# IE360: CAD/CAM

## Computer Aided Design and Computer Aided Manufacturing

## Lecture (3)
## Geometric Modeling Systems

Dr Ashraf Afifi

Industrial Engineering Department

King Saud University

# Outline

- Introduction
- Wireframe Modeling Systems
- Surface Modeling Systems
- Solid Modeling Systems
- Modeling Functions
  - Primitive creation functions
  - Sweeping and skinning functions
  - Rounding (or blending) and lifting functions
  - Boundary modeling
  - Feature-based modeling
  - Parametric modeling
- Data Structures
  - CSG tree structure
  - B-Rep data structure

## Introduction:

- A model is an abstract representation of some thing (a machine, a building, a hurricane, etc.)

- Several types of models are available, including: geometric models (drawings), mathematical models, prototypes, etc.

- The designer needs models that represent his design for two main reasons:
  - Recording the design for reviewing, modifying, manipulating, etc.
  - Communication between designers, manufacturers, managers, etc.

- In engineering fields, geometric models (drawings) are the most appropriate representation method.

- Geometric modeling deals with the mathematical representation of curves, surfaces and solids necessary in the definition of complex physical or engineering objects.

- It attempts to provide a complete, flexible, and unambiguous (clear) representation of the object, so that the shape of the object can be:
  - Easily visualized (rendered),
  - Easily modified (manipulated),
  - Increased in complexity,
  - Converted to a model that can be analyzed computationally,
  - Manufactured and tested.

- Geometric modeling is the basic of many applications such as:
  - Mass property calculations,
  - Mechanism analysis,
  - Finite element modeling,
  - NC programming.

- Requirements of geometric modeling include:
  - Completeness of the part representation,
  - The modeling method should be easy to use by designers,
  - Rendering capabilities (which means how fast the entities can be accessed and displayed by the computer).

- The basic geometric modeling approaches available to designers on CAD/CAM systems are:
  - Wire-frame modeling,
  - Surface modeling,
  - Solid modeling.

## Wireframe Modeling:

➢ Wireframe modeling, developed in the early 1960's, is one of the earliest geometric modeling techniques.

➢ It uses points, lines, arcs and circles, conics and curves to represent 3D objects.

➢ An example of a surface model is given in Figure 1.

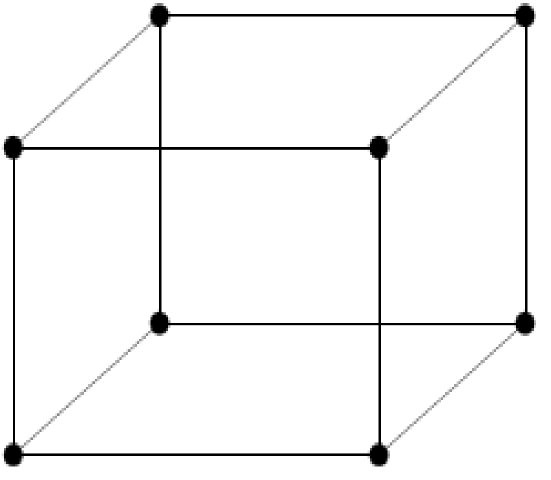
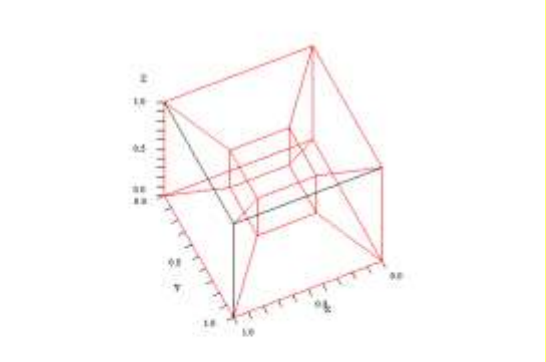


Figure 1: Wireframe model of a cube.

- Advantages of Wireframe Modeling
  - Easy to use,
  - The most economical of the 3D schemes in terms of computer time and memory requirements.



