all times.

In host-based mail systems, such as Unix and Web mail, SMTP is the only protocol the server uses. Received messages are stored locally and retrieved from the local file system by the mail program. In the case of Web mail, the message is then translated into HTML and transmitted to your

browser. SMTP is the only protocol for transferring messages between servers. How they're then stored varies from system to system.

Post Office Protocol

POP is a message-retrieval protocol used by many PC mail clients to get messages from a server, typically your ISP's mail server. It only allows you to download all messages in your mailbox at once. It works in 'pull' mode, the receiving PC initiating the connection. PC-based POP3 mail clients can do this automatically at a preset interval. When you use your Web mail account to access a POP3 mailbox, the mail server opens a connection to the POP3 server just as a PC-based application would. The messages are then copied into your Web mailbox and read via a browser.

Since POP3 downloads all the messages in your mailbox, there's an option to leave messages on the server, so that they can be picked up from different machines without losing any. This does mean that you'll get every message downloaded every time you connect to the server. If you don't clean out your mailbox regularly, this could mean long downloads. When using a Web mail account to retrieve POP3 mail, be careful about leaving messages on the server – if too many build up, each download will take a long time and fill up your inbox. Many Web mail systems won't recognise messages you've already downloaded, so you'll get duplicates of ones you haven't deleted.

Internet Mail Access Protocol

IMAP is similar in operation to POP, but allows you more choice over what messages you download. Initially, only message headers are retrieved giving information about the sender and subject. You can then download just those messages you want to read. You can also delete individual messages from the server, and some IMAP4 servers let you organise your mail into folders. This makes download times shorter and there's no danger of losing messages.

Использованные термины

attachment – присоединение
ASCII (American Standard Code
for Information) – американский стандартный код для обмена
информацией
authentication – аутентификация; опознание
DNS (Domain Name Service) – сервер имён доменов
blind carbon copy – слепая копия
bulletin board – доска объявлений
gibberish – «мусор»
IMAP (Internet Message Access Protocol) – протокол доступа к
сообщениям
impostor – мошенник
mailing list – список адресов
newsgroup – телеконференция
Password – пароль
private key – секретный ключ
public key – открытый ключ
rather than – а не
snail mail – «черепашья» почта
stand-alone – автономный
tamper-proof – защита от копирования

Задания к текстам

1. Выпишите новые слова в свой словарь и, пользуясь их транскрипцией, произнесите их вслух.
2. Прочитайте первый абзац вслух.
3. Ответьте на вопросы:
 - какие новые сведения вы получили в прочитанном тексте?
 - какими программами электронной почты из названных в тексте вы пользуетесь?
4. Найдите в Интернете дополнительную информацию по обсуждаемой в тексте теме и сделайте о ней сообщение на занятии.

5. Подготовьте 2–3 вопроса по содержанию прочитанного текста.

6. (Работа в парах). На занятии задайте партнеру подготовленный дома вопрос и выслушайте его ответ; поменяйтесь ролями.

7. Используя имеющуюся у вас информацию, подготовьте письменный ответ на вопрос «Передача информации» (вопрос реальный) на экзамене по «Информатике» в конце этого семестра.

7. GRAPHICS / DESIGN

7.1. Graphics and design

Computer graphics are pictures created, changed or processed by computers. There are two categories:

- **Bitmapped graphics** represent images as **bitmaps**; they are stored as pixels and can become a bit distorted when they are manipulated. The density of dots, known as the resolution and expressed in dots per inch, determines how sharp the image is.
- **Vector graphics** represent images as mathematical formulae, so they can be changed or scaled without losing quality. They are ideal for high-resolution output.

There are different types of graphics software:

- **Image manipulation programs** let you edit your favourite images. For example, you can scan a picture into your PC or transfer a photo from your camera and then add different effects, or filters.
- **Painting and drawing programs**, also called illustration packages, offer facilities for freehand drawing, with a wide choice of pens and brushes, colours and patterns. One example is Windows Paint.
- **Business graphics programs**, also called presentation software, let you create pie charts, bar charts and line graphs of all kinds for slide shows and reports. You can import data from a database or spreadsheet to generate the graphs.
- **Computer-aided design (CAD)** is used by engineers and architects to design everything from cars and planes to buildings and furniture. First they make a wireframe, a drawing with edges and contour lines. Then if they want to colour the objects and add texture, they create a surface for the object; this is called ‘filling the surface’. Finally, the design is rendered to make the object look realistic. Rendering is a process that

adds realism to graphics by using shading, light sources and reflections.

- **Desktop publishing (DTP)** is based around a page layout program, which lets you import text from a word processor, clip-art (ready-made pictures) from graphics packages, and images from scanners or cameras, and arrange them all on a page. It is used to design and publish books, newspapers, posters, advertisements, etc.
- **Digital art**, or **computer art**, is done with applets that use mathematical formulae to create beautiful bright shapes called fractals. A fractal is a geometrical figure with special properties. Fractals can also be used to model real objects like clouds, coastlines or landscapes.
- **Computer animation** uses graphics programs (e.g. digital cartooning systems) to create or edit moving pictures. Each image in a sequence of images is called a ‘frame’.
- **Geographic information systems (GIS)** allow cartographers to create detailed maps.

