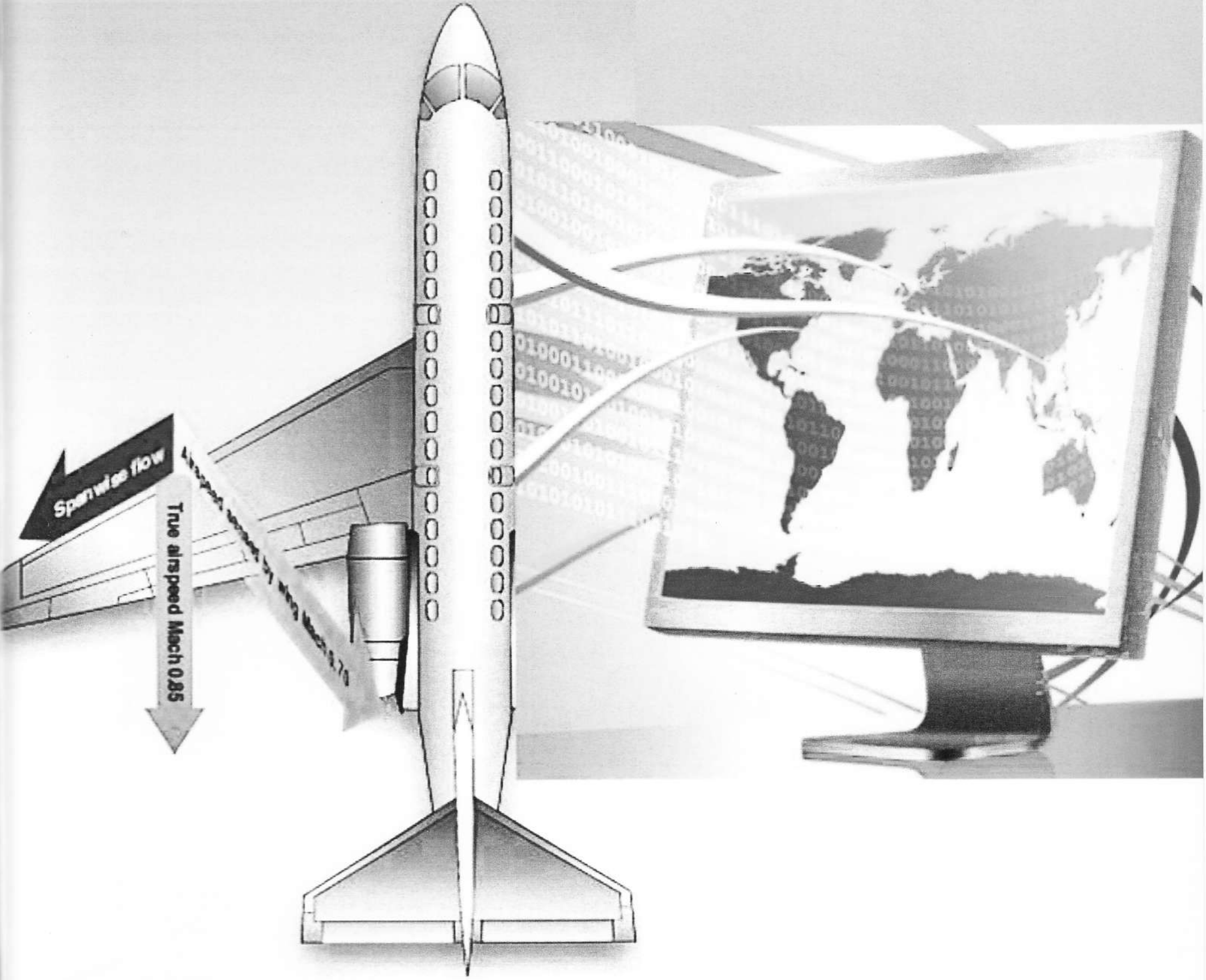
AIRCRAFT LIFE CYCLE SUPPORTING COMPUTER TECHNOLOGIES



STUDENT LABGUIDE

NATIONAL AVIATION UNIVERSITY
INFORMATION TECHNOLOGIES CENTER ITC

To my parents
with great gratitude and respect

L.A. Zhuravlova

AIRCRAFT LIFE CYCLE SUPPORTING COMPUTER TECHNOLOGIES

STUDENT LABGUIDE

KYIV 2012

UDC 629.735.017.1:004(076.5)

LBC 053-05я7
A 28

Reader: Doctor of Science O.Tamargazin. – Professor of Aerospace Institute of National Aviation University of Ukraine.

A 28 Aircraft Life Cycle Supporting Computer Technologies: Student Laboratory Guide / L.A. Zhuravlyova.-K.:ITC, 2012.-77 p.

ISBN

This Laboratory Guide is intended for mastering the computer information technologies which have regular application in the course of an aircraft life cycle. The LabGuide covers the general principles of investigating the structure of a product life cycle, and in particular, the structure of an aircraft life cycle. It also covers such themes as getting skills in CAD/CAE technologies, general principles of developing databases, using spreadsheets for the life cycle cost assessment and preparing business presentations. This LabGuide is intended for students of engineering specialties.

ISBN LBC 053-05я7

CONTENTS

PREFACE

Product lifecycle management (PLM) originated from two roots. One is enterprise management, which can be further subdivided into material resource planning (MRP), enterprise resource planning (ERP), customer relationship management (CRM), and supply chain management (SCM). Because of the visibility across the complete lifecycle of a product, the attendant risks can be estimated. In this sense, PLM serves as a decision support tool.

The other root is the management of product information throughout the entire lifecycle of the product. In this context, system integration facilitates the collaboration among virtual enterprises.

Global Aircraft Information Life Cycle Support plays a major role here.

The Aircraft information space nowadays involves a lot of computer information technologies and software products. The Computer-Aided Design and Manufacturing(CAD/CAM), product data management (PDM) systems, Aviation databases, Computer Based Training, Virtual environment tools are among them.

The aircraft information space contains different types of information: numerical, verbal, graphical, text, linguistic and mixed data.

The use of PDM systems required engineering knowledge.

With the advent of the Internet, web-based PDM systems became more accessible to suppliers and other parties outside of the enterprise.

PLM extends PDM beyond engineering and manufacturing into more strategic areas like marketing, finance, and after-sales service throughout the lifecycle of the product .

During the life of a product, a large amount of data is generated. This includes CAD data, specifications, quality documents, bill of materials, engineering simulations, etc., which are shared throughout the extended enterprise . All participants of Life Cycle create, manage, disseminate, share and use product information.

This Laboratory Guide is intended for mastering students in major computer information technologies and software products, which are the members of product data management, and aircraft in particular.

LABORATORY WORK # 1. Investigating the structure of an aircraft life cycle and developing the LC conceptual model

Key words: product life cycle, LC phases, LC management, LC cost, LC members, ontology, problem domain, information space, LC support, design stages, conceptual design, outsourcing

Objective of the laboratory work: to develop the conceptual model of an aircraft life cycle using different information sources: both technical papers and Internet resources.

Preliminary knowledge and skills

***Student must know*:**

- Aircraft operation.
- Flight operation.
- Technical operation.
- Aircraft handling, ground servicing.
- General notions on aircraft maintenance.

***Student is to be able*:**

To perform the information search in the Internet.

Knowledge after the performing the laboratory work

Student must know:

- The Aircraft LC structure and phases.
- Basic notions: information space, ontology, problem domain, conceptual model.

***Student is to be able*:**

- To define the problem domain.
- To develop the ontology of the problem domain.
- to develop the generic aircraft life cycle conceptual model.

Theoretical background

Information space is the set of concepts and relations among them held by an information system; it describes the range of possible values or meanings an entity can have under the given rules and circumstances. Information spaces surround us. When we retrieve a file from our computer, we are browsing through an information space; when we use a search engine we are sifting through an information space; and when we visit a website we are moving through yet another information space.

Another definition is that the information space are the total results of the semantic activity of the humanity, "the world of names and titles", conjugated to the ontological world.

Ontology (from onto-, from the Greek ὤν, ὄντος "being; that which is", present participle of the verb εἰμί "be", and -λογία, -logia: *science*, *study*, *theory*) is the philosophical study of the nature of *being, existence*, or *reality*, as well as the basic categories of being and their relations. Traditionally listed as a part of the major branch of philosophy known as metaphysics, ontology deals with questions concerning what entities exist or can be said to exist, and how such entities can be grouped, related within a hierarchy, and subdivided according to similarities and differences.

Principal questions of ontology are "What can be said to exist?", "Into what categories, if any, can we sort existing things?", "What are the meanings of being?", "What are the various modes of being of entities?".[]

One common approach is to divide the extant subjects and predicates into groups called categories. Such an understanding of ontological categories, however, is merely taxonomic, classificatory. The categories are, properly speaking, the ways in which a being can be addressed simply as a being, such as what it is (its 'whatness', quidditas or essence), how it is (its 'howness' or qualitativeness), how much it is (quantitativeness), where it is, its relatedness to other beings, etc.

A ***problem domain*** is the area of expertise or application that needs to be examined to solve a problem. A problem domain is simply looking at only the topics you are interested in, and excluding everything else. For example, if you were developing a system trying to measure good practice in medicine, you wouldn't include carpet drawings at hospitals in your problem domain. In this example the domain refers to relevant topics solely within your interest: medicine.

In mathematics, the term defines a domain where the parameters defining the boundaries of the domain and sufficient mappings into a set of ranges including itself are not well enough understood to provide a systematic description of the domain.

In this context see information theory as the idea behind a domain as a minimal set of sources for mappings relative to the problem a specific instance of applying.

Having defined a specific problem domain with sufficient parameters and mappings for consultation, a systematic approach to the solution can be developed in accordance with the Parker Rule. Using this rule, it is implied that any topics not directly associated with the initial problem domain, and its immediate mappings should not be included within the problem domain, but should be considered as parameters of the secondary mappings of any associated domains.

A ***domain model*** in problem solving and software engineering is a conceptual model of all the topics related to a specific problem. It describes the various entities, their attributes, roles, and relationships, plus the constraints that govern the problem domain. The domain model is created in order to represent the vocabulary and key concepts of the problem domain. The domain model also identifies the relationships among all the entities within the scope of the problem domain, and commonly identifies their attributes. A domain model that encapsulates methods within the entities is more properly associated with object oriented models.

An important advantage of a domain model is that it describes and constrains the scope of the problem domain, and can be effectively used to verify and validate the understanding of the problem domain among various stakeholders.

Conceptual model is a representation of some phenomenon, data or theory by logical and mathematical objects such as functions, relations, tables, stochastic processes, formulas, axiom systems, rules of inference etc. A conceptual model has an ontology that is the set of expressions in the model which are intended to denote some aspect of the modeled object.

Preliminary unsupervised student's work (4 hours)

1. To carry out the information retrieval, and literature search on the subject. To investigate aircraft life cycle problem domain. To generate the ontology of the problem domain.
2. To define the ontology of the problem domain.
3. To develop the ontograph of the ontology defined.

Progress of work

1. To select the concrete aircraft. To investigate the specifics of its LC. In order to do this, follow the criteria, given below:
   1.1.An aircraft has designed without the prototype (absolutely new idea) or according to the existing aircraft. If the aircraft is designed according to the prototype, carry out its generic analysis (type, general operational characteristics).
   1.2.Actual position on the market.
   1.3.Perspectives of operation in future.
   1.4.Location of outsourcers and their general characteristics.

1.5.Airlines-operators.

1.6.Computer information communication technologies used in the process of operation.

1.7.Using of CALS-technologies.

1.8.LC cost.

1.9.Accidents.

2. To arrange the Introduction to the First homework "Conceptual model of the aircraft ...." (3 pages minimum).
3. Using the ontograph, to develop the conceptual model of the problem domain.
4. Draw the sketch of the conceptual model.

Control questions

1. Give the definitions of key notions: product life cycle, LC management, LC cost, LC members, ontology, problem domain, information space, LC support, design stages, conceptual design, and outsourcing.
2. Explain the purpose of each LC phase.
3. Give the examples of using computer information technologies in the frame of an aircraft LC phases.
4. Give the generic characteristic of an aircraft life cycle cost assessment.
5. Explain the essence of outsourcing.
6. Explain the essence of Green Design.
7. Give the general characteristic of CALS technologies.
8. Give the general characteristic of the problem "Aircraft at the end of life sector".

References to the Laboratory Work #1 are given in the folder "References to the LW#1" on CD.

LABORATORY WORK # 2. Basics of CAD/CAE technologies

Key words: computer-aided design, computer-aided engineering, engineering drawings

Objective of the laboratory work: to develop the technical drawing and the conceptual 3D model of an object.

Preliminary knowledge and skills

***Student must know*:**

- Fundamentals of technical drawing
- General effective drawing standards

***Student is to be able*:**

- Understand technical drawings.
- To draw manually

Knowledge after the performing the laboratory work

Student must know:

Fundamentals of technical drawing by means of Computer Aided Design System.

***Student is to be able*:**

To create simple drawings and 3D models by means of Computer Aided Design System.

Theoretical background

Google SketchUp is a program which lets you sketch in 3D. It is excellent for creating buildings, houses, and mechanical designs.

Google SketchUp basics.

1. To run the product: double-click the Google SketchUp icon

The start screen is shown on Figure 2.1.

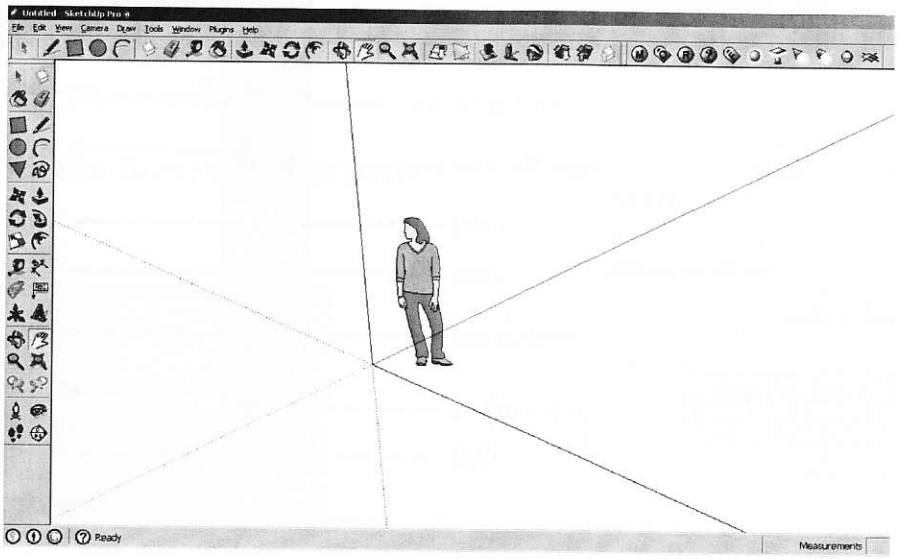
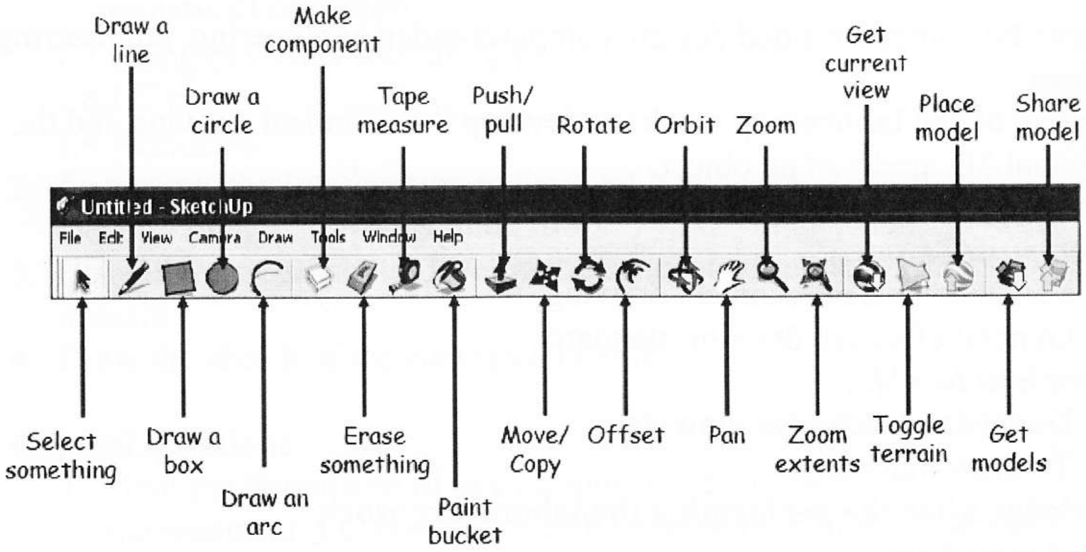


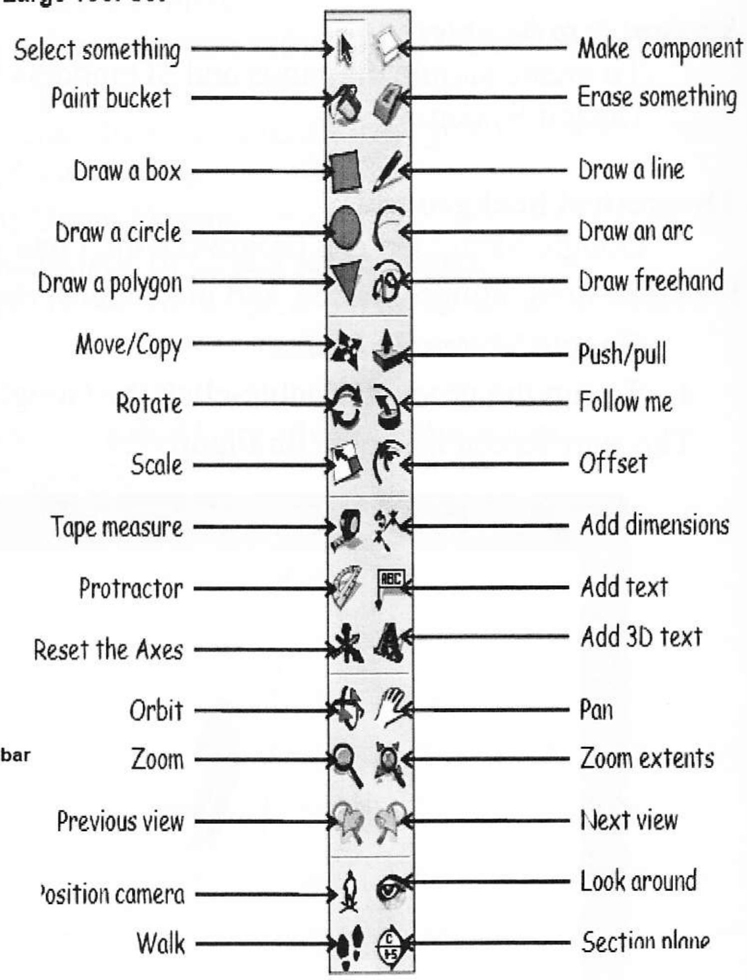
Figure 2.1.-Google SketchUp start screen

2. Panels of tools:



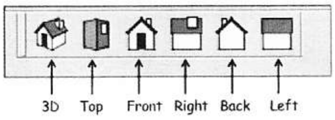
Large Toolset Toolbar

Select **View→Toolbars→Large Tool Set**



The Views Toolbar

Select **View →Toolbars →Views**



3. SketchUp "Inferences"

SketchUp uses Right Hand Coordinate system.

One of SketchUp's key strengths is that it doesn't require you to enter every little piece of information as many 3D computer programs do. Instead, it tries to infer what you really mean by how you do things. Oftentimes it uses colors to tell you what it is inferring.

— Green dots = Endpoints

— Red dots = On an edge

— Cyan dots = Midpoints of edges

— Blue dots = On a surface

— Red line = X axis

— Blue line = Y axis

— Green line = Z axis

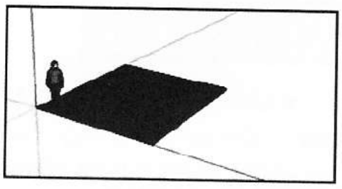
— Magenta line = something is parallel or perpendicular to an edge

Hold SHIFT to capture and lock an Inference.

4. Drawing a 2D Box

Click on the **Draw-a-Box** icon, then click on the origin, and while holding down the mouse, drag in this direction: → (Figure 2.2) .

The result:



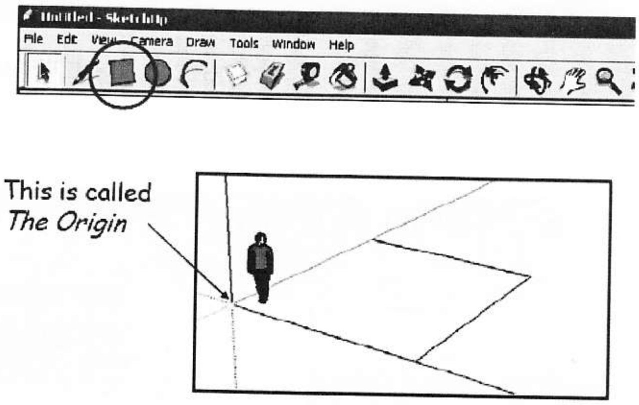
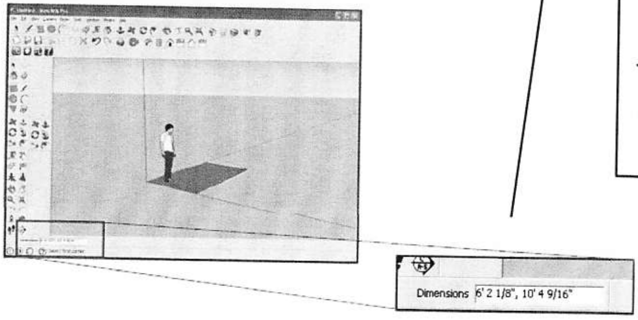


Figure 2.2.- Location of the **Draw-a-Box** icon

Notice the Bottom of the Screen

This is the **Measurement Toolbar**, or **MTB.** It is used to show you the dimensions, size, angle, etc. that you are currently setting
It can also be used to set exact values – just type into it while you are sizing with the mouse. But, if inputting length, be sure to use units: ' for feet and " for inches.



Dimensions 6' 2 1/8", 10' 4 9/16"

5. Extruding into a 3D Box

Click on the **Push/pull** icon (Figure 2.3), then click on the box you just created, and while holding down the mouse, drag in this direction↑

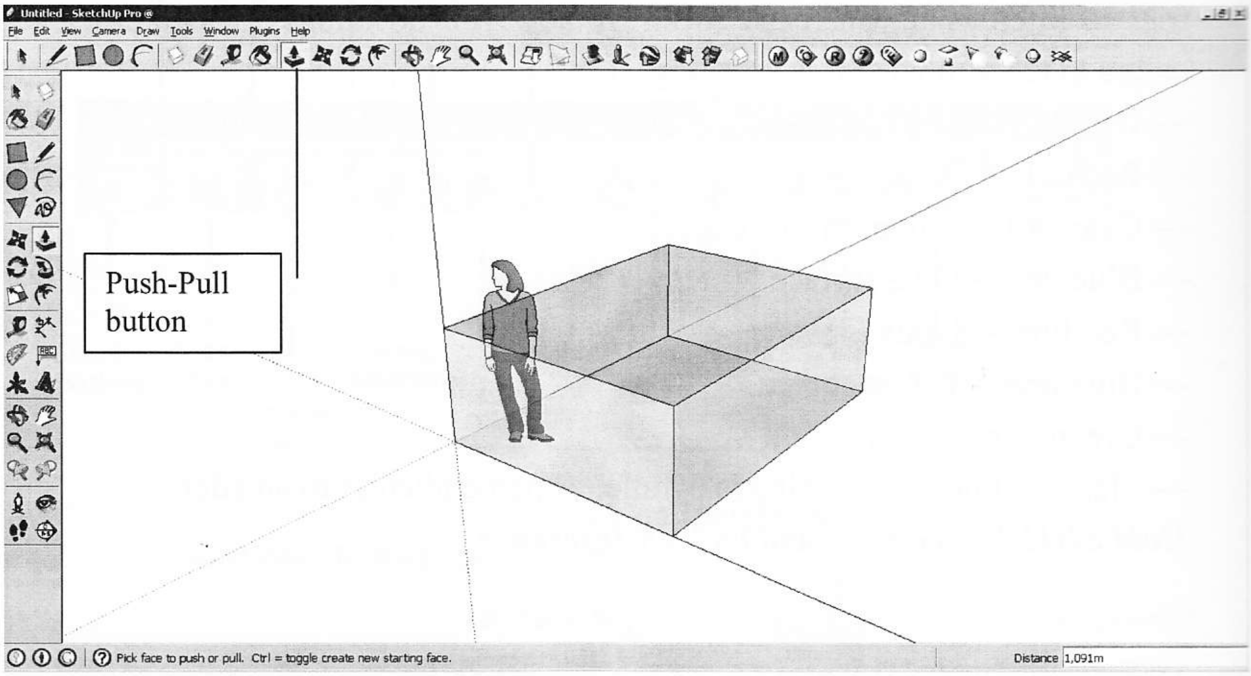
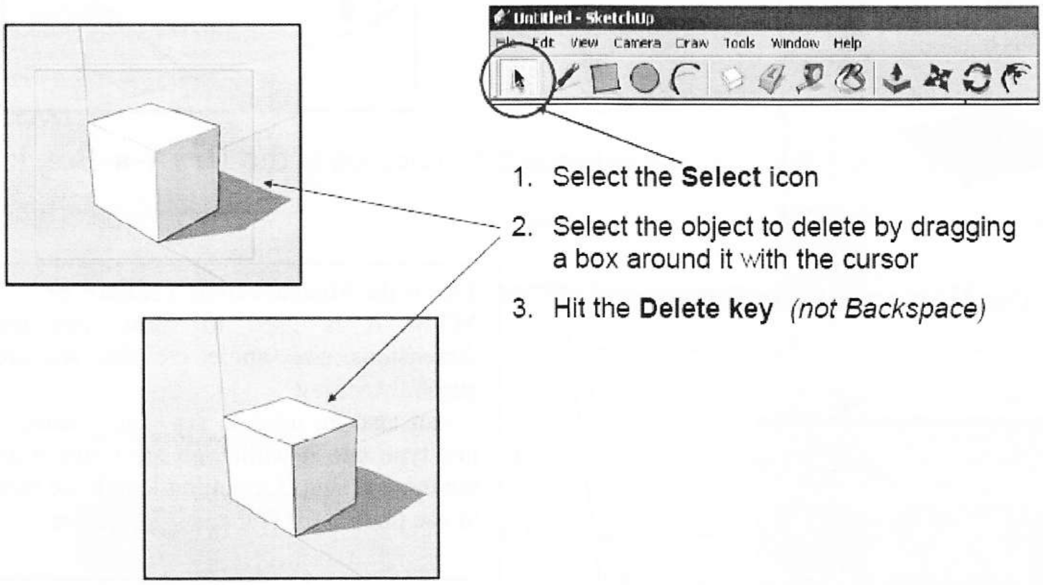

Untitled - SketchUp Pro
File Edit View Camera Draw Tools Window Plugins Help
Push-Pull button
Pick face to push or pull. Ctrl = toggle create new starting face.
Distance 1,091m

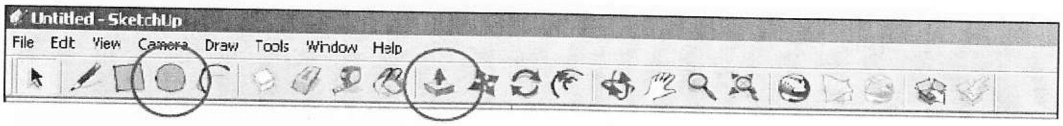
Figure 2.3.- Extruding into a 3D Box

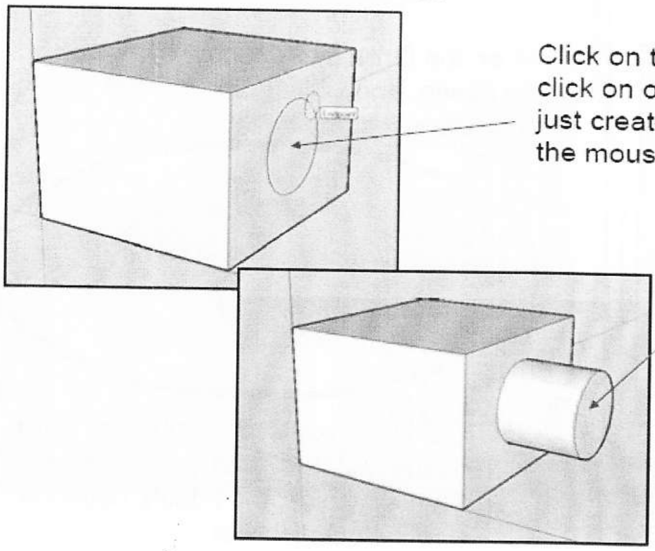
6. Deleting an Object



1. Select the **Select** icon
2. Select the object to delete by dragging a box around it with the cursor
3. Hit the **Delete key** *(not Backspace)*

7. Adding more detail to an existing face

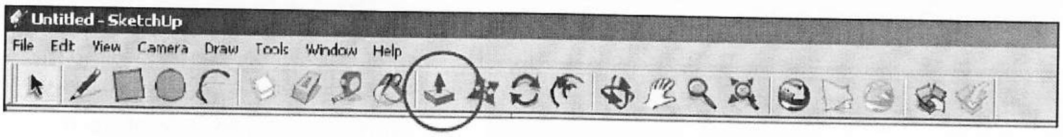


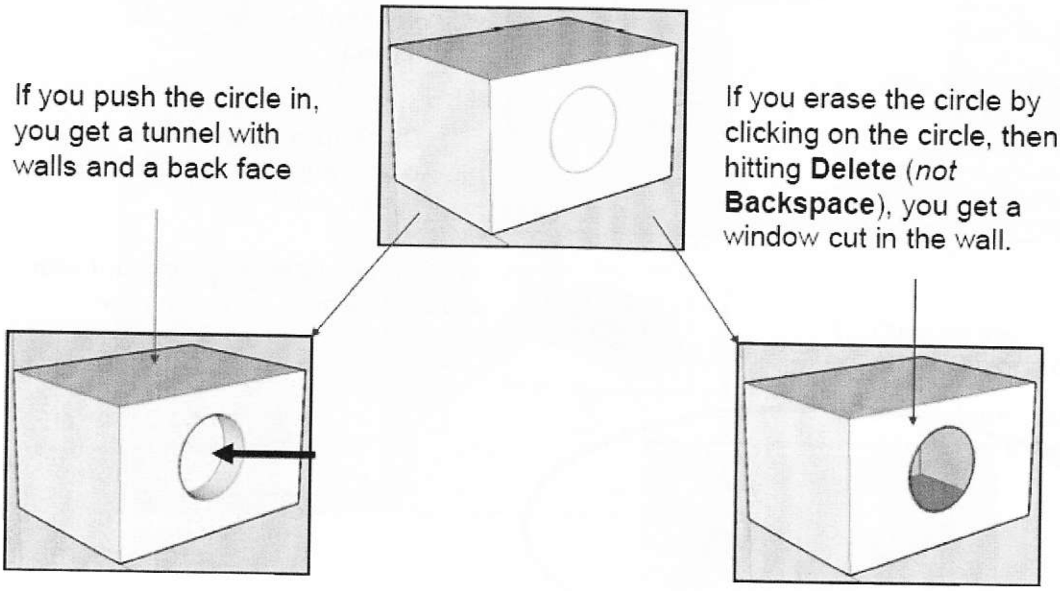


Click on the **Draw-a-circle** icon, then click on one face of the 3D solid you just created, and while holding down the mouse, drag in some direction

Click on the **Push/pull** icon, then click on the circle you just created, and while holding down the mouse, drag in this direction

Pushing a hole and cutting a hole

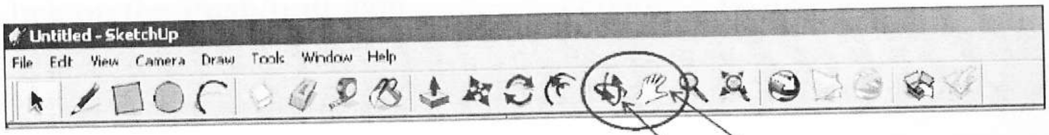


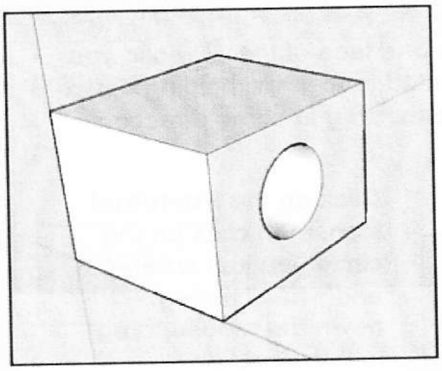


If you push the circle in, you get a tunnel with walls and a back face

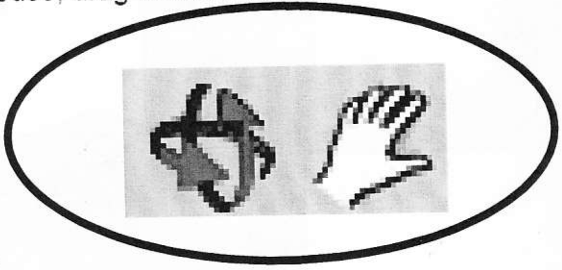
If you erase the circle by clicking on the circle, then hitting **Delete** (*not* **Backspace**), you get a window cut in the wall.

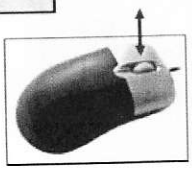
8. Change the point of view





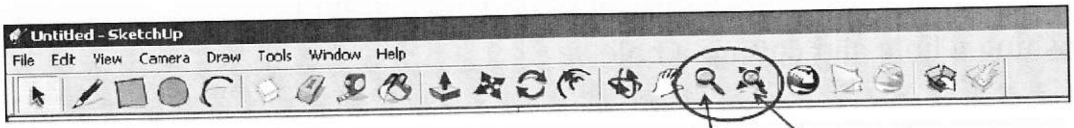
Click on the **Orbit** or **Pan** icon, then click in the scene, and while holding down the mouse, drag in some direction

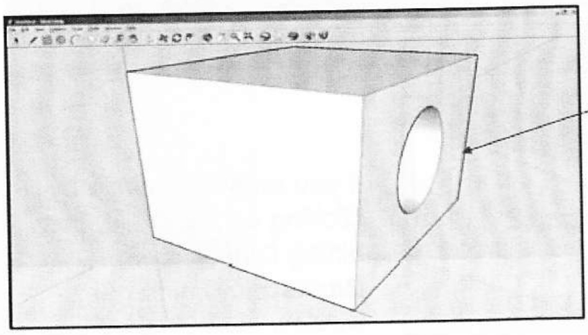




You can also **Orbit** by pushing down on the middle button on the mouse. On many mice, the middle button is also the scroll wheel.