- Disadvantages of Wireframe Modeling
  - Ambiguity,
  - Complex models are difficult to interpret,

  - Limited ability to calculate mechanical properties or geometric intersections,
  - Cannot be used as a basis for manufacturing or analysis.

## Surface Modeling:

➢ Surface modeling techniques, developed in the late 1960's, is more sophisticated than wireframe modeling in that it defines not only the edges of a 3D object, but also its surfaces.

➢ In surface modeling, objects are defined by their bounding faces as shown in Figure 2.

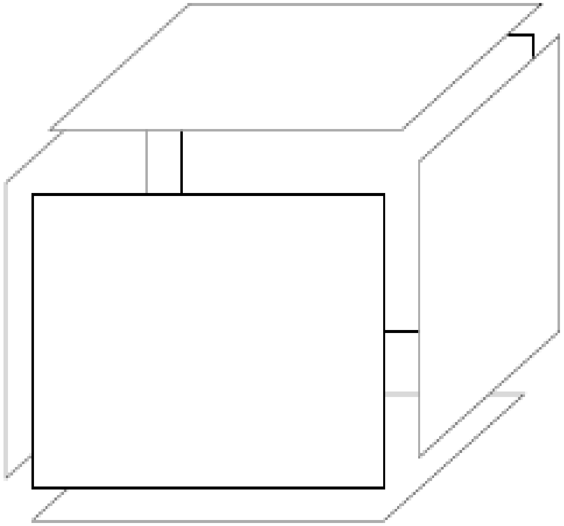


Figure 2: Surface model of a cube.

## Advantages of Surface Modeling

- Allows graphic display and numerical control machining of carefully constructed models.

## Disadvantages of Surface Modeling

- More computationally demanding than wireframe,
- Requires more skill in their construction,
- Surfaces may be discontinuous or may intersect with themselves or with each other.

## Solid Modeling:

- Solid modeling, first introduced in the early 1970's, explicitly or implicitly contains information about the closure and connectivity of the volumes of solid shapes.

- It give designers  a complete descriptions of constructs, shape, surface, volume, and density.

- An example of a solid model is given in Figure 3.

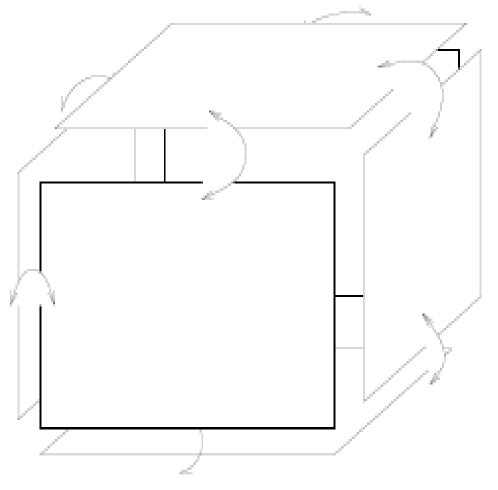
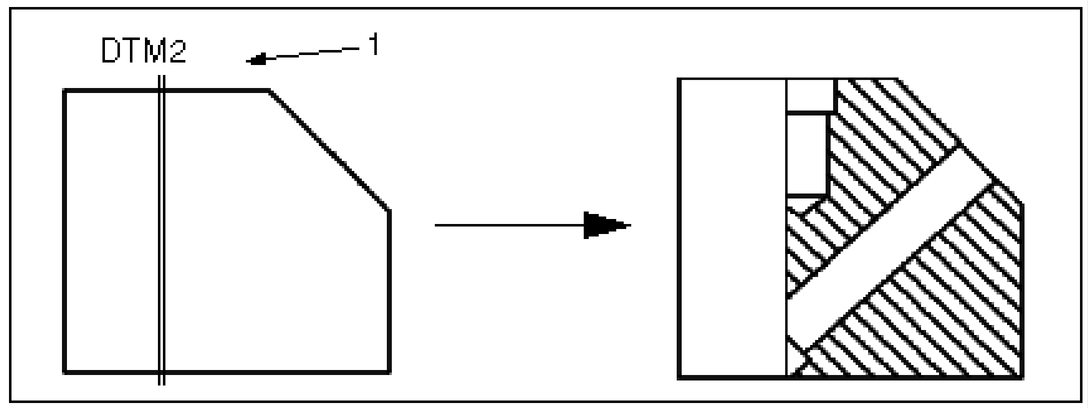