7.2. Word processing

People use word processors for writing all kinds of documents, such as letters, school papers and reports. Word processors have many advantages over handwriting and manual typewriters. Word processing is faster and easier than writing by hand and you can store documents on your computer, which you cannot usually do on a typewriter. This makes it easier to review and rewrite your documents. You have more formatting choices with a **word processor** and the spelling, grammar and language tools are useful, too. You can also print copies of your documents, which look neater than handwritten ones. Many language students use word processors to improve their writing skills and because they help them feel proud of their work.

Word processors do have disadvantages, however. First, it is not easy to read long documents on a computer screen. Second, sometimes the printer does not print an exact copy of what you see on the screen. Not all

word processors can read each other's files, which is another disadvantage. Finally, word processors do not always work well with e-mail. If you paste a word-processed letter into an e-mail it may lose a lot of its formatting. Many people use a **text editor** for the Internet, which is similar to a word processor but has fewer formatting features and cannot use graphics. Text editors, such as Notepad, use a simple coding system called ASCII (American Standard Code for Information Interchange), as does e-mail.

7.3. Desktop Publishing

Desktop Publishing (DTP) programs let you work with graphics: you can draw shapes, fill them with text or colour, insert graphics or special characters from the program, or import them from another program, and you can move them all easily around the page.

While DTP programs and word-processing programs have a lot of similar commands and tools, DTP programs have one important advantage: what you see on the screen is exactly what you get when you print your document.

There are many things you can do with text and graphics: you can use an align command to put them in a straight line, horizontally or vertically, and a rotate tool lets you turn them around. You can bring text to the front of a shape or graphic or send it to the back so that you can't see it. You can also wrap text around a picture or inside a shape.

Desktop Publishing programs, such as Adobe PageMaker and QuarkXPress, let you combine text and graphics in creative ways to produce stylish greeting cards, holiday brochures, business cards, newsletters, sales catalogues, calendars and many other documents.

The tools and commands in DTP programs give you a great deal of control over text. For example, you can make word and character changes, such as changing the space between words in a text without changing the font size, or changing the space between characters to make them look neater. These choices are useful when you only have a small space to work in.

These programs also let you make a template of your document so you do not have to remake the whole document each time you want to

change the text or the pictures. Many DTP programs let you change the file format of your design into a Web page, too.

7.4. Computer-aided design

Computer-aided design (CAD), also known as computer-aided drafting (CAD) or computer-aided design and drafting (CADD), is the use of computer systems to assist in the creation, modification, analysis, or optimization of a design. Computer-aided drafting describes the process of creating a technical drawing with the use of computer software. CAD software is used to increase the productivity of the designer, improve the quality of design, improve communications through documentation, and to create a database for manufacturing. CAD output is often in the form of electronic files for print or machining operations. CAD software uses either vector based graphics to depict the objects of traditional drafting, or may also produce raster graphics showing the overall appearance of designed objects.

CAD often involves more than just shapes. As in the manual drafting of technical and engineering drawings, the output of CAD must convey information, such as materials, processes, dimensions, and tolerances, according to application-specific conventions.

CAD may be used to design curves and figures in two-dimensional (2D) space; or curves, surfaces, and solids in three-dimensional (3D) space.

CAD is an important industrial art extensively used in many applications, including automotive, shipbuilding, and aerospace industries, industrial and architectural design, prosthetics, and many more. CAD is also widely used to produce computer animation for special effects in movies, advertising and technical manuals. Because of its enormous economic importance, CAD has been a major driving force for research in computational geometry, computer graphics (both hardware and software), and discrete differential geometry.

The design of geometric models for object shapes, in particular, is occasionally called computer-aided geometric design (CAGD).

While the goal of automated CAD systems is to increase efficiency, they are not necessarily the best way to allow newcomers to understand the

geometrical principles of Solid Modeling. For this, scripting languages such as PLaSM (Programming Language of Solid Modeling) are more suitable.

Beginning in the 1980s computer-aided design programs reduced the need of draftsmen significantly especially in small to mid-sized companies. Their affordability and ability to run on personal computers also allowed engineers to do their own drafting work, eliminating the need for entire departments. In today's world, many students in universities do not learn manual drafting techniques because they are not required to do so. The days of hand drawing for final drawings are all but over. Universities no longer require the use of protractors and compasses to create drawings, instead there are several classes that focus on the use of CAD software.

Current computer-aided design software packages range from 2D vector-based drafting systems to 3D solid and surface modelers. Modern CAD packages can also frequently allow rotations in three dimensions, allowing viewing of a designed object from any desired angle, even from the inside looking out. Some CAD software is capable of dynamic mathematical modeling, in which case it may be marketed as CADD.