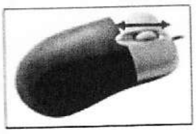
9. Zooming



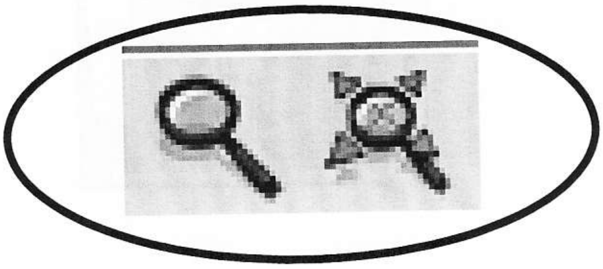


The **Zoom extents** icon will zoom in as much as possible without making any of your object disappear off the screen

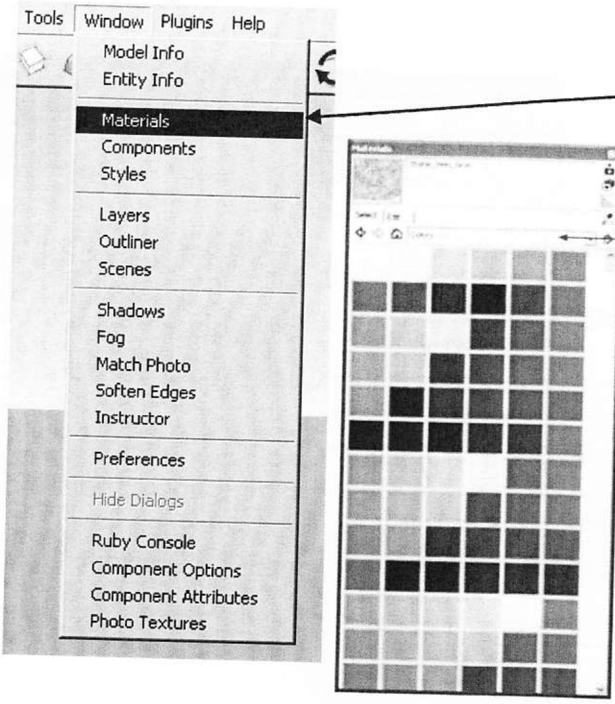
The **Zoom** icon will allow you to zoom as much or as little as you want



You can also **Zoom** in and out with the scroll wheel on the mouse



10.Coloring

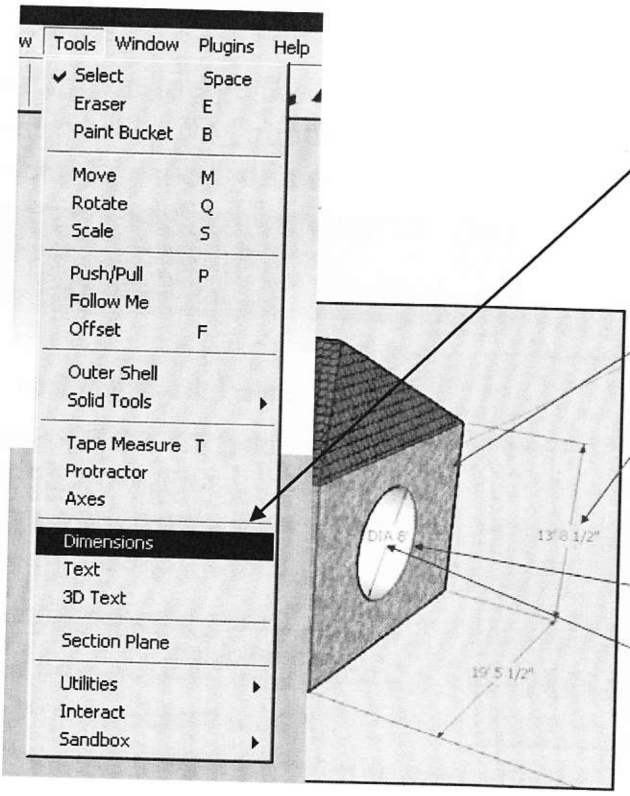


Click **Window** →**Materials**

1. Select **Colors**
2. Treat the color just like you did the material

Scroll up and down to get more colors

11.Dimensions



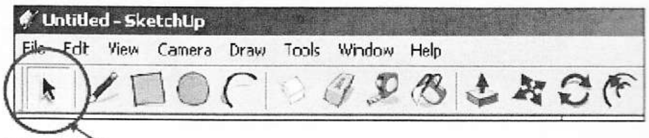
Click **Tools**→**Dimensions**

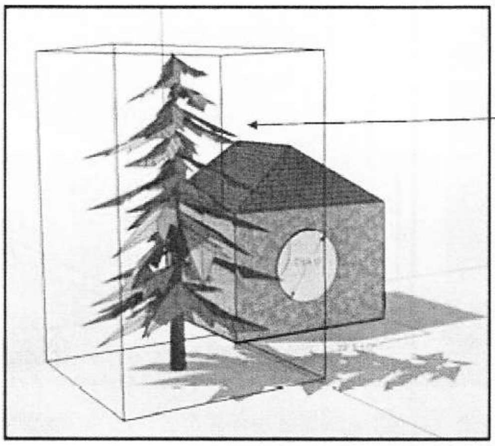
1. Click on an edge
2. Drag where you want the dimension to be drawn

Dimensions are useful if you are giving your design to someone so that they can build it

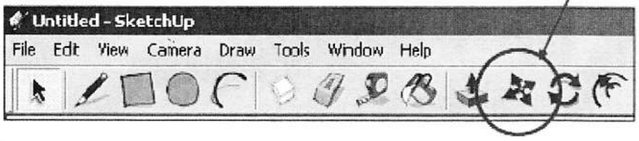
1. Click on the circumference of a circle
2. Drag where you want the dimension to be drawn

12.Moving and copying

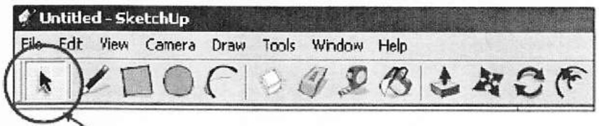


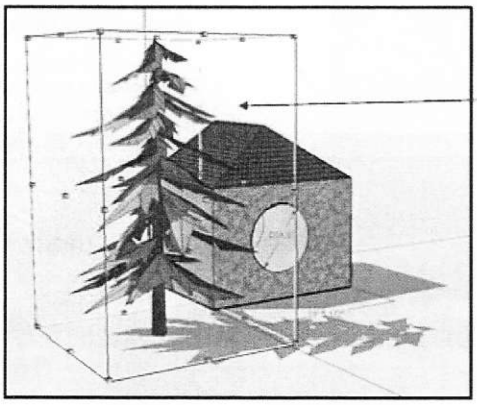


1. Select the **Select** icon
2. Select the object to move
3. Click the **Move** icon
4. Move the object
5. Holding down the **Control key** will result in a *Copy* instead of a *Move*



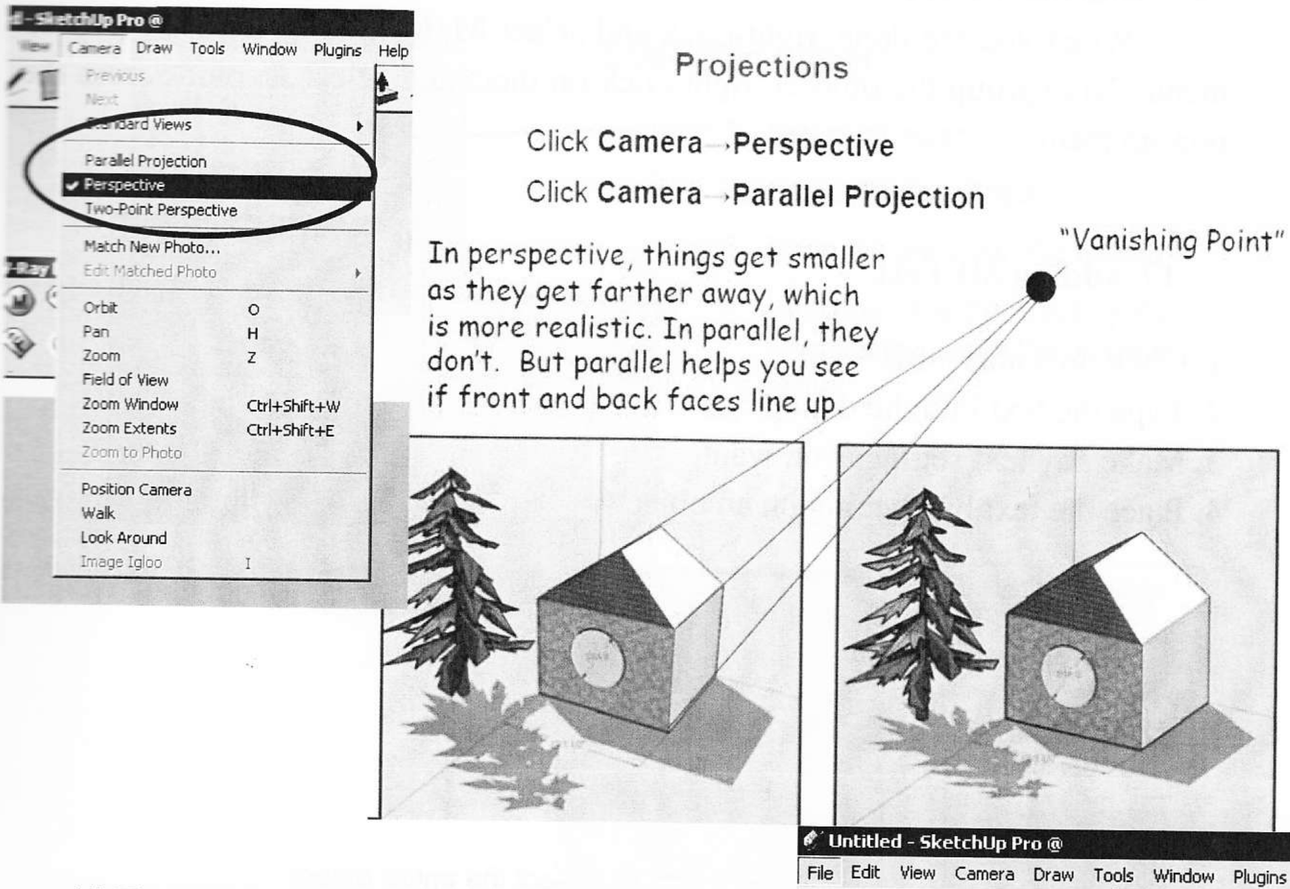
13.Scaling





1. Select the **Select** icon
2. Select the object to scale
3. Select **Tools**→**Scale** or click the **Scale** icon in the **Large Toolset** toolbar
4. Grab a grip point and scale the object

14.Perspective



Projections

Click **Camera** → **Perspective**

Click **Camera** → **Parallel Projection**

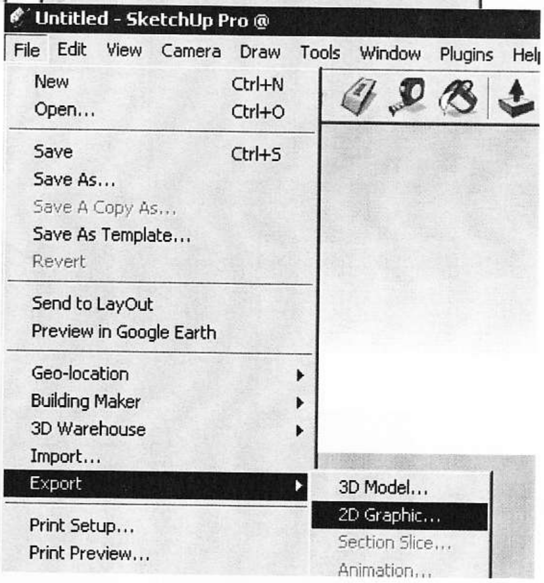
In perspective, things get smaller as they get farther away, which is more realistic. In parallel, they don't. But parallel helps you see if front and back faces line up.

15.Exporting an image file

Click File→Export →2D Graphic

Your image can be exported in one of 4 formats:

1. BMP
2. JPEG
3. TIF
4. PNG



16.Creating groups

Sometimes you would like to collect several pieces of geometry together and be able to treat them as a single unit (Figure 2.4). This is called a SketchUp **Group**. Create a SketchUp Group by first clicking on the **Select** icon.

Then, click on the first object you want in the Group. It will turn blue. Then, hold down the **Shift** key and click on all other objects you want in the group. They will also turn blue. If you select the wrong item, just click it again to un-select it.

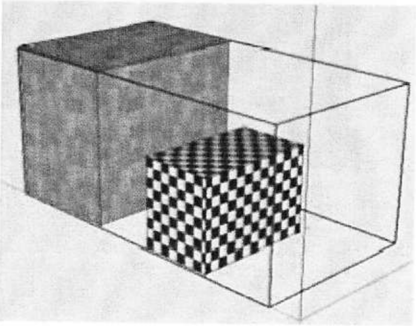


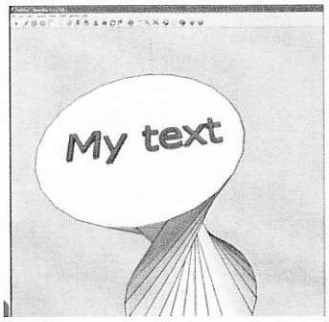
Figure 2.4.- A group

You can select many things at once by creating a rectangle around all of them with the Select cursor.

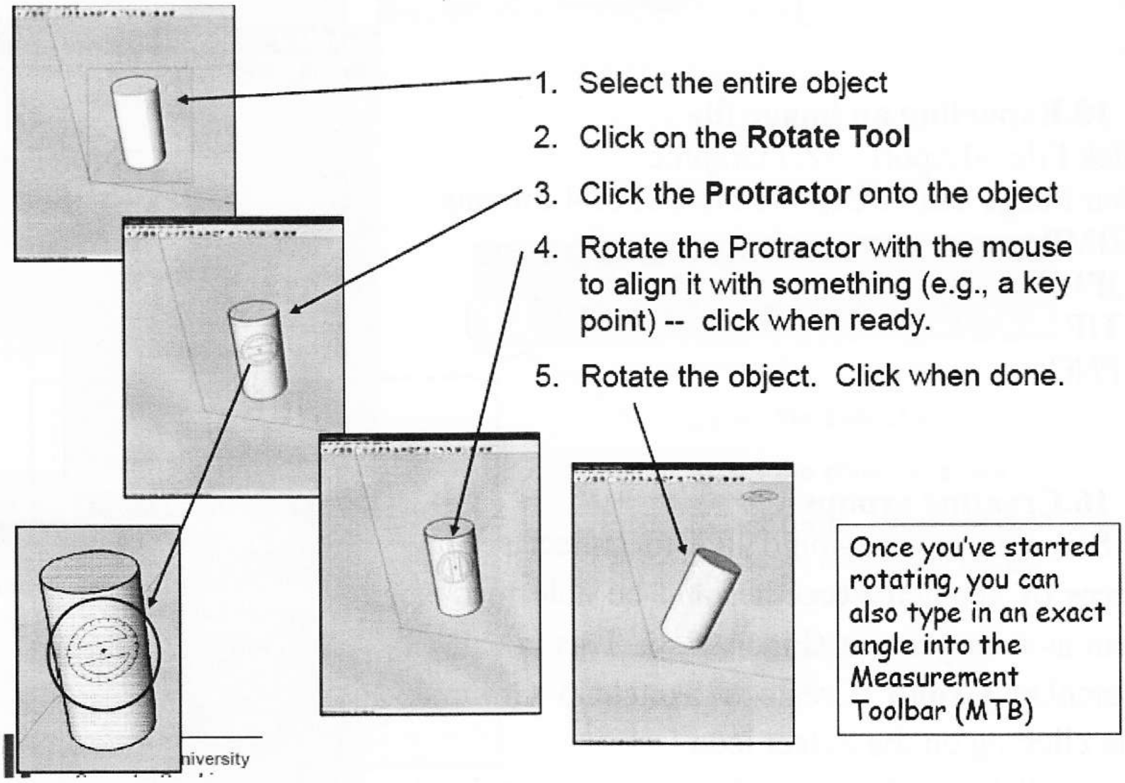
When you are done, right-click and select **Make Group** from the pop-up menu. To ungroup the objects, right click on them and select **Explode** from the pop-up menu.

17.Adding 3D Text

1. Click on **Tools→3D Text**
2. Type the text into the dialog box
3. Make any text settings you want
4. Place the text by clicking on an object



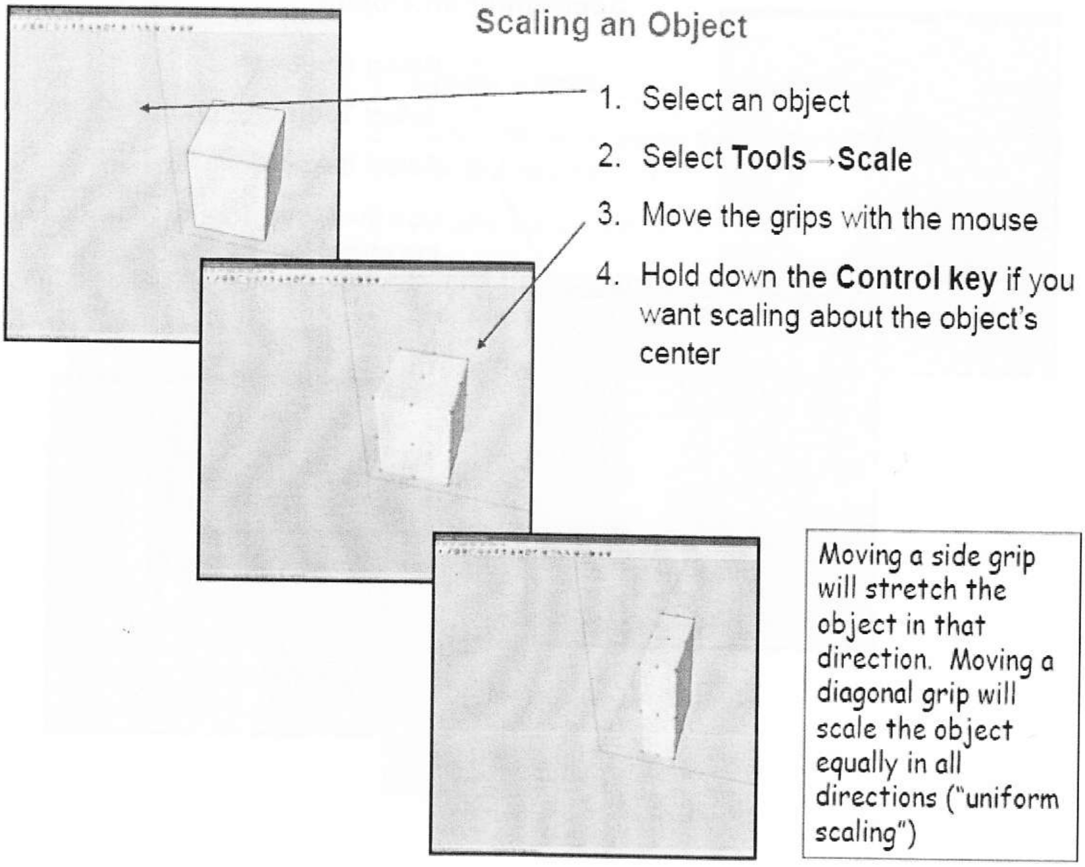
18.Rotation an object



1. Select the entire object
2. Click on the **Rotate Tool**
3. Click the **Protractor** onto the object
4. Rotate the Protractor with the mouse to align it with something (e.g., a key point) -- click when ready.
5. Rotate the object. Click when done.

Once you've started rotating, you can also type in an exact angle into the Measurement Toolbar (MTB)

19.Scaling an object



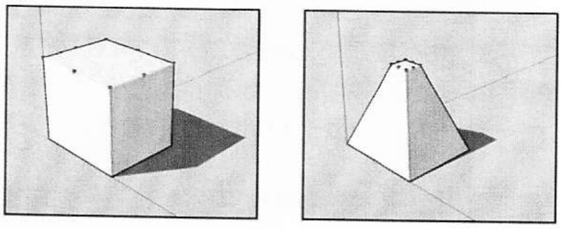
Scaling an Object

1. Select an object
2. Select **Tools**→**Scale**
3. Move the grips with the mouse
4. Hold down the **Control key** if you want scaling about the object's center

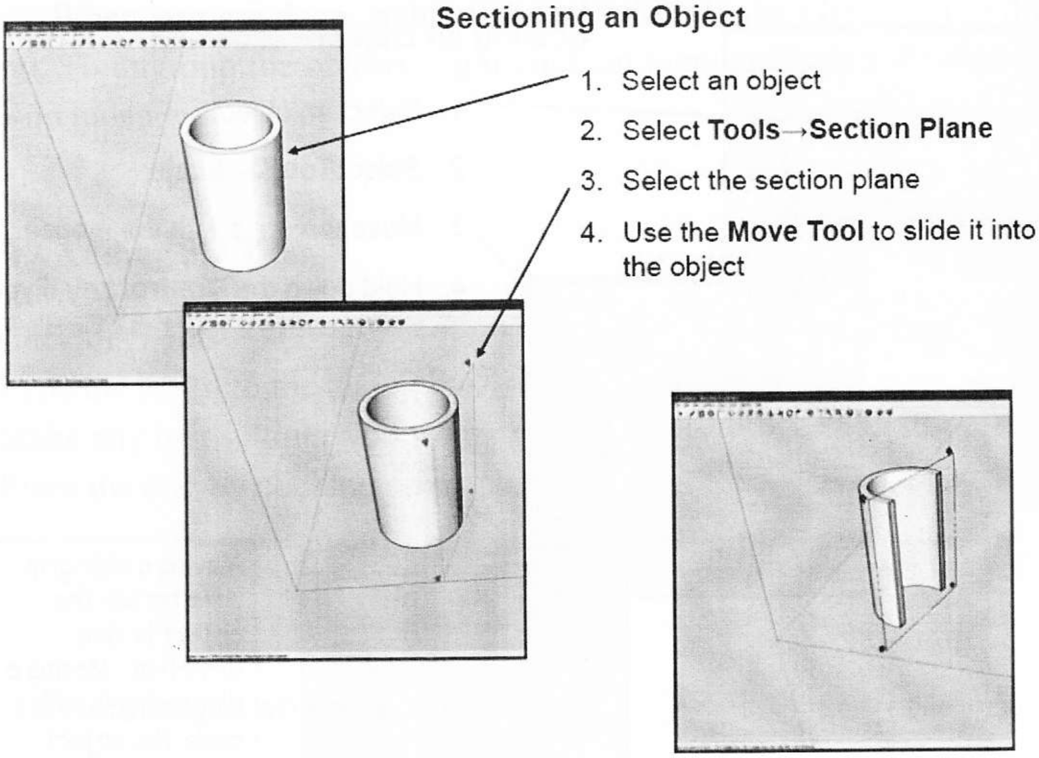
Moving a side grip will stretch the object in that direction. Moving a diagonal grip will scale the object equally in all directions ("uniform scaling")

20.Scale just a face

1. Select a face
2. Select **Tools→Scale**
3. Move the grips with the mouse
4. Hold down the **Control key** if you want scaling about the object's center



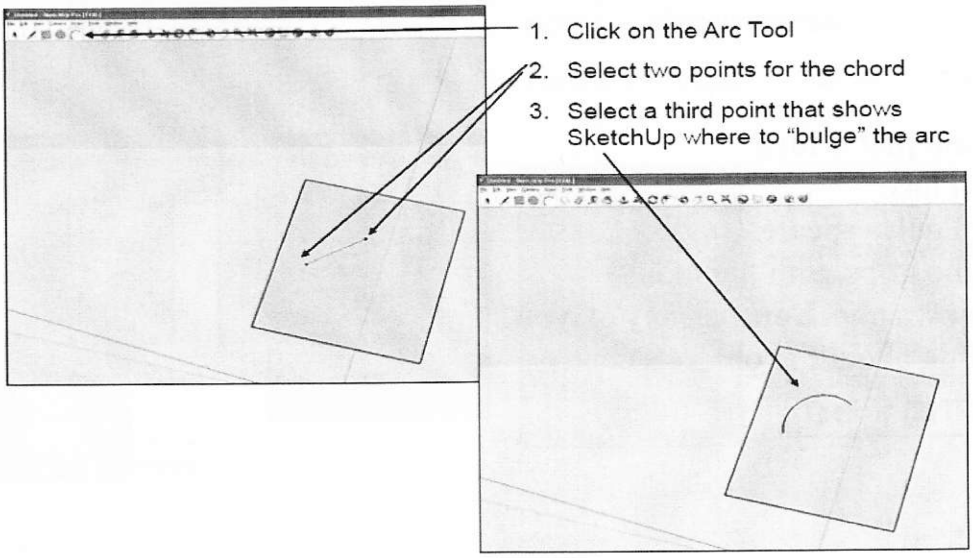
21.ctioning the object



Sectioning an Object

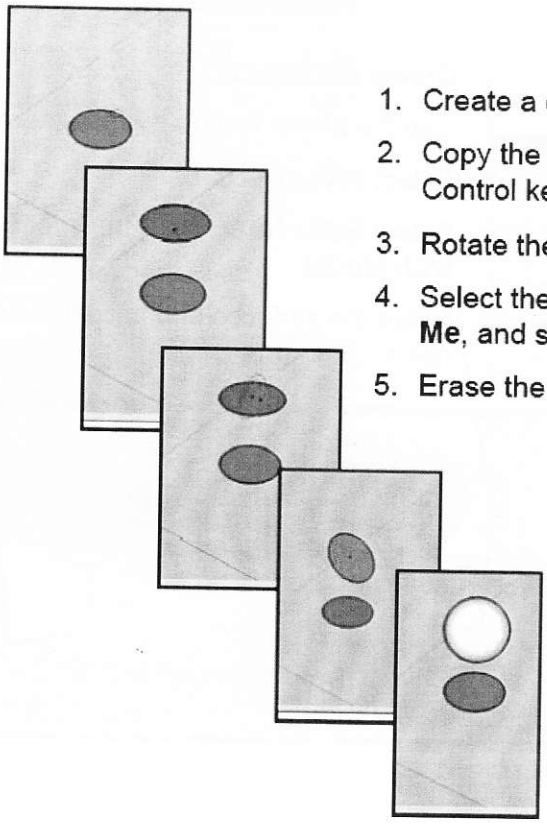
1. Select an object
2. Select **Tools→Section Plane**
3. Select the section plane
4. Use the **Move Tool** to slide it into the object

22.Drawing an arc



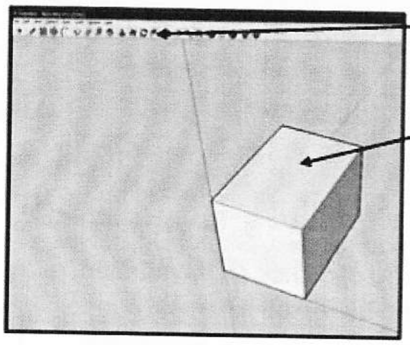
1. Click on the Arc Tool
2. Select two points for the chord
3. Select a third point that shows SketchUp where to "bulge" the arc

23.Drawing sphere

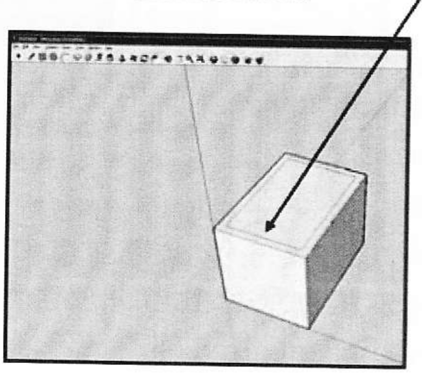


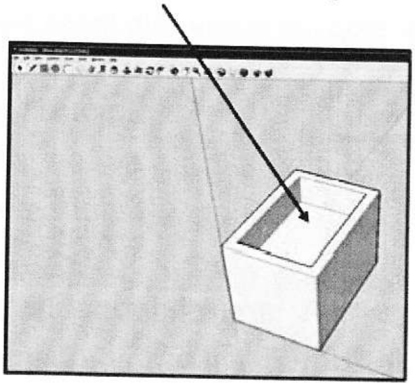
1. Create a circle
2. Copy the circle using the Move Tool with the Control key held down
3. Rotate the top circle 90°
4. Select the bottom circle, select **Tools→Follow Me**, and select the top circle
5. Erase the bottom circle and Delete it

24.Offsetting

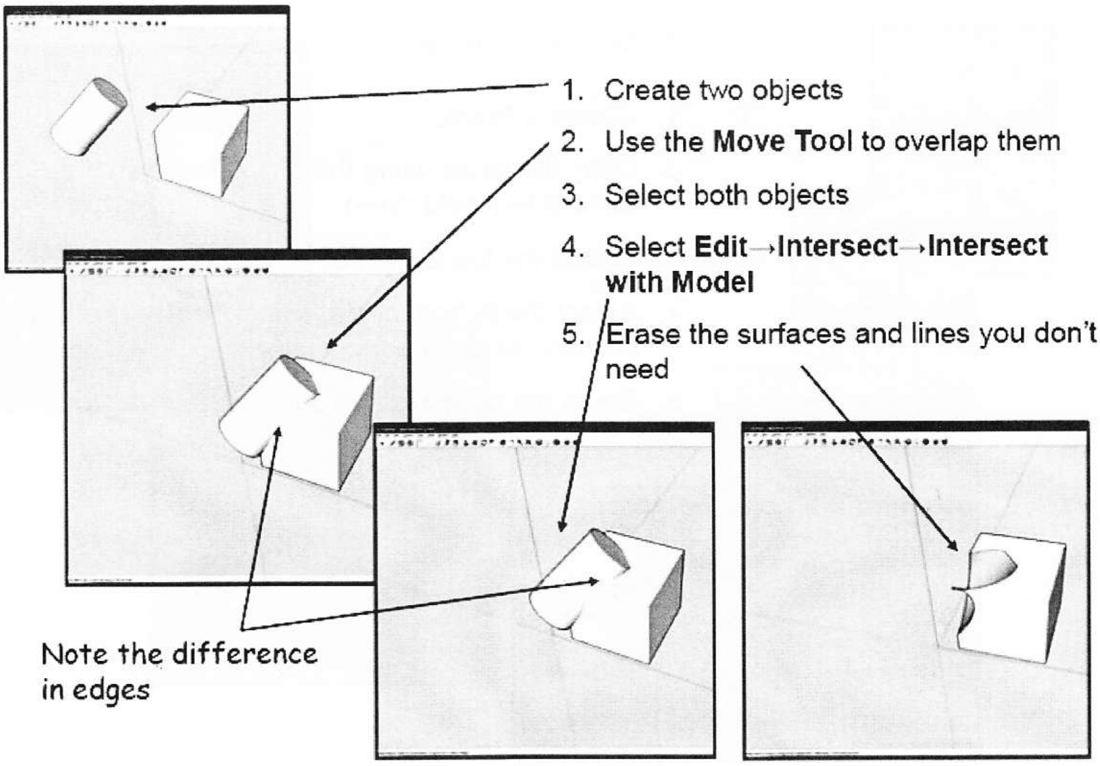


1. Click on the Offset Tool
2. Select the surface to offset on
3. Move the mouse to show how much to offset – click when ready
4. This only creates offset edges – you need to use the Push-Pull Tool to do something with it.





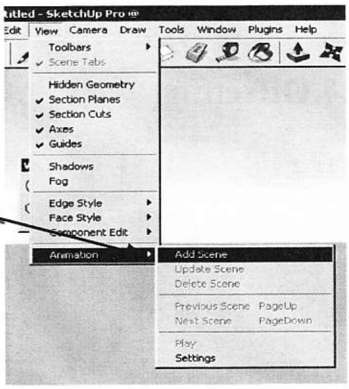
25.Subtracting one surface from another



1. Create two objects
2. Use the **Move Tool** to overlap them
3. Select both objects
4. Select **Edit→Intersect→Intersect with Model**
5. Erase the surfaces and lines you don't need

Note the difference in edges

26.Animation

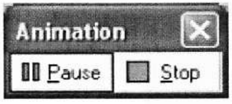


To create an animation:

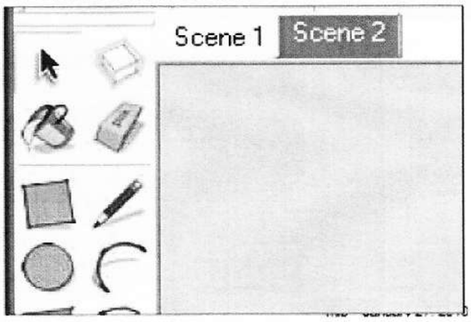
1. Create a view of the scene
2. Select **View→Animation→Add Scene**
3. Create a different view
4. Select **View→Animation→Add Scene**
5. . . .

To play the full animation:

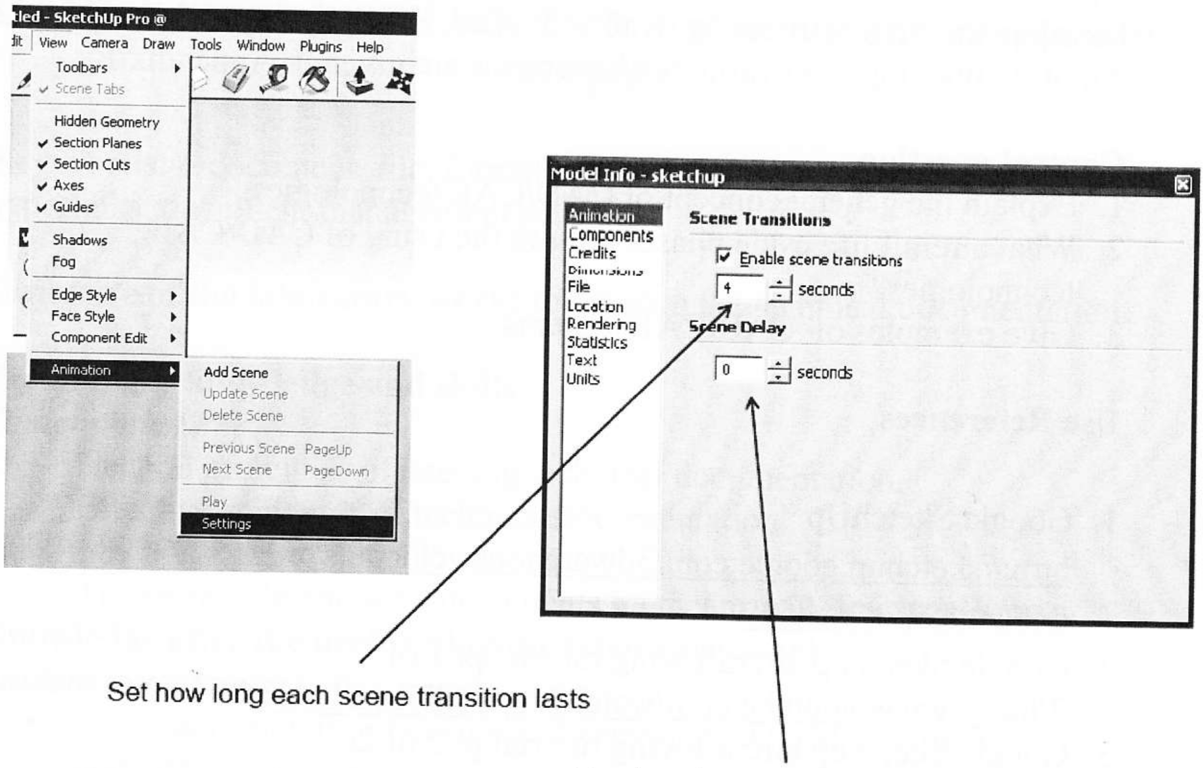
1. Select **View→Animation→Play**
2. Pause or stop the scene with these buttons



As you add scenes, SketchUp will list them. You can click on any of them to get back to that view.



27.Animation settings



Scene Transitions

☑ Enable scene transitions

4 seconds

Scene Delay

0 seconds

Set how long each scene transition lasts

How long to wait before starting the animation

28.Exporting animation

1. Select File→Export→Animation
2. Save as an AVI file

To save an animation to a file: Double-click on your AVI file

To play the animation file:

1. Select Insert→Movies and Sounds→Movie from File

To import your animation into PowerPoint:

1. Double-click on the image when editing the slide
2. Click on the image in Slide Show mode

Preliminary unsupervised student's work

1. To revise general technical drawing principles.
2. To familiarize with generic technology of an aircraft conceptual design.
3. Investigate the basic techniques of creating 3D models with Google SketchUp (use the sources, given on CD).
4. Prepare the draft for the 3D conceptual model design.
5. Investigate the basic techniques of creating drawings with AutoCad (use the sources, given on CD).

Progress of work

1. Develop the 3D model of the selected object with Google SketchUp.
2. Develop the simple drawing with AutoCad. The examples of simple 2D AutoCad drawings and variants of exercises are given in Appendix E.

Control questions

1. Explain the general concept of CAD/CAE technologies.
2. What aircraft life cycle phases require the using of CAD/CAE technologies?
3. Give examples of CAD/CAE systems.

Use References

1. Google SketchUp http://sketchup.google.com/3dwarehouse/cldetails?mid=8299198acb0062d65ded8939d9aed569
2. Google Sketchup aircraft wing tutorial pt 1 of 2/http://www.youtube.com/watch?v=pv7dULmJUiU
3. Google Sketchup aircraft wing tutorial pt 2 of 2/ http://www.youtube.com/watch?v=Jyj-eGo-uRQ
4. Aircraft Modeling Tutorial - Part 3 "Accuracy)/ http://www.youtube.com/watch?v=ZjdWNzGOHhU&feature=related
5. 5 Year Old Designing An Airplane on Google Sketchup / http://www.youtube.com/watch?v=Iwr8UXS-k1g
6. Useful Google Sketchup Plugins/ http://www.youtube.com/watch?v=5Fdo3telGaU&feature=related
7. Aircraft Modeling Tutorial - Part 2 "Tracing) / http://www.youtube.com/watch?v=yK8K1A9w_YM&feature=related
8. Google Sketchup tutorial : How to do a car in 3D./ http://www.youtube.com/watch?v=Hgrix3m119k&feature=related
9. how to make a car in google sketchup/ http://www.youtube.com/watch?v=GR6rgXxVGPw&feature=related
10. Building a Caravelle jet airliner model using Google Sketchup 8/ http://www.youtube.com/watch?v=7vH8BSCaS5M&feature=related
11. Mini tutorial: 2D to 3D using the free app Google SketchUp/ http://www.youtube.com/watch?v=1FJi1nmVGKg&feature=fvwrel

References to the Laboratory Work#2 also are given in the folder “Sources to the LW#2” on CD.

LABORATORY WORK # 3. Technical documents design and keeping

Key words: e-document, file, format of file, text documents, hypertext, hyperlinks, document management system

Objective of the laboratory work: to develop the technical documentation.

Preliminary knowledge and skills

Student must know:

- General principles of keeping technical documentation.
- Effective standards of technical documentation.

Student is to be able:

To create, edit and save documents by means of Microsoft Word.

Knowledge after the performing the laboratory work

Student must know:

- General principles of design and keeping technical electronic documentation.
- Formats of files for saving and keeping e-documents on computer.

Student is to be able:

- To design the templates of technical documents.
- To design, edit, save and arrange technical e-documents.

Theoretical background

Technical documentation follows the specific standards. First of all the technical document has to be of the proper structure. As usual it contains the following elements:

1. The title.
2. The abstract.
3. The contents.
4. Introduction.
5. The main body, subdivided on some parts.
6. Conclusions.
7. References.

The title is the first page of the document. It includes the actual title of the document, the name of the author, the year. The example of the Homework title is given in Appendix A.

Abstract represents the general idea of the document, the objectives, specific aspects presented, target groups. Abstract as usual is about 1 page.

The contents should be arranged in structure mode.

The introduction represents the results of the situation investigation, includes the problem statement, target settings and the expected approaches to the problem solving.

The main body of the document contains some parts, which elucidate the problem, methods of problem solving and estimating of the results.

Conclusions contain the most significant positions of the results, suggestion, guidelines.

References contain the list of information sources.

The In the second place is the paper work and making up of the e-document.

General principles of technical documentation making up are the next:

- Text and figures layout.
- Underlines.
- Pages orientation and numeration.
- Headers and footers.
- Fonts (type, color, size).
- Line spaces.
- Borders of the page.
- Indentation.

Table 3.1 gives the recommended settings.

Table 3.1

#	Parameter	Meaning
1.	Font	Times New Roman
2.	Size	14
3.	Borders of the page	Left – 2, upper – 2, lower – 2, right – 1.5
4.	Indentation	1.25
5.	Line space	1.15
6.	Header	Title of the document
7.	Footer	Page number

All pages of the technical documentation have frames.

The 1st page of the chapter has big frame, others - small (Appendix B).

Preliminary unsupervised student's work (4 hours)

1. Revise the skills of operating Microsoft Word (use resources to the 3rd Laboratory Work from CD).
2. Get to know the advanced MS Word tools (creating templates and contents).

3. To design the structure of the first Homework as technical document. The Homework #1 is the description of the selected aircraft life cycle conceptual model, including the 3D Sketch Up model, as the result of the preliminary design. Recommendations to the performing Homeworks are given in Appendix C.
4. To prepare the draft of the Homework #1. In order to do this it's necessary to collect the information for the performing the Homework #1, namely:
   - The current market situation and the position of the selected aircraft.
   - General operational characteristics of the selected aircraft and its positioning on the world marker (rating).
   - Main positions of the LC management strategy.
   - The ontology and onto graph of the problem domain.
   - The diagram of the LC conceptual model.
   - The steps of the 3D model design.
5. To prepare the abstract and introduction, to define and arrange the chapters as separate MS Word documents. Download and/or create drawings and pictures and save them as graphic files. Arrange the list of references.
6. To create the templates for the pages of the Homework #1.