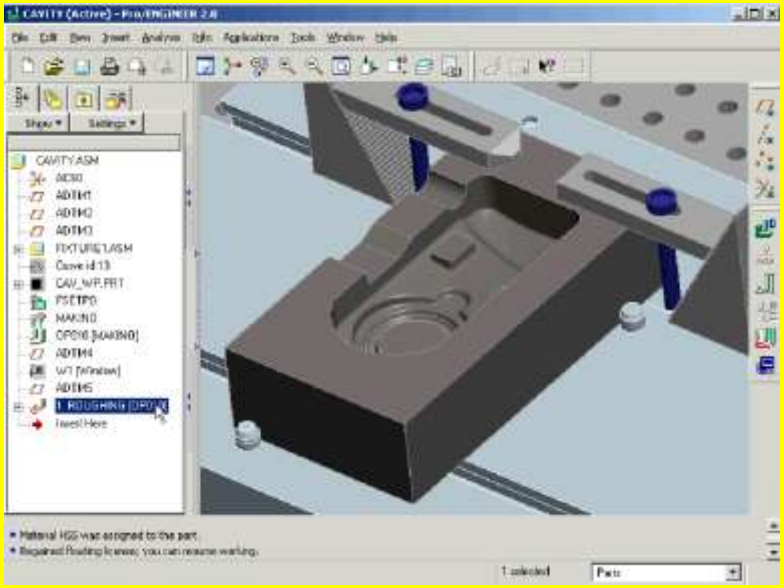
Figure 3: Solid model of a cube.

## ➢ Advantages of Solid Modeling

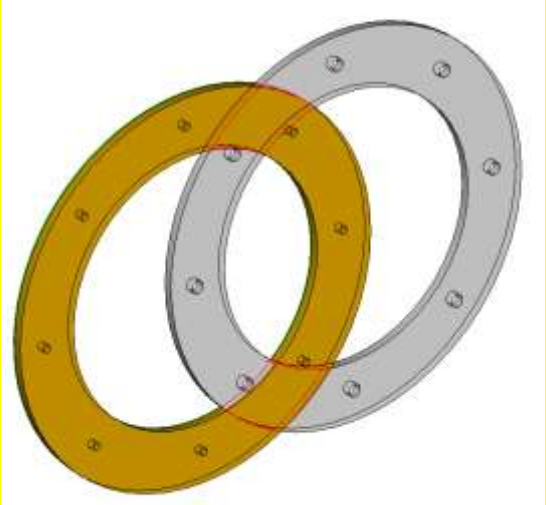
- Every surface boundary is always directly adjacent to one other surface boundary, guaranteeing a closed volume,
- Solid models, unlike surface models, enable a modeling system to distinguish the outside of a volume from the inside,
- It allows integral property analysis for the determination of volume, center of volume or gravity, moments of inertia, etc.,
- It enables accurate detailing,
  - Hidden lines
  - Section views
  - Auxiliary views



- It allows making tool paths directly from the model.




- It enables interference checking.



## Modeling Functions:

- The developers of solid modeling systems provide simple and natural modeling functions so that users can manipulate the shape of a solid as they do for a physical model without having to consider the details of mathematical description.

- The modeling functions supported by solid modeling systems can generally be classified as six groups:
  - Primitive creation functions
  - Sweeping and skinning functions
  - Rounding (or blending) and lifting functions
  - Boundary modeling
  - Feature-based modeling
  - Parametric modeling

- **Primitive creation functions:** These functions retrieve a solid of a simple shape from among the primitive solids stored in the program in advance and create a solid of the same shape but of the size specified by the user. Figure 4 illustrates the primitives supported by most solid modeling techniques.