CAD is used in the design of tools and machinery and in the drafting and design of all types of buildings, from small residential types (houses) to the largest commercial and industrial structures (hospitals and factories).

CAD is mainly used for detailed engineering of 3D models and/or 2D drawings of physical components, but it is also used throughout the engineering process from conceptual design and layout of products, through strength and dynamic analysis of assemblies to definition of manufacturing methods of components. It can also be used to design objects. Furthermore many CAD applications now offer advanced rendering and animation capabilities so engineers can better visualize their product designs.

CAD has become an especially important technology within the scope of computer-aided technologies, with benefits such as lower product development costs and a greatly shortened design cycle. CAD enables designers to layout and develop work on screen, print it out and save it for future editing, saving time on their drawings.

Uses

Computer-aided design is one of the many tools used by engineers and designers and is used in many ways depending on the profession of the user and the type of software in question.

CAD is one part of the whole Digital Product Development (DPD) activity within the Product Lifecycle Management (PLM) processes, and as such is used together with other tools, which are either integrated modules or stand-alone products, such as:

- Computer-aided engineering (CAE) and Finite element analysis (FEA)
- Computer-aided manufacturing (CAM) including instructions to Computer Numerical Control (CNC) machines
- Photo realistic rendering
- Document management and revision control using Product Data Management (PDM).

CAD is also used for the accurate creation of photo simulations that are often required in the preparation of Environmental Impact Reports, in which computer-aided designs of intended buildings are superimposed into photographs of existing environments to represent what that locale will be like were the proposed facilities allowed to be built. Potential blockage of view corridors and shadow studies are also frequently analyzed through the use of CAD

CAD has been proven to be useful to engineers as well. Using four properties which are history, features, parameterization, and high level constraints, the construction history can be used to look back into the model's personal features and work on the single area rather than the whole model. Parameters and constraints can be used to determine the size, shape, and the different modeling elements. The features in the CAD system can be used for the variety of tools for measurement such as tensile strength, yield strength, also its stress and strain and how the element gets affected in certain temperatures.

Types

There are several different types of CAD, each requiring the operator to think differently about how to use them and design their virtual components in a different manner for each.

There are many producers of the lower-end 2D systems, including a number of free and open source programs. These provide an approach to the drawing process without all the fuss over scale and placement on the drawing sheet that accompanied hand drafting, since these can be adjusted as required during the creation of the final draft.

3D wireframe is basically an extension of 2D drafting (not often used today). Each line has to be manually inserted into the drawing. The final product has no mass properties associated with it and cannot have features directly added to it, such as holes. The operator approaches these in a similar fashion to the 2D systems, although many 3D systems allow using the wireframe model to make the final engineering drawing views.

3D «dumb» solids are created in a way analogous to manipulations of real world objects (not often used today). Basic three-dimensional geometric forms (prisms, cylinders, spheres, and so on) have solid volumes added or subtracted from them, as if assembling or cutting real-world objects. Two-dimensional projected views can easily be generated from the models. Basic 3D solids don't usually include tools to easily allow motion of components, set limits to their motion, or identify interference between components.

3D parametric solid modeling requires the operator to use what is referred to as «design intent». The objects and features created are adjustable. Any future modifications will be simple, difficult, or nearly impossible, depending on how the original part was created. One must think of this as being a «perfect world» representation of the component. If a feature was intended to be located from the center of the part, the operator needs to locate it from the center of the model, not, perhaps, from a more convenient edge or an arbitrary point, as he could when using «dumb» solids. Parametric solids require the operator to consider the consequences of his actions carefully.

Some software packages provide the ability to edit parametric and non-parametric geometry without the need to understand or undo the design intent history of the geometry by use of direct modeling functionality. This ability may also include the additional ability to infer the correct relationships between selected geometry (e.g., tangency, concentricity)

which makes the editing process less time and labor intensive while still freeing the engineer from the burden of understanding the models. These kind of non-history based systems are called Explicit Modellers or Direct CAD Modelers.

Top end systems offer the capabilities to incorporate more organic, aesthetics and ergonomic features into designs. Freeform surface modeling is often combined with solids to allow the designer to create products that fit the human form and visual requirements as well as they interface with the machine.

Today, CAD systems exist for all the major platforms (Windows, Linux, UNIX and Mac OS X); some packages even support multiple platforms.

Right now, no special hardware is required for most CAD software. However, some CAD systems can do graphically and computationally intensive tasks, so a modern graphics card, high speed (and possibly multiple) CPUs and large amounts of RAM may be recommended.

The human-machine interface is generally achieved, via a computer mouse but can also be via a pen and digitizing graphics tablet. Manipulation of the view of the model on the screen is also sometimes done with the use of a Spacemouse / SpaceBall. Some systems also support stereoscopic glasses for viewing the 3D model.