Progress of work

1. To perform the integration of the technical document, using the results of the preliminary work.
2. At the end arrange the contents.

Control questions

1. Give the definition of key notions: : e-document, file, format of file, text documents, hypertext, hyperlinks, document management system.
2. Explain the structure of file specification.
3. Give examples of file formats.
4. Explain the general idea of hypertext.
5. Explain the general idea and functions of e-documents Management System

Information sources to the Laboratory Work #3 are given in the folder "Sources to the LW3" on CD, and Boeing 737 AMM CD.

LABORATORY WORK # 4. Developing the Computer Based Training System for technicians

Key words: computer based training, e-learning, WEB-based learning, distance learning

Objective of the laboratory work: to develop the electronic manual for aviation maintenance technicians.

Preliminary knowledge and skills

***Student must know*:**

1. General professional requirements to the aviation engineers and aviation maintenance technicians (AMT)
2. The notions of information space, problem domain, ontology.
3. The basic principles of developing ontology, algorithms and flow-charts.
4. General concept of WEB-based learning, e-learning, computer-based training

***Student is to be able*:**

1. To operate with Microsoft Word options, such as lists, tables and graphics.

Knowledge after the performing the laboratory work

Student must know:

1. General principles of creating hypertext documents.

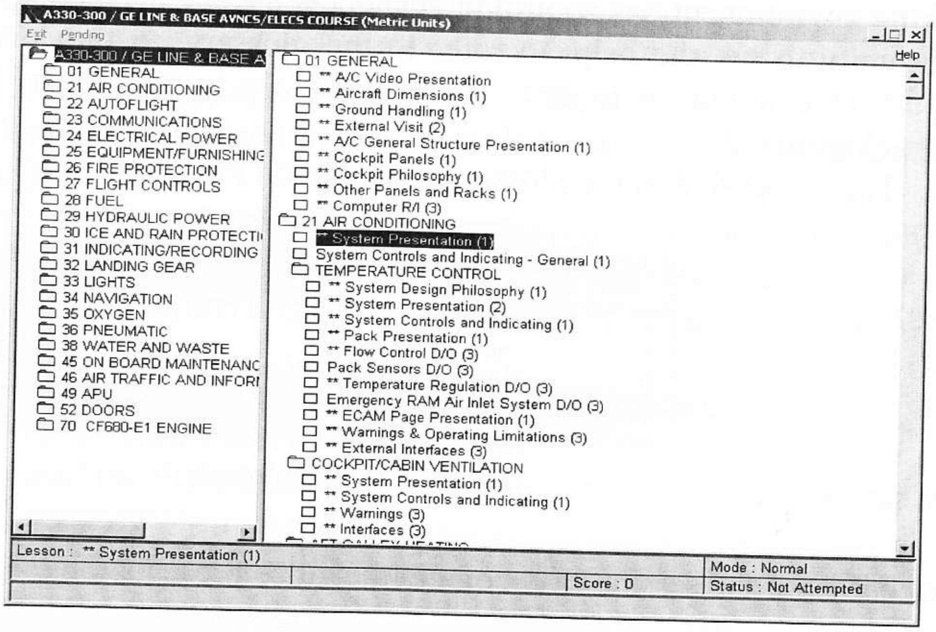
***Student is to be able*:**

1. To investigate the problem domain of the concrete technical operation task (for instance, handling or maintenance operations).
2. To develop the ontology of the problem domain.
3. To develop the algorithm of the operation.
4. To create the hypertext training manual on the selected operation.

Theoretical background

The CBT is a specific environment, which is realized by the mean of some computer system (software). The background of this system contains the multi-branched algorithm of training situations. The regular CBT system gives the opportunity to study the new theme in the interactive mode, using text, video, slideshow, verbal explanation. At the end of each theme the testing exercises are offered.

As usual the CBT session starts from the selection the current training theme. The list of themes is interactive, and it has hierarchical structure (Figure 4.1).



A330-300 / GE LINE & BASE AVNCS/ELECS COURSE (Metric Units)

A330-300 / GE LINE & BASE A
- 01 GENERAL
- 21 AIR CONDITIONING
- 22 AUTOFLIGHT
- 23 COMMUNICATIONS
- 24 ELECTRICAL POWER
- 25 EQUIPMENT/FURNISHING
- 26 FIRE PROTECTION
- 27 FLIGHT CONTROLS
- 28 FUEL
- 29 HYDRAULIC POWER
- 30 ICE AND RAIN PROTECTI
- 31 INDICATING/RECORDING
- 32 LANDING GEAR
- 33 LIGHTS
- 34 NAVIGATION
- 35 OXYGEN
- 36 PNEUMATIC
- 38 WATER AND WASTE
- 45 ON BOARD MAINTENANC
- 46 AIR TRAFFIC AND INFOR
- 49 APU
- 52 DOORS
- 70 CF680-E1 ENGINE

01 GENERAL
- □ ** A/C Video Presentation
- □ ** Aircraft Dimensions (1)
- □ ** Ground Handling (1)
- □ ** External Visit (2)
- □ ** A/C General Structure Presentation (1)
- □ ** Cockpit Panels (1)
- □ ** Cockpit Philosophy (1)
- □ ** Other Panels and Racks (1)
- □ ** Computer R/I (3)

21 AIR CONDITIONING
- □ ** System Presentation (1)
- □ System Controls and Indicating - General (1)
- TEMPERATURE CONTROL
  - □ ** System Design Philosophy (1)
  - □ ** System Presentation (2)
  - □ ** System Controls and Indicating (1)
  - □ ** Pack Presentation (1)
  - □ ** Flow Control D/O (3)
  - □ Pack Sensors D/O (3)
  - □ ** Temperature Regulation D/O (3)
  - □ Emergency RAM Air Inlet System D/O (3)
  - □ ** ECAM Page Presentation (1)
  - □ ** Warnings & Operating Limitations (3)
  - □ ** External Interfaces (3)
- COCKPIT/CABIN VENTILATION
  - □ ** System Presentation (1)
  - □ ** System Controls and Indicating (1)
  - □ ** Warnings (3)
  - □ ** Interfaces (3)

Lesson : ** System Presentation (1)

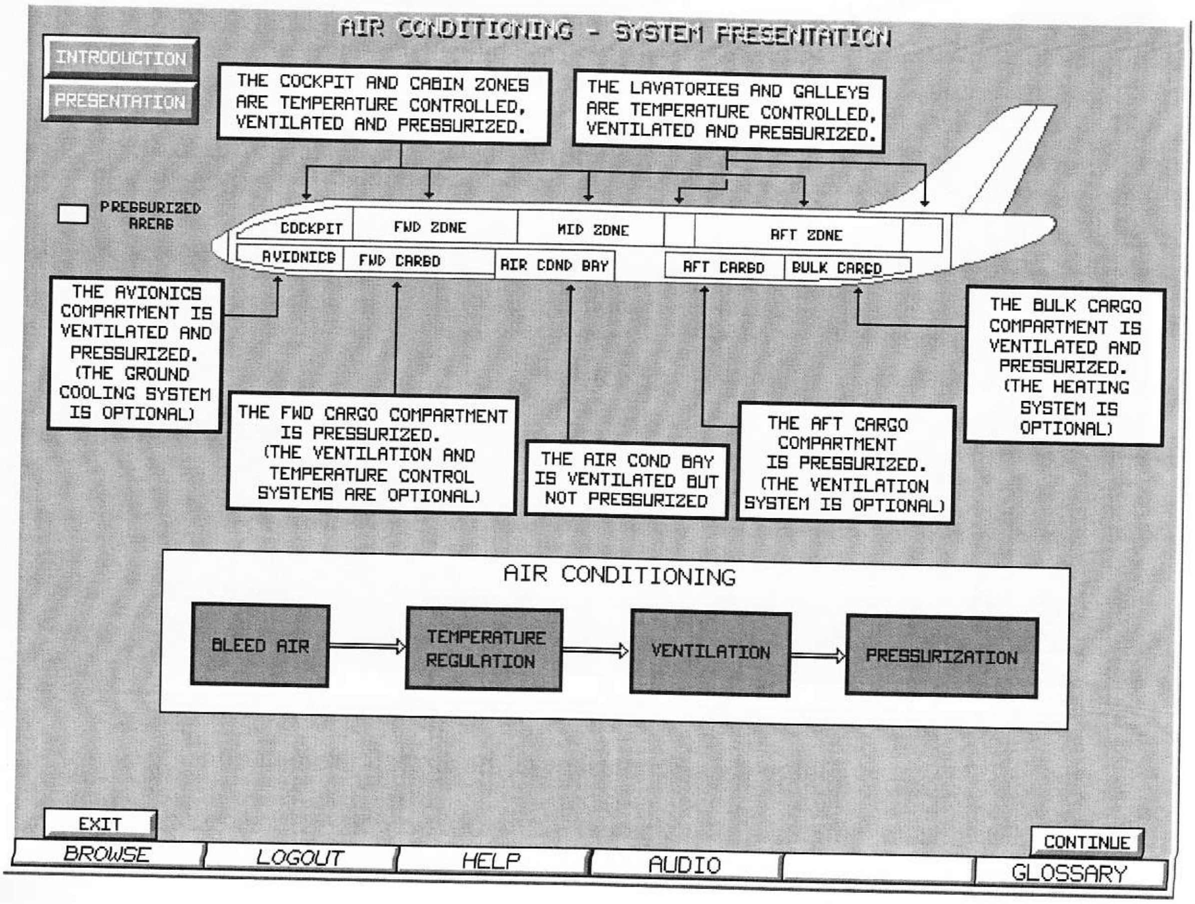
Score : 0

Mode : Normal

Status : Not Attempted

Figure 4.1.- List of themes

Each element of the structure has the hyperlink to the corresponding fragment of text, audio and video explanation (Figure 4.2).



AIR CONDITIONING - SYSTEM PRESENTATION

THE COCKPIT AND CABIN ZONES ARE TEMPERATURE CONTROLLED, VENTILATED AND PRESSURIZED.

THE LAVATORIES AND GALLEYS ARE TEMPERATURE CONTROLLED, VENTILATED AND PRESSURIZED.

THE AVIONICS COMPARTMENT IS VENTILATED AND PRESSURIZED. (THE GROUND COOLING SYSTEM IS OPTIONAL)

THE FWD CARGO COMPARTMENT IS PRESSURIZED. (THE VENTILATION AND TEMPERATURE CONTROL SYSTEMS ARE OPTIONAL)

THE AIR COND BAY IS VENTILATED BUT NOT PRESSURIZED

THE AFT CARGO COMPARTMENT IS PRESSURIZED. (THE VENTILATION SYSTEM IS OPTIONAL)

THE BULK CARGO COMPARTMENT IS VENTILATED AND PRESSURIZED. (THE HEATING SYSTEM IS OPTIONAL)

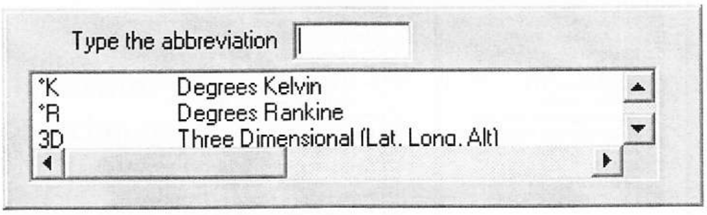
AIR CONDITIONING

BLEED AIR → TEMPERATURE REGULATION → VENTILATION → PRESSURIZATION

Figure 4.2.- The screenshot of the training session fragment

The arrangement of the training session demands to control it. For this the working environment has control elements (as usual – widgets), which give the opportunity to get the help, to refer to the glossary, to turn on/off the audio support, to continue or to exit. The widgets are located at the bottom of the screen (Figure 4.2).

The screenshot of the glossary is shown on Figure 4.3.

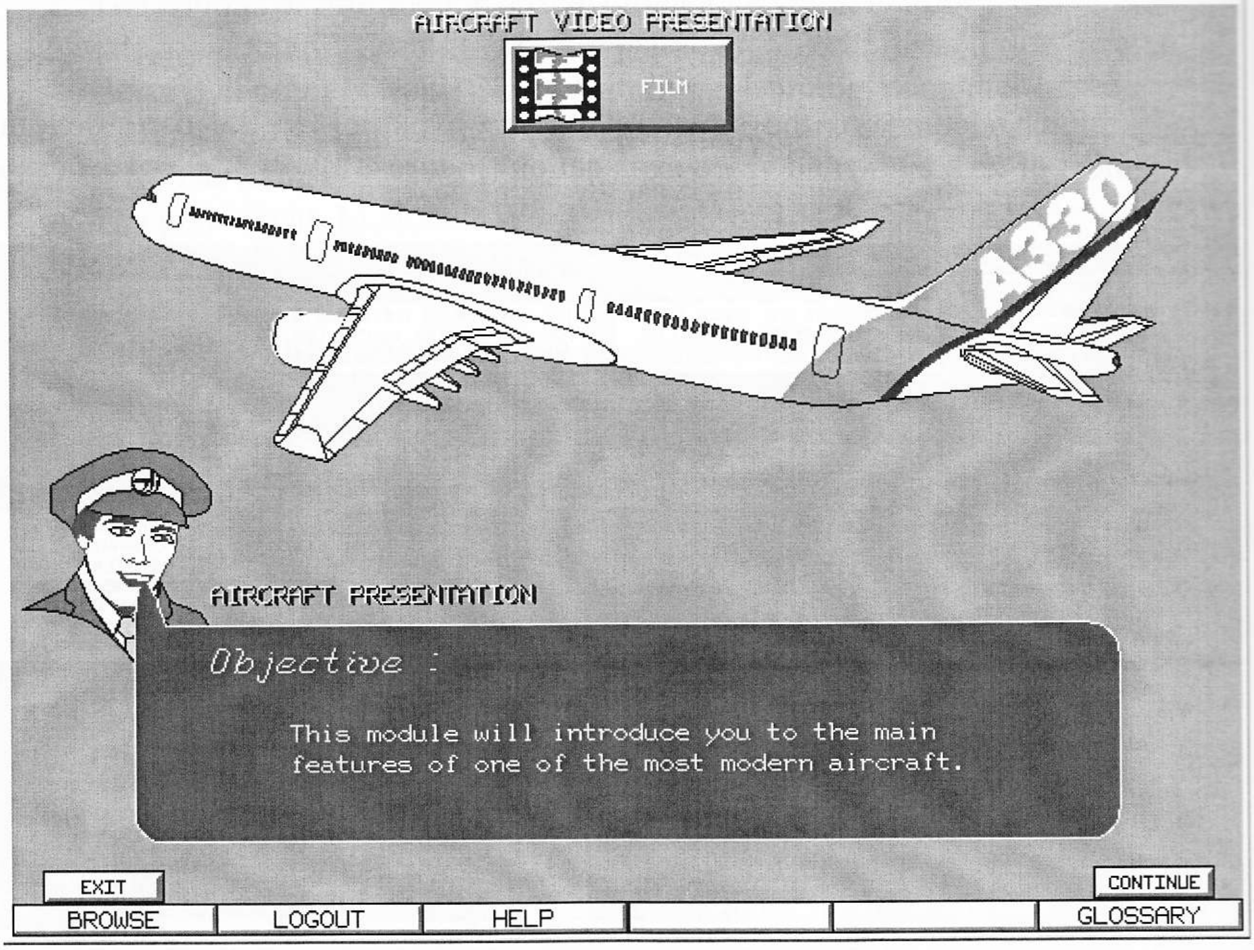


Type the abbreviation

°K	Degrees Kelvin
°R	Degrees Rankine
3D	Three Dimensional (Lat. Long. Alt)

Figure 4.3.- The glossary

The content of the session may contain the video presentation (Figure 4.4).



Objective :

This module will introduce you to the main features of one of the most modern aircraft.

Figure 4.4.- Screenshot of the aircraft presentation

Each theme ends with the quiz.

Preliminary unsupervised student's work (4 hours)

1. To revise the skills of operating Microsoft Word.
2. Get to know the advanced MS Word tools (creating hypertext documents, statistics).
3. To select one type of maintenance operations (transit check, daily check, aircraft handling operation etc.), performed on the selected aircraft.
4. To prepare the detailed text description of the operation as the MS Word file.
5. To create the ontology of the selected problem domain.
6. To develop the flow-chart of the operation.
7. To download the illustrations to the operation and save them as graphical files.
8. To define the key words of the theme.
9. To develop the conceptual model of the training process.
10. To develop the algorithm of the training process.
11. To develop the architecture of the training system. For this arrange the set of files, which contain the elements of the training material.
12. To prepare the questions to the final quiz.

Progress of work

1. Investigate the CBT technology, using the CBT systems for different aircraft.
2. Create the hypertext document, which realizes the links between the elements of the training material.
3. Develop the final quiz to the theme.

Control questions

1. Explain the general concept of computer-based training
2. Specify the general features of the CBT system.
3. Specify the technology of developing the e-learning manual.

Information sources to the Laboratory Work #4 are given in the folder "Sources to the LW4" on CD.

LABORATORY WORK # 5. Developing the database

Key words: data, structure of data, types of data, field, record, table, query, entities, attributes, data model, database

Objective of the laboratory work: to develop the aircraft database.

Preliminary knowledge and skills

***Student must know*:**

- Types of information and structures of data.
- Notions of table and operations with tables.

***Student is to be able*:**

- To make the analysis of data and data classification.
- To arrange data into tables.

Knowledge after the performing the laboratory work

Student must know:

- The general concept of database.
- Database classification.
- General principles of developing data models.

***Student is to be able*:**

- To develop the data model.
- To transform the data model into tables.
- To develop the database by means of Microsoft Access.

Theoretical background

A database is an *ordered collection* of *related data elements* intended to meet *the information needs* of an organization and designed to be *shared* by multiple users.

Note the key terms in the definition:

Ordered collection. A database is a collection of data elements. Not just a random assembly of data structures, but a collection of data elements put together deliberately with proper order. The various data elements are linked together in the most logical manner.

Related data elements. The data elements in a database are not disjointed structures without any relationships among them. These are related among themselves and also pertinent to the particular organization.

Information needs. The collection of data elements in a database is there for a specific purpose. That purpose is to satisfy and meet the information needs of the organization.

Shared. All authorized users in an organization can share the information stored in its database. Integrated information is kept in the database for the purpose of sharing so that all user groups may collaborate and accomplish the organization's objectives.

A database is a simple, yet flexible and powerful tool for storing and retrieving data. Every company, every website, has lots of data. The more of your data that you keep in your database - the better.

Whichever database you use - it'll be a *relational* database. This is the industry standard design these days. Relational databases use the principles of set theory. Set theory is a field of mathematics that describes how to deal with sets of data.

Here are the basics:

- All data is held in *tables.*
- A table has *columns -fields* (along the top) and *rows - records.*
- You create the tables you need. You define the *table names.*
- You define what the *fields names* are in each table.
- You define what *type* of data the fields contain.

Types of data

There are a number of different data types available which represent the different aspects of the real life. They are:

- Numerical types, i.e. numbers. There are fundamentally two types: integer and float. Integers are whole numbers; floats are numbers with decimal places.
- String types, i.e. text. There are two types: fixed length, and variable length.
- Date and Time types.
- Binary data. This is arbitrary data, could be images, programs, absolutely anything.

Often it is necessary to specify the ***precision*** of the fields too. This is the number of digits / decimal places / characters etc.

Also it is necessary to define which column is the ***primary key***. This is the column that is *guaranteed different* for all the rows in the table. In the example table above, it would be the 'order_id' column.

You populate the tables with ***rows*** of ***data***. Tables are ***related*** to each other by the *columns they have in common*-hence *relational* database!

The database developing process is composed of the specified steps.

The first step is the problem domain definition.

The second step is the design of the problem domain ontology.

The third step is the identification of the necessary data (entities) to be stored in the DB, concerned with the problem domain, and defined by the ontology.

The fourth step is the definition of the relations between entities.

The fifth step is the developing of the data model (entity-relational diagram).

The six step is the DB program implementation.

Entity-Relationship (ER) Diagrams

An Entity-Relationship (ER) diagram provides a graphical model of the things that the organization deals with (entities) and how these things are related to one another (relationships). An ER diagram is a high-level, logical model used by both end users and database designers to document the data requirements of an organization. The model is classified as "high-level" because it does not require detailed information about the data. It is called a "logical model" because it provides a conceptual understanding of the data and as opposed to actually defining the way the data will be stored in a database (which is referred to as the "physical" model).

Entities and Attributes

Entities are things. They can be tangible things, like a classroom or a part, or they can be intangible things, like a purchase or a meeting. Attributes (or fields) represent facts about an entity.

For example, an Employee entity might be described by social security number, name, and address attributes.

Entities are described by at least two attributes. One attribute is a unique identifier (often referred to as the key field). The value of this attribute must be unique for each occurrence of an entity. For example, if there were dozens of employees, then there would be no duplicate values of the key field for any of the employees. In addition to the key field, there must be at least one other attribute that provides a fact about an entity. That is, an entity would never be described solely by a key field.

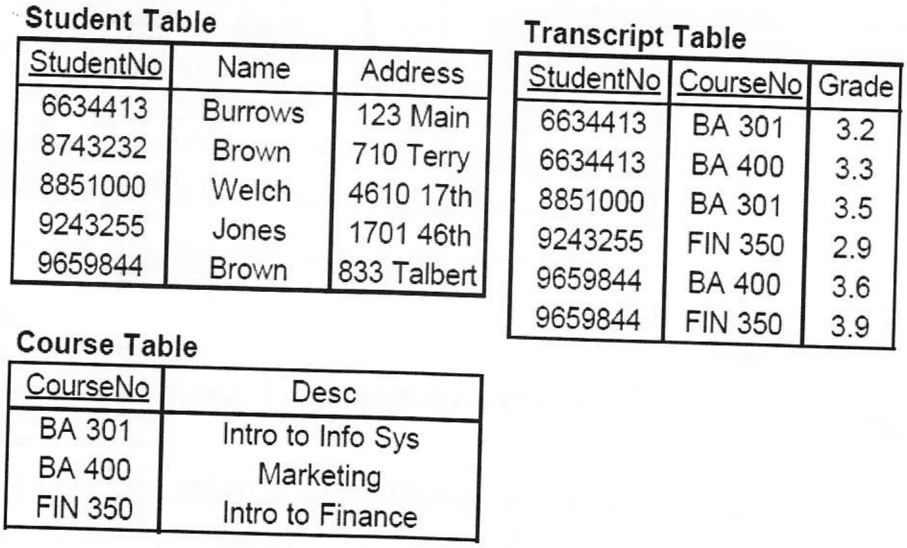
Relationships

Different entities can be related to one another. For example, in a university, the Student and the Course entities are related because students "enroll in" courses. A database must store not only information about the Student and Course entities, but it must also store the relationship between Student and Course. For example, the user of the database should be able to ask for a list of courses taken by a specific student or ask for a list of students currently enrolled in a specific course.

The relationship between a Student and Course is called a binary relationship because it relates to two entities. A relationship between three entities is called a ternary relationship; when four or more entities participate in a relationship, it is generally called an n-ary relationship. We restrict our discussion here to binary relationships but you should be aware that more complex relationships exist.

Relational Databases

A relational database is made up of a set of tables (files). For example, Figure 5.1 shows three tables from a hypothetical relational database that supports a many-to-many relationship between students and courses.



Student Table

StudentNo	Name	Address
6634413	Burrows	123 Main
8743232	Brown	710 Terry
8851000	Welch	4610 17th
9243255	Jones	1701 46th
9659844	Brown	833 Talbert

Transcript Table

StudentNo	CourseNo	Grade
6634413	BA 301	3.2
6634413	BA 400	3.3
8851000	BA 301	3.5
9243255	FIN 350	2.9
9659844	BA 400	3.6
9659844	FIN 350	3.9

Course Table

CourseNo	Desc
BA 301	Intro to Info Sys
BA 400	Marketing
FIN 350	Intro to Finance

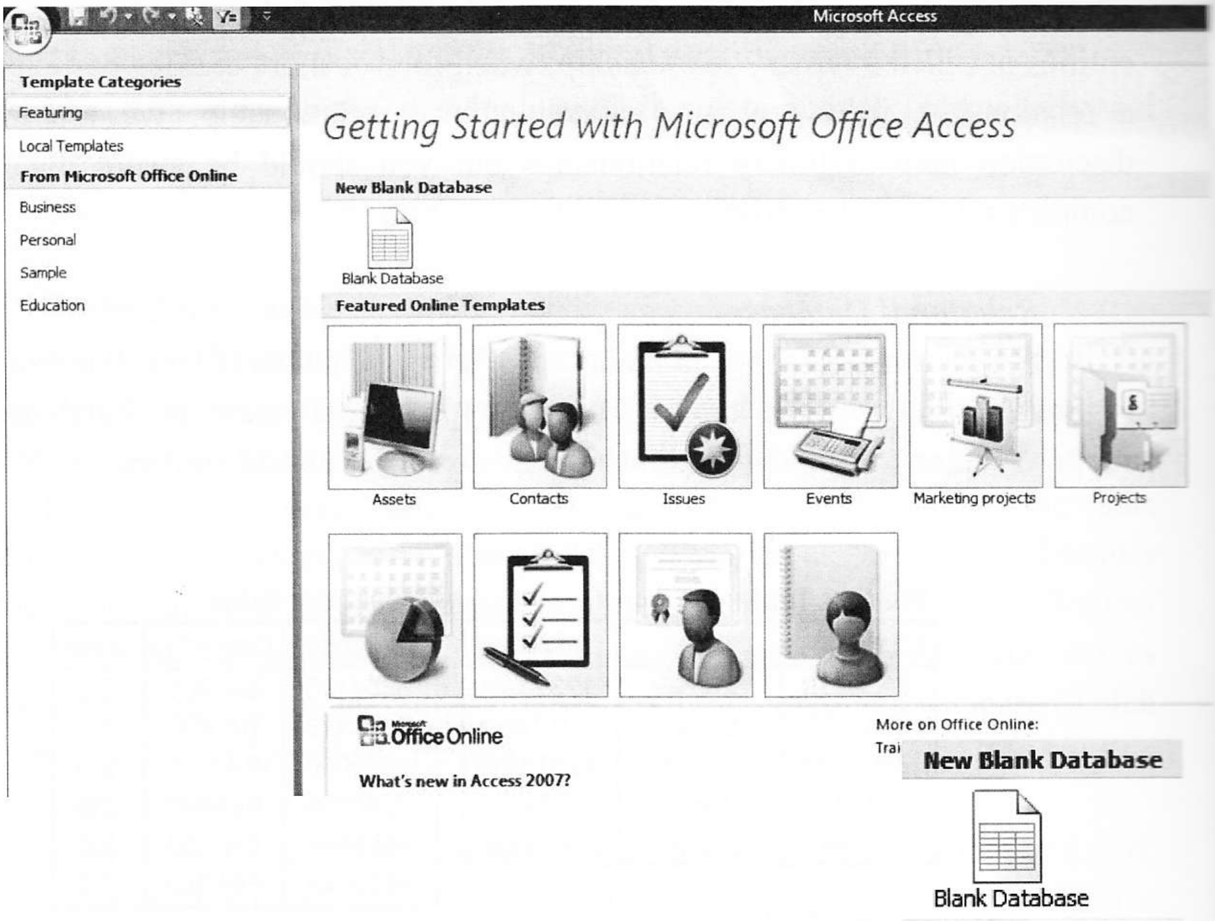
Figure 5.1.- Tables of the database

Each table is made up of a series of rows (records) and attributes (fields). Key fields, whose values are unique from one row to another, are underlined in the figure.

When converting an ER diagram into a relational database, each entity becomes a separate table. In addition, attributes must be added to the tables and new tables must be created so that the sets of relationships inherent in the ER diagram are maintained. Finally, a key field must be defined for each table.

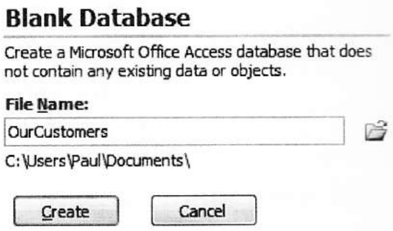
Microsoft Access basics

Load up MS Access 2007 and the opening screen will appear.



Click the option 'New Blank Database'

At the bottom right of the screen enter a name for the new database and then click the create button.

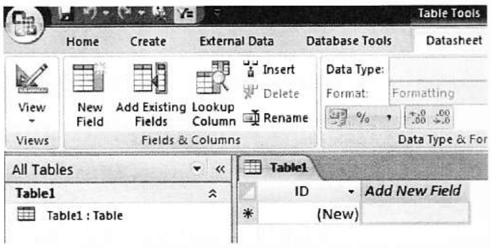

Blank Database

Create a Microsoft Office Access database that does not contain any existing data or objects.

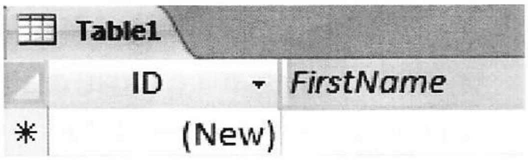
File Name: OurCustomers

C:\Users\Paul\Documents\

MS Access will now open at a screen for you to enter some data.



You could at this point double click the 'Add New Field' heading and then type in a column name.



Then click in the actual cell and type in some data.

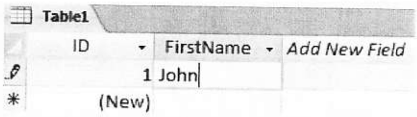
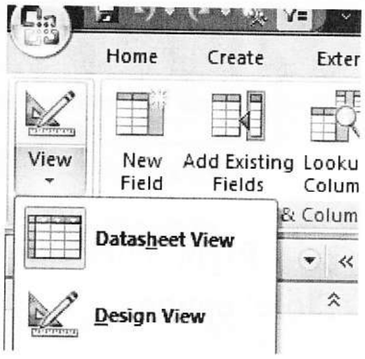


Table1

	ID	FirstName	Add New Field
	1	John	
*	(New)		

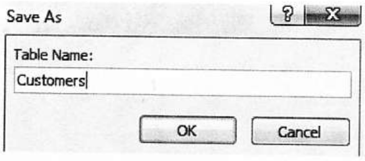
At the top left click the 'View' button and select the 'Design View' option.

This will take you into design view of the table.



Access uses tables to store the data. A table in Access is a container with columns such as name, address etc. Tables are the most important object in MS Access and are a useful starting point for any new database.

Access will ask you to save the table and to enter a name for it. Enter a name and click the ok button.

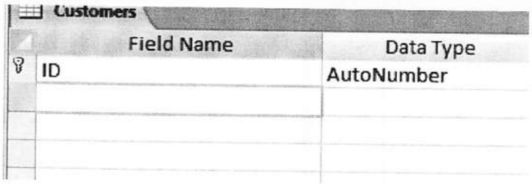


Save As

Table Name: Customers

OK Cancel

Now you are in table design view. The two columns to be concerned with are 'Field Name' and also 'Data Type'. Access has already created a column called 'ID'. This is fine and should be left here.



Customers

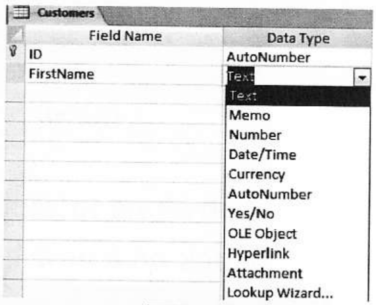
Field Name	Data Type
ID	AutoNumber

Click on the second line and enter a column name. I will be entering 'FirstName'. You can add a space between the words, but there are good reasons not to.

Tab along to the 'Data Type' column and click the down arrow.

You will then see a list of possible data types to use for the 'FirstName' column.

In this case for a 'FirstName' the most suitable data type would be 'Text'.

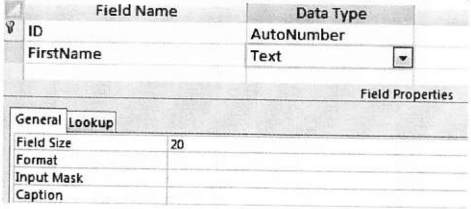


Customers

Field Name	Data Type
ID	AutoNumber
FirstName	Text

Text
Memo
Number
Date/Time
Currency
AutoNumber
Yes/No
OLE Object
Hyperlink
Attachment
Lookup Wizard...

At the bottom of the screen you can set how many characters should be entered for this 'FirstName' column. I set it to 20.



Field Name	Data Type
ID	AutoNumber
FirstName	Text

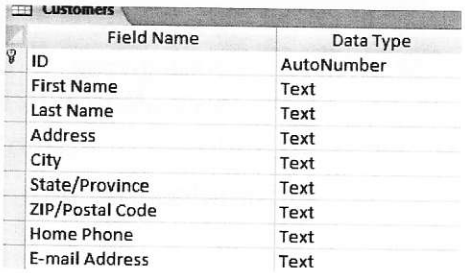
Field Properties

General Lookup

Field Size	20
Format	
Input Mask	
Caption	

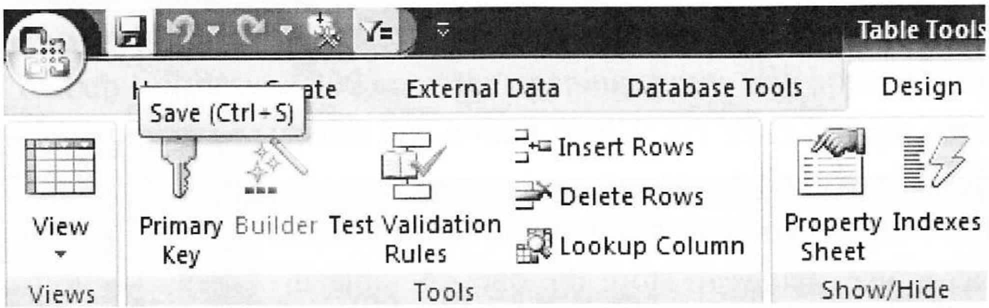
Enter some more columns and choose the appropriate data type and column size.

When you are done click the save button.

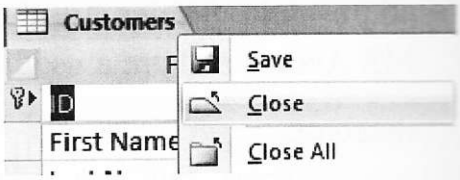


Customers

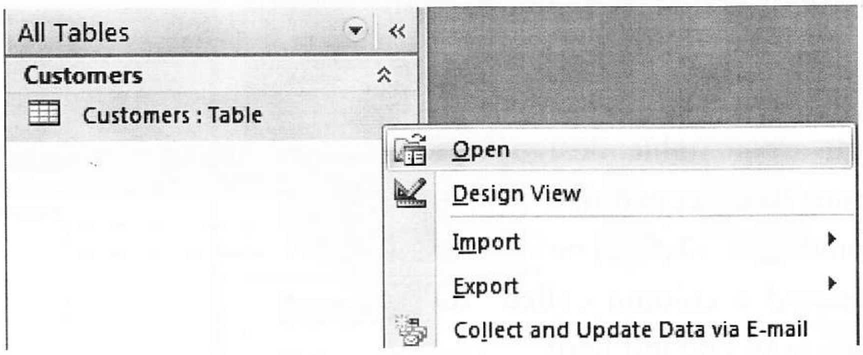
Field Name	Data Type
ID	AutoNumber
First Name	Text
Last Name	Text
Address	Text
City	Text
State/Province	Text
ZIP/Postal Code	Text
Home Phone	Text
E-mail Address	Text



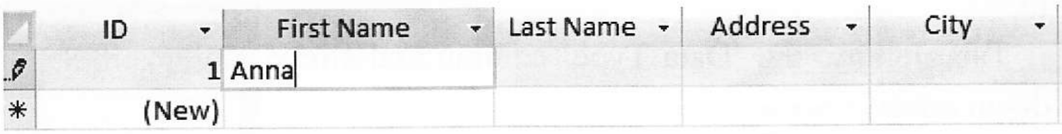
Right click the table tab and select the 'Close' option.



Right click the table and select the 'Open' option.



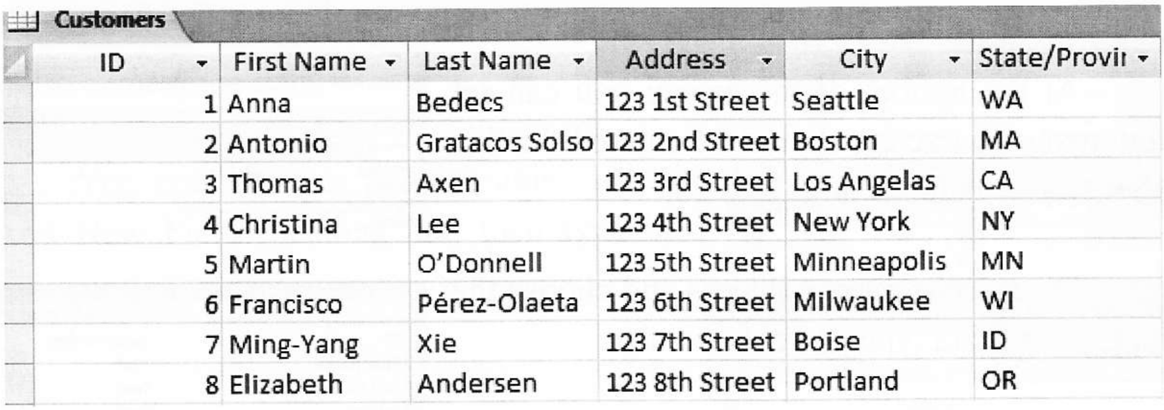
Click in the 'FirstName' column and enter some data.



ID	First Name	Last Name	Address	City
1	Anna			
(New)				

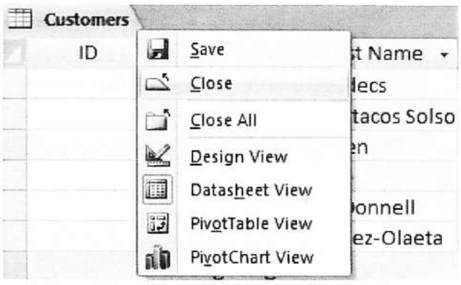
Tab along to each field and enter more information.

Click on a new line to enter a new record.