7.5. Product data management

Product data management (PDM) is the business function often within product life cycle management that is responsible for the management and publication of product data. In software engineering, this is known as version control.

PDM stems from traditional engineering design activities that created product drawings and schematics on paper and using CAD tools to create parts lists (Bills of Material structures – BOM). The PDM and BOM data is used in enterprise resource planning (ERP) systems to plan and coordinate all transactional operations of a company (sales order management, purchasing, cost accounting, logistics, etc.)

PDM is a subset of a larger concept of product lifecycle management (PLM). PLM encompasses the processes needed to launch new products (NPI), manage changes to existing products (ECN/ECO) and retire products at the end of their life (End of Life).

Product data management is the use of software or other tools to track and control data related to a particular product. The data tracked usually involves the technical specifications of the product, specifications for manufacture and development, and the types of materials that will be required to produce goods. The use of product data management allows a company to track the various costs associated with the creation and launch of a product. Product data management is part of product life cycle management, and is primarily used by engineers.

Within PDM the focus is on managing and tracking the creation, change and archive of all information related to a product. The information being stored and managed (on one or more file servers) will include engineering data such as computer-aided design(CAD) models, drawings and their associated documents.

Product data management (PDM) serves as a central knowledge repository for process and product history, and promotes integration and data exchange among all business users who interact with products – including project managers, engineers, sales people, buyers, and quality assurance teams.

The central database will also manage metadata such as owner of a file and release status of the components. The package will control check-in and check-out of the product data to multi-user; carry out engineering change management and release control on all versions/issues of components in a product; build and manipulate the product structure bill of materials (BOM) for assemblies; and assist in configurations management of product variants.

This enables automatic reports on product costs, etc. Furthermore, PDM enables companies producing complex products to spread product data into the entire PLM launch-process. This significantly enhances the effectiveness of the launch process.

Product data management is focused on capturing and maintaining information on products and/or services through its development and useful life. Typical information managed in the PDM module include:

- Part number
- Part description
- Supplier/vendor
- Vendor part number and description
- Unit of measure
- Cost/price
- Schematic or CAD drawing
- Material data sheets

7.6. Product lifecycle management

In industry, **product lifecycle management (PLM)** is the process of managing the entire lifecycle of a product from its conception, through design and manufacture, to service and disposal. PLM integrates people, data, processes and business systems and provides a product information backbone for companies and their extended enterprise

PLM systems help organizations in coping with the increasing complexity and engineering challenges of developing new products for the global competitive markets.

Product lifecycle management (PLM) should be distinguished from 'Product life cycle management (marketing)' (PLCM). PLM describes the engineering aspect of a product, from managing descriptions and properties of a product through its development and useful life; whereas. PLCM refers to the commercial management of life of a product in the business market with respect to costs and sales measures.

Product lifecycle management is one of the four cornerstones of a corporation's information technology structure. All companies need to manage communications and information with their customers (CRM-customer relationship management), their suppliers (SCM-supply chain management), their resources within the enterprise (FRP-enterprise

resource planning) and their planning (SDLC-systems development life cycle). In addition, manufacturing engineering companies must also develop, describe, manage and communicate information about their products.

One form of PLM is called people-centric PLM. While traditional PLM tools have been deployed only on release or during the release phase, people-centric PLM targets the design phase.

As of 2009, ICT development has allowed PLM to extend beyond traditional PLM and integrate sensor data and real time 'lifecycle event data' into PLM, as well as allowing this information to be made available to different players in the total lifecycle of an individual product (closing the information loop). This has resulted in the extension of PLM into closed-loop lifecycle management (CLM).

Many software solutions have developed to organize and integrate the different phases of a product's lifecycle. PLM should not be seen as a single software product but a collection of software tools and working methods integrated together to address either single stages of the lifecycle or connect different tasks or manage the whole process. Some software providers cover the whole PLM range while others single niche application. Some applications can span many fields of PLM with different modules within the same data model. An overview of the fields within PLM is covered here. It should be noted however that the simple classifications do not always fit exactly, many areas overlap and many software products cover more than one area or do not fit easily into one category. It should also not be forgotten that one of the main goals of PLM is to collect knowledge that can be reused for other projects and to coordinate simultaneous concurrent development of many products. Although PLM is mainly associated with engineering tasks it also involves marketing activities such as product portfolio management (PPM), particularly with regards to new product development. There are several life-cycle models in industry to consider, but most are rather similar. What follows below is one possible life-cycle model; while it emphasizes hardware-oriented products, similar phases would describe any form of product or service, including non-technical or software-based products.

The first stage in idea is the definition of its requirements based on customer, company, market and regulatory bodies' viewpoints. From this specification of the products major technical parameters can be defined. Parallel to the requirements specification the initial concept design work is carried out defining the aesthetics of the product together with its main functional aspects. For the industrial design many different media are used from pencil and paper, clay models to 3D CAID computer-aided industrial design software.