Customers

ID	First Name	Last Name	Address	City	State/Provi
1	Anna	Bedecs	123 1st Street	Seattle	WA
2	Antonio	Gratacos Solso	123 2nd Street	Boston	MA
3	Thomas	Axen	123 3rd Street	Los Angelas	CA
4	Christina	Lee	123 4th Street	New York	NY
5	Martin	O'Donnell	123 5th Street	Minneapolis	MN
6	Francisco	Pérez-Olaeta	123 6th Street	Milwaukee	WI
7	Ming-Yang	Xie	123 7th Street	Boise	ID
8	Elizabeth	Andersen	123 8th Street	Portland	OR



Note: When you enter data into a table or even a form, MS Access saves it automatically

Close the table.

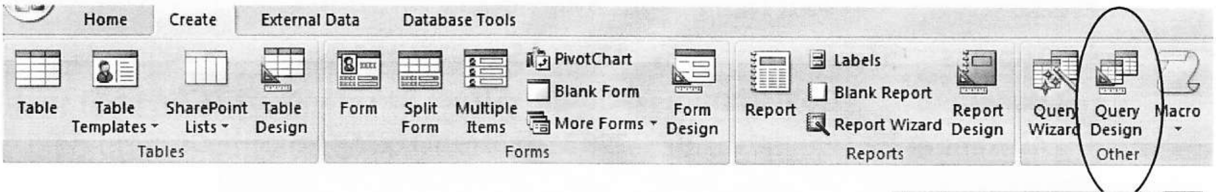
Right click the table tab and select the close option.

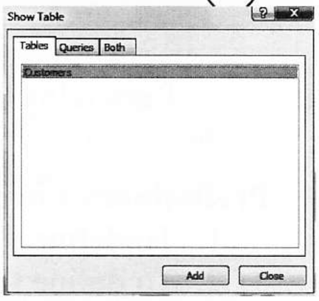
Create a query

A query is used to ask questions and get answers from the data.

Click the Create tab on the Ribbon.

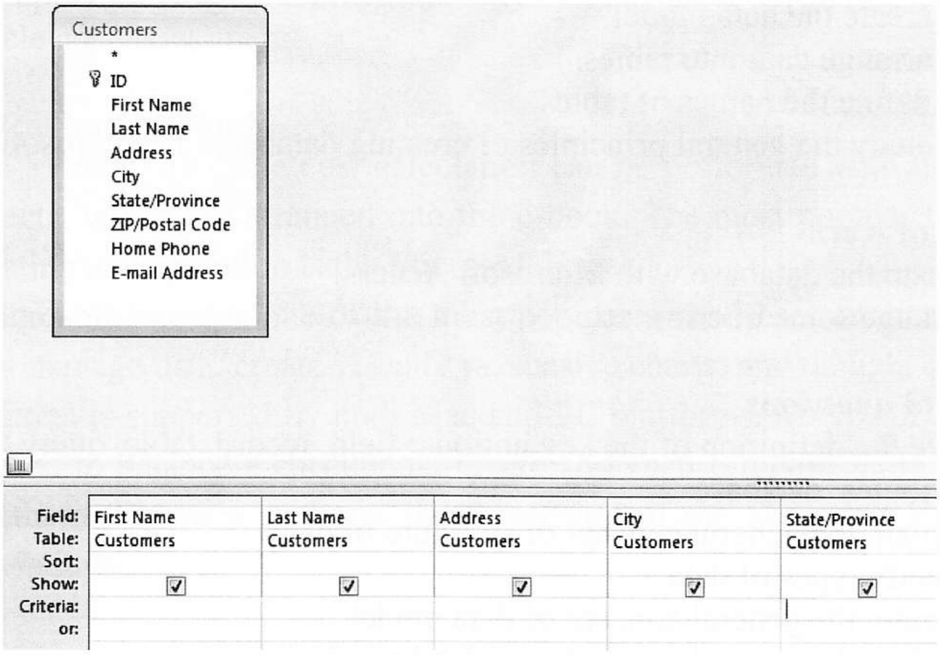
At the section called 'Other' select the option called 'Query Design'.





Select the table to use in the query and then click the 'Add' button.

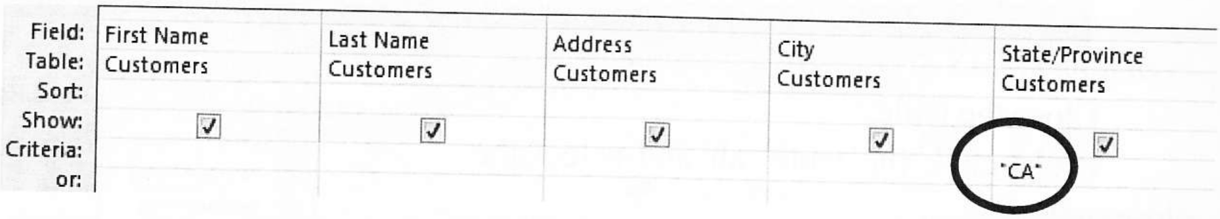
Double click the column names to use in the query



Field:	First Name	Last Name	Address	City	State/Province
Table:	Customers	Customers	Customers	Customers	Customers
Sort:					
Show:	☑	☑	☑	☑	☑
Criteria:					
or:					

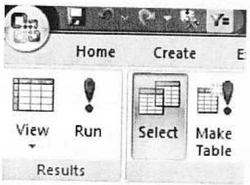
Each column will drop into the query design grid.

In this example we must extract all customers in the state of 'CA'. We do this by entering 'CA' on the criteria line under the state column.

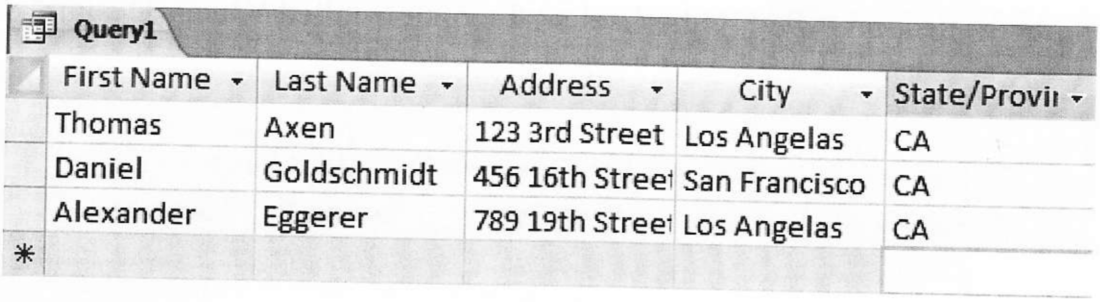


Field:	First Name	Last Name	Address	City	State/Province
Table:	Customers	Customers	Customers	Customers	Customers
Sort:					
Show:	☑	☑	☑	☑	☑
Criteria:					"CA"
or:					

Now run the query by clicking the 'Run' button on the Ribbon.



3 records have been returned for the state of 'CA'.



Query1

First Name	Last Name	Address	City	State/Provi
Thomas	Axen	123 3rd Street	Los Angelas	CA
Daniel	Goldschmidt	456 16th Street	San Francisco	CA
Alexander	Eggerer	789 19th Street	Los Angelas	CA
*				

Removing the criteria from the query grid would return all records.

Preliminary unsupervised student's work (4 hours)

1. To define the problem space for the DB developing.
2. To define the entities and relations.
3. To create the data model.
4. To arrange data into tables.
5. To define the names of tables.
6. To study the general principles of creating databases in Microsoft Access.

Progress of work

1. Create the database with Microsoft Access.
2. Arrange some queries.

Control questions

1. Give the definition of the key notions: field, record, table, query, entities, attributes, database
2. Explain the general concept of structure of data
3. Specify types of data
4. Explain the general concept of data model
5. Explain the general principles of database design
6. Give examples of aviation databases

Information sources to the Laboratory Work #5 are given in the folder "Sources to the LW5" on CD.

LABORATORY WORK # 6. Aircraft life cycle cost assessment

Key words: life cycle management, life cycle cost assessment, spreadsheets

Objective of the laboratory work: to develop the e-book for the LCC.

Preliminary knowledge and skills
Student must know:

- General concept of aircraft life cycle.
- Life cycle stages.
- The elements and specifics of each stage.
- General principles of spreadsheets operating.

Student is to be able:

To create, edit and save Microsoft Excel spreadsheets

Knowledge after the performing the laboratory work
Student must know:

- General principles of creating, editing and using Microsoft Excel e-books.
- The general methodology of aircraft life cycle cost assessment

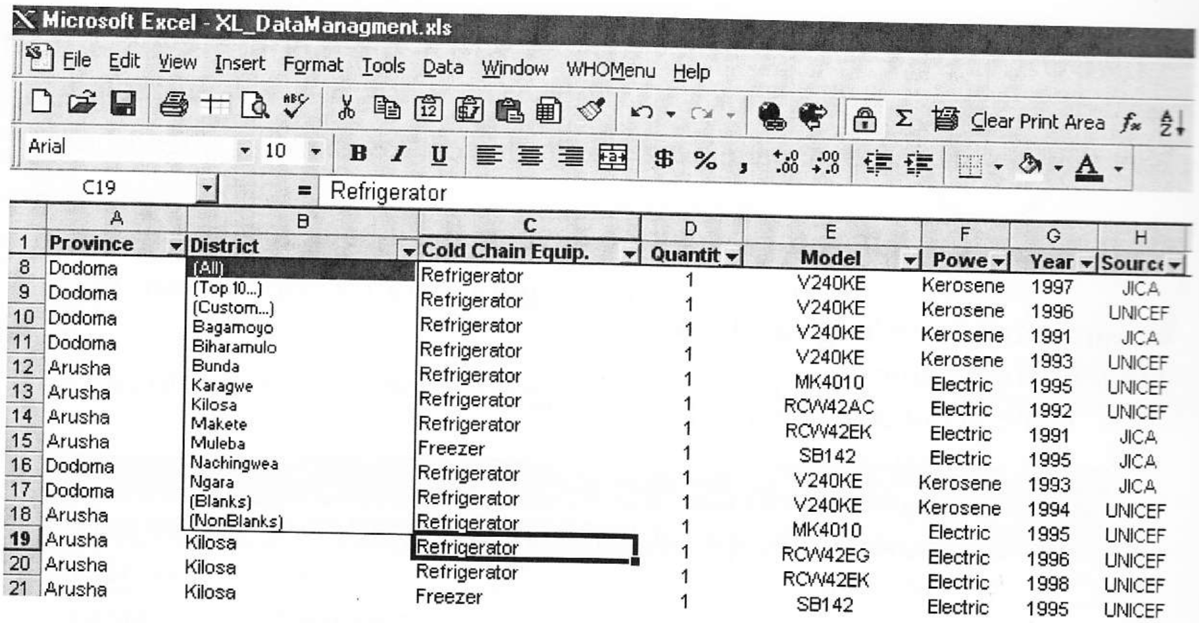
Student is to be able:

- To create, edit and use Microsoft Excel e-books.
- To apply practically the methodology of a technical object or system life cycle cost assessment

Theoretical background

An aircraft life cycle cost calculation can be performed with the help of spreadsheets, which are arranged into the e-book. The most popular tool for e-book creating and operation is the Microsoft Excel.

Microsoft Excel is one of the most popular spreadsheet applications that helps you manage data, create visually persuasive charts, and thought-provoking graphs. Excel is supported by both Mac and PC platforms. Microsoft Excel can also be used to balance a checkbook, create an expense report, build formulas, and edit them. Figure 6.1. shows the screenshot of the Microsoft Excel 2007 regular window.



C19 = Refrigerator

	A	B	C	D	E	F	G	H
1	Province	District	Cold Chain Equip.	Quantit	Model	Powe	Year	Sourc
8	Dodoma		Refrigerator	1	V240KE	Kerosene	1997	JICA
9	Dodoma		Refrigerator	1	V240KE	Kerosene	1996	UNICEF
10	Dodoma		Refrigerator	1	V240KE	Kerosene	1991	JICA
11	Dodoma		Refrigerator	1	V240KE	Kerosene	1993	UNICEF
12	Arusha		Refrigerator	1	MK4010	Electric	1995	UNICEF
13	Arusha		Refrigerator	1	RCW42AC	Electric	1992	UNICEF
14	Arusha		Refrigerator	1	RCW42EK	Electric	1991	JICA
15	Arusha		Freezer	1	SB142	Electric	1995	JICA
16	Dodoma		Refrigerator	1	V240KE	Kerosene	1993	JICA
17	Dodoma		Refrigerator	1	V240KE	Kerosene	1994	UNICEF
18	Arusha		Refrigerator	1	MK4010	Electric	1995	UNICEF
19	Arusha	Kilosa	Refrigerator	1	RCW42EG	Electric	1996	UNICEF
20	Arusha	Kilosa	Refrigerator	1	RCW42EK	Electric	1998	UNICEF
21	Arusha	Kilosa	Freezer	1	SB142	Electric	1995	UNICEF

Figure 6.1.- Screenshot of the Microsoft Excel window

Toolbars

In Microsoft Excel 2007 for a PC, the toolbars are automatically placed as tabs at the top of the screen. Within these tabs you will find all of your options to change text, data, page layout, and more. To be able access all of the certain toolbars you need to click on a certain tab that is located towards the top of the screen.

The ***Home Tab***: (Figure 6.2). This is one of the most common tabs used in Excel. You are able to format the text in your document, cut, copy, and paste information.

Change the alignment of your data, insert, delete, and format cells. The ***Home Tab*** also allows you to change the number of your data (i.e. currency, time, date).

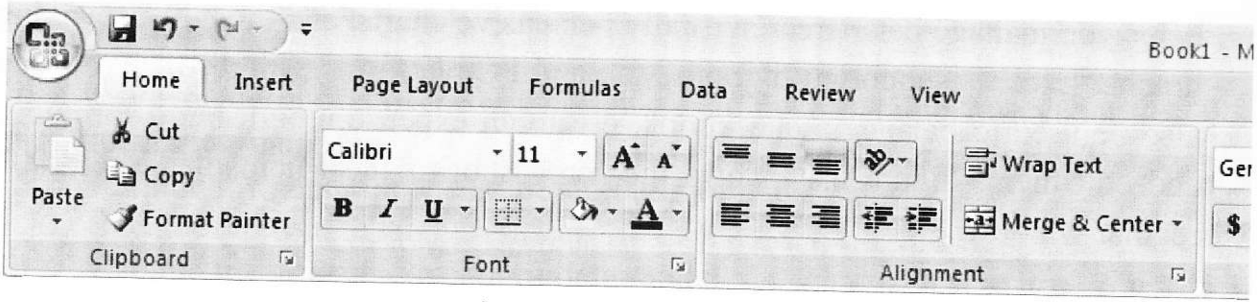


Figure 6.2.- Home Tab

The ***Insert Tab***: (Figure 6.3). This tab is mainly used for inserting visuals and graphics into your document. There are various different things that can be inserted from this tab such as pictures, clip art, charts, links, headers and footers, and word art.

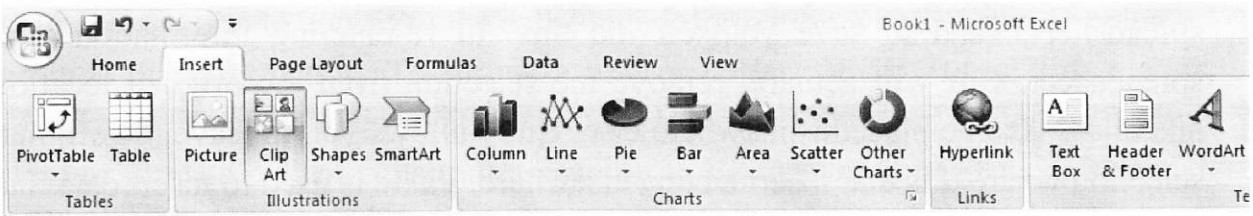


Figure 6.3.-. Insert Tab

The ***Page Layout Tab***: (Figure 6.4). Here you are able to add margins, themes to your document, change the orientation, page breaks, and titles. The scale fit of your document is also included as a feature within this tab, if needed.

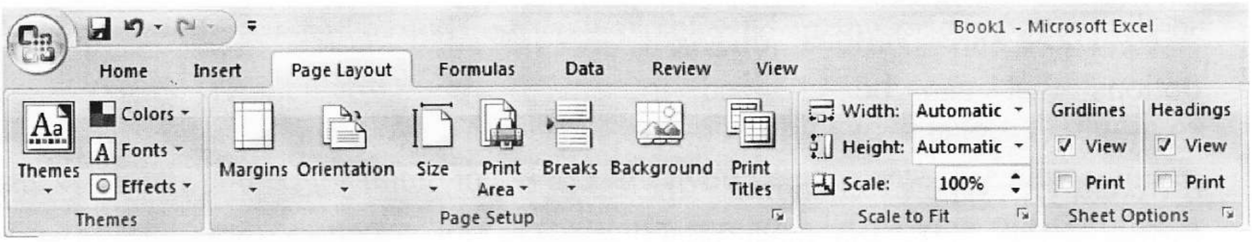


Figure 6.4.-. Page Layout Tab

Working with cells

Cells are an important part of any project being used in Microsoft Excel. Cells hold all of the data that is being used to create the spreadsheet or workbook. To enter data into a cell you simply click once inside of the desired cell, a black border will appear around the cell (Figure 6.5). This border indicates that it is a selected cell. You may then begin typing in the data for that cell.

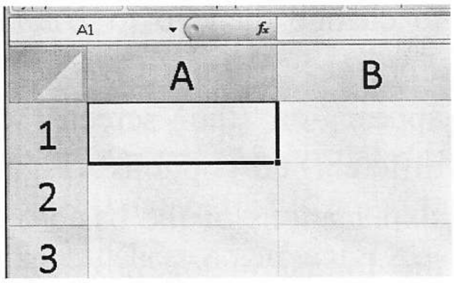


Figure 6.5 - Entering Data

Changing an entry within a cell

You may change an entry within a cell two different ways:

- Click the cell one time and begin typing. The new information will replace any information that was previously entered.
- Double click the cell and a cursor will appear inside. This allows you to edit certain pieces of information within the cells instead of replacing all of the data.

Cut, copy, and paste

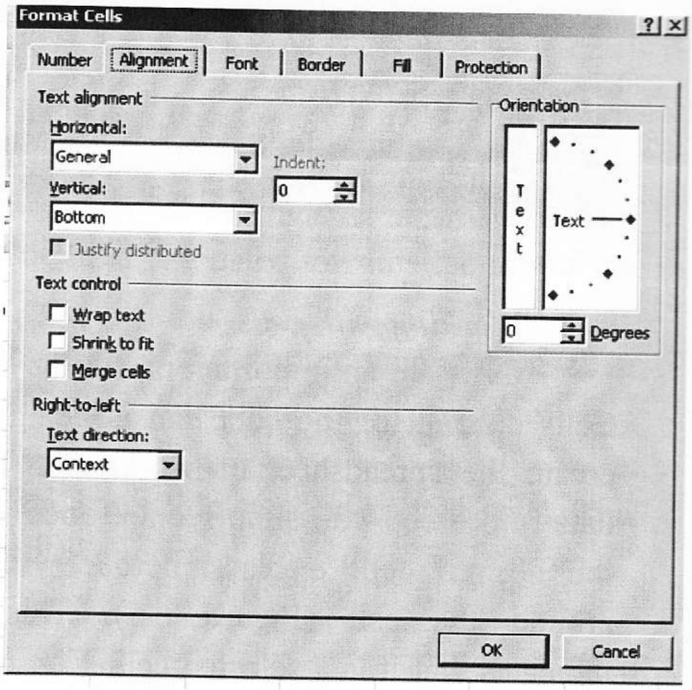
You can use the *Cut, Copy* and *Paste* features of Excel to change the data within your spreadsheet, to move data from other spreadsheets into new spreadsheets, and to save yourself the time of re-entering information in a spreadsheet. **Cut** will actually remove the selection from the original location and allow it to be placed somewhere else. **Copy** allows you to leave the original selection where it is and insert a copy elsewhere. **Paste** is used to insert data that has been cut or copied.

To Cut or Copy: highlight the data or text by selecting the cells that they are held within. Go to the ***Home Tab*** > ***Copy*** (CTRL + C) or ***Home Tab*** > ***Cut*** (CTRL + X). Click the location where the information should be placed. Go to ***Home Tab*** > ***Paste*** (CTRL + V) to be able to paste your information.

Formatting cells

There are various different options that can be changed to format the spreadsheets cells differently. When changing the format within cells you must select the cells that you wish to format.

To get to the ***Format Cells*** dialog box select the cells you wish to change then go to ***Home Tab*** > ***Format*** > ***Format Cells***. A box will appear on the screen with six different tab options (Figure 6.6). Explanations of the basic options in the format dialog box are bulleted below.



Horizontal: General

Indent: 0

Vertical: Bottom

□ Justify distributed

□ Wrap text

□ Shrink to fit

□ Merge cells

0 Degrees

Text direction: Context

Figure 6.6.- Formatting Cells

Number allows you to change the measurement in which your data is used. (If your data is concerned with money the number that you would use is currency).

Alignment. This allows you to change the horizontal and vertical alignment of your text within each cell. You can also change the orientation of the text within the cells and the control of the text within the cells as well.

Font gives the option to change the size, style, color, and effects.

Border. gives the option to change the design of the border around or through the cells.

Formatting rows and columns

When formatting rows and columns you can change the height, choose for your information to autofit to the cells, hide information within a row or column, un-hide the information. To format a row or column go to ***Home Tab > Row Height*** (or Column Height), then choose which height you are going to use (Figure 9). The cell or cells that are going to be formatted need to be selected before doing this.

When adding a row or column you are inserting a blank row or column next to your already entered data. Before you can add a ***Row*** you are going to have to select the row that you wish for your new row to be placed. (Rows are on the left hand side of the spreadsheet) once the row is selected it is going to highlight the entire row that you chose. To insert the row you have to go to ***Home Tab > Insert > Insert Sheet Rows***. The row will automatically be placed on the spreadsheet and any data that was selected in the original row will be moved down below the new row.

Before you can add a ***Column*** you are going to have to select a column on the spreadsheet that is located in the area that you want to enter the new column.

(Columns are on the top part of the spreadsheet.) Once the column is selected it is going to highlight the entire row that you chose. To insert a column you have to go to ***Home Tab > Insert > Insert Sheet Column***. The column will automatically be place on the spreadsheet and any data to the right of the new column will be moved more to the right.

Working with charts

Charts are an important part to being able to create a visual for spreadsheet data. In order to create a chart within Excel the data that is going to be used for it needs to be entered already into the spreadsheet document. Once the data is entered, the cells that are going to be used for the chart need to be highlighted so that the software knows what to include. Next, click on the ***Insert Tab*** that is located at the top of the screen. (Figure 6.7).

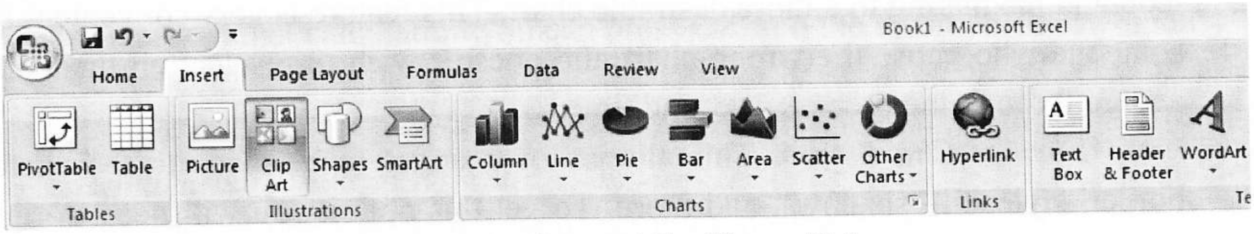
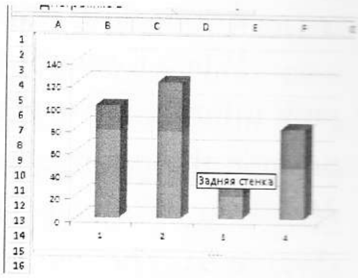


Figure 6.7.- Charts Tab



You may choose the chart that is desired by clicking the category of the chart you will use. Once the category is chosen the charts will appear as small graphics within a drop down menu. To choose a particular chart just click on its icon and it will be placed within the spreadsheet you are working on. To move the chart to a page of its own, select the border of the chart and ***Right Click***. This will bring up a drop down menu; navigate to the option that says ***Move Chart***. This will bring up a dialog box that says ***Chart Location***. From here you will need to select the circle next to ***As A New Sheet*** and name the sheet that will hold your chart. The chart will pop up larger in a separate sheet, but in the same workbook as your entered data.

Chart design

There are various different features that you can change to make your chart more appealing. To be able to make these changes you will need to have the chart selected or be viewing the chart page that is within your workbook. Once you have done that the ***Design Tab*** will appear highlighted with various different options to format your graphic (Figure 6.8).

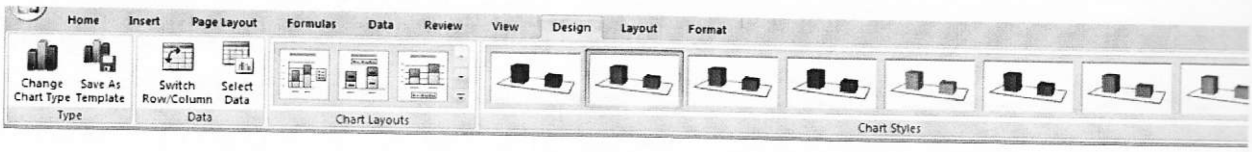


Figure 6.8.- Design Tab for chart design

Chart options

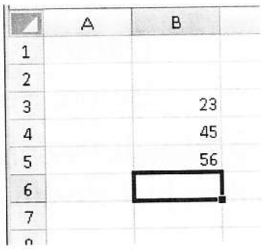
Titles. To add titles to a chart of graphic you have to click on the Insert Tab. Once you have done this, click on the Text Box Icon. This will insert a text box that you can type the title and place anywhere you wish on the chart.

Change Chart Type. You can change your chart easily by selecting this icon and navigating to a more desirable chart. This feature is very convenient for someone who chose the wrong chart and doesn't wish to reselect all their data and go through the process a second time.

Format Chart Area. This allows for changes to be made to the chards border, style, fill, shadows, and more. To get this option you will need to right click on the charts border and navigate to the ***Format Chart Area*** option. Once this is clicked a dialog box will appear.

You are able to change the color of the bars that are within your chart.

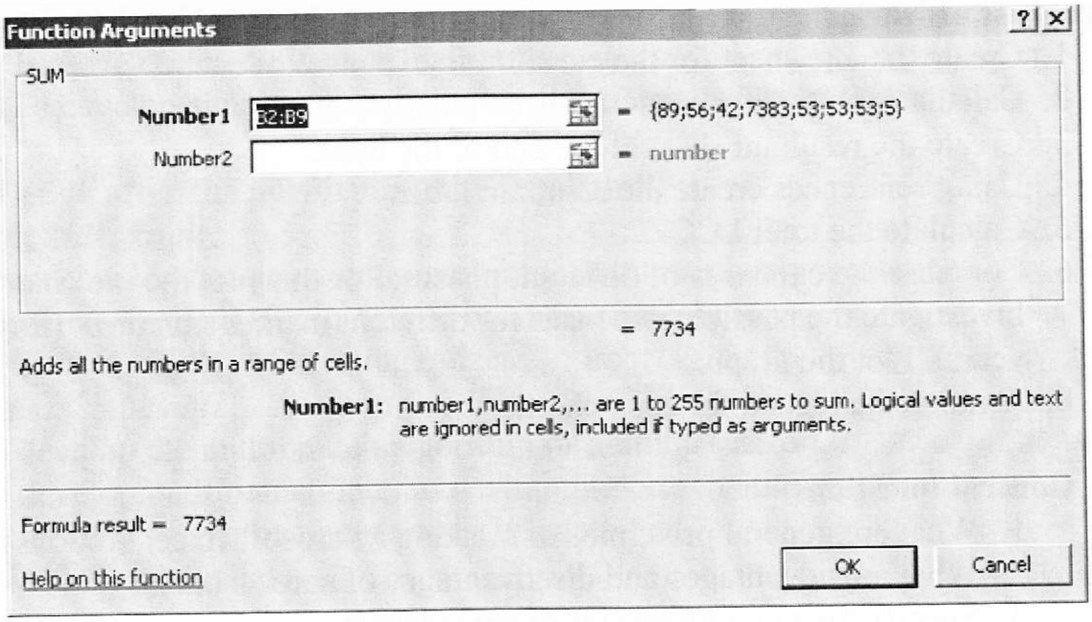
Creating functions



	A	B
1		
2		
3		23
4		45
5		56
6		
7		

When creating a function in Excel you must first have the data that you wish to perform the function with selected. Select the cell that you wish for the calculation to be entered in (i.e.: if I want to know the sum of B1:B5 I will highlight cell B6 for my sum to be entered into). Once you have done this you will need to select the ***Formulas Tab*** located at the top of the screen. A list of ***Most Recently Used, Financial, Logical, Text, Date and Time, Math and Trig*** formulas will appear. To choose one of the formulas click the icon that holds the formula you are looking for.

Once you have clicked your formula this will display a dialog box on your screen. (Figure 6.9)



Function Arguments

SUM

Number1 B2:B9 = {89;56;42;7383;53;53;53;5}

Number2 = number

= 7734

Adds all the numbers in a range of cells.

Number1: number1,number2,... are 1 to 255 numbers to sum. Logical values and text are ignored in cells, included if typed as arguments.

Formula result = 7734

Figure 6.9.- First calculation display

In this screen it lists the cells that are being calculated, the values within the cells, and the end result.

To accept that calculation you can press OK and the result will show up in the selected cell.

Undo and redo

In order to undo an action, you can click on the blue arrow icon that is pointing to the left at the top of the screen. To redo an action, you can click on the blue arrow icon pointing to the right. It is important to note that not all actions are undoable, thus it is important to save before you make any major changes in your document so you can revert back to your saved document.

Quitting

Before you quit, it's a good idea to save your document one final time. You will need to choose the ***Office Icon*** and choose ***Exit Excel.*** This is better than just closing the window, as it insures your document quits correctly.

Preliminary unsupervised student's work (4 hours)

1. To revise the skills of operating Microsoft Excel.
2. To investigate the information about the cost of the phases of selected aircraft life cycle.
3. To define the positions of expenses for each phase.

Progress of work

1. Create the one sheet for the calculating expenses of each LC phase.
2. Calculate the total expenses of each phase.
3. Create the resultant sheet at the end of the book.
4. Using references create the resultant table.
5. Calculate the total LCC.
6. Compare the expenses of different phases. For this plot the pie chart.
7. Investigate the possible expenses for different market situations (for three years). Plot the graph.
8. Arrange the legend for each chart.

Control questions

1. What are general principles of creating Microsoft Excel e-books?
2. What are advantages and disadvantages of spreadsheets?
3. Specify the main properties of spreadsheet.
4. Explain the principles of creating links between the worksheets.
5. Give the brief characteristic of the life cycle cost assessment methodology.
6. Make the analysis of main problems of an aircraft life cycle assessment.

Information sources to the Laboratory Work #6 are given in the folder "Sources to the LW6" on CD.

LABORATORY WORK # 7. Developing training and business presentations with computer technologies

Key words: presentation, slide, palette, color schemes, multibranch

Objective of the laboratory work: to develop the multibranch interactive presentation, training presentation.

Preliminary knowledge and skills

***Student must know*:**

- The notion of algorithm
- General principles of developing multibranch algorithms
- General principles of creating flowcharts
- Basic principles of creating presentations with Microsoft Power Point

***Student is to be able*:**

- To develop the multibranch algorithm
- To develop simple presentations with Microsoft Power Point

Knowledge after the performing the laboratory work

Student must know:

- The technology of developing multibranch interactive presentations
- Requirements to the training and business presentations
- General principles of creating quizes

***Student is to be able*:**

- To develop the multibranch algorithm of training session
- To develop multibranch interactive training presentation
- To develop proper business presentations
- To develop the quiz

Theoretical background

Microsoft PowerPoint is one of the most popular presentation programs supported by both Mac and PC platforms. Microsoft PowerPoint can be used to create interactive presentations for classroom, business, or personal use.

Toolbars

Microsoft PowerPoint uses several toolbars to allow you to modify your document. When you first open PowerPoint, usually the toolbars discussed below will open.

The ***Home*** bar (Figure 7.1). This toolbar has features that allow you to edit ***Slides, Fonts, Paragraph, Drawing and Editing***. This is the only toolbar that you can edit these options from.

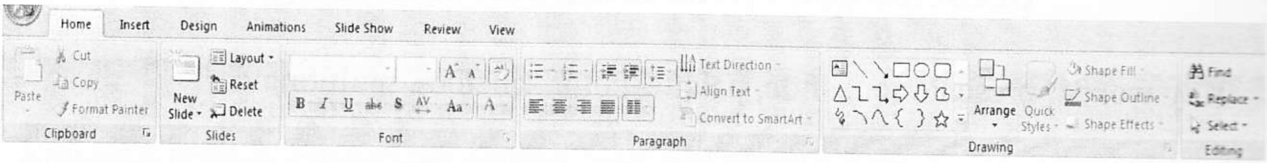


Figure 7.1.- Home Tab selected

The ***Insert*** bar (Figure 7.2). This toolbar has features that allow you to add ***Tables, Illustrations, Links, Text, and Media Clips***.

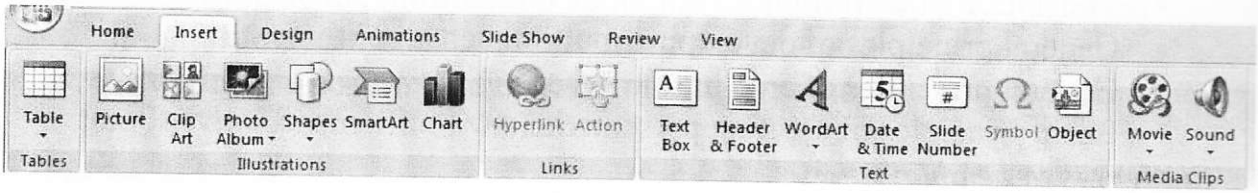


Figure 7.2.- Insert Tab selected

The ***Design*** bar (Figure 7.3). This toolbar has features that allow you to edit how your presentation will look. It includes features such as ***Page Setup, Themes****, and* ***Background.***

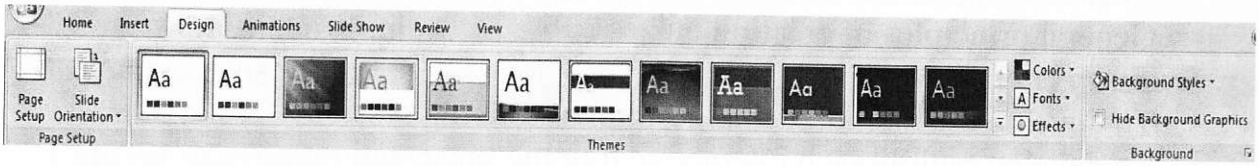


Figure 7.3.- Design Tab Selected

The ***Animations*** bar (Figure 7.4). This toolbar has features that allow you to add custom animations to your presentation. You can select from ***Preview, Animations, and Transitions***. By selecting ***Custom Animation*** *in the* ***Animations*** box, you can apply effects to individual elements in your presentation.

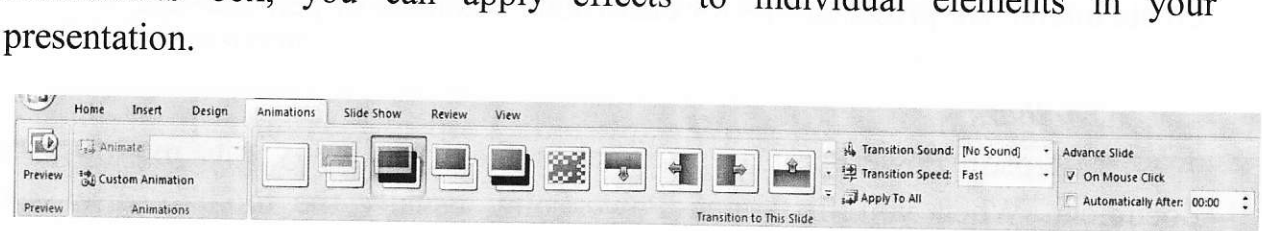


Figure 7.4.- Animations Tab Selected

The ***Slide Show*** bar (Figure 7.5). This toolbar has features that allow you to select how your presentation will be displayed. From here, you can ***Start your Slide Show, Set Up your Slide Show***, and ***Adjust Monitor Settings***.

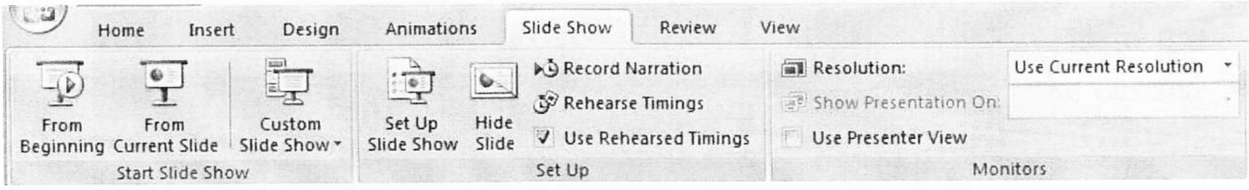


Figure 7.5. Slide Show Tab Selected.

The ***Review*** bar (Figure 7.6). This toolbar has features that allow you to ***Proofread, Comment Slides***, and ***Protect (Lock)*** the Presentation.