In some concepts, the investment of resources into research or analysis-of-options may be included in the conception phase – e.g. bringing the technology to a level of maturity sufficient to move to the next phase. However, life-cycle engineering is iterative. It is always possible that something doesn't work well in any phase enough to back up into a prior phase – perhaps all the way back to conception or research. There are many examples to draw from.

Design: Describe, define, develop, test, analyze and validate.

This is where the detailed design and development of the product's form starts, progressing to prototype testing, through pilot release to full product launch. It can also involve redesign and ramp for improvement to existing products as well as planned obsolescence. The main tool used for design and development is CAD. This can be simple 2D drawing / drafting or 3D parametric feature based solid/surface modeling. Such software includes technology such as Hybrid Modeling, Reverse Engineering, KBE (knowledge-based engineering), NDT (Nondestructive testing). Assembly construction.

This step covers many engineering disciplines including: mechanical, electrical, electronic, software (embedded), and domain-specific, such as architectural, aerospace, automotive. Along with the actual creation of geometry there is the analysis of the components and product assemblies. Simulation, validation and optimization tasks are carried out using CAE (computer-aided engineering) software either integrated in the CAD package or standalone. These are used to perform tasks such as: Stress analysis, FEA (finite element analysis): kinematics: computational fluid dynamics (CFD); and mechanical event simulation (MES). CAQ

(computer-aided quality) is used for tasks such as Dimensional tolerance (engineering) analysis. Another task performed at this stage is the sourcing of bought out components, possibly with the aid of procurement systems.

Realize: Manufacture, make, build, procure, produce, sell and deliver.

Once the design of the product's components is complete the method of manufacturing is defined. This includes CAD tasks such as tool design; creation of CNC Machining instructions for the product's parts as well as tools to manufacture those parts, using integrated or separate CAM computer-aided manufacturing software. This will also involve analysis tools for process simulation for operations such as casting, molding, and die press forming. Once the manufacturing method has been identified CPM comes into play. This involves CAPE (computer-aided production engineering) or CAP/CAPP – (production planning) tools for carrying out factory, plant and facility layout and production simulation. For example: press-line simulation; and industrial ergonomics; as well as tool selection management. Once components are manufactured their geometrical form and size can be checked against the original CAD data with the use of computer-aided inspection equipment and software. Parallel to the engineering tasks, sales product configuration and marketing documentation work take place. This could include transferring engineering data (geometry and part list data) to a web based sales configurator and other desktop publishing systems.

Service: Use, operate, maintain, support, sustain, phase-out, retire, recycle and disposal.

The final phase of the lifecycle involves managing of in service information. Providing customers and service engineers with support information for repair and maintenance. as well as waste management/recycling information. This involves using tools such as Maintenance, Repair and Operations Management (MRO) software. There is an end-of-life to every product.

Product lifecycle: Communicate, manage and collaborate

None of the above phases can be seen in isolation. In reality a project does not run sequentially or in isolation of other product development projects. Information is flowing between different people and systems. A major part of PLM is the co-ordination and management of product definition data. This includes managing engineering changes and release status of components; configuration product variations; document management; planning project resources and timescale and risk assessment.

For these tasks graphical, text and metadata such as product bills of materials (BOMs) needs to be managed. At the engineering departments level this is the domain of PDM – (product data management) software, at the corporate level EDM (enterprise data management) software, these two definitions tend to blur however but it is typical to see two or more data management systems within an organization. These systems are also linked to other corporate systems such as SCM, CRM, and ERP.

This central role is covered by numerous collaborative product development tools which run throughout the whole lifecycle and across organizations. This requires many technology tools in the areas of conferencing, data sharing and data translation.

Использованные термины

applet – апплет
application-specific – специализированный
bar chart – столбчатая диаграмма
bitmap – растр
CADD автоматизированное
(computer-aided design and drafting) – проектирование и оформление чертежей
clip-art – иллюстрированная вставка; клип
CNC (Computer Numerical Control) machines – станки с ЧПУ (числовым программным управлением)
extended enterprise – расширенное предприятие
fractal – фрактал
frame – кадр; рамка
launch process – процесс запуска
line graph – линейный график
multi-user – многопользовательская система
PDM (Product Data Management) – управление сертификацией выпускаемой продукции
pie chart – круговая диаграмма
PLM (Product Lifecycle Management) – управление жизненным циклом продукции
release status – конечное состояние
rendering – визуализация
Solid Modeling – пространственное моделирование
spreadsheet – электронная таблица
surface modelers – средства моделирования поверхностей
wireframe – каркасное представление
word processor – текстовый процессор

Задания к текстам

1. Выпишите незнакомые слова и выражения в свой словарь и, пользуясь их транскрипцией, произнесите их вслух.
2. Прочитайте вслух абзацы 1 и 2 .
3. Составьте (письменно) 2–3 вопроса к тексту.

4. Подготовьте краткое (5 мин) письменное сообщение о прочитанном тексте.

5. Организуйте и проведите (старосты) мини-конференцию по прочитанным текстам раздела «Графика и проектирование»:

- организуйте объединённое занятие 2–3 групп;
- пригласите преподавателя, читающего лекции по «Информатике»;
- выступите с подготовленными сообщениями;
- задайте выступающим подготовленные вопросы.

6. Используйте подготовленные сообщения на экзамене по «Информатике» в конце этого семестра.

8. FUTURE

8.1. Future developments

A number of technologies are being developed to increase processing speed. Gallium arsenide chip, which will allow speed five to six times faster than silicon chips, should soon be commercially available. Chips whose switching devices operate in a vacuum – thus offering no resistance to the flow of electrons – might each contain billions of such devices. And researchers are now learning how to store information at the molecular level in the chemical bonds between atoms.

Optical disks, similar to the computer disks now used to record popular music, should offer continued huge increases in storage capacity. An optical disk today can store about 1,000 megabytes of data, which is more than sufficient to handle the contents of any multivolume encyclopedia. If present trends continue, entire libraries could soon be accessed electronically for little more than the cost of a few ordinary compact discs.

The culmination of these developments may be the optical computer, which would operate with light waves. Such a machine could boost computing power by several orders of magnitude. While much of the scientific community is skeptical that optical computing can be developed

in less than a generation, if at all, it should be remembered that all great advances have met such doubts.

Nanotechnology, the science of creating and using materials or devices at molecular and atomic sizes, is going to represent a new technological revolution. These devices will fall in the range of 1 nanometre, which is equal to one billionth of a metre, to 100 nanometres (nm).

Nanobots, robots formed from molecules or molecular components, will be used in medicine to control and diagnose diseases. For example, they will be injected and will move through blood vessels destroying cholesterol molecules or cancer.

Nanocomputers, molecule-sized computers, may have the power of 100 workstations but only be the size of a grain of sand. There will be two main types of molecular computers:

Quantum computers, based on quantum mechanics, may be millions of times faster than current computers. They will be so fast because they will be able to examine all possible answers to a query at the same time. This capability is made possible by qbits, quantum bits, which can be 0 or 1, or something in between, simultaneously.

DNA computers will use DNA biochips to perform the same functions as silicon microchips do today but at a much faster speed.

The relationship between people and computers will be closer.

Computers will be embedded, or hidden, in a variety of items, for example, we'll have **wearable computers** that will be embedded in a belt or a piece of jewellery, etc

Computer chips can be injected under the skin: **RFID, radio-frequency identification tags**, might be used to track or identify people or to store information, such as medical data, although there are concerns about privacy and personal safety.

User interfaces, the systems that, facilitate communication between people and computers, will resemble human communication There will be **gesture interfaces** based on facial-hand recognition systems.

8.2. Architecture of future information systems

The installation of high-speed networks using optical fiber and high bandwidth message forwarding gateways is changing the physical capabilities of information systems. These capabilities must be complemented with corresponding software systems advances to obtain a real benefit. Without smart software we will gain access to more data, but not improve access to the type and quality of information needed for decision making.

To develop the concepts needed for future information systems we model information processing as an interaction of data and knowledge. This model provides criteria for a high-level functional partitioning. These partitions are mapped into information processing modules. The modules are assigned to nodes of the distributed information systems. A central role is assigned to modules that mediate between the users' workstations and data resources. Mediators contain the administrative and technical knowledge to create information needed for decision-making. Software which mediates is common today, but the structure, the interfaces, and implementations vary greatly, so that automation of integration is awkward.

By formalizing and implementing mediation we establish a partitioned information systems architecture, which is of manageable complexity and can deliver much of the power that technology puts into our reach. The partitions and modules map into the powerful distributed hardware that is becoming available. We refer to the modules that perform these services in a sharable and composable way as mediators.

We will present conceptual requirements that must be placed on mediators to assure effective large-scale information systems. The modularity in this architecture is not only a goal, but also enables the goal to be reached, since these systems will need autonomous modules to permit growth and enable them to survive in a rapidly changing world.

Computer-based information systems, connected to world-wide high-speed networks provide increasingly rapid access to a wide variety of data resources. This technology opens up possibilities of access to data, requiring capabilities for assimilation and analysis which greatly exceed what we now have in hand. Without intelligent processing these advances will

have only a minor benefit to the user at a decision-making level. That brave user will be swamped with ill-defined data of unknown origin.

In essence, the gap between information and data will yet be wider than it is now. Knowing that information exists, and is accessible creates expectations by end-users. Finding that it is not available in a useful form or that it cannot be combined with other data creates confusion and frustration. We believe that the objection made by some users about computer-based systems, that they create information overload, is voiced because those users get too much of the wrong kind of data.