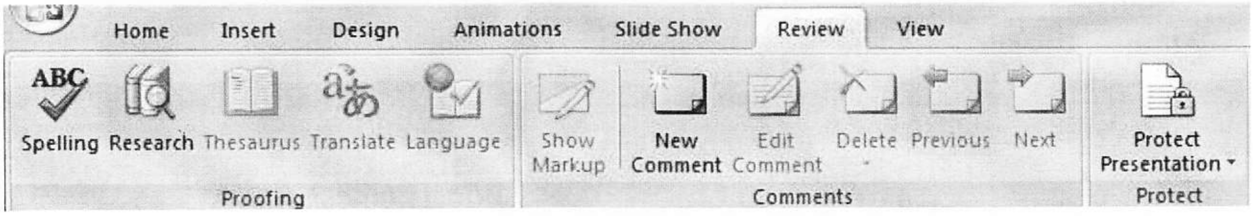


Figure 7.6.- Review Tab Selected

The ***View*** bar (Figure 7.7). This toolbar has features that allow you to set the ***View of your Presentation, Show/Hide Features, Zoom, Switch from Color to Grayscale, Adjust Windows, and Add Macros.***

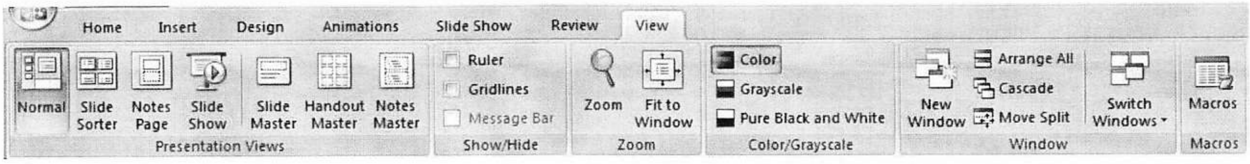


Figure 7.7.- View Tab Selected

Formatting the presentation

The default slide design for a ***Microsoft PowerPoint*** document is a blank slide. If you want your slides to have a design, you must add one. To do this, go to ***Tool Ribbon > Design Tab*** (Figure 7.8) and select the design that you like. The design is instantly applied when you click on it.

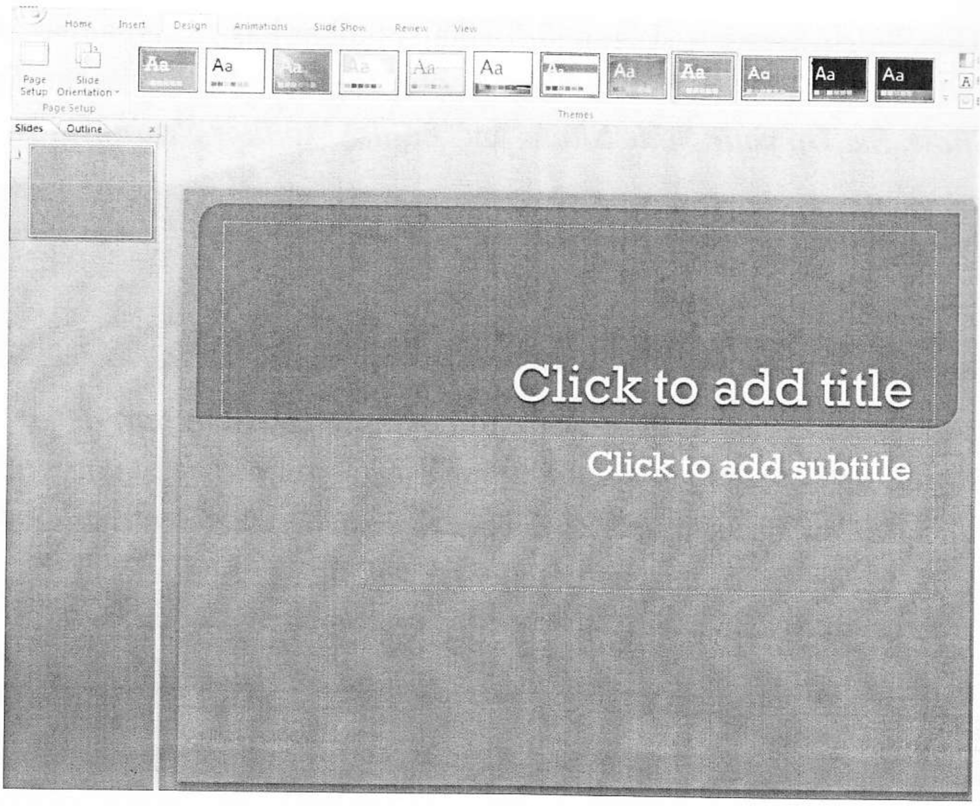


Figure 7.8.- Applying Slide Themes

Formatting the slide layout

From the **Home** tab, select **Layout** (Figure 7.9) from the Slide box and select your preferred layout. From here, you can specify how the content on your slide is displayed. To apply a layout to your slide, click on the preferred style.

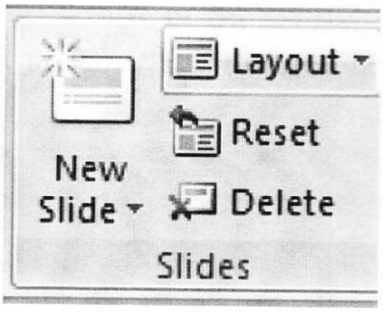
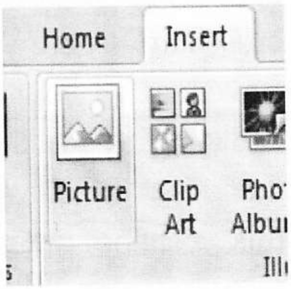


Figure 7.9.- Adjusting Slide Layout

Inserting and adding objects

Clip art. When trying to enhance your word document you may want to include ***Clip Art*** and/or ***Word Art***. Microsoft PowerPoint comes with a ***Clip Gallery*** that contains a large variety of images including pictures, borders, and backgrounds. To find a desired image, you can either click on topics or type in the search box to find exactly what you are looking for. To insert ***Clip Art***: Go to the ***Insert Tab*** and select ***Clip Art***. and then select the desired picture in the Clip Gallery and click insert. The picture will be inserted at the location of your cursor within your document. If you need to modify your ***Clip Art***, click on it once to select it, and small boxes will appear around the corners. Once your ***Clip Art*** is selected, you can resize your picture by clicking and dragging on the boxes. Holding ***SHIFT*** while clicking and dragging will resize the ***Clip Art*** proportionately.

WordArt is inserted similar to ***Clip Art***. To insert ***WordArt***: Go to the ***Insert Tab*** > ***WordArt*** Select the desired style and click ***OK***. Type the desired text and click ***OK***. To select your ***WordArt***, click on it, and small boxes will appear in the corners. Moving the yellow box on a selected ***WordArt*** allows the user to change the slant of the ***WordArt***. ***WordArt*** can be resized and deleted similarly to ***Clip Art***.



Pictures. To insert a picture select the ***Insert Tab*** > ***Picture***. Locate the image that you want to put into your document and click ***Insert***.

Inserting slides from a previously created file

To insert a slide from another presentation go to the ***Home tab*** > ***New Slide*** > ***Reuse Slides***.(Figure 7.10.), select the slide that you wish to insert into your presentation.

Creating a table

To create a table within your document, go to the ***Insert Tab*** > ***Table***. Choose the desired table size by dragging your mouse over the squares, and click. The table will be inserted at the cursor's location within your document. To navigate within your table, use the arrow keys.

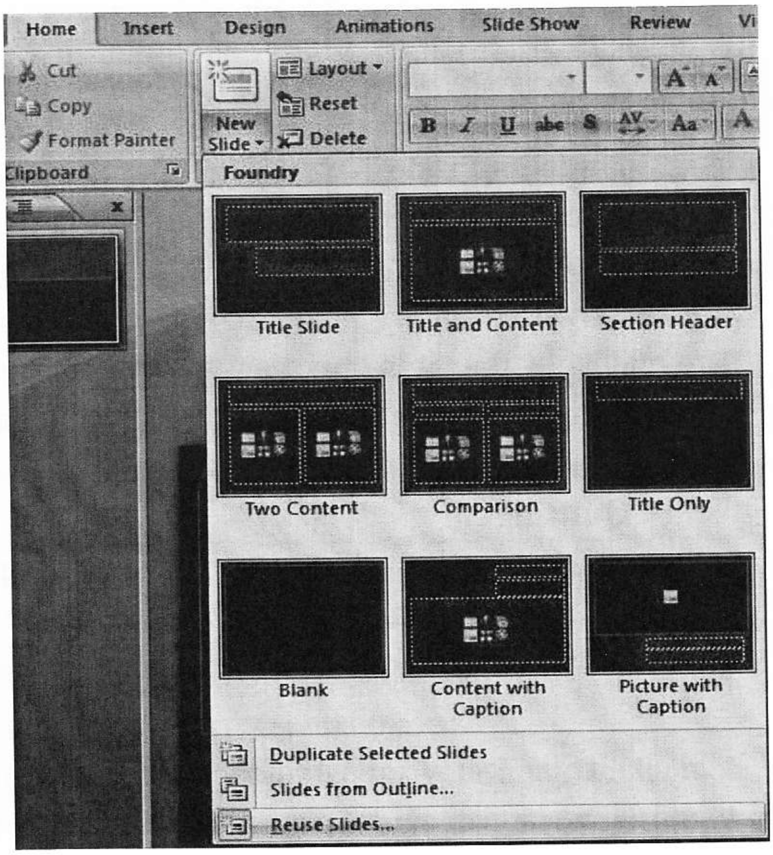


Figure 7.10.- Inserting a Slide from Another Presentation

Flowcharts

`Flowcharts are used to create diagrams in Microsoft PowerPoint. To draw a ***Flowchart*** go to the ***Home Tab*** and click the ***Down Arrow*** to access **Flowchart** (Figure 7.11). Once the ***Drawing*** toolbar is open, click and hold on the ***AutoShapes*** button until a side menu appears. Continue holding down the mouse while you drag the cursor over to ***Flowchart***, and then to the desired shape (Figure 7.11). The cursor will appear as crosshairs. Position these crosshairs where you want to place the flowchart in your document and click and drag to the desired size. You can then use the ***Drawing*** toolbar to fill this shape with words and/or pictures/color. To draw connecting lines between the shapes: Click and hold on the ***Lines*** button on the ***Drawing*** toolbar until the side menu appears. Continue holding down the mouse and select the desired line style. Click and drag your line from the first shape to the connecting shape.

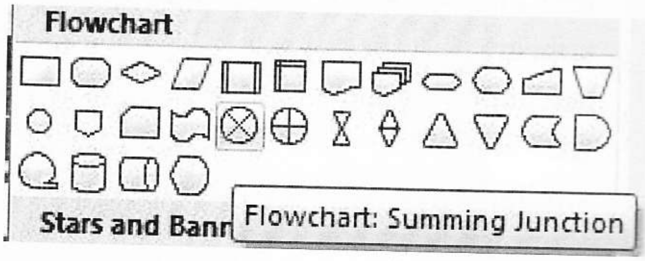


Figure 7.11.- Inserting Flowcharts

Undo and redo

In order to undo an action, go to the top of the window on the left and ***click on the Arrow Pointing to the Left (Undo)***. To redo an action, go to ***click on the arrow Pointing to the Right***. It is important to note that not all actions are undoable, thus it is important to save before you make any major changes in your document so you can revert back to your saved document.

Quitting

Before you quit, it's a good idea to save your document one final time. Then, go to ***Office Button > Quit PowerPoint***. This is better than just closing the window, as it insures your document quits correctly.

Shortcut keys for Microsoft office

The shortcut keys listed below can be a great help when using Microsoft Office products. Not only are they quick and easy, they are also amazing time savers.

CTRL+K	Create a Hyperlink
CTRL+ /	Display HTML Tags
CTRL+T	Create an Auto Thumbnail Of Picture
CTRL+SHIFT+B	Preview Page in Web Browser
SHIFT+ALT+F11	Display Script Editor
CTRL+N	Create a New Slide
CTRL+B	Bold
CTRL+I	Italic
CTRL+U	Underline
CTRL+C	Copy
CTRL+V	Paste
CTRL+S	Save
CTRL+P	Print
CTRL+O	Open

General guidelines for PowerPoint Presentation writing, design, text, graphics and color

***Before the Presentation*:**

- Check to see if the equipment will work properly
- See if the projector's resolution is the same as that on your computer
- Determine how far you will stand from the audience and how far the audience will be from the screen. This will help determine your font size, which can account for 90 percent of your success.
- Generally use only equipment you are familiar with. Take your own, if possible.
- If using another PC, make sure PowerPoint is installed or that it is the same version as the one you created the presentation in.
- Cut off screensaver. You don't want to apologize for your home pictures.
- Cut off standby. If you are talking and the screen goes into standby if may take a while to get back to the presentation or it may require you to reboot. (Start, Settings, Control Panel, Power, Power Schemes, System Standby – Never)

Writing Guidelines

- Organize the entire presentation before you create the PowerPoint presentation
- Determine the key points you want to make
- Make sure you thoroughly researched your topic
- Determine the purpose of the presentation – entertain, inform, persuade or sell. Is it lighthearted or formal – this will also help you determine colors, clip art and templates.
- Remember you are not there to overwhelm the audience or demonstrate every bell and whistle of the software.
- Don't get caught up in the gee-whiz.
- Write an outline for each slide.
- Eliminate unnecessary words on slides. Think about writing headlines.

Design Guidelines

- Make a pact with the devil – remember 666 – no more than six word slides in a row, no more than 6 bullets on a slide and 6 words to each bullet
- A long list can overwhelm an audience.
- All text should be 18 to 20 points or larger.
- Fonts should be easy to read. Try Arial, Helvetica or Tahoma they work well.
- Choose one font and font size for headlines and another font and font size for body text. Headline fonts should be san serif and body copy should be serif type. Remember, serif is with feet. San serif without feet.
- Do not use more than 3 fonts.
- Use no more than two levels of bullets (Level 1 headings for main topics and Level 2 headings for subtopics).
- Organize the elements of the slides in advance. Decide which elements are the most important and the least important and organize them accordingly.
- Emphasize text with bold, size, color, and spacing formatting. Be careful with italic, it can be hard to read.
- Consider creating a visual theme by selecting colors and graphics related to the topic of the presentation.
- For slides containing a lot of text, choose simple backgrounds.
- Use bright colors in small areas for emphasis.

- After designing the slides, stand back and examine them. Make any needed adjustments. Do the drop test to see if you can read the slide. Print out a copy. Drop the paper to the floor. Stand and see if you can read the material. If you can't, you need to start over.

Text Guidelines

- Avoid using all caps. (ALL CAPS ARE MORE DIFFICULT TO READ THAN UPPER AND LOWER CASE TYPE.) Research shows that it takes 57 percent longer to read all caps.
- Choose readable fonts. Also use standard fonts. If you create a presentation and use a non-standard font and use a different PC for the presentation, the machine will substitute a font for you.
- If using WordArt, use it sparingly. Prefer not use WordArt at all.
- Write phrases, not sentences.
- Less is more. Keep the line length of text to no more than 45 to 55 characters, including spaces. Longer lines are difficult for the audience to read.
- Avoid underlining text. Use italic or bold type instead. Use both sparingly.
- If text is placed over a color graphic, make sure the text is readable on all parts of the slide.
- Be careful when using rotated and vertical text.
- Allow for margins on slides. Make sure everything on the slide can be seen or read.

Graphics Guidelines

- Limit the use of lines to three or four lines per slide.
- Choose graphics that relate to the topic.
- Make use of white space.
- Limit the number of graphics per slide. Keep graphics throughout the presentation in the same family.
- Simplify data labels.
- Use photographs sparingly. They take up a lot of space and if you use a laptop with limited space, you may have to run the presentation from the disc, making everything slower.
- If you have clipart on every slide, you are doing something wrong.
- Don't animate unless you need to make a point.

Color Guidelines

- Use a limited number of colors.
- Choose colors that contrast for text and background.
- Because some people have problems distinguishing certain colors, avoid using certain color combinations including: red/green, brown/green, blue/black, and blue/purple.
- Use cool and muted colors for backgrounds. Bright warm colors are hard to look at for a long period of time.
- Light backgrounds with dark text can be used to create a soft look.
- Use PowerPoint's built-in color schemes.
- Colors help set a tone. Generally blue, green and gray say professional. Red and orange are high energy, but can be difficult to look at for long periods. Use them sparingly.
- For text colors white and yellow or a black background are best. Cyan and bright green on a dark background are second best. There is a reason that traffic signs are yellow and black. They are easy to see and read.

The Slide Master

The Slide Master of a PowerPoint presentation can be a very helpful tool in creating and carrying out a presentation design. It is a part of the design template of a slide that stores information about the template, including font styles, placeholder sizes and positions, background design, and color schemes. It functions as the skeleton of a presentation. The Slide Master defines the options for every slide in a presentation.

To make the most of the Slide Master:

- Format the background - Create a background for the slides in a presentation. A PowerPoint template can be used, a background can be created using PowerPoint tools, or a background can be imported from another source. Once the background is added to the slide master, it will appear on every slide in the presentation.
- Define the color scheme - PowerPoint's built-in color schemes can be used or a color scheme can be created.
- Select fonts and bullets - By applying fonts to the Slide Master, font formatting will be applied to all slides in the presentation.
- Add logos or other elements to each slide - Any graphic elements that need to appear on each slide can be added here.

Using Charts and Graphs

Charts and graphs can be a very effective part of a presentation, but each chart and graph should have a purpose that adds to the presentation's overall effectiveness. A common mistake is adding unnecessary charts and graphs to a presentation. Use charts to highlight key points of a presentation. When designing charts and graphs for presentations:

- *Guide the eye to the main point* - The main point of the chart or graph should jump out at the audience. Arrows, animation, or color can be used to draw the eye to the main point.
- *Limit the number of lines* - Make charts and graphs as simple as possible. A single data series, such as a line or row of bars, per chart or graph is best. Too many lines can be confusing.
- *Use either an axis scale or data points* - Do not use both. Keep charts and graphs as simple as possible.
- *Remove details* - Remove details including grid lines, footnotes, and other details whenever possible. They can distract from the main point.

Slide Transition

Use PowerPoint slide transitions wisely. Many transition options are available, but they should be used with reserve. A good solution is to choose a simple transition and apply it to every slide in the presentation. Rules to follow for slide transitions include:

- When using sections to divide a presentation, a second transition can be used for the introductory slide for each section.
- Slide transitions, like animations, lose their effectiveness when used more than a few times.

Slide transitions should not distract from the content of the presentation, so the best ones blend into the overall presentation.

More recommendations are given in Appendix D.

Preliminary unsupervised student's work (4 hours)

1. Study the general principles of creating training interactive presentations and games.
2. Develop the algorithm of the interactive training session. For this use the previously prepared interactive e-guide.
3. Choose some pictures and video to illustrate the teaching material.
4. Develop the algorithm for implementation the interactive quiz.

5. Correspond the portions of text and video materials with training situation, which the branches of algorithm create.
6. Provide the control elements in the presentation.
7. Develop the draft of business presentation on the subject of 1st or 2nd Homework. Follow requirements to the business presentations.

Progress of work

1. Create the interactive training presentation, including the quiz.

Control questions

1. What are general demands to the business presentations?
2. Compare the specifics of training and business presentations.

Information sources to the Laboratory Work #7 are given in the folder "Sources to the LW7" on CD.

Appendix A

NATIONAL AVIATION UNIVERSITY
AEROSPACE INSTITUTE
AIRCRAFT FACULTY
AURCRAFT AIRWORTHINESS RETAINING DEPARTMENT

HOMEWORK #1

ON THE DISCIPLINE

"AIRCRAFT LIFE CYCLE SUPPORTING COMPUTER TECHNOLOGIES"

CONCEPTUAL MODEL OF AIRBUS 380 LIFE CYCLE

Performed by student of 305 FLA Litvinova Svetlana

Checked by L. Zhuravlova

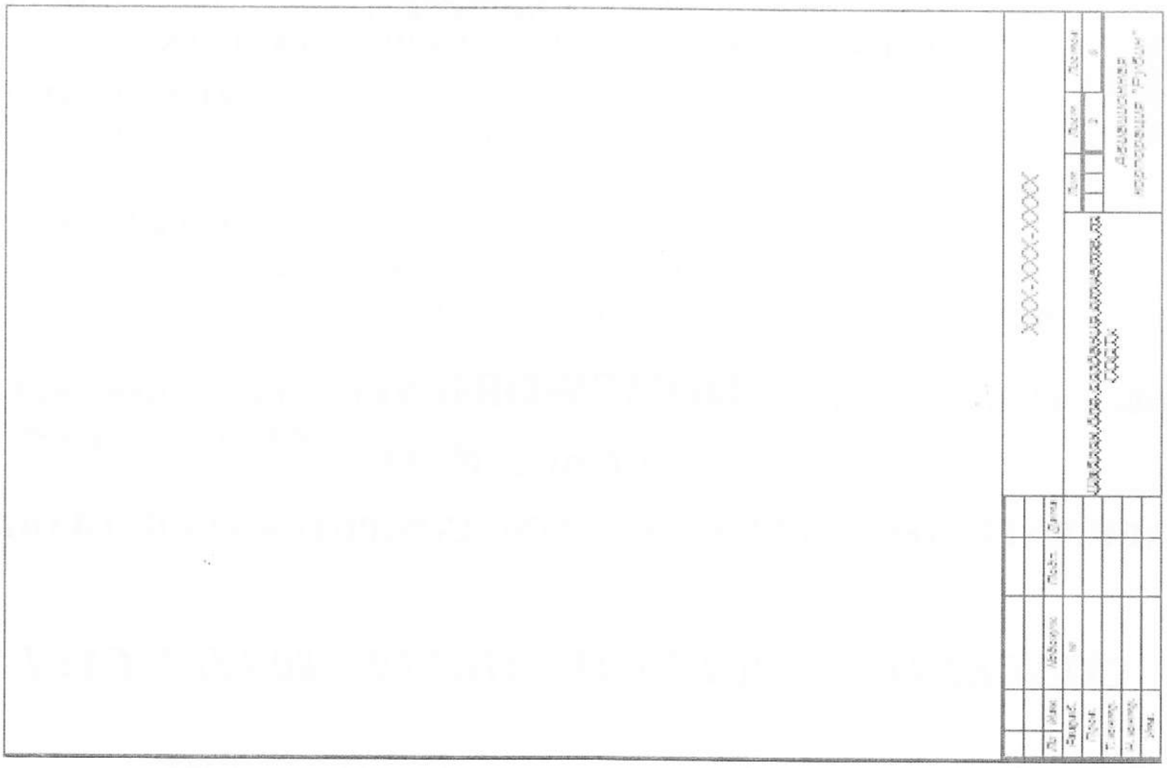
Date ____________________

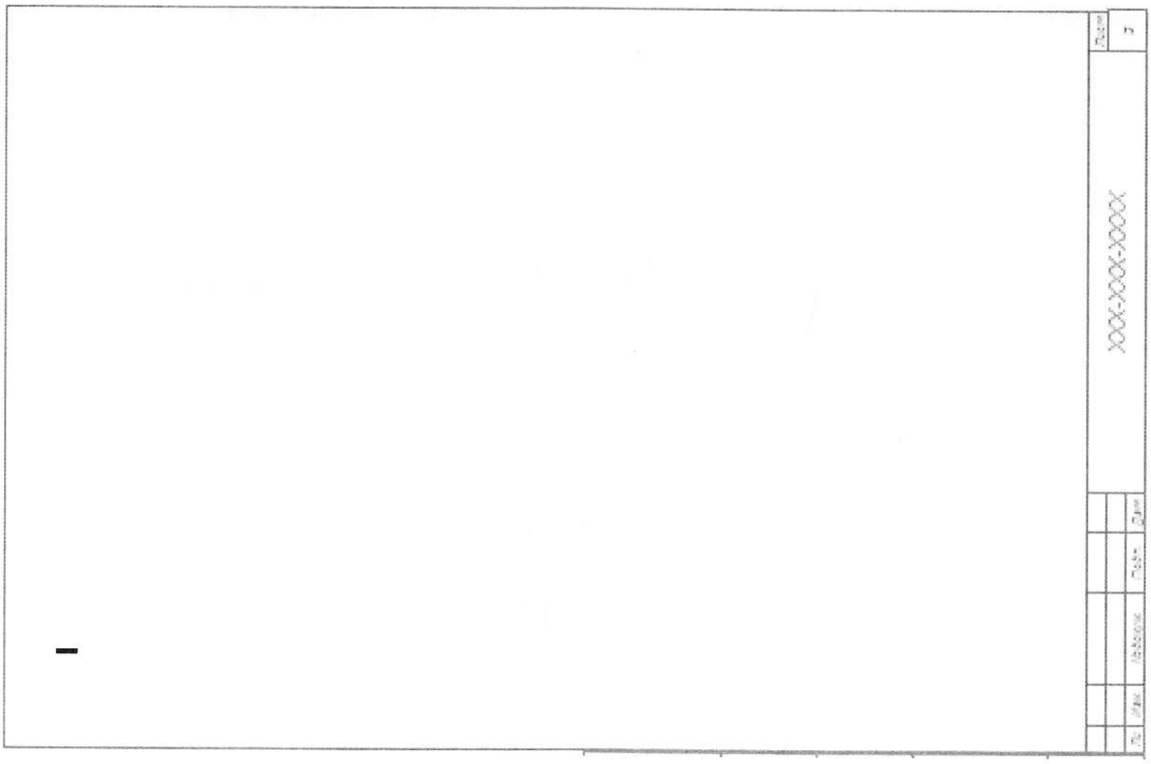
Mark ____________________ Signature ______________________________

Kyiv

2012

Appendix B





Appendix C

Methodical recommendations to the performance of the Homework #1

"CONCEPTUAL MODEL OF <concrete aircraft> LIFE CYCLE"

All information, collected during the performing of Laboratory works, can be used for the arrangement of the Homework #1.

The main steps are the next.

1. Choose the type of an aircraft
2. Investigate its market position
3. Investigate the life cycle strategy
4. Estimate the cost of each life cycle stage.
5. Investigate, what computer technologies are applied on each stage of the aircraft life cycle. Estimate the cost of software developed (or purchased). Pay attention on databases.
6. Define the problem domain of the aircraft life cycle.
7. Create the ontology of the problem domain. Restrict it, if possible. Define the criteria of restriction.
8. Design the ontograph of the problem domain.
9. Define the key notions, which correlate with the most important objects of the problem domain.
10. Specify the relations between objects.
11. Specify the input (inputs) and output (outputs) of the conceptual model.
12. Specify the key characteristics of the elements of the conceptual model.
13. Develop the algorithm of the conceptual model functioning.
14. Create the flow-chart of the algorithm.
15. Arrange the conceptual model guide, following the requirements to the technical documentation.

Methodical recommendations to the performance of the Homework #2 "Developing the Business Power Point presentation "Aircraft life cycle""

Use the Homework #1 for the performance of this Homework.

1. Specify the general objective of the work. Define the local goals of the work.
2. Develop the algorithm of the presentation.
3. Correlate each step of the algorithm with the actual information. Define the form of presentation this information: text, diagram, video, or picture.
4. Transform the portions of information to the slides of the presentation.
5. Follow the recommendations for creating proper presentations.

Appendix D

Tips for Effective Presentations

- **Allow one minute per slide** - An audience has a short attention span and if you spend longer than one minute per side, you risk losing the attention of the audience.
- **Don't read your slides** - Put your key points on your slides and use speaker notes for your presentation.
- **When using charts and graphs, highlight the bottom line** - Re-enforce your message by highlighting the benefits or bottom line results.
- **Dim the lights** - Use dark backgrounds with light-color text. This will provide good contrast for projection in a light-dimmed room.
- **Maintain eye contact with your audience** - Practice your presentation until you are able to discuss your subject without reading your slides. Position yourself so that you can have eye contact with most of the people in the room.

Some General Tips:

- Keep in mind, most people read from left to right and from top to bottom. They also tend to notice dark or bright areas before light ones. Consider this when placing elements on a slide.
- Consider using backgrounds that are neither very light nor very dark. Very dark backgrounds are usually used with white or yellow text for contrast, but the result can be hard on the eye. To create a softer effect, use a medium green or blue, but be sure there is sufficient contrast with the color of the text.
- A watermarked company logo can work well as a slide background.

7 Steps for Preparing Powerful Presentation

1. State the objectives and write them down – at the end of the presentation I want the audience to think… I want her to do …. I want them to say… I want him to feel.
2. Answer the basic questions – why am I making the presentation? What am I going to say? What do the audience want to hear? Who am I saying it to? Where will I be saying it?
3. Use a clear and logical structure – break your material into manageable “chunks” like the chapters in a book. Share the structure with your audience, it will help them too.
4. Tell the audience what they want to hear, nothing more – they don’t want to hear everything that you know, or have done. They want to hear the parts that are relevant to them.

5. Think of your presentation as telling as story – there should be a beginning, a middle and an end. Each part should flow logically from the previous part and it should be told in your natural style of speech.

6. Spend time on the opening words of the presentation – you need to grab the audience's attention while you have the change, once you have lost their attention it is very hard to get it back.

7. Rehearse with someone who doesn't know the subject – they can tell you whether or not it is clear, whether you have missed something and most importantly whether it meets the objectives you set in Step 1.

10 Power Tips for Calming Presentation Butterflies

1. Arrive early and walk briskly around the room before anyone is there, making it your own.

2. Greet audience members as they come in, shaking hands and building friendship bridges.

3. Go to an out-of-sight area and bend from the waist, letting your hands and arms drop to the ground.

4. Take deep breaths, letting the air out slowly.

5. Tense and relax the muscles in your face, arms, stomach and legs.

6. Stretch your neck, arms and legs.

7. Concentrate on your success.

8. Visualize the audience as you want then to act immediately after the presentation.

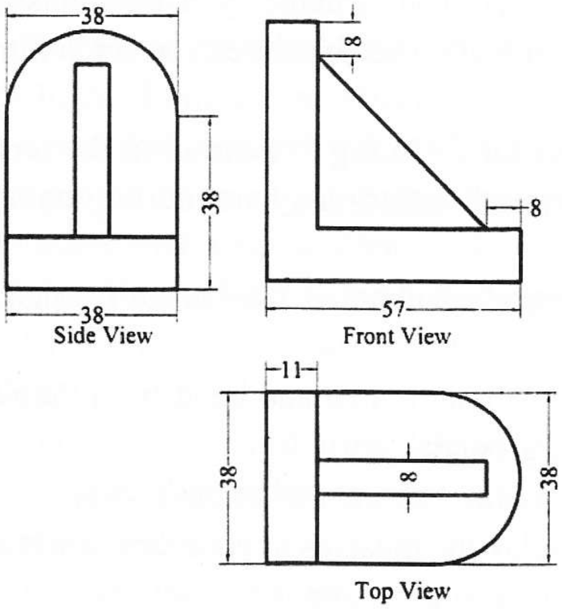
9. Engage in positive self-talk about your excellent presentation.

10. Tell a story or anecdote very early on that will require you to move around, make large gestures or raise your voice – any of which will serve as a release for your nervous energy.