We find two types of problems: for single databases a primary hindrance for end- user access is the volume of data that is becoming available, the lack of abstraction, and the need to understand the representation of data.

To manage the variety of knowledge, we employ specialists, and to manage the volume of data, we segment our databases. In these partitions partial results are produced, abstracted, and filtered. The problem of making decisions is now reduced to the issue of choosing and evaluating the significance of the pieces of information derived in those partitions and fusing the important portions.

For example, an investment decision for a manufacturer will depend on the fusion of information on its own production capability versus that of others, of its sales experience in related products, of the market for the conceived product at a certain price, of the cost-to- price ratio appropriate for the size of the market, and its cost of the funds to be invested. Specialists will consider these diverse topics, and each specialist will consult multiple data resources to support their claims. The decision maker will fuse that information, using considerations of risk and long-range objectives in combining the results.

An effective architecture for future information systems must support automated information acquisition processes for decision-making activities. By default, we approach the solution in a manner analogous seen in human-based support systems. While we model the system based on abstractions from the world of human information processing, we do not constrain the architecture to be anthropomorphic and to mimic human

behavior. There are many aspects of human behavior that we have not yet been able to capture and formalize adequately. Our goal is merely to define an architecture wholly composed of pieces of software that are available or appear to be attainable in a modest timeframe, say ten years. The demands of the software in terms of processing cost should be such that modern hardware can deal with it.

8.3. The future of information technology

We are in the midst of convergence. At the hardware layer, computers, phones and consumer electronics are converging. At the applications layer, we see convergence of information, entertainment, communications, shopping commerce, and education.

Computers are rapidly catching up in capability with the human brain. We can expect human-machine equivalence by about 2015. But after this, computers will continue to get smarter. There is a noticeable positive feedback loop in technology development, with each generation of improved computers giving us more assistance in the design and development of the next. Ultimately, they will design their offspring with little or no human involvement. This technology development will push every field of knowledge forwards, not just computing. It will be almost as though extraterrestrials had landed in 2020 and given us all their advanced technology overnight.

But we will never get far unless we can solve the interface problem. In the near future we may have electronic pets, with video camera eyes and microphone ears, linked by radio to the family computer. With voice and language recognition we will have easy access to all that the Internet can provide. We can tell the pet what we want and it will sort it out for us. It will be impossible to be technophobic about such an interface, and the only IT skill needed will be to speak any major language.

Telecoms applications will soon be bundled together in much the same way as office application suites are today. A major example is the electronic marketplace, which will bring customers and suppliers together in smart databases and virtual environments, with ID verification, encryption and translation. It will then implement the billing, taxation and

electronic funds transfer, while automatically producing accounts and auditing. The whole suite of services will be based on voice processing, allowing a natural voice interface to talk to the computer, all the AI to carry out the request, and voice synthesis and visualisation technology to get the answer out.

Electronic money will be very secure but much more versatile than physical alternatives. E-cash can be completely global and could be used as a de facto standard. It does not have to be linked to any national currency, so can be independent of local currency fluctuations. Its growing use on the Net will lead to its acceptance on the street and we may hold a large proportion of our total funds in this global electronic cash. People will increasingly buy direct from customised manufacturers. Shops will be places where people try on clothes, not buy them. Their exact measurements can be sent instantly to the manufacturer as soon as they have chosen an outfit. The shops may be paid by the manufacturer instead.

Employment patterns will change, as many jobs are automated and new jobs come into existence to serve new technologies. Some organisations will follow the virtual company model, where a small core of key employees is supported by contractors on a project by project basis, bringing together the right people regardless of where they live. The desks they will use will have multiple flat screens, voice interfaces, computer programs with human-like faces and personalities, full-screen videoconferencing and 3D sound positioning. All this will be without any communication cables since the whole system uses high capacity infrared links. The many short-term contractors may not have enough space in their homes for an office and may go instead to a new breed of local telework centre.

Of course, workers can be fully mobile, and we could see some people abandon offices completely, roaming the world and staying in touch via satellite systems. Even in trains and planes there may be infrared distribution to each seat to guarantee high bandwidth communication. One tool they may have in a few years is effectively a communicator badge. This will give them a voice link to computers across the network, perhaps on their office desk. Using this voice link, they can access their files and email

and carry out most computer-based work. Their earphones will allow voice synthesisers to read out their mail, and glasses with a projection system built into the arms and reflectors on the lenses will allow a head-up display of visual information. These glasses could be replaced by an active contact lens that writes pictures directly onto the retina using tiny lasers.

Another important development is the growth of the information infrastructure. Integrated information networks will greatly expand the communications traffic flowing today on space satellites.

In the near future we'll be able to swim in the immersive Internet, a technology that will change the two-dimensional world of the Internet into a 3-D experience with three-dimensional sound and images and even the sense of touch.

By the year 2040 there might be intelligent robots, machines that will be able to think creatively. The processing power of computers may have reached 1,000,000,000 MIPS (millions of instructions per second), the estimated speed of human thought.