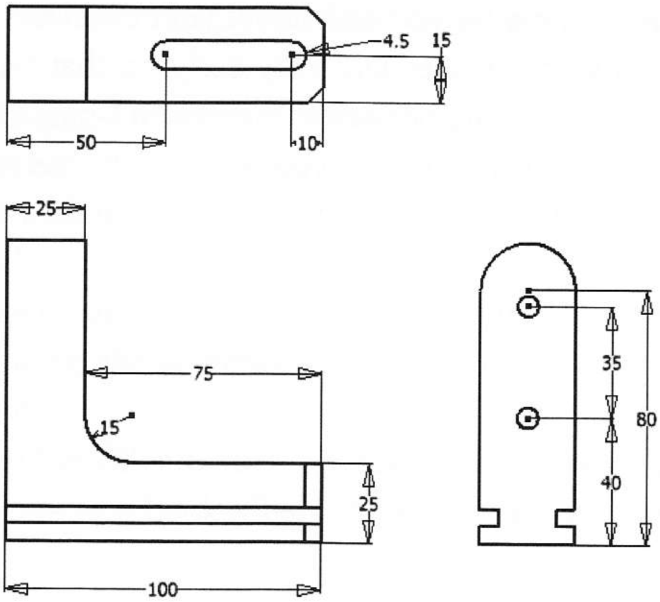
Simple 2D AutoCad drawings

Level 1

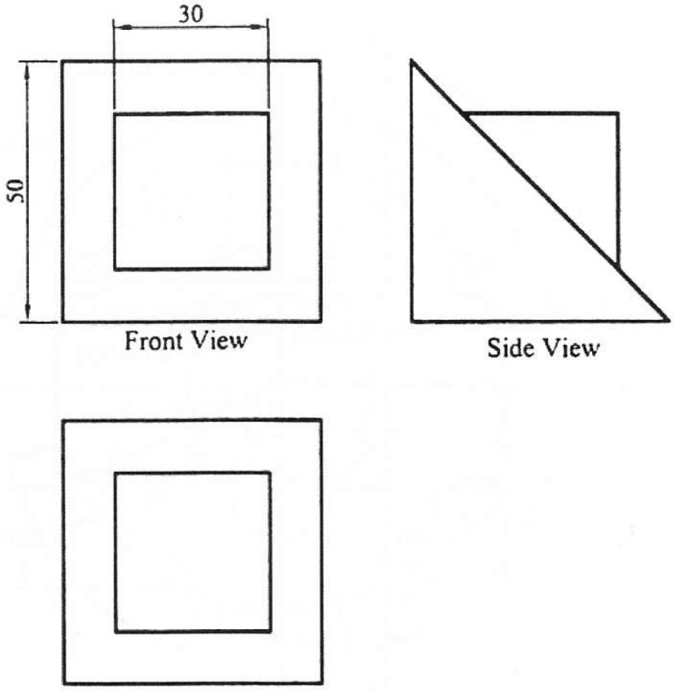
Variant 1.1



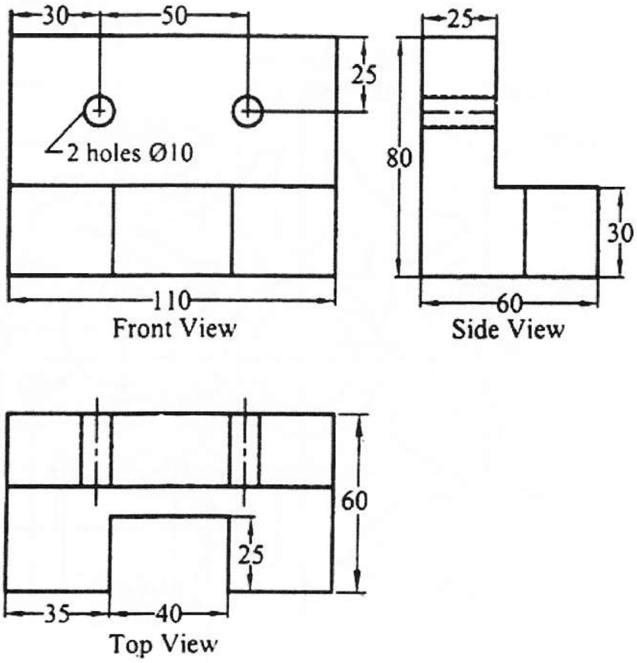
Variant 1.2



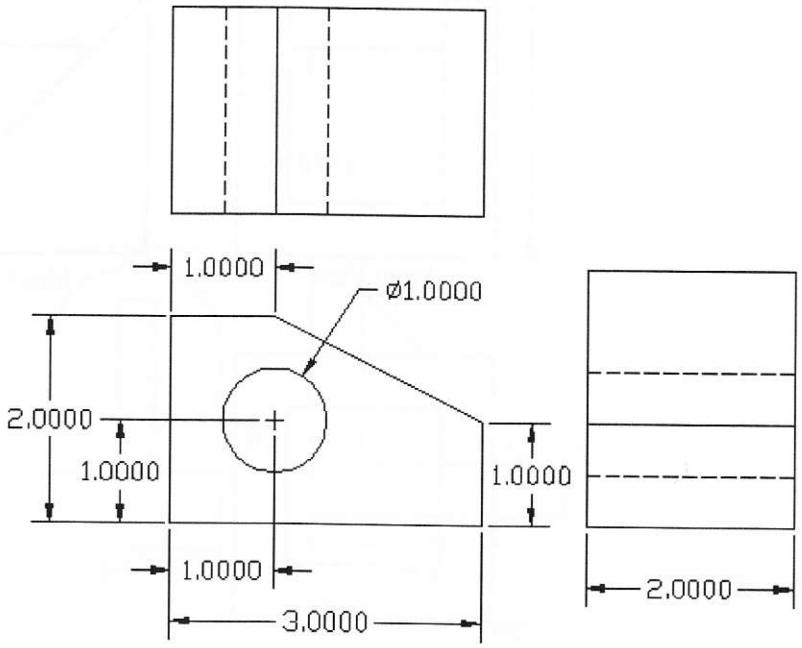
Variant 1.3



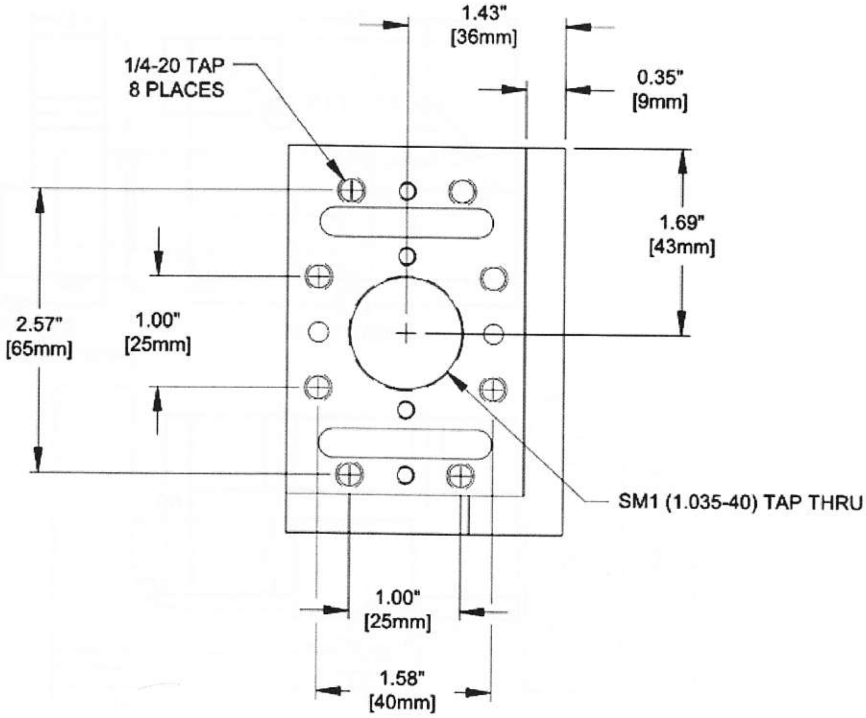
Variant 1.4



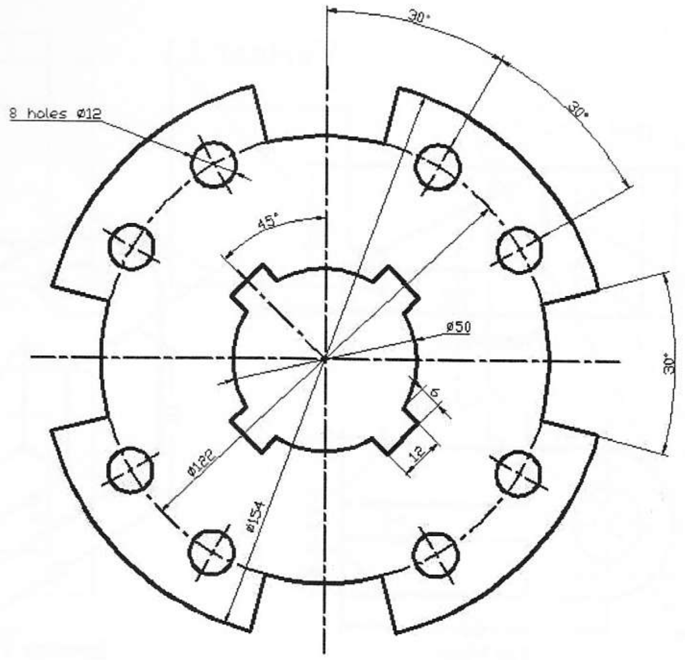
Variant 1.5



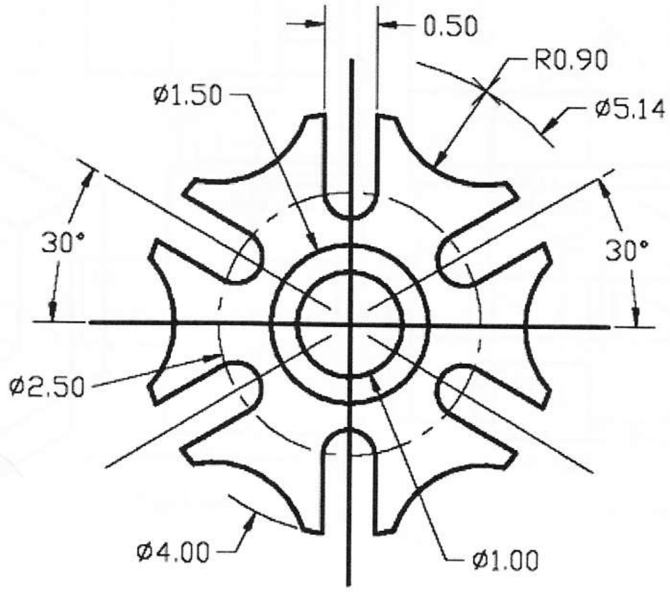
Variant 1.6



Variant 1.7

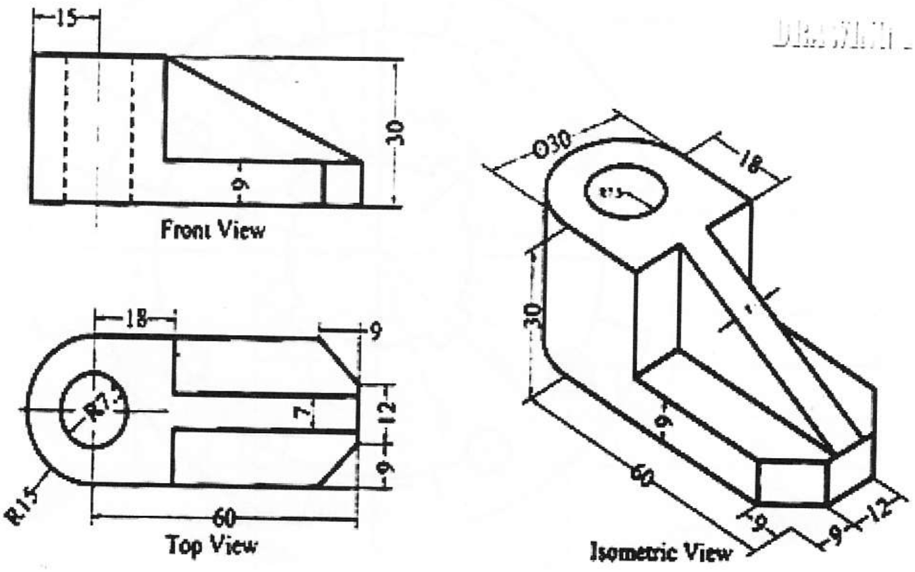


Variant 1.8



Level 2

Variant 2.1

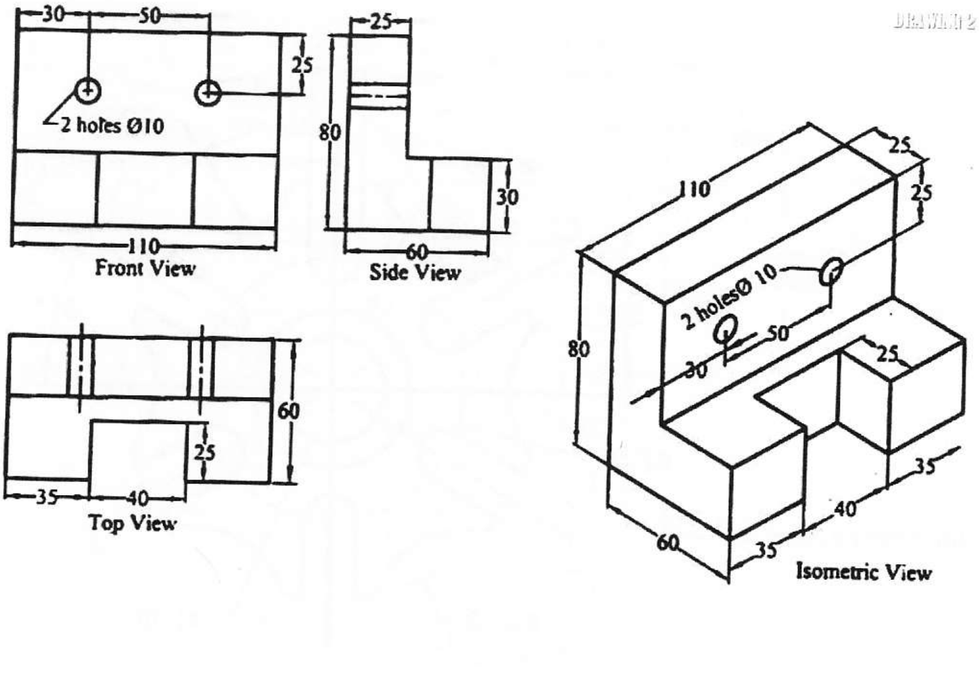


Front View

Top View

Isometric View

Variant 2.2



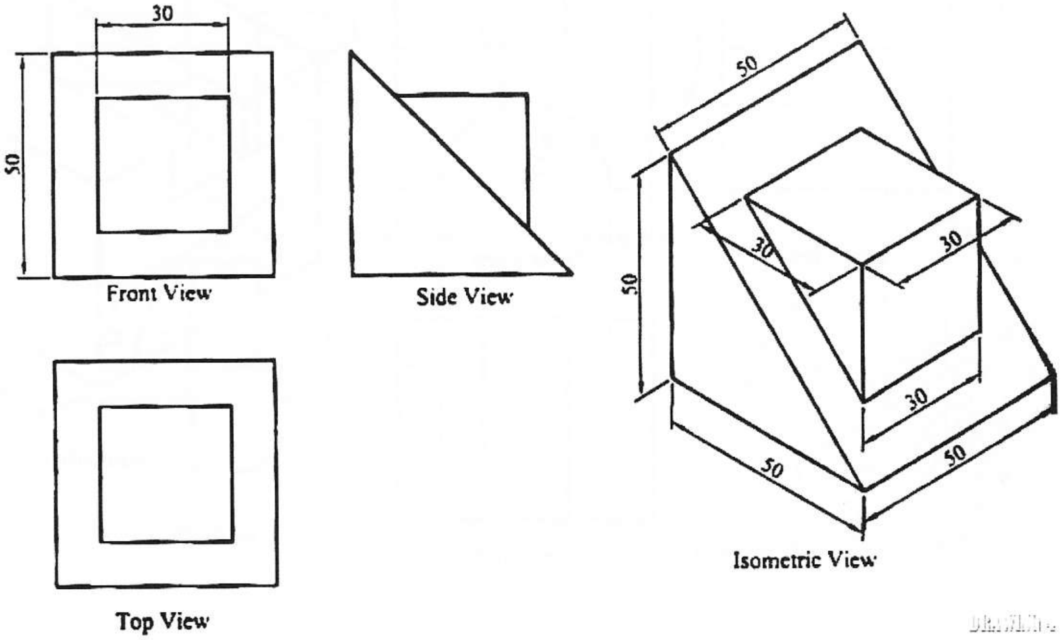
Front View

Side View

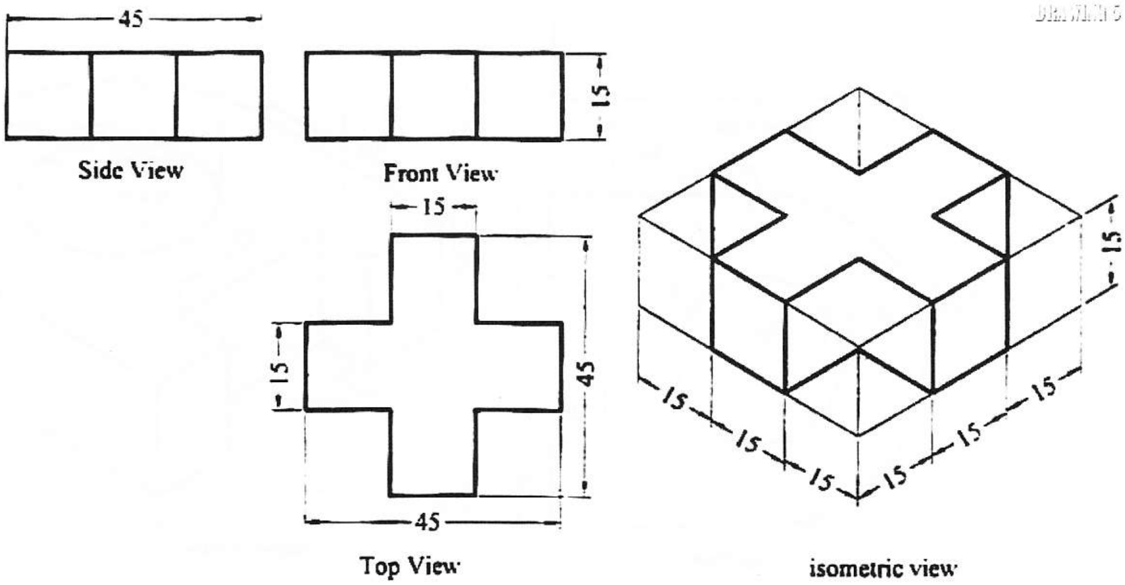
Top View

Isometric View

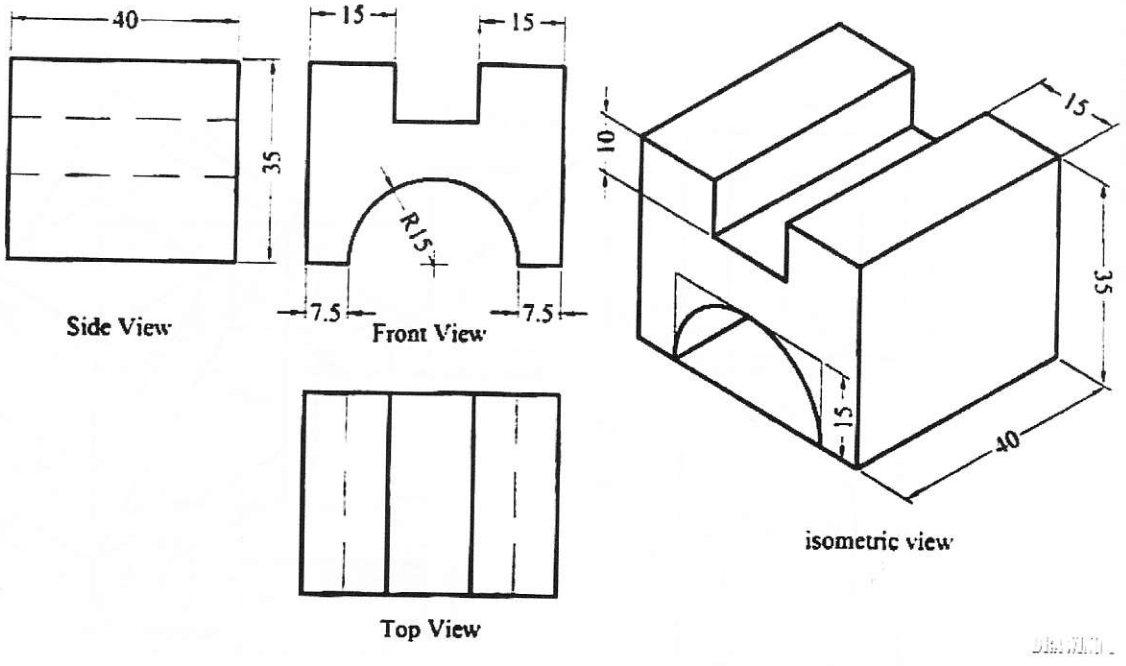
Variant 2.3



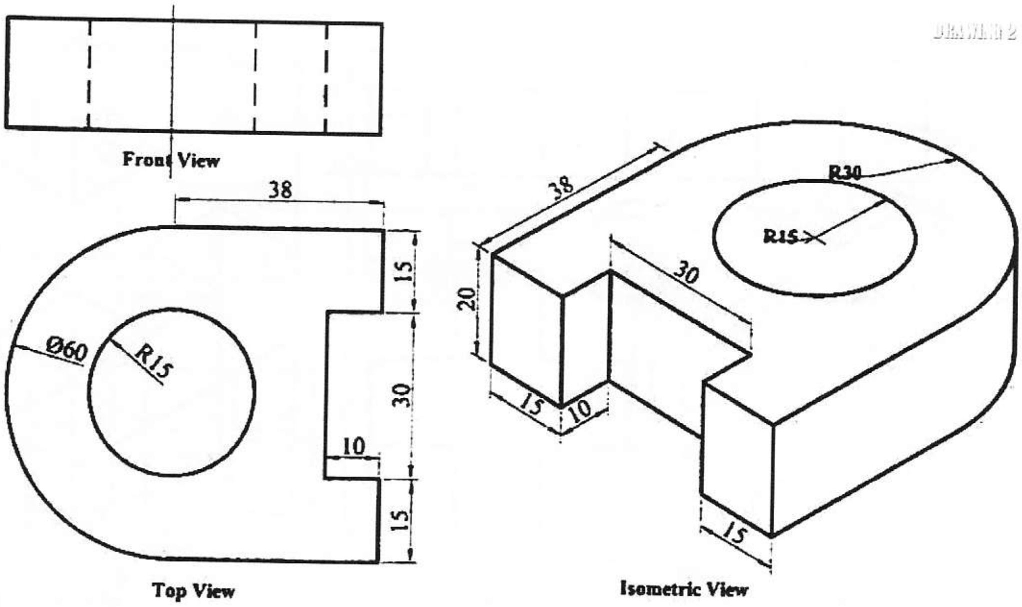
Variant 2.4



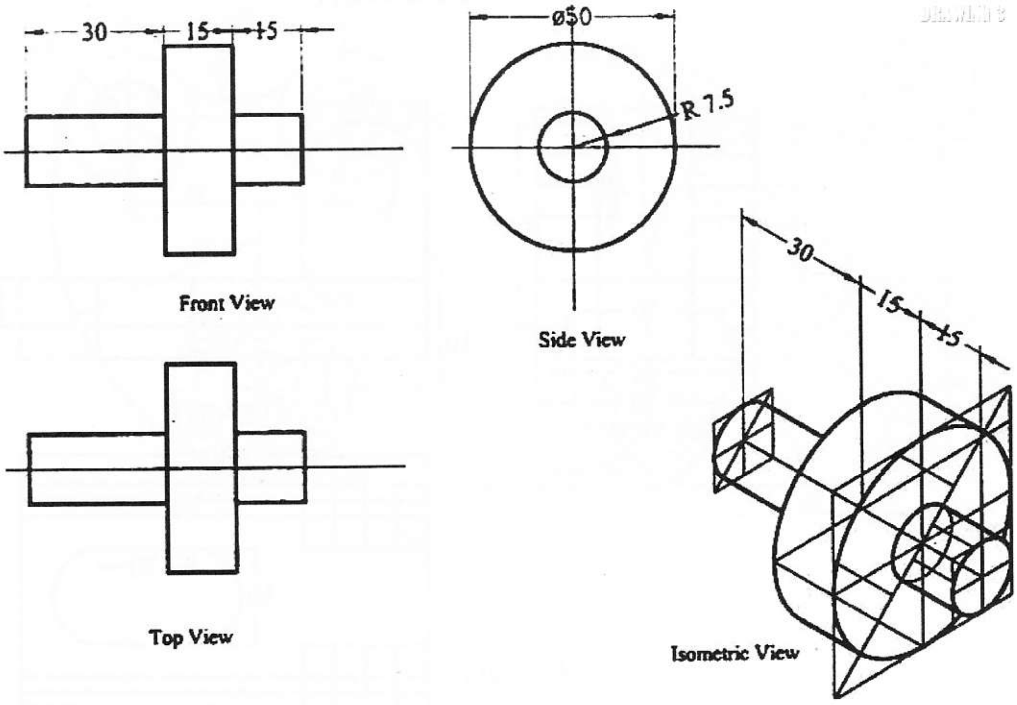
Variant 2.5



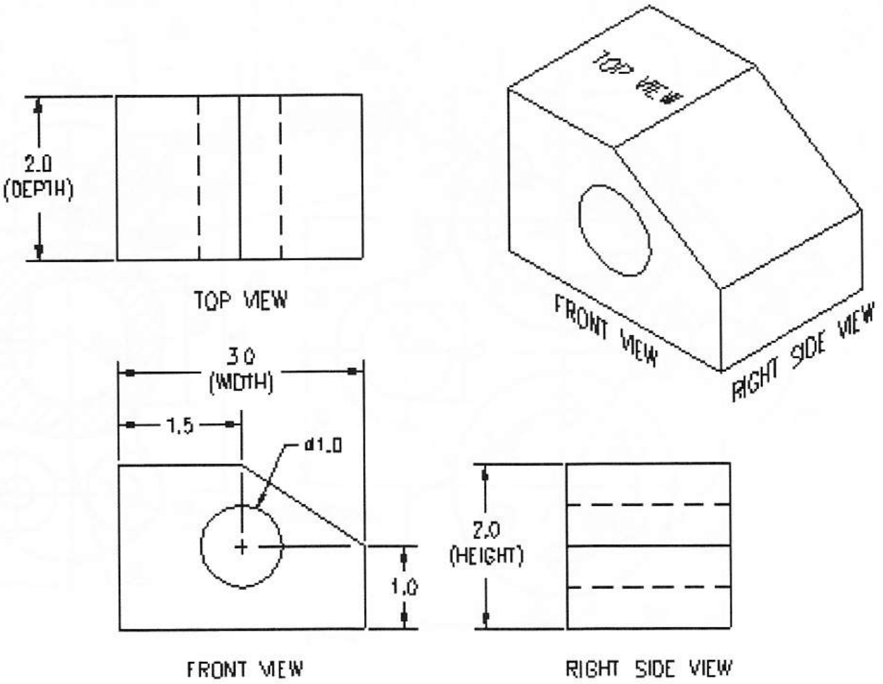
Variant 2.6



Variant 2.7

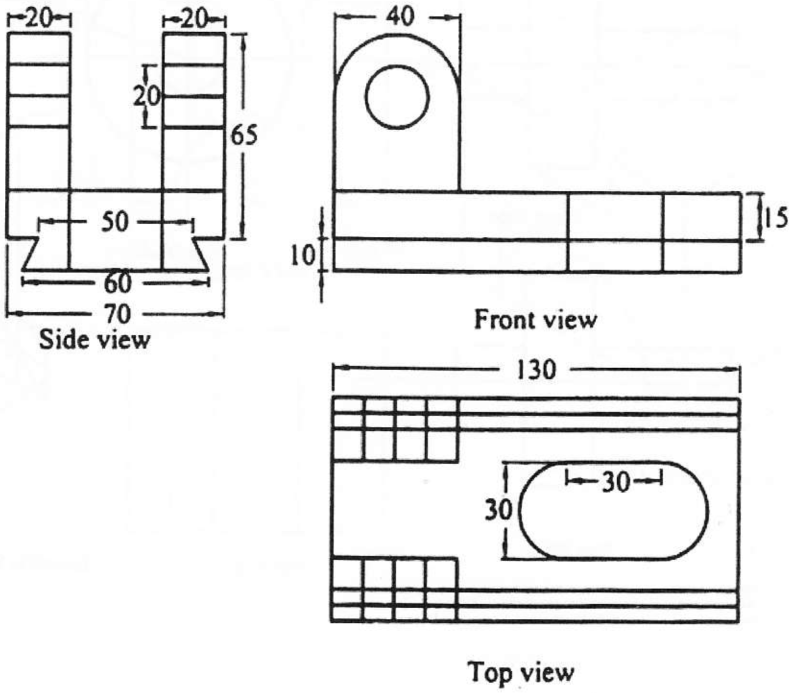


Variant 2.8

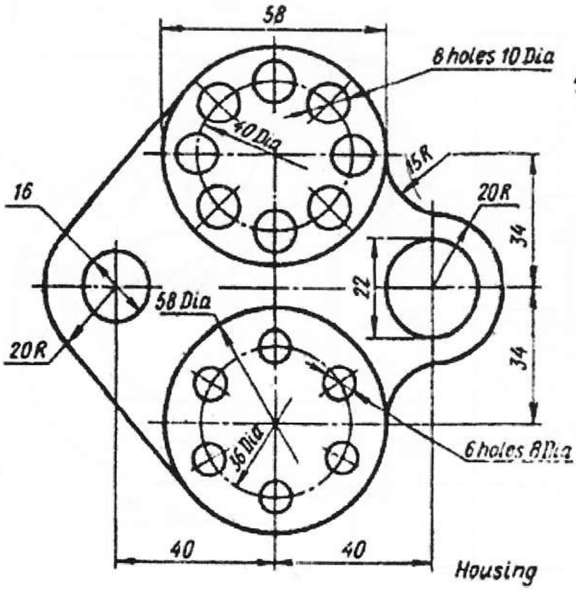


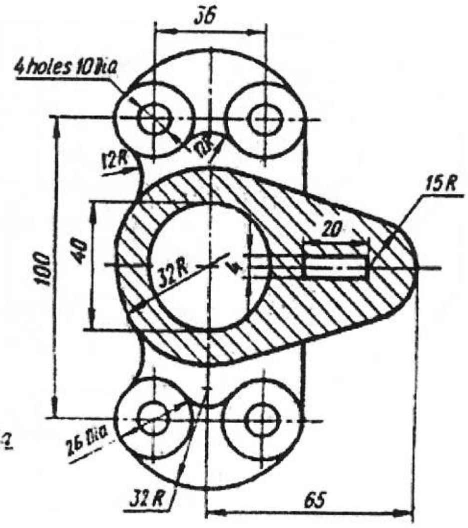
Level 3

Variant 3.1

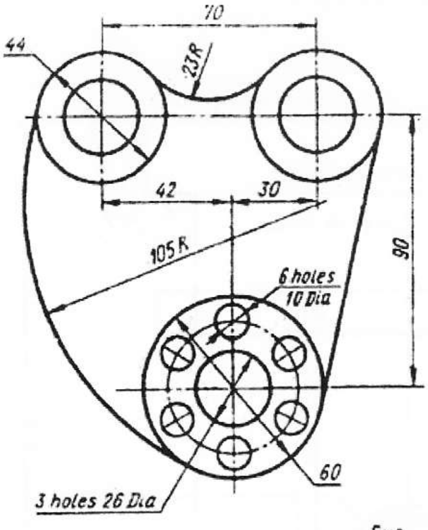


Variant 3.2

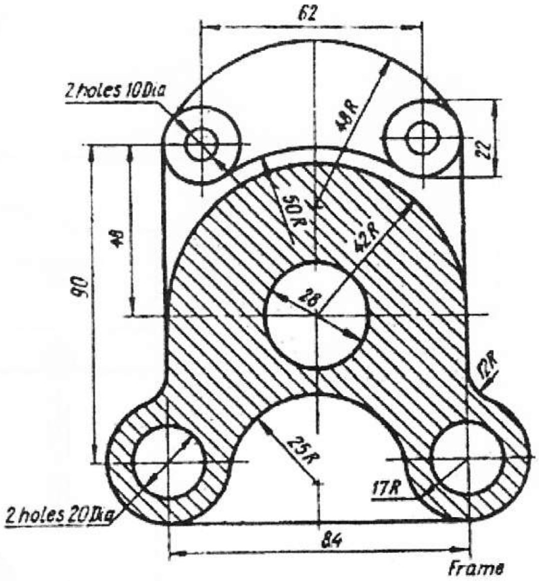




Variant 3.3

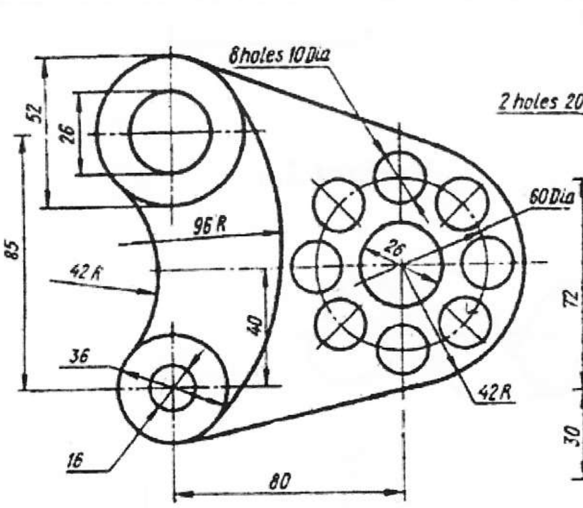


Eye

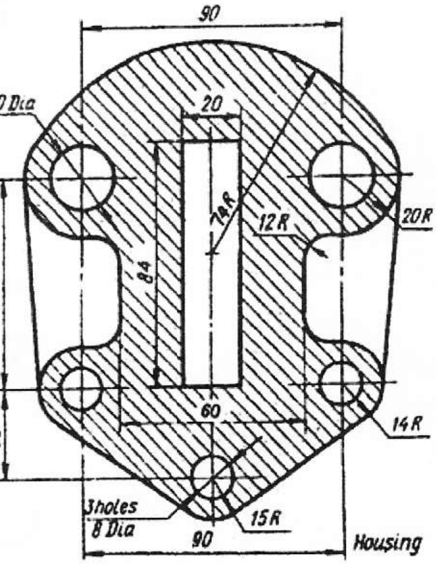


Frame

Variant 3.4

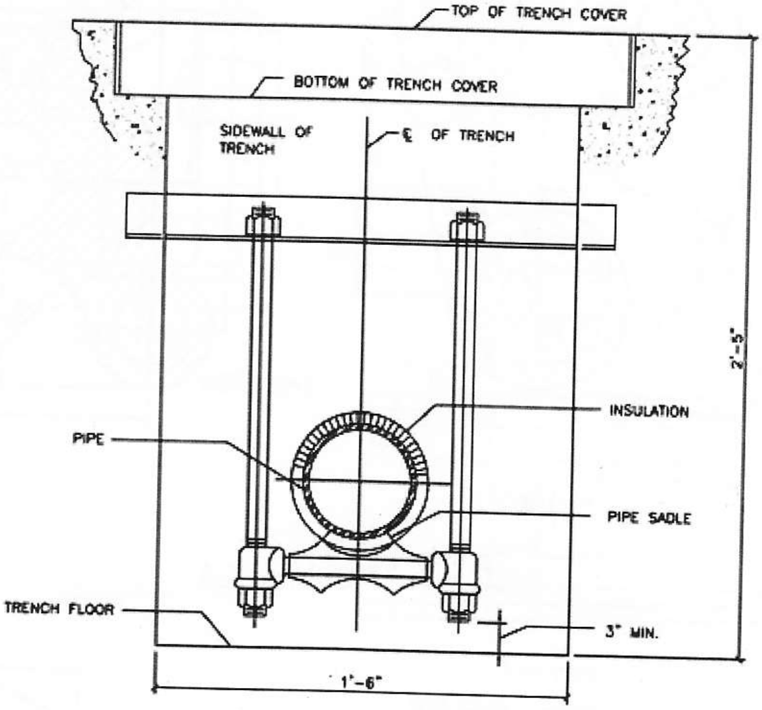


Plate

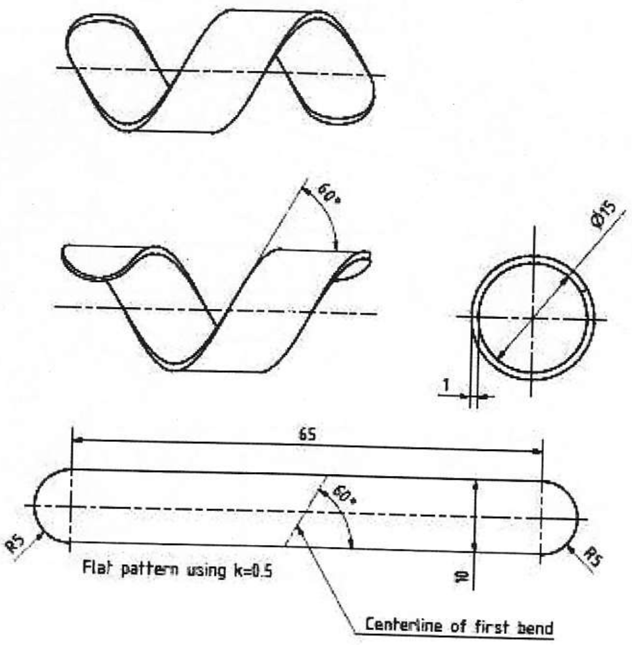


Housing

Variant 3.5



Variant 3.6



improve processing and input/output speeds by using multiple CPUs and disks in parallel. Centralized and client–server database systems are not powerful enough to handle such applications. In parallel processing, many operations are performed simultaneously, as opposed to serial processing, in which the computational steps are performed sequentially.

Parallel databases can be roughly divided into two groups, the first group of architecture is the multiprocessor architecture, the alternatives of which are the followings :

- Shared memory architecture, where multiple processors share the main memory space, as well as mass storage (e.g. hard disk drives).
- Shared disk architecture, where each node has its own main memory, but all nodes share mass storage, usually a storage area network. In practice, each node usually also has multiple processors.
- Shared nothing architecture, where each node has its own mass storage as well as main memory.

The other architecture group is called hybrid architecture, which includes:

- Non-Uniform Memory Architecture (NUMA), which involves the Non-Uniform Memory Access.
- Cluster (shared nothing + shared disk: SAN/NAS), which is formed by a group of connected computers.

Real-time databases

A real-time database is a processing system designed to handle workloads whose state is constantly changing. This differs from traditional databases containing persistent data, mostly unaffected by time. For example, a stock market changes very rapidly and is dynamic. The graphs of the different markets appear to be very unstable and yet a database has to keep track of current values for all of the markets of the New York Stock Exchange. Real-time processing means that a transaction is processed fast enough for the result to come back and be acted on right away. Real-time databases are useful for accounting, banking, law, medical records, multi-media, process control, reservation systems, and scientific data analysis. As computers increase in power and can store more data, they are integrating themselves into our society and are employed in many applications.

Spatial databases

A spatial database is a database that is optimized to store and query data that is related to objects in space, including points, lines and polygons. While typical databases can understand various numeric and character types of data, additional functionality needs to be added for databases to process spatial data types. These are typically called geometry or feature.

Temporal databases

A temporal database is a database with built-in time aspects, for example a temporal data model and a temporal version of Structured Query Language.

More specifically the temporal aspects usually include valid-time and transaction-time. These attributes go together to form bitemporal data.

Valid time denotes the time period during which a fact is true with respect to the real world.

Transaction time is the time period during which a fact is stored in the database.

Bitemporal data combines both Valid and Transaction Time.

Unstructured-data databases

Unstructured Data (or unstructured information) refers to information that either does not have a pre-defined data model and/or does not fit well into relational tables. Unstructured information is typically text-heavy, but may contain data such as dates, numbers, and facts as well. This results in irregularities and ambiguities that make it difficult to understand using traditional computer programs as compared to data stored in fielded form in databases or annotated (semantically tagged) in documents.

An unstructured-data database is intended to store in a manageable and protected way diverse objects that do not fit naturally and conveniently in common databases. It may include email messages, documents, journals, multimedia objects etc. The name may be misleading since some objects can be highly structured. However, the entire possible object collection does not fit into a predefined structured framework. Most established DBMSs now support unstructured data in various ways, and new dedicated DBMSs are emerging.

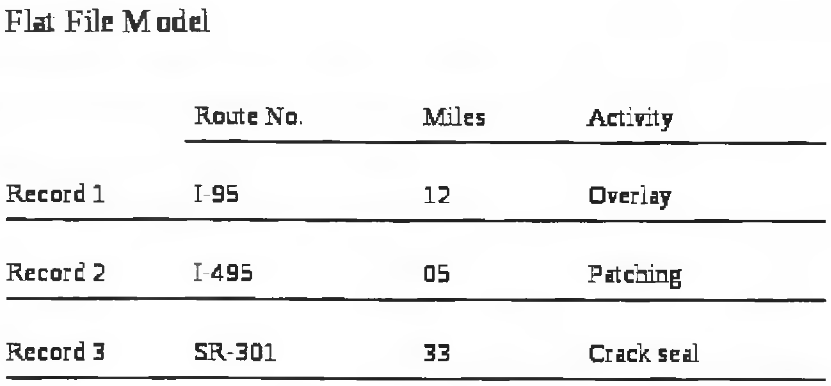
5.2. Databases models

Various techniques are used to model data structure. Most database systems are built around one particular data model, although it is possible for products to offer support for more than one model. For any logical model various physical implementations may be possible, and most products will offer the user some level of control in tuning the physical implementation, since the choices that are made have a significant effect on performance.

A database model is the theoretical foundation of a database and fundamentally determines in which manner data can be stored, organized, and manipulated in a database system. It thereby defines the infrastructure offered by a particular database system. The most popular example of a database model is the relational model. A database model is a theory or specification describing how a database is structured and used.

Flat model

The flat (or table) model (Figure 5.2) consists of a single, two-dimensional array of data elements, where all members of a given column are assumed to be similar values, and all members of a row are assumed to be related to one another. For instance, columns for name and password that might be used as a part of a system security database. Each row would have the specific password associated with an individual user. Columns of the table often have a type associated with them, defining them as character data, date or time information, integers, or floating point numbers. This may not strictly qualify as a data model, as defined above.



Flat File Model

	Route No.	Miles	Activity
Record 1	I-95	12	Overlay
Record 2	I-495	05	Patching
Record 3	SR-301	33	Crack seal

Figure 5.2.- Flat Database Model

Hierarchical model

In a hierarchical model (Figure 5.3), data is organized into a tree-like structure, implying a single upward link in each record to describe the nesting, and a sort field to keep the records in a particular order in each same-level list. Hierarchical structures were widely used in the early mainframe

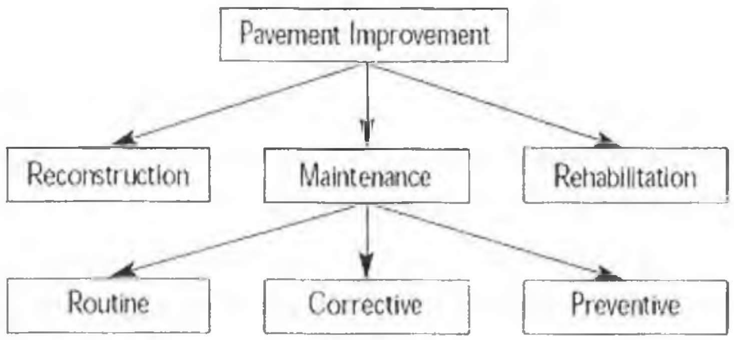


Figure 5.3.- Hierarchical Database Model

database management systems, such as the Information Management System (IMS) by IBM, and now describe the structure of XML documents. This structure allows one 1:M relationship between two types of data. This structure is very efficient to describe many relationships in the real world; recipes, table of contents, ordering of paragraphs/verses, any nested and sorted information. However, the hierarchical structure is inefficient for certain database operations when a full path (as opposed to upward link and sort field) is not also included for each record.

Mother–child relationship: Child may only have one mother but a mother can have multiple children. Mothers and children are tied together by links called "pointers". A mother will have a list of pointers to each of her children.

Network model

The network model (Figure 5.4) (defined by the CODASYL specification) organizes data using two fundamental constructs, called records and sets. Records contain fields (which may be organized hierarchically, as in the programming language COBOL). Sets (not to be confused with mathematical sets) define one-to-many relationships between records: one owner, many members. A record may be an owner in any number of sets, and a member in any number of sets.

The network model is a variation on the hierarchical model, to the extent that it is built on the concept of multiple branches (lower-level structures) emanating from one or more nodes (higher-level structures), while the model differs from the hierarchical model in that branches can be connected to multiple nodes. The network model is able to represent redundancy in data more efficiently than in the hierarchical model.

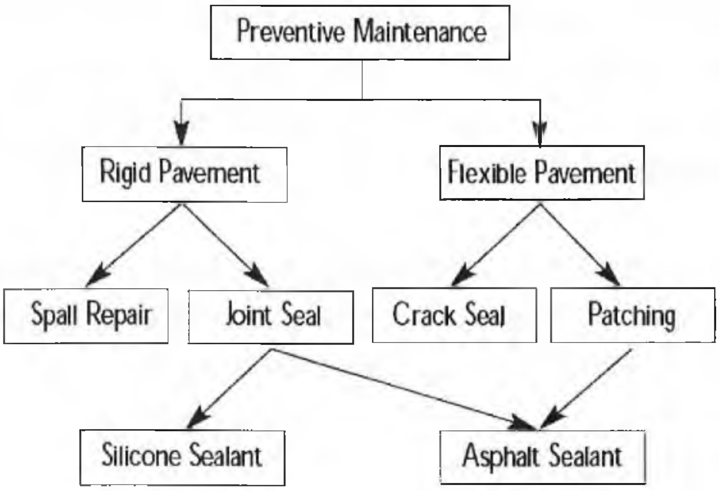


Figure 5.4.- Network Database Model

The operations of the network model are navigational in style: a program maintains a current position, and navigates from one record to another by following the relationships in which the record participates. Records can also be located by supplying key values.

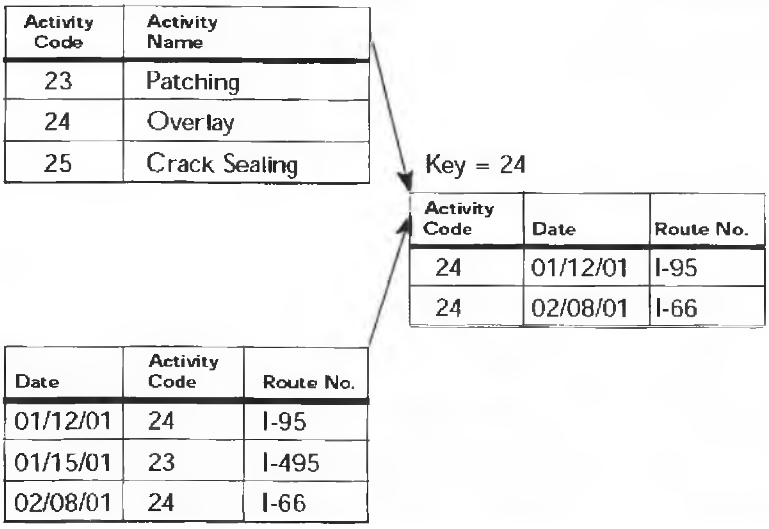
Although it is not an essential feature of the model, network databases generally implement the set relationships by means of pointers that directly address the location of a record on disk. This gives excellent retrieval performance, at the expense of operations such as database loading and reorganization.

Most object databases use the navigational concept to provide fast navigation across networks of objects, generally using object identifiers as "smart" pointers to related objects. Objectivity/DB, for instance, implements named 1:1, 1:many, many:1 and many:many named relationships that can cross databases. Many object databases also support SQL, combining the strengths of both models.

Relational model

The relational model (Figure 5.5) was introduced by E.F. Codd in 1970 as a way to make database management systems more independent of any particular application. It is a mathematical model defined in terms of predicate logic and set theory.

The products that are generally referred to as relational databases in fact implement a model that is only an approximation to the mathematical model defined by Codd. Three key terms are used extensively in relational database models: relations, attributes, and domains. A relation is a table with columns and rows. The named columns of the relation are called attributes, and the domain is the set of values the attributes are allowed to take.



Activity Code	Activity Name
23	Patching
24	Overlay
25	Crack Sealing

Activity Code	Date	Route No.
24	01/12/01	I-95
24	02/08/01	I-66

Date	Activity Code	Route No.
01/12/01	24	I-95
01/15/01	23	I-495
02/08/01	24	I-66

Figure 5.5.- Relational Database Model

The basic data structure of the relational model is the table, where information about a particular entity (say, an employee) is represented in rows (also called tuples) and columns. Thus, the "relation" in "relational database" refers to the various tables in the database; a relation is a set of tuples. The columns enumerate the various attributes of the entity (the employee's name, address or phone number, for example), and a row is an actual instance of the entity (a specific employee) that is represented by the relation. As a result, each tuple of the employee table represents various attributes of a single employee.

All relations (and, thus, tables) in a relational database have to adhere to some basic rules to qualify as relations. First, the ordering of columns is immaterial in a table. Second, there can't be identical tuples or rows in a table. And third, each tuple will contain a single value for each of its attributes.

A relational database contains multiple tables, each similar to the one in the "flat" database model. One of the strengths of the relational model is that, in principle, any value occurring in two different records (belonging to the same table or to different tables), implies a relationship among those two records. Yet, in order to enforce explicit integrity constraints, relationships between records in tables can also be defined explicitly, by identifying or non-identifying parent-child relationships characterized by assigning cardinality (1:1, (0)1:M, M:M). Tables can also have a designated single attribute or a set of attributes that can act as a "key", which can be used to uniquely identify each tuple in the table.

A key that can be used to uniquely identify a row in a table is called a primary key. Keys are commonly used to join or combine data from two or more tables. For example, an Employee table may contain a column named Location which contains a value that matches the key of a Location table. Keys are also critical in the creation of indexes, which facilitate fast retrieval of data from large tables. Any column can be a key, or multiple columns can be grouped together into a compound key. It is not necessary to define all the keys in advance; a column can be used as a key even if it was not originally intended to be one.

A key that has an external, real-world meaning (such as a person's name, a book's ISBN, or a car's serial number) is sometimes called a "natural" key. If no natural key is suitable (think of the many people named Brown), an arbitrary or surrogate key can be assigned (such as by giving employees ID numbers). In practice, most databases have both generated and natural keys, because generated keys can be used internally to create links between rows that cannot break, while natural keys can be used, less reliably, for searches and for integration with other databases. (For example, records in two independently developed databases could be matched up by social security number, except when the social security numbers are incorrect, missing, or have changed.)

Dimensional model

The dimensional model (Figure 5.6) is a specialized adaptation of the relational model used to represent data in data warehouses in a way that data can be easily summarized using OLAP queries. In the dimensional model, a database schema consists of a single large table of facts that are described using dimensions and measures.

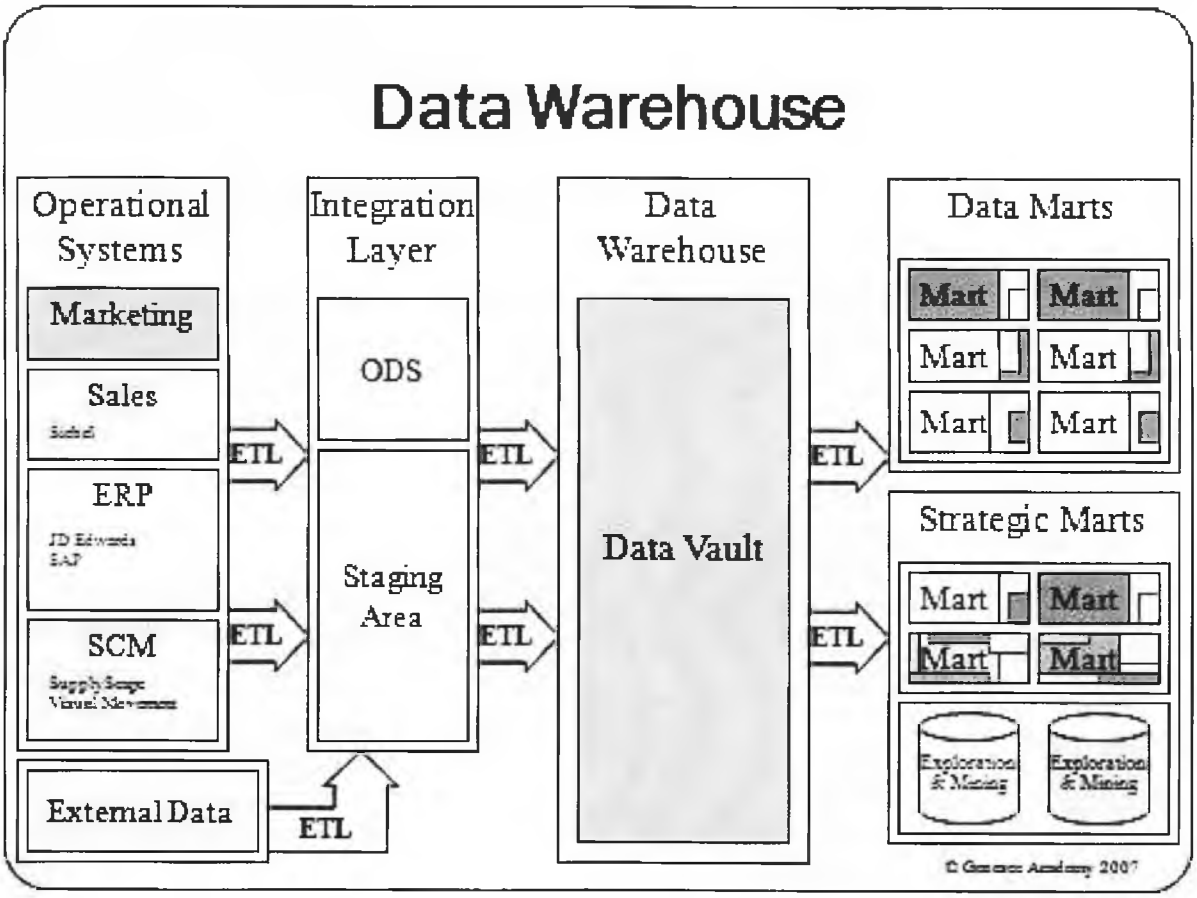


Figure 5.6.- Dimensional Database Model

A dimension provides the context of a fact (such as who participated, when and where it happened, and its type) and is used in queries to group related facts together. Dimensions tend to be discrete and are often hierarchical; for example, the location might include the building, state, and country. A measure is a quantity describing the fact, such as revenue. It's important that measures can be meaningfully aggregated--for example, the revenue from different locations can be added together.

In an OLAP query, dimensions are chosen and the facts are grouped and aggregated together to create a summary.

The dimensional model is often implemented on top of the relational model using a star schema, consisting of one highly normalized table containing the facts, and surrounding denormalized tables containing each dimension. An alternative physical implementation, called a snowflake schema, normalizes multi-level hierarchies within a dimension into multiple tables.

A data warehouse can contain multiple dimensional schemas that share dimension tables, allowing them to be used together. Coming up with a standard set of dimensions is an important part of dimensional modeling.

Objectional database models

In recent years, the object-oriented paradigm has been applied to database technology, creating a new programming model known as object databases (Figure 5.7). These databases attempt to bring the database world and the application programming world closer together, in particular by ensuring that the database uses the same type system as the application program. This aims to avoid the overhead (sometimes referred to as the impedance mismatch) of converting information between its representation in the database (for example as rows in tables) and its representation in the application program (typically as objects). At the same time, object databases attempt to introduce the key ideas of object programming, such as encapsulation and polymorphism, into the world of databases.

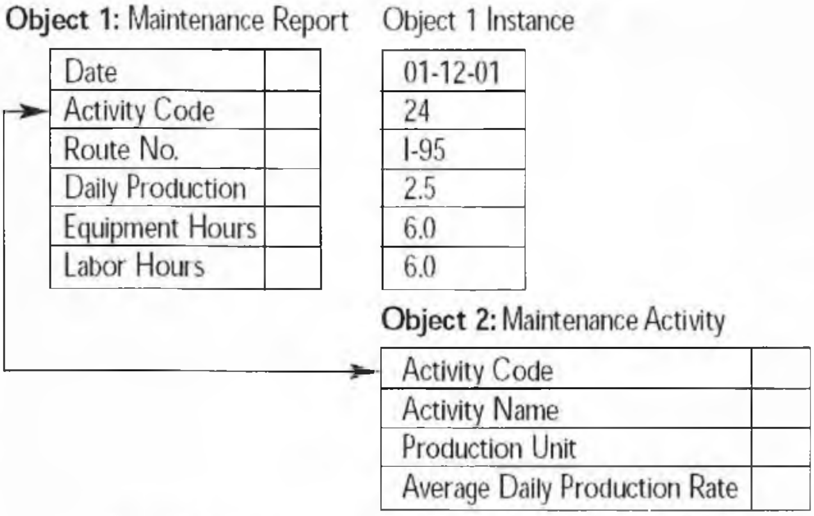


Figure 5.7.- Objectional Database Model

A variety of these ways have been tried for storing objects in a database. Some products have approached the problem from the application programming end, by making the objects manipulated by the program persistent. This also typically requires the addition of some kind of query language, since conventional programming languages do not have the ability to find objects based on their information content. Others have attacked the problem from the database end, by defining an object-oriented data model for the database, and defining a database programming language that allows full programming capabilities as well as traditional query facilities.

Object databases suffered because of a lack of standardization: although standards were defined by ODMG, they were never implemented well enough to ensure interoperability between products. Nevertheless, object databases have been used successfully in many applications: usually specialized applications such as engineering databases or molecular biology databases rather than mainstream commercial data processing. However, object database ideas were picked up by the relational vendors and influenced extensions made to these products and indeed to the SQL language.

5.3. Database Design

Generic representation of the database design process is shown on Figure 5.8.

The First Step

The first step in designing the database is to know what are the objects or so called "Entities". How to determine what the entities are? The textbook says: "Read the problem statement or the scenario and try to identify nouns." From a collection of nouns it would be much clearer to determine what the entities are in your system. This is called the "ROOT CAUSE ANALYSIS".

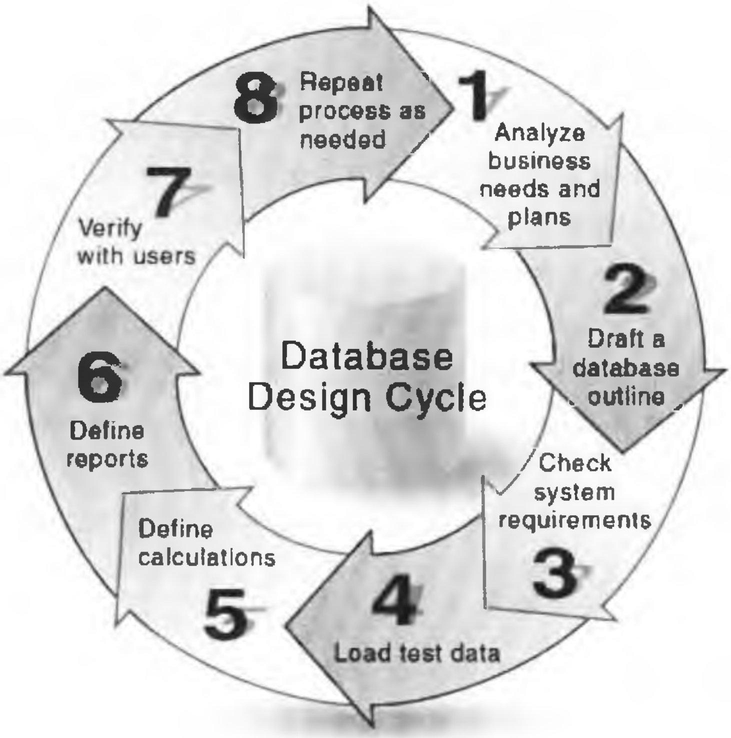

8 Repeat process as needed
1 Analyze business needs and plans
2 Draft a database outline
Check system requirements 3
4 Load test data
Define calculations 5
6 Define reports
7 Verify with users
Database Design Cycle

Figure 5.8.- Database Design Process

Example. If someone wants to develop an online forum and have to design the database for this, it is necessary to generate a problem statement. The requirement would be vaguely "A discussion forum has many categories. In each category there are many topics and each topic may have many messages by different users. Old messages (let's say 10 days old) are deleted automatically by the system. User can post new messages and edit their message" what would the database design be?It is evident with the problem statement that you need to keep track of categories, messages (post) and user. An entity in the system might be a strong or a weak entity. A Weak entity is one which cannot exist on its own in a system and requires a relationship with a strong entity (the one with a primary key) to identify its records. Such an entity is described by a double lined rectangle in the model. What attributes to store? What is a primary key? What is a partial Key? What are candidate keys? What is a super key? How to select them: Simply ask yourself what is the information that the system should store for a particular entity, importantly how that information would be identified at a later time? Is there a way, an attribute which uniquely identifies any record/row/ tuple in the table? What are some of the other important attributes the system has to store?

The answer to these questions would be very important in terms of database design. Primarily all entities would be converting to tables so what are the columns of that table. They are the characteristics of the entity. e.g. In our scenario what are the important attributes of a User (in this example we only require a loginid and a password but in a slightly complex scenario like a university registration system there might be many attributes like address , phone number , date of birth , previous college , social security etc Similarly what is the data about the categories we want to store (I guess the category name , Description , we would like to know the number of messages in the category and the topic with maximum posts and stuff like that but lets not talk about that here)The set of attributes which can together uniquely identify an entity instance constitute the Super key. The minimum set of attributes (subset of super key) which can uniquely identify a record in the database is called the candidate key. There can be more than one candidate key for the entity e.g. A Student number and the Social security number we use only one as the primary key of the table and the candidate key which we use to uniquely identify any record in the system is called the Primary Key (The programmer assigned candidate key). Note in case of weak entities the identifying attribute is called the partial key instead of the primary key.

Step 2: What are relationships and why? What is cardinality and multiplicity?

Once the entities have identified, all of them think of the relationships and the cardinalities or multiplicity of the relationship. For the sake of simplicity check each of the entity against all others in order to clarify the design. Start with the User. How is the user related to Categories? A user can create a new category? If yes then we have a relationship. Then there are 2 things that need to be answered.1. Can a Single User create more than 1 category2. Can a Single Category be created by more than one user These two questions are the key to determine the cardinalities which would indicate how we resolve the database design in terms of tables. The relationship can have the following cardinalities:

· 1 to 1· 1 to Many· Many to 1· Many to Many

Similarly then compare if the user and topic has a relationship. That would be pretty similar to the first one. Now also compare the user with all the other entities. Note that relationship may also have attributes like the date/time when the message was posted etc. Relationships are indicated as diamonds in the design. A relationship with one of the participating entities is weak is called a weak relationship.

Figure 5.9 shows the diagram of relationships between entities.

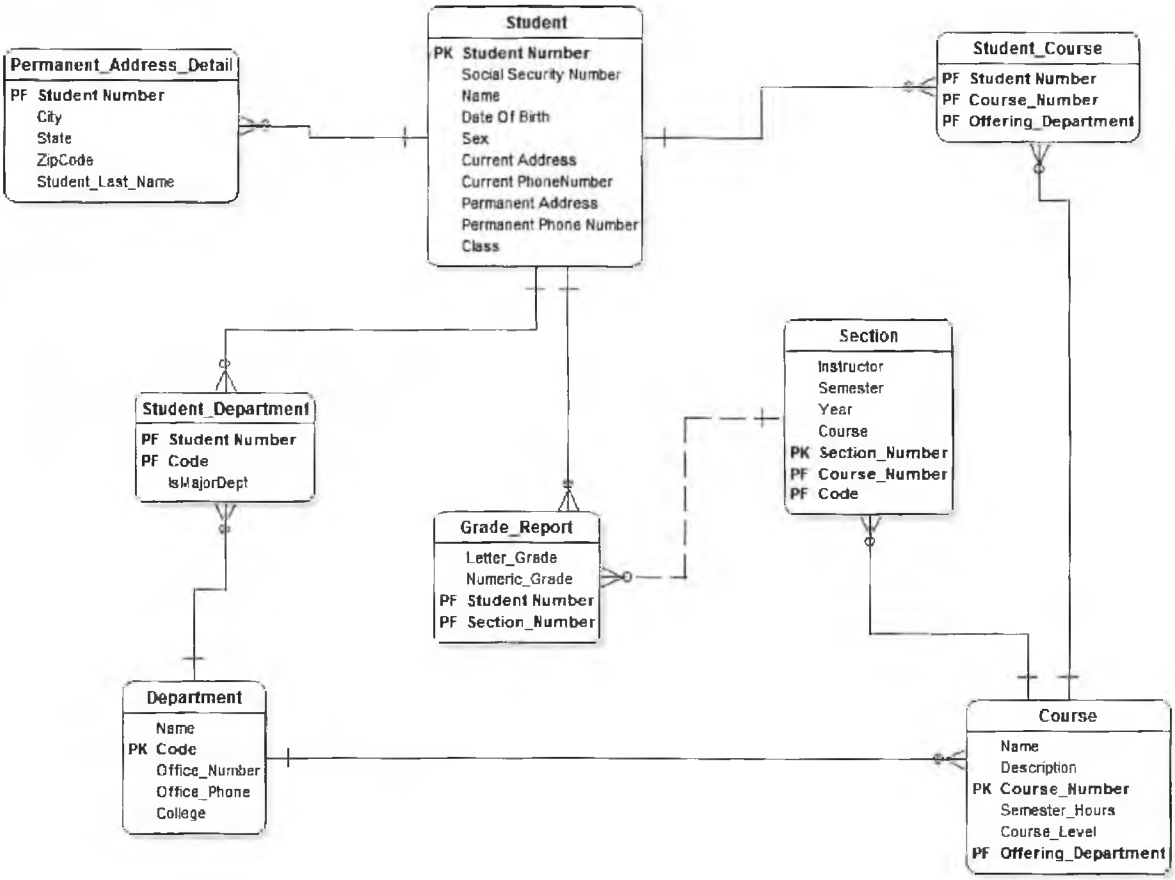


Figure 5.9.- The Diagram of Relationships between Entities

For 1 to 1 and many to many relationships create a new table for the relationship. Include the primary key of each participating entry. The union of all the primary keys which are referenced as foreign keys can uniquely identify any record in the table. So if the cardinality of both the participating entries is the same make a separate table. However if it is not take the primary key of the entity where cardinality is 1 and include it in the table where it is many as a foreign key. This is also called an identifying relationship.

Chapter 5 is prepared on the basis of information sources [29-42].

6. DATA STRUCTURES

Introduction

In computer science, a ***data structure*** is a particular way of storing and organizing data in a computer so that it can be used efficiently.

Array

In computer science, an array data structure or simply array is a data structure consisting of a collection of elements (values or variables), each identified by at least one index. An array is stored so that the position of each element can be computed from its index tuple by a mathematical formula.

An array stores a number of elements of the same type in a specific order. They are accessed using an integer to specify which element is required (although the elements may be of almost any type). Arrays may be fixed-length or expandable.

Matrix.

In mathematics, a matrix (plural matrices, or less commonly matrixes) is a rectangular array of numbers, symbols, or expressions. The individual items in a matrix are called its elements or entries.

Record (also called tuple or struct) Records are among the simplest data structures. A record is a value that contains other values, typically in fixed number and sequence and typically indexed by names. The elements of records are usually called fields or members.

A ***hash*** (or dictionary or map) is a more flexible variation on a record, in which name-value pairs can be added and deleted freely.

Union. A union type definition will specify which of a number of permitted primitive types may be stored in its instances, eg "float or long integer". Contrast with a record, which could be defined to contain a float and an integer; whereas, in a union, there is only one value at a time.

A ***tagged union*** (also called a variant, variant record, discriminated union, or disjoint union) contains an additional field indicating its current type, for enhanced type safety.

A ***set*** is an abstract data structure that can store certain values, without any particular order, and no repeated values. Values themselves are not retrieved from sets, rather one tests a value for membership to obtain a boolean "in" or "not in".

An ***object*** contains a number of data fields, like a record, and also a number of programe code fragments for accessing or modifying them. Data structures not containing code, like those above, are called plain old data structure.

Lists. In computer science, a doubly linked list is a linked data structure that consists of a set of sequentially linked records called nodes (Figure6.1).

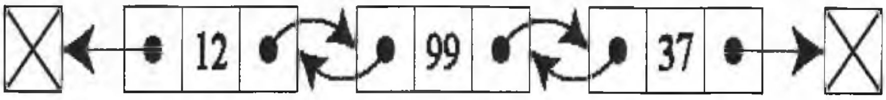


Figure 6.1.- The List

Each node contains two fields, called links, that are references to the previous and to the next node in the sequence of nodes. The beginning and ending nodes' previous and next links, respectively, point to some kind of terminator, typically a sentinel node or null, to facilitate traversal of the list. If there is only one sentinel node, then the list is circularly linked via the sentinel node. It can be conceptualized as two singly linked lists formed from the same data items, but in opposite sequential orders.

Binary trees

In computer science, a binary tree is a tree data structure in which each node has at most two child nodes, usually distinguished as "left" and "right" (Figure 6.2). Nodes with children are parent nodes, and child nodes may contain references to their parents. Outside the tree, there is often a reference to the "root" node (the ancestor of all nodes), if it exists. Any node in the data structure can be reached by starting at root node and repeatedly following references to either the left or right child.

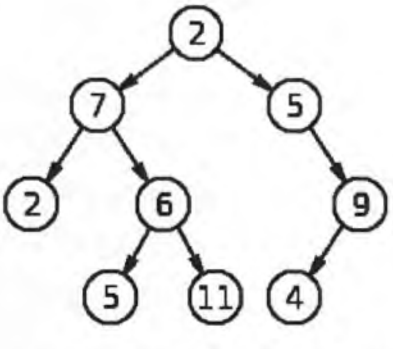


Figure 6.2.- The Binary Tree

Ropes

In computer programming a rope, or cord, is a data structure for efficiently storing and manipulating a very long string. For example, a text editing program may use a rope to represent the text being edited, so that operations such as insertion, deletion, and random access can be done efficiently.

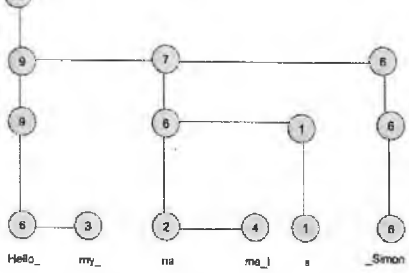


Figure 6.3.- The Rope

Graphs

In computer science, a graph is an abstract data structure that is meant to implement the graph and hypergraph concepts from mathematics.

A graph data structure consists of a finite (and possibly mutable) set of ordered pairs, called edges or arcs, of certain entities called nodes or vertices (Figure 6.4). As in mathematics, an edge (x,y) is said to point or go from x to y. The nodes may be part of the graph structure, or may be external entities represented by integer indices or references.

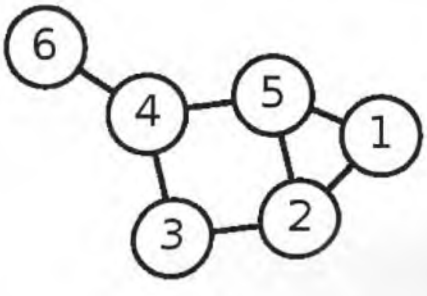
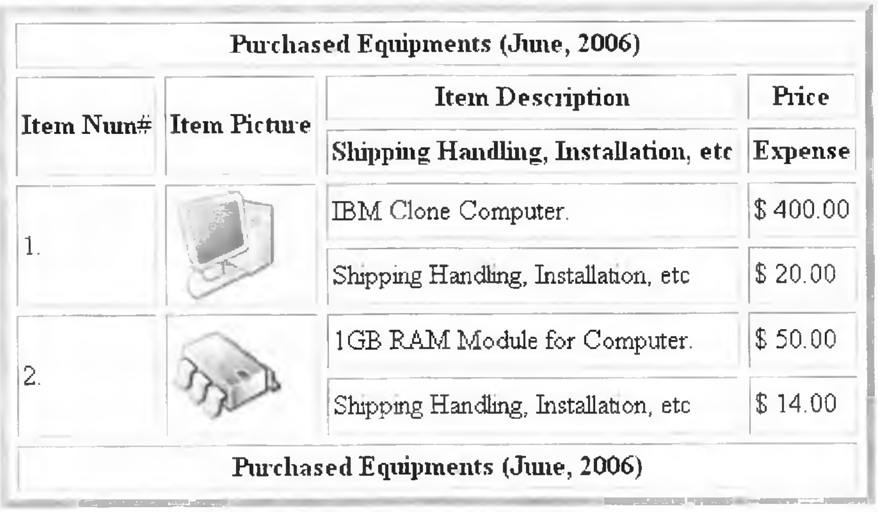


Figure 6.4.- The Graph

A graph data structure may also associate to each edge some edge value, such as a symbolic label or a numeric attribute (cost, capacity, length, etc.).

Tables

A table is a means of arranging data in rows and columns (Figure 6.5).



Purchased Equipments (June, 2006)			
Item Num#	Item Picture	Item Description	Price
		Shipping Handling, Installation, etc	Expense
1.		IBM Clone Computer.	$ 400.00
		Shipping Handling, Installation, etc	$ 20.00
2.		1GB RAM Module for Computer.	$ 50.00
		Shipping Handling, Installation, etc	$ 14.00
Purchased Equipments (June, 2006)			

Figure 6.5.- The Table

Object

In computer science, an object is any entity that can be manipulated by the commands of a programming language, such as a value, variable, function, or data structure. In this way, while primitive or simple data types are still just single pieces of information, object-oriented objects are complex types that have multiple pieces of information and specific properties (or attributes). Instead of merely being assigned a value, (like int =10), objects have to be "constructed". In the real world, if a Ford Focus is an "object" - an instance of the car class, its physical properties and its function to drive would have been individually specified.

6.1. Data Modeling

A data model is not just a way of structuring data: it also defines a set of operations that can be performed on the data. The relational model, for example, defines operations such as selection (project) and joining. Although these operations may not be explicit in a particular query language, they provide the foundation on which a query language is built.

Data modeling in software engineering is the process of creating a data model by applying formal data model descriptions using data modeling techniques (Figure 6.6).

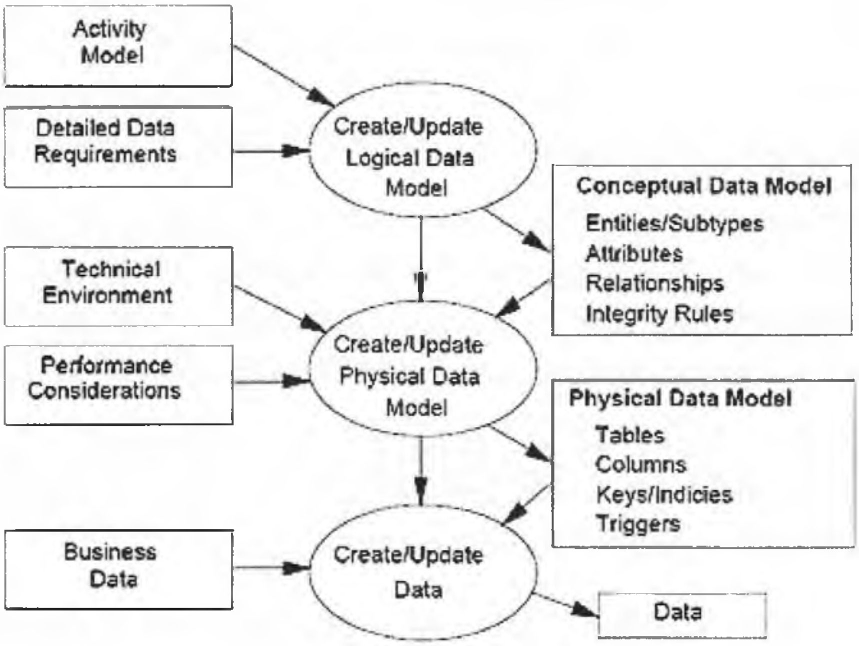


Figure 6.6.- Data Modeling Techniques

The figure illustrates the way data models are developed and used today. A conceptual data model is developed based on the data requirements for the application that is being developed, perhaps in the context of an activity model. The data model will normally consist of entity types, attributes, relationships, integrity rules, and the definitions of those objects. This is then used as the start point for interface or database design.

The data modeling process

Data modeling is a method used to define and analyze data requirements needed to support the processes. The data requirements are recorded as a conceptual data model with associated data definitions. Actual implementation of the conceptual model is called a logical data model. To implement one conceptual data model may require multiple logical data models. Data modeling defines not just data elements, but their structures and relationships between them. Data modeling techniques and methodologies are used to model data in a standard, consistent, predictable manner in order to manage it as a resource.

Data modeling may be performed during various types of prcesses and workflows, and in multiple phases of processes. For instance, in systems analysis logical data models are created as part of the development of new databases. It is sometimes called database modeling because a data model is eventually implemented in a database.

Data models for different systems are arbitrarily different. The result of this is that complex interfaces are required between systems that share data. These interfaces can account for between 25-70% of the cost of current systems.

A data model instance may be one of three kinds (according to ANSI in 1975):

1. *Conceptual schema*: describes the semantics of a domain, being the scope of the model. A conceptual schema specifies the kinds of facts or propositions that can be expressed using the model. In that sense, it defines the allowed expressions in an artificial 'language' with a scope that is limited by the scope of the model.

2. *Logical schema*: describes the semantics, as represented by a particular data manipulation technology. This consists of descriptions of tables and columns, object oriented classes, and XML tags, among other things.

3. *Physical schema*: describes the physical means by which data are stored. This is concerned with partitions, CPUs, tablespaces, and the like.

4. The actual database design is the process of producing a detailed data model of a database. Figure 6.7 shows the process of data modeling will result in database generation.

A fully attributed data model contains detailed attributes for each entity. The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the

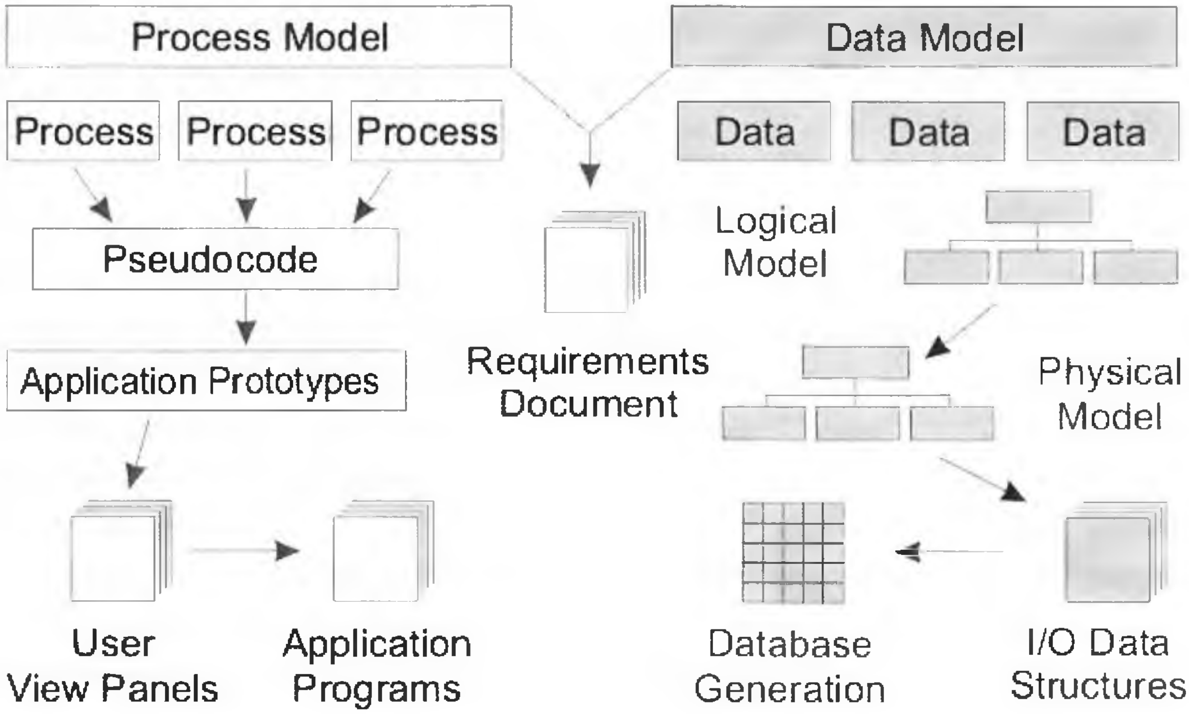


Figure 6.7.- Data modeling Process

logical design of the base data structures used to store the data. In the relational model these are the tables and views. In an Object database the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the Database Management System or DBMS.

6.2. Modeling methodologies

Data models represent information areas of interest. While there are many ways to create data models, only two modeling methodologies stand out: top-down and bottom-up.

Bottom-up models are often the result of a reengineering effort. They usually start with existing data structures forms, fields on application screens, or reports. These models are usually physical, application-specific, and incomplete from an enterprise perspective. They may not promote data sharing, especially if they are built without reference to other parts of the infrastructure.

Top-down logical data models, on the other hand, are created in an abstract way by getting information from people who know the subject area. A system may not implement all the entities in a logical model, but the model serves as a reference point or template.

Sometimes models are created in a mixture of the two methods: by considering the data needs and structure of an application and by consistently referencing a subject-area model.

6.3. Entity relationship diagrams

Several techniques have been developed for the design of data models. While these methodologies guide data modelers in their work, two different people using the same methodology will often come up with very different results. Most notable are: Bachman diagrams, Barker's Notation, Chen, Data Vault Modeling, Extended Backus–Naur form, IDEF1X, Object-relational mapping, Object Role Modeling, Relational Model.

Figure 6.8 demonstrates a sample entity-relationship diagram.