Использованные термины

bandwidth – пропускная способность
DNA – цифровая ДНК
extraterrestrials – инопланетяне
ID verification – проверка идентификации
facial-hand recognition system – система распознавания лица и руки
feedback loop – цикл обратной связи
forwarding – передача; переадресация
fuse – объединять
mediate – посредничать
offspring – потомство
partition – раздел; разделение; разделять
query – запрос

Задания к текстам

1. Выпишите новые слова в свой словарь и, пользуясь их транскрипцией, произнесите их вслух.
2. Прочитайте первый абзац вслух.
3. Переведите второй абзац письменно.
4. Ответьте на вопросы:
 - какую новую информацию вы получили из прочитанного текста?
 - какие из описанных в тексте будущих достижений в области информационных технологий производят на вас наибольшее впечатление?
5. Приготовьте 2–3 вопроса к прочитанному тексту.
6. (Работа в парах). На занятии задайте партнеру подготовленный дома вопрос и выслушайте его ответ; поменяйтесь ролями.

Frequently Used Words

access – доступ
access point – точка доступа
account – учёт
application – приложение
artificial intelligence – искусственный интеллект
back up – копировать; резервная копия
backbone – магистраль
bandwidth – пропускная способность
bridge – мост
broadband – широкополосный
browse – просматривать
bug – ошибка
cache – кэш
cache hit – кэш-попадание
checksum – контрольная сумма
cleanse – чистить
cluster – кластер
command interpreter – интерпретатор команд
compatible – совместимый
compile – компилировать
computer-aided design – компьютерное проектирование
computing – вычисление
cursor – курсор
data mining – поиск информации
decision tree – дерево решений
default – значение по умолчанию
desktop /(computer) – настольный компьютер
directory – каталог
download – загружать
email address – адрес электронной почты
encrypt – шифровать
Ethernet – стандарт локальной сети

execute – выполнять
FAQ – часто задаваемые вопросы
fault tolerance – отказоустойчивость
fibre-optics cable – волоконно-оптический кабель
frame – кадр
gateway – шлюз
handheld (computer) – портативный компьютер
hang – «зависать»
home button – кнопка исходной позиции
homepage – начальная страница
host – главная машина
hub – концентратор
hyperlink – гиперссылка
hypertext markup language (HTML) – гипертекстовый язык описания документа
icon – пиктограмма
inbox – входной почтовый ящик
inheritance – наследование
input – ввод
input device – устройство ввода
interpreter – интерпретатор
intranet – интрасеть
ISP – поставщик интернет-сервиса
junk email – ненужное почтовое сообщение
kernel – ядро
keyboard – клавиатура
keyword – ключевое слово
laptop (computer) – компактный портативный компьютер
load – загружать
local area network (LAN) – локальная сеть
machine code – машинный код
mainframe (computer) – базовый компьютер
markup language – язык разметки

menu bar – зона заголовков меню
message-authentication code (MAC) – код аутентификации сообщений
neural net (work) – нейронная сеть
object-oriented programming – объектно-ориентированное программирование
output – выход
palmtop (computer) – миниатюрный портативный компьютер
password – пароль
peripheral – периферийное устройство
printed circuit board – печатное плато
private key – личный ключ (секретный)
public key – открытый ключ
random access memory (RAM) – оперативное запоминающее устройство
raw data – необработанные данные
record – запись
resolution – разрешающая способность
router – маршрутизатор
rule – правило
run – запускать
scrollbar – зона прокрутки
search engine – система поиска
serial – последовательный
smart card – смарт-карта
source code – исходный код
spam – спам
storage device – устройство памяти
supervisor (program) – управляющая программа
tag – тэг; ярлык
telnet – телетайпная сеть
text editor – текстовый редактор
throughput – пропускная способность
traffic – трафик
troubleshooting – поиск и исправление неисправностей

upload – загружать
upstream – от абонента к центральному узлу
utility (program) – утилита
virus-check – проверка на вирус
voice recognition – распознавание речи
Web address – веб-адрес
Web mail – веб-почта
window – окно
wireless network – беспроводная сеть
wordprocessing – текстообработка
wordprocessor – текстовый редактор
worm – «червь»
write-back cache – кэш с задержанной записью
write-through cache – кэш с немедленной записью
X (windowing) – система управления окнами

Учебное издание

Балахонов Александр Сергеевич,
Лыков Анатолий Николаевич

КОМПЬЮТЕРНЫЕ
И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Практикум

Технический редактор *И.Н. Жеганина*

Подписано в печать 16.08.13. Формат 60×90/16.
Усл. печ. л. 8,0. Тираж 100 экз. Заказ № 175/2013.

Издательство
Пермского национального исследовательского
политехнического университета.
Адрес: 614990, г. Пермь, Комсомольский пр., 29, к. 113.
Тел. (342) 219-80-33.