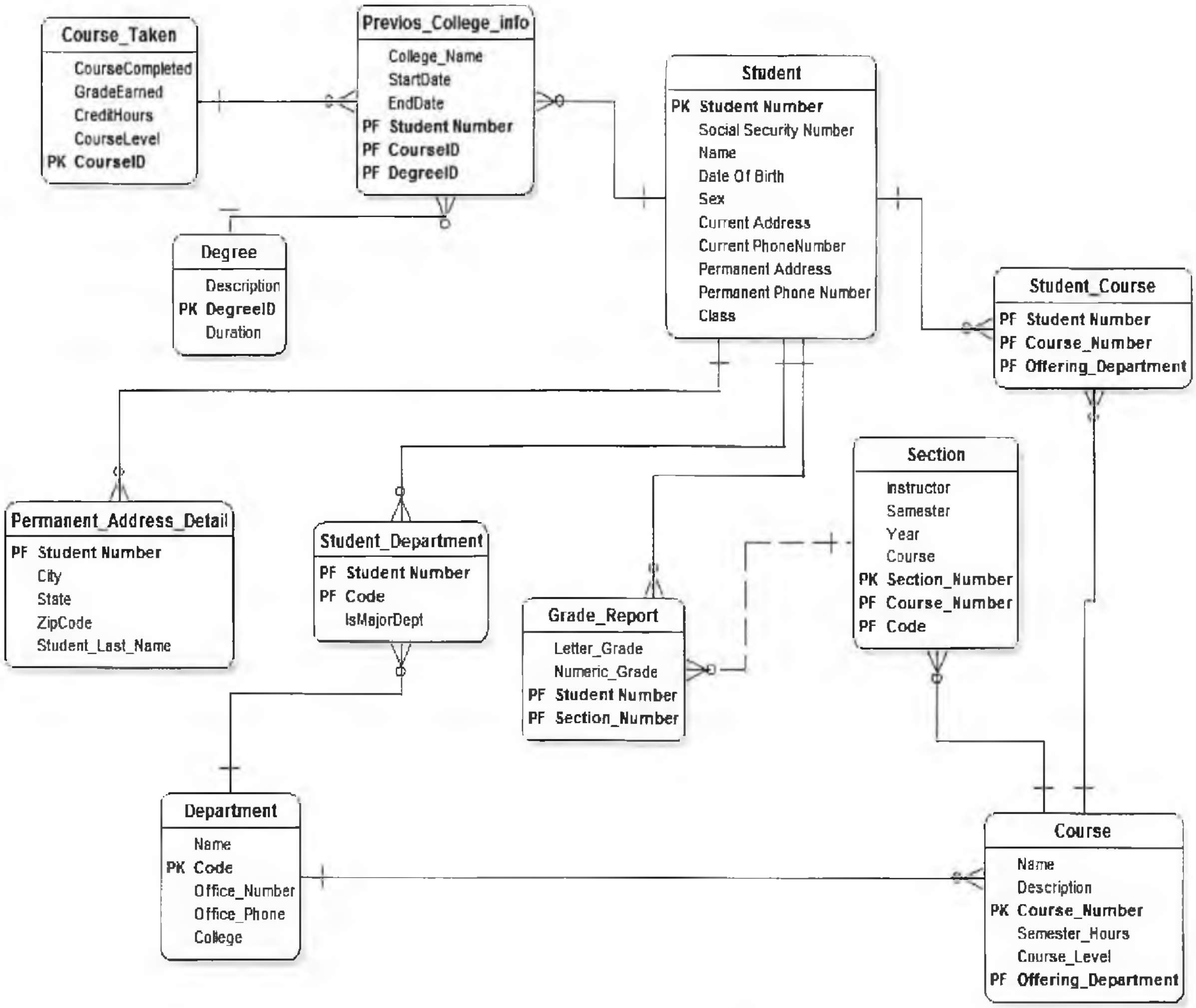


Figure 6.8.- Entity-Relationship Diagram

The *actual model* is frequently called "Entity relationship model", because it depicts data in terms of the entities and relationships described in the data. An entity-relationship model (ERM) is an abstract conceptual representation of structured data. Entity-relationship modeling is a relational schema database modeling method, used in software engineering to produce a type of conceptual data model (or semantic data model) of a system, often a relational database, and its requirements in a top-down fashion. These models are being used in the first stage of information system design during the requirements analysis to describe information needs or the type of information that is to be stored in a database. The data modeling technique can be used to describe any ontology (i.e. an overview and classifications of used terms and their relationships) for a certain universe of discourse i.e. area of interest.

Generic data model

Generic data models are generalizations of conventional data models (Figure 6.9). They define standardized general relation types, together with the kinds of things that may be related by such a relation type. The definition of generic data model is similar to the definition of a natural language. For example, a generic data model may define relation types such as a 'classification relation', being a binary relation between an individual thing and a kind of thing (a class) and a 'part-whole relation', being a binary relation between two things, one with the role of part, the other with the role of whole, regardless the kind of things that are related.

Given an extensible list of classes, this allows the classification of any individual thing and to specify part-whole relations for any individual object. By standardization of an extensible list of relation types, a generic data model enables the expression of an unlimited number of kinds of facts and will approach the capabilities of natural languages. Conventional data models, on the other hand, have a fixed and limited domain scope, because the instantiation (usage) of such a model only allows expressions of kinds of facts that are predefined in the model.

Semantic data modeling

The need to define data from a conceptual view has led to the development of semantic data modeling techniques. That is, techniques to define the meaning of data within the context of its interrelationships with other data. As illustrated in the figure the real world, in terms of resources, ideas, events, etc., are symbolically defined within physical data stores. A semantic data model is an abstraction which defines how the stored symbols relate to the real world. Thus, the model must be a true representation of the real world.

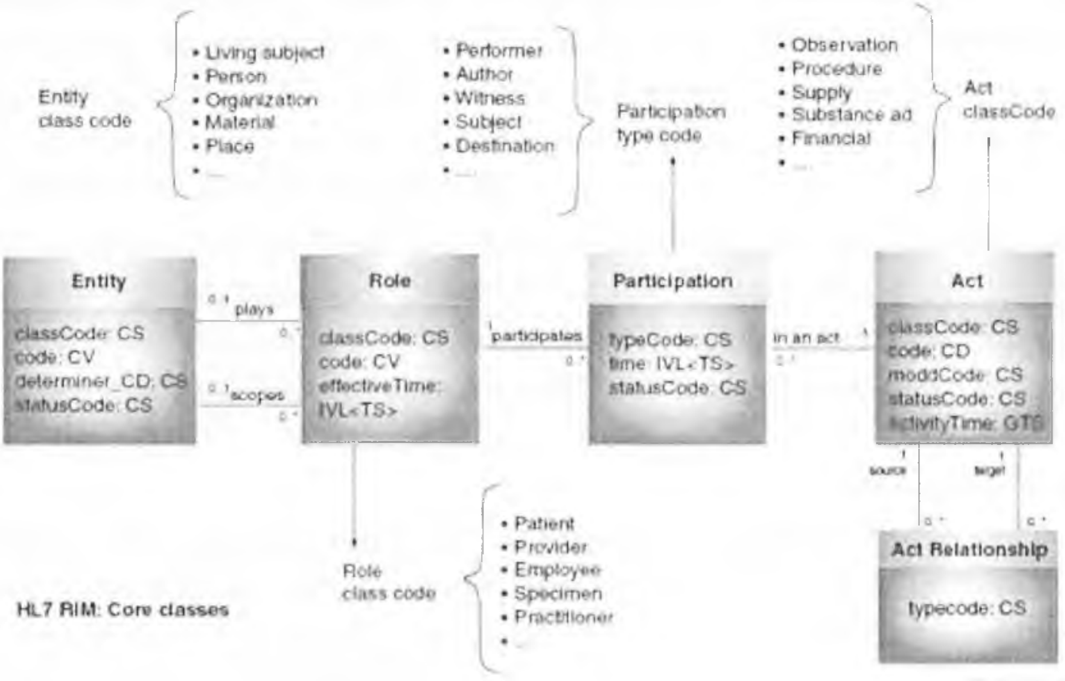


Figure 6.9.-Generic Data Model

Legend: CD – Concept Descriptor; CS – Coded Simple Value; GTS – General Timing Specification; HL7 – Health Level 7; IVL – Interval; RIM – Reference Information Model; TS – Point in Time.

A semantic data model can be used to serve many purposes, such as: planning of data resources; building of shareable databases; evaluation of vendor software, integration of existing databases.

The overall goal of semantic data models is to capture more meaning of data by integrating relational concepts with more powerful abstraction concepts known from the Artificial Intelligence field. The idea is to provide high level modeling primitives as integral part of a data model in order to facilitate the representation of real world situations.

Chapter 6 is prepared on the basis of information sources [35-41].

7. COMPUTER BASED TRAINING

Introduction

The Computer Based Training (CBT) is a pedagogical software, which allows a person to study at his own pace and according to his own schedule.

CBT is just one of the many terms to express just this. CBT is also referred to as:

CAI (computer assisted instruction),
CAL (computer aided learning),
CBE (computer based education),
CBI (computer based instruction),
CBL (computer based learning) etc.

The beginnings of CBT are mostly traced back to a big project called *Programmed Logic for Automated Teaching Operations* (PLATO) which was developed at the University of Illinois and marketed by the Control Data Corporation (CDC). PLATO was begun in 1959. The early PLATO lessons ran on a large mainframe host. Later CDC pulled back from their PLATO effort and sold the name to The Roach Organization (TRO).

Computer based training is intended to be *self-directed* and *self-paced* by the student. In its most simple form, CBT could be based on software that does little more than displaying a text. In such a simple CBT device, the sole student interaction would be to 'turn the page'.

Software for CBT purposes is called *courseware*. Preferably, CBT is based on *interactive courseware* (ICW). The interactive courseware allows an exchange between the student and the computer program. The interaction may begin with a mouse click, a key press, a screen touch, or any other input by the student. The program may react visually, aurally, or in any other way that can be perceived by the student.

If appropriate to the training contents, the CBT courseware may include *multimedia elements*. Multimedia is the ability to simultaneously present on a single workstation two or more dissimilar media types such as:

- text,
- still images,
- computer drawn graphics,
- audio,
- computer animation,
- motion video.

Often CBT is based on *hypertext*. Hypertext uses "a method of information delivery that provides multiple connected pathways through a body of information, allowing the user to jump easily from one topic to related or supplementary material".

A hypertext-type document that contains text and other multimedia elements as well as triggers, or pre-defined areas on the screen that can link to sections of the file that contain information in different media is call *hypermedia*.

Producing CBT courseware is a very time consuming and hence costly job. What are the costs for the development of a training course that will take an average student 1 hour to work through? Required development time is quoted to be roughly between 90 hours and 260 hours, costs are estimated to be between 25000 US$ and 50000 US$. Development time and costs will increase if more interactive features and more multimedia elements get included into the courseware. For instance, the *Lilienthal Project* (Multimedia Off- and Online Distance Learning for European Pilot Training) is an educational package with about 700 pages based on the *Hyper-Text Markup Language* (HTML). 100 student hours of its CBT course required an investment of 2.5 million euros; this is roughly 25000 US$ per course hour. The development of one course hour in the PLATO project required an investment of up to 20000 US$. Although these are only rough figures that deviate somewhat, they give an indication of the immense costs involved in CBT development.

These development costs may be justified if they can be distributed over a large number of students as in the case of the abovementioned Europe-wide pilot training. Yet, the number of students in aircraft design education is small compared to the number of students in other fields of the aviation training business. For this reason, it does not come as a surprise that computer based training has not made much impact in aircraft design education today.

7.1. CBT in aviation

Traditionally, most effort has been spent on computer applications for pilot training. Over the past decade CBT has seen a widespread adoption within the pilot training sphere. A *training pyramid* evolved with CBT forming the base. The *flight training devices* (FTD) - also known as fixed based simulators (FBS) - are the middle segment including part task trainers (PTT), cockpit system trainers (CST), cockpit systems simulators (CSS), and instrument flight trainers (IFT). The apex is formed by the *airplane simulator* also known as *full flight simulator* (FFS).

An "*airplane simulator* is a full-size replica of a specific type or make, model, and series airplane cockpit, including the assemblage of equipment and computer programs necessary to represent the airplane in ground and flight operations, a visual system providing an out-of-the-cockpit view, and a force cuing system ...".

An "airplane *flight training device* is a full scale replica of an airplane's instruments, equipment, panels, and controls including computer software programs necessary to represent the airplane in ground and flight conditions does not require a force (motion) cueing or visual system" .

Another branch are the *PC aviation training devices* (PCATD). They come from a background of commercial game software and the general growth in PC processing power.

The FFS achieves the ultimate when it comes to interactivity and multimedia elements.

Nevertheless, it is only a sparring partner for an experienced pilot. The FFSs, FTDs, and PCATDs lack to deliver the lecture and the necessary "instructional experience" to the student as demanded by the CBT definition given above. For that reason, these systems cannot be considered as CBT devices.

In a practical sense, CBT devices are associated with low-cost desktop computers.

These CBT devices have evolved through the "use as flight management systems trainers (FMST) whereby the addition of three dimensional mode control panels (MCP) and control display units (CDU) the desktop CBT system has become the FMST PTT." The next step is "to begin surrounding these basic CBT kernels in 3D cockpit hardware including throttle quadrants, partially working instrumentation, primary flight displays (PDF) and navigational displays (ND), seats and cockpit surrounds". If these CBT devices with added hardware retain their capability of delivering a lecture to the student, their interactivity and closeness to reality is a definite advantage.

7.2. Aviation Maintenance Computer Based Training

CBT is used for various levels of aviation maintenance training. CBT is used to teach:

1. aircraft maintenance fundamentals,
2. generic aircraft systems concepts,
3. aircraft type-specific maintenance,
4. troubleshooting by means of simulation-type computer based training.

In addition, troubleshooting can be taught using maintenance training simulators (MTS), flight training devices (FTD), full flight simulators (FFS), or (of course) the real aircraft.

Some examples of CBT products are given below.

TRO Learning (UK) produced the *jet aircraft maintenance fundamentals (JAMF) course* for *Lufthansa Technical Training* (LTT) investing 20 man-years on course development.

The 150 hours of courseware consists of high resolution graphics, digital audio, scanned images and digital video. Examples are taken from Boeing and Airbus aircraft.

VEGA in cooperation with *The Dutch Aviation Training College* (NLC) developed the PC-based *aircraft systems concepts (ASC) trainer* for KLM. Simulations and multimedia content can be used as an aid to instructors in teaching the basic concepts of operation and behavior of the major systems of large passenger jet aircraft. The courseware can be used by students in a self-paced mode to assist in their understanding of system behavior.

When performing troubleshooting, student actions are logged, allowing safety critical actions to be flagged to the instructor. The system is hosted on commercially available Pentium PCs running Windows 95 or NT.

Both Airbus and Boeing have developed in-house *aircraft type-specific maintenance CBT* courses for their respective aircraft. The *Boeing* 737-600/700/800 CBT is a course consisting of both instructor-lead CBT (ILCBT) lessons and student-paced CBT (SPCBT) lessons. The instructor-lead lessons include tutorial information with photographs, animations, and interactive diagrams and schematics. These are designed to support a classroom instructor's presentation, but may also be used by individual students when the text box option is turned on. The SPCBT lessons focus on the performance of exercises and procedures by individuals.

The *Airbus* video and computer based instruction (VACBI) system is intended for SPCBT lesson with diagrams, schematics, aural explanations, text boxes, and questions. A separate optical disk and a separate monitor show photos and video sequences. When the aural explanations are switched off, the VACBI courseware can also be used for ILCBT lessons.

The *B737 maintenance virtual workplace* from *Wicat Systems, Inc* simulates the mechanic's working environment, allowing real-time access to all of the resources, tools, and parts that are normally available to troubleshoot and maintain the aircraft. The virtual workplace has a "game-like" interface including 'troubleshooting tips' that contain data about components which have a high no-faultfound (NFF) removal rate. Each covered system also contains a CBT tutorial on system components, normal operation, non-normal conditions, built-in test equipment (BITE), and system-specific troubleshooting procedures. The Boeing 737 maintenance virtual workplace can be used as stand-alone instruction, delivered over a network, or may be used as instructor-led CBT. The troubleshooting simulations and fault-free practice allow the learner to engage in self-paced discovery learning through experimentation within the simulated environment. *Wicat* offers a similar cockpit maintenance operations simulation (CMOS) for the A320.

Computer based training is also extensively used in other branches of the aviation training business:

- cabin crew training,
- x-ray interpretation training;
- general topic training (safety, emergency, security).

7.3. CBT Developers

The big players in professional aviation CBT development are:

VEGA Group PLC. VEGA has bought TRO's Commercial Aviation Division and continues TRO's products alongside its own.

It claims "VEGA has the largest library of computer-based training products for the aviation training world."

Wicat Systems, Inc. It claims that "more than half of the world's largest airlines use proven Wicat CBT products"

Airbus Industrie and *Flight Safety Boeing Training International* develop to a large extend their own CBT courseware.

Airlines like Lufthansa do some CBT development also in-house. Both manufacturers and airlines cooperate with specialized CBT developers.

Many aircraft design computer programs have been written with the primary intention to support research in the field. More appropriate to support aircraft design education are smaller packages like e.g.

Advanced Aircraft Analysis (AAA) or *RDS Aircraft Design Software* written specifically to address student's needs.

The Advanced Aircraft Analysis (AAA) program displays a menu tree showing the design path selected by the user. After supplying input data in one window, several output parameters can be calculated and are displayed in another window. In the input/output windows, the user has access to information about the theory used. Context sensitive help is provided which helps the user in working with the program.

A closer look however reveals that these programs are quite good in aircraft analysis but lack support in the synthesis phase. In addition, they make no attempt in teaching aircraft design. On the contrary, they already assume a sound knowledge of aircraft design fundamentals and about strategies in aircraft design synthesis. For this reason, these valuable programs cannot be considered as a substitute for CBT in aircraft design education.

7.4. PC based flight simulation in aircraft design education

PCs have become extremely fast making it viable to perform realistic flight training. Pilot interface devices that are compatible with PCs (artificial stick, yoke and pedals) have been developed to supply the growing market of flight schools, universities and research organizations, who have a need for realistic, but inexpensive flight simulation. Low cost personal computer aviation training devices (PCATD) have sparked a lot of interest recently. The *Federal Aviation Administration* (FAA) has commissioned a study at the University of Illinois into the effectiveness of PCATDs, with favorable results.

A study at the University of Limerick points out two PCATDs that allow the input of individually designed aircrafts:

· X-Plane with its feature 'planemaker' can easily be handled but today still lacks many necessary details in modeling aircraft dynamics.

· The Aviator Visual Design Simulator (AVDS) utilizes a full nonlinear six degree of freedom model. However, all input files have to be written in C++ and require good programming skills. To make the package user-friendlier, a front-end module is needed.

7.5. Distance Learning and Web Based Training

Distance learning is defined as "The use of computers and communications technology to allow the interaction of teachers and students who are not co-located". Distance learning may be based on one-way or two-way audio/video teleconferencing, video broadcast, or correspondence courses. Distance learning may be delivered in real time, or it may be delayed and is called to be asynchronous.

Web based training (WBT) is one specific approach to distance learning. WBT is computer based training (CBT) that makes use of technologies and methodologies of the World Wide Web (WWW), the Internet, or intranets. Web based learning (WBL), web based instruction (WBI) and Internet based instruction (IBI) are just other terms used for WBT.

Chat, discussion groups, and e-mail can be a part of WBT. When appropriately used, these technologies can enhance learning and can improve student's performance. Discussion groups (private forums or Usenet newsgroups) are similar to text chat, only asynchronous.

When one poses a question to the group, or answers another, all members of the group benefit, though not instantaneously. It's the group dynamic of these information sharing media that adds perspective perhaps not available in one-on-one instruction. When email lists are used in sending private group messages, the learning effect is the same.

It has to be acknowledged that today the traditional sequence of "school ® university ® work" applies to ever fewer people. It becomes more common to study, after some time of work experience has been gathered. Also, the necessity may arises to work and study at the same time which calls for part time studies.

Quite often, students are bound to a certain place (due to a work or personal situation), which may not be at the location of the university. All these facts call for more flexible forms of teaching. Web based training (WBT) can be one step in this direction and should therefore be considered from the beginning in innovative course design.

7.6. Computer Based Training Tools

First of all it must be stressed that tool selection is only of secondary importance to CBT development. Establishing a formal development process is very important: listen, plan, design, test, build, deliver, observe, and refine. CBT development, in the first place, requires a good instructor.

Having said this, much depends on the selected tool, though. A discussion of the pros and cons of CBT tools can be a subject in its own right. The intention here is simply to provide a first insight for all those who have had very little or no exposure to CBT development.

If course notes are already available on the computer, they could be used in their original format. "CBT could be based on software which does little more than displaying a text". Such simple "CBT", however, would rarely fulfill today's requirements. Possibilities for user interaction and multimedia elements can already be added to office documents produced e.g. in *Microsoft Word* or *Microsoft PowerPoint*. If delivered over the Internet, such a document format would assume that users work on a PC and possess Microsoft software. In addition, problems can arise with different versions of the software. The latest Microsoft versions allow automatic generation of HTML files. The results are not always as expected, especially if highly interactive documents get converted that make extensive use of multimedia elements.

Adobe Acrobat converts any document into Adobe's Portable Document Format (PDF). The layout of the original document is maintained and the PDF file can be read on almost any platform with the free *Acrobat Reader* that is available over the Internet.

Multimedia elements and links can be added to PDF files. However, PDF offers no possibilities to add platform independent computing capabilities.

If computing capabilities are required e.g. for preliminary sizing, conceptual aircraft design, or simple simulation-type CBT, spreadsheets like *Microsoft Excel* allow to crank out impressive results very efficiently.

Unfortunately, the same restrictions and drawbacks apply as stated above for *Microsoft Word* and *Microsoft PowerPoint*. These restrictions may be acceptable if the software is intended to be used by a limited group of people.

Hypermedia CBT can be produced professionally with generic or specialized editors directly for the Internet, intranets or for a CD ROM using the Hyper-Text Markup Language (HTML). Interactivity and limited computing power can be added with *JavaScript* - a simple programming language. JavaScript code gets included directly into the HTML file.

Java applets provide the possibility to include and offer also complex calculations with the courseware. The universal platform-independent approach of HTML-based courseware has a few drawbacks: The detailed layout of the pages depends on the type of browser on the user's machine. Different versions of JavaScript exist; so courseware designed for one browser may not run properly with another browser.

Programming in JavaScript is much more time consuming than spreadsheet programming.

Professional CBT courseware is often produced with authoring systems like *Macromedia Director*. *Macromedia Director* is the leading authoring system. Its integrated language *Lingo* allows the specification of complex user interactions. Lingo is an object oriented programming language with features comparable to other programming languages.

As such, it should be possible to apply Lingo also for the programming of aircraft design specific problems. The courseware runs on PCs with Windows as well as on Apple computers and is primarily intended to be distributed on a CD ROM. Moreover, conversion of the courseware is possible for use on the Internet.

This requires the shockwave-plug-in to be present on the user's computer. The free plug-in is available over the Internet. The development of courseware with *Macromedia Director* is quite complex. Considerable time is required to get acquainted with this powerful tool.

The evaluation of software tools can often become a matter of faith in one system over another. The best thing to do is to get started and to judge for oneself.

The domain of aircraft maintenance is rapidly becoming the focus of computer-based training (CBT) aids. With the use of desktop computers with multimedia packages, new maintenance job aids have been developed to teach technical skills to maintenance technicians. AMTs may learn a variety of skills from CBT that range from scheduling preventive maintenance to applying expert systems for fault diagnosis and repair.

In a typical maintenance environment, the technician is challenged to work autonomously but still be a team member, communicating and coordinating the activities with other technicians and inspectors. Though the advantages of teamwork are widely recognized in the airline industry, the work culture assigns responsibility for faulty work on individual AMTs rather than on the teams in which they work. The reasons for this could be the individual licensing process and personal liability, both of which often results in AMTs and their supervisors being less willing to share their knowledge and work across shifts with less experienced or less skilled colleagues. The problem is further compounded since the more experienced inspectors and mechanics are retiring and being replaced by a much younger and less experienced workforce.

Furthermore, the AMTs are not trained to work as team members.

Research on civil aircraft inspection and maintenance has shown the importance of teamwork in the aircraft maintenance environment. In addition, training has been identified as one of the primary intervention strategies in improving team performance. With improvements in technology-based training devices, it may be possible to provide aircraft maintenance technicians with training tools to help enhance their team skills and improve performance with the aircraft maintenance environment. This research examines the effectiveness and applicability of computer-based multimedia team training for aircraft maintenance technicians. Using a computer-based multimedia training tools the Aircraft Maintenance Team Training (AMTT) software is controlled study was conducted to:

- evaluate the use of computers in acquiring knowledge on team skills,
- determine the ease of use of a computer-based multimedia tool,
- determine if computer-based training was as effective as traditional instructor-based training.

With computer-based technology becoming cheaper, the future will bring an increased application of advanced technology to training. Over the past decade, instructional technologists have provided numerous technology-based training devices with the promise of improved efficiency and effectiveness. Examples of such technology include computer simulation, interactive video discs and other derivatives of computer based application. Several of these have been employed for diagnostic maintenance training. Furthermore, multimedia has assisted in teaching difficult and complex skills.

Because of the advantages offered, computer-based training may have a role to play in team training in the aircraft maintenance environment. It is important, therefore, to examine the effectiveness and applicability of computer-based multimedia team training for aircraft maintenance technicians.

To facilitate the study, a computer-based multimedia training tool for the Aircraft Maintenance Team Training (AMTT) software was developed. AMTT provided the AMTs instruction on team skills which are necessary to function cooperatively and effectively in the aircraft maintenance environment.

Specifically designed for training aircraft maintenance technicians in basic team skills, AMTT uses a multimedia presentational approach with interaction opportunities between the user and the computer. The multimedia presentation includes: full motion video which provide real life examples of proper and improper team behavior, photographs and animation that illustrate difficult concepts and voice recordings coupled with visual presentations of the main contextual material. Since the software was developed as a training and research tool, the software facilitates the collection of pre-training and post-training performance data.

AMTT is divided into four major programs (see Fig. 7.1): (1) Team Skills Instructional program, (2) Instructor's program, (3) Printing program and (4) the Supplemental program.

While the Team Skills Program and the Supplemental Program have been designed for use by the aircraft maintenance technician undergoing team training, the remaining two programs are for use by the instructor/supervisor. An aircraft maintenance technician (AMT), interacting with the AMTT software, first uses the Team Skills Instructional program which initially provides an introduction to the software. Following this step, the AMT is provided with instruction on basic team skills through four separate team skill submodules: communication, leadership, decision making and interpersonal relationships. These submodules not only emphasize and cover generic material related to these skills but also relate the importance and use of the specific skills within the aircraft maintenance environment.

The new approach to the team training opportunities are so-called Virtual Operating Room (VOR).

The Virtual Operating Room (VOR) is an operating room training environment capable of supporting team training. One of the challenges of creating sucha team training environment is assembling a suitable team of participants for a training session. In the VOR, simulated agents may be

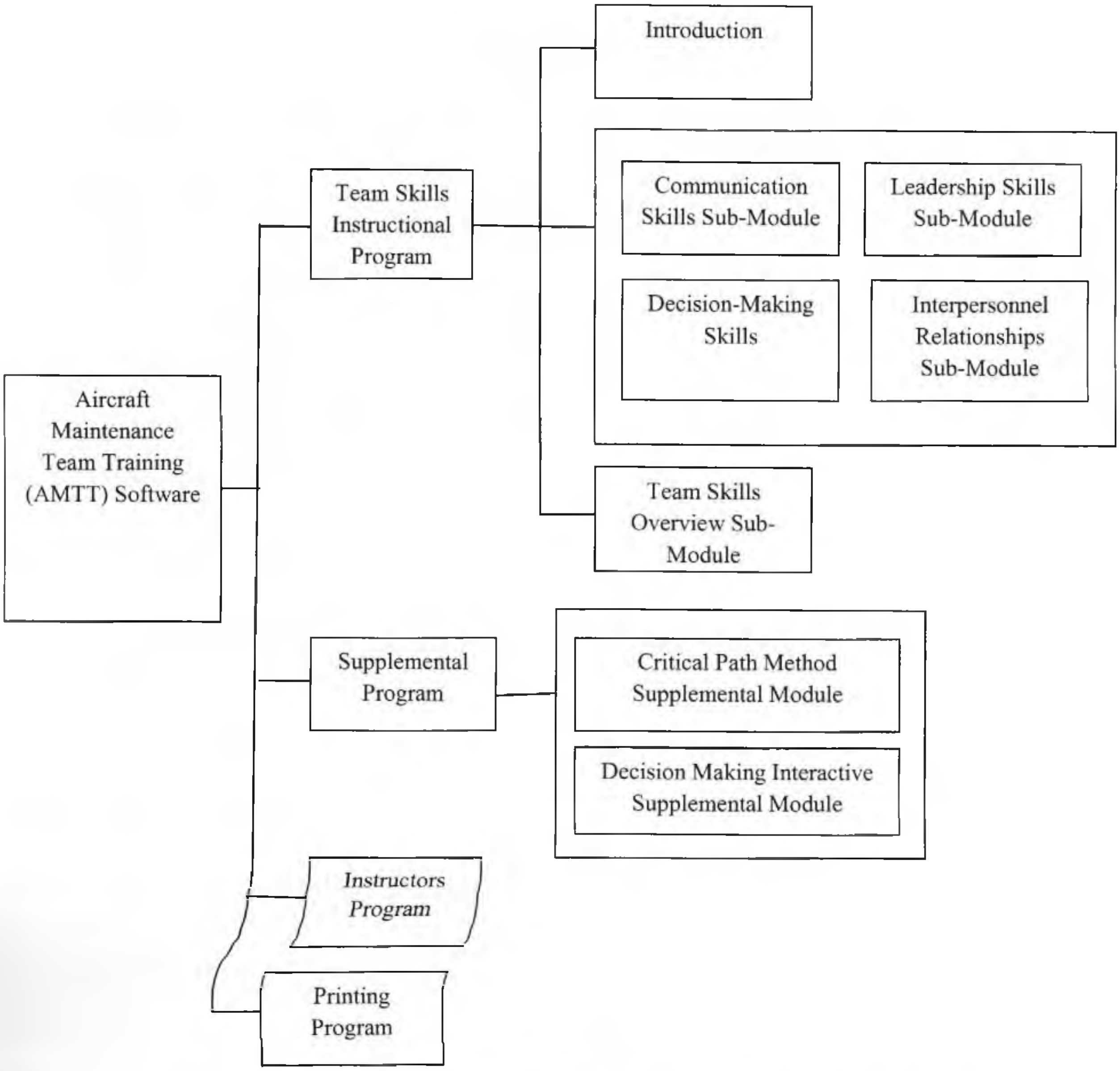

Introduction
Team Skills Instructional Program
Communication Skills Sub-Module
Leadership Skills Sub-Module
Decision-Making Skills
Interpersonnel Relationships Sub-Module
Aircraft Maintenance Team Training (AMTT) Software
Team Skills Overview Sub-Module
Supplemental Program
Critical Path Method Supplemental Module
Decision Making Interactive Supplemental Module
Instructors Program
Printing Program

Figure 7.1.- Layout of Aircraft Maintenance Team Training (AMTT) Software

substituted for actual participants. This provides the ability to conduct training sessions with a few or even one participant. The VOR has computer-based agent architecture. The agent automaton in VOR interacts with others agents present in the environment through specific inputs and outputs such as voice recognition, environment sensors, and equipment simulators. In the VOR simulation architecture, the primary components in the system such as voice recognition and rendering are decoupled and therefore allow for modular development and abstraction. The potential interactions between the simulated agents and the trainee(s) are based on an underlying scenario script. The script provides the potential sequence of steps, interchanges, branching (deviations) that may occur in the training.

Chapter 7 is prepared on the basis of information sources [42-54].

REFERENCES

1. Life Cycle Management for Design and Maintenance.- www.logtech.unc.edu
2. Life Cycle Assessment of the Airbus A330-200 Aircraft J Vasco O. F. L..- https://dspace.ist.utl.pt/bitstream/2295/792342/1/Tese_JoaoVascoLopes.pdf.
3. Airbus 380 Life Cycle Cost Concepts and Design Economics .- http://esminfo.prenhall.com/engineering/sullivan/Chapter2_crop.pdf
4. The Aircraft at End of Life Sector: a Preliminary Study.- http://users.ox.ac.uk/~pgrant/Airplane%20end%20of%20life.pdf
5. Department of defense. Handbook acquisition logistics.- http://www.barringer1.com/mil_files/MIL-HDBK-502.pdf.
6. Logistic and acquisition.- http://acta.fih.upt.ro/pdf/2012-3/ACTA-2012-3-24.pdf.
7. Aircraft Conceptual Design / Sadraey M.- Daniel Webster College.- http://faculty.dwc.edu/sadraey/Chapter%203.%20Aircraft%20Conceptual%20Design.pdf
8. Feature -Based Hierarchical Knowledge Engineering for Aircraft Life Cycle Design Decision Support / Dissertation / Wei Zhao .- http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.85.5916&rep=rep1&type=pdf
9. Product lifecycle management in aviation maintenance, repair and overhaul/S.G. Lee, Y.-S.Maa, G.L. Thimm, J. Verstraeten .- http://www.deepdyve.com/lp/elsevier/product-lifecycle-management-in-aviation-maintenance-repair-and-korDghdeOK
10. LCC-OPS: Life Cycle Cost Application in Aircraft Operations/ Dissertation / Suwondo E..- http://repository.tudelft.nl/view/ir/uuid%3Ad6b06571-051b-44d4-88ef-5a4e71251388/
11. Life Cycle Cost Analysis Handbook .- http://www.eed.state.ak.us/facilities/publications/lccahandbook1999.pdf
12. A Life Cycle Cost Summary / H. Paul Barringer, P.E./ Barringer & Associates, Inc., Humble, Texas .-USA .- http://www.barringer1.com/pdf/LifeCycleCostSummary.pdf.
13. Domain Modeling / Paul Oldfield .- http://www.aptprocess.com/whitepapers/DomainModelling.pdf
14. Ontology Development 101: A Guide to Creating Your First Ontology / Natalya F. Noy and Deborah L. McGuinnessl.- http://www.citeulike.org/user/wnpx/article/909628
15. Mind games: the ontology of aviation safety and its consequences Winfred J., Garst JR. .- http://scholar.lib.vt.edu/theses/available/etd-04192009-111720/unrestricted/dissertation.pdf

16. Microsoft excel 2007 tutorial .- http://www.bgsu.edu/downloads/cio/file15626.pdf
17. Advanced Excel Skills.- https://extranet.who.int/aim_elearning/en/excel//pdf/ExcelAdvanced.pdf
18. Microsoft Excel 2010 Tutorial .- http://www.goodwin.edu/computer_resources/pdfs/exfcel_2010_tutorial.pdf
19. Excel 2003 – Tutorial .- http://psych.utoronto.ca/users/reingold/courses/resources/handouts_excel/Excel2003/Excel2003.TutorialPart1.pdf
20. Introduction to Systems Engineering .- http://www.argospress.com/sample-chapters/fEaS_sample-chapter.pdf
21. The Internet Explained, Vincent Zegna & Mike Pepper, Sonet Digital, November 2005, Pages 1 - 7.- http://www.southbourne.com/articles/internet-explained
22. Internet .- http:/en.wikipedia.org/wiki/Internet
23. 2D Modeling in AutoCad.- http://edina.ac.uk/projects/mapscholar/casestudies/murison/2D-ex.pdf
24. AutoCad Basics .- http://cms.cerritos.edu/uploads/engt/autocad%20basics.pdf
25. AUTOCAD Exercise Workbook.- http://minos.materials.uoc.gr/~marigianna/Chapters.pdf
26. SketchUp Basics .- http://faculty.delhi.edu/hultendc/A250-Week1-SketchUp-S2008.pdf
27. Sketch Up Basic Exercises .- http://www.3dvinci.net/PDFs/SU7BasicsSampler.pdf
28. Tips on How to Make a PowerPoint Presentations .- http://www.healthsci.utas.edu.au/trsu/pdf/powerpoint.pdf
29. Microsoft Access Tutorial.- http://www.itu.dk/people/slauesen/UID/AccessTutorial.pdf
30. Example Application of Aviation Quality Database .- http://flightsafety.org/files/AQD_application.pdf
31. Basic Database Concepts .- http://media.wiley.com/product_data/excerpt/74/04712187/0471218774.pdf
32. Dtabase Basics .- http://www.mhprofessional.com/downloads/products/0072255404/0072255404_ch01.pdf
33. Database Fundamentals / Robert J Robbins / http://www.esp.org/db-fund.pdf
34. Data Modeling .- http://www.liberty.edu/media/1414/[6330]ERDDataModeling.pd
35. Data Modeling Techniques for Data Warehousing.- http://www.redbooks.ibm.com/redbooks/pdfs/sg242238.pdf
36. Introduction to Data Modeling / Mamchenko J.- http://gama.vtu.lt/biblioteka/Information_Resources/i_part_of_information_resources.pdf

37. Guide to Data Modeling .- http://faculty.washington.edu/burrows/IS300/dataModelingGuide.pdf
38. A Brief Review of Data Models .- http://www.neesgrid.org/documents/TR_2004_01.pdf
39. Generic Data Modeling / Dan Collins .- http://www.dama-nj.org/presentations/Kalido_Generic_Data_Modeling.pdf
40. Data Modeling Versus Ontology Engineering / Spyns P., Meersman R., Jarrar M. / http://lsdis.cs.uga.edu/SemNSF/SIGMOD-Record-Dec02/Meersman.pdf
41. Bsic Data Modeling Concepts / Gerhard Beck.- http://members.verizon.net/~gtbeck/data_modeling.pdf
42. Aviation Maintenance Incident Database .- http://www.hf.faa.gov/docs/508/docs/maint_MEIMS.pdf
43. Aviation CBT .- http://www.sphaera.co.uk/aviation.htm
44. Computer Based Training in Aircraft Design Education .- http://www.fzt.haw-hamburg.de/pers/Scholz/paper/ICA0173Paper.pdf
45. Effect of team training on aircraft maintenance technicians: computer-based training versus instructor-based training David C. Krausa, Anand K. Gramopadhye .- http://www.hf.faa.gov/hfmaint/Portals/1/science5.pdf
46. Team training: role of computers in the aircraft maintenance environment David Krausa,*, Anand K. Gramopadhye .- http://www.hf.faa.gov/hfmaint/Portals/1/science8.pdf
47. Virtual Operating Room Team Training via Computer–Based Agents / Baydogan E, Belfore L.A., Scerbo M., Mazumdar S. / international journal of intelligent control and systems.- VOL. 14, NO. 1, MARCH 2009, 115-122..- http://www.ezconf.net/IJICS/files/160/11-ijics-Baydogan-main.pdf
48. Cooper T. *Glossary of terms related to computer-based training*. Rev. 2.0, Aviation Industry CBT Committee, 1997. - Doc. No. CRS002, URL: http://www.aicc.org/pages/down-docsindex.
49. Brendel H. *Computer based training*. 1 st edition, IWT, 1990 : http://www.pilotschool.net
50. Adriaanse H, Broekhuizen P, Geerligs T, van Oorde K, Zaremba T. Problem-based learning. *Civil Aviation Training*, Vol. 9, No. 3, pp 40-43, 1998.
51. Doyle A. Maintenance training. *Flight International*, 3-9 March, pp 58-62, 1999 : http://www.wicat.com/Maintenance/b737mvwp.htm
52. Figiel A. Computer-based training is a cost effective solution to constraints on time, training budgets. *ICAO Journal*, Vol. 53, No. 10, 1998.
53. Young T. PC-based flight simulation in educating aeronautical engineers. *Third European Workshop on Aircraft Design Education*, Bristol, UK, 1998: http://www.X-plane.com
54. Reichow D. Application of computer-based training for improved maintenance training efficiency. In: Eighth Federal Aviation Administration Meeting on Human Factors Issues in Aircraft Maintenance and Inspection. Alexandria, VA: FAA, 1994.

INDEX