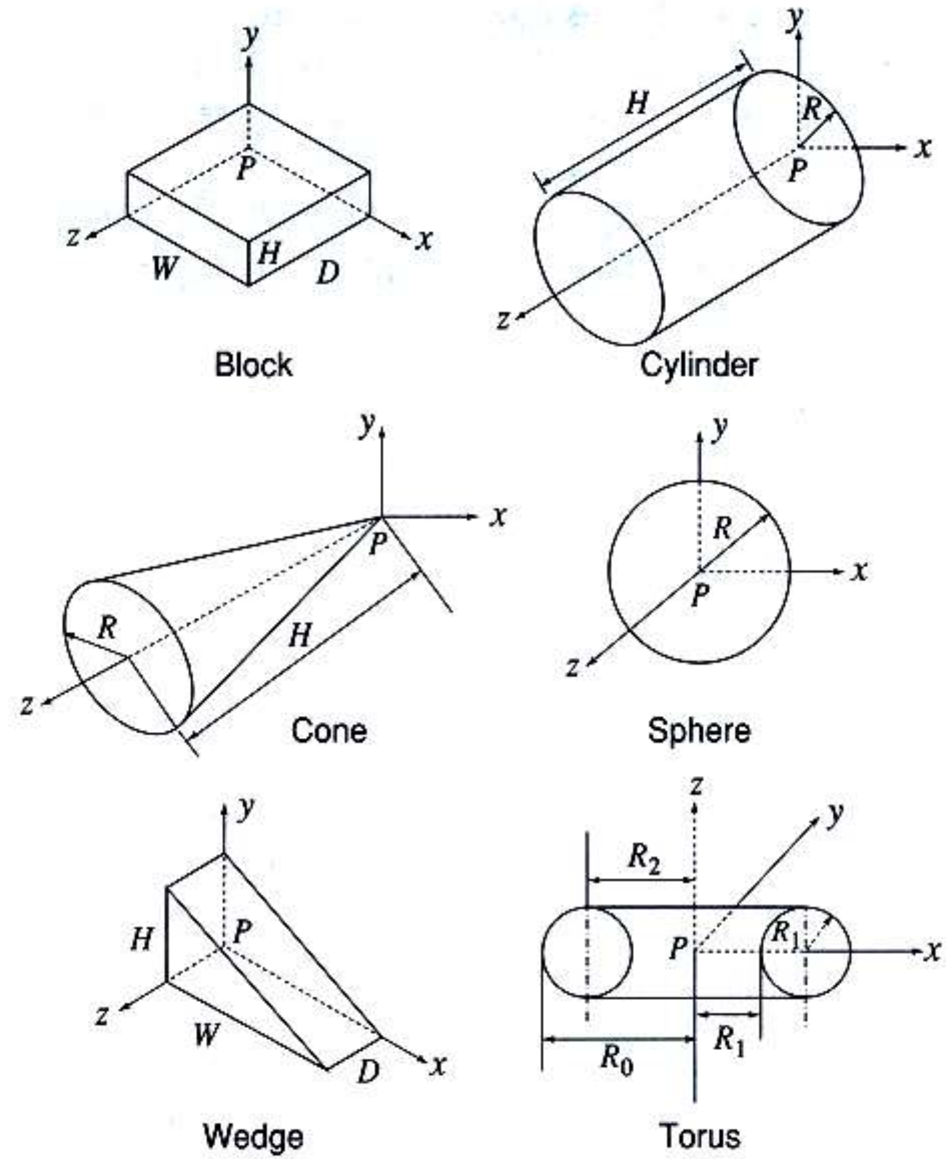



Figure 4: Primitives generally supported.

➢ *Boolean operations*: The Boolean operations of set theory can be applied to create numerous solid models by combining the primitives in Figure 4. The Boolean operations supported by most solid modeling systems are union, intersection, and difference, as illustrated in Figure 5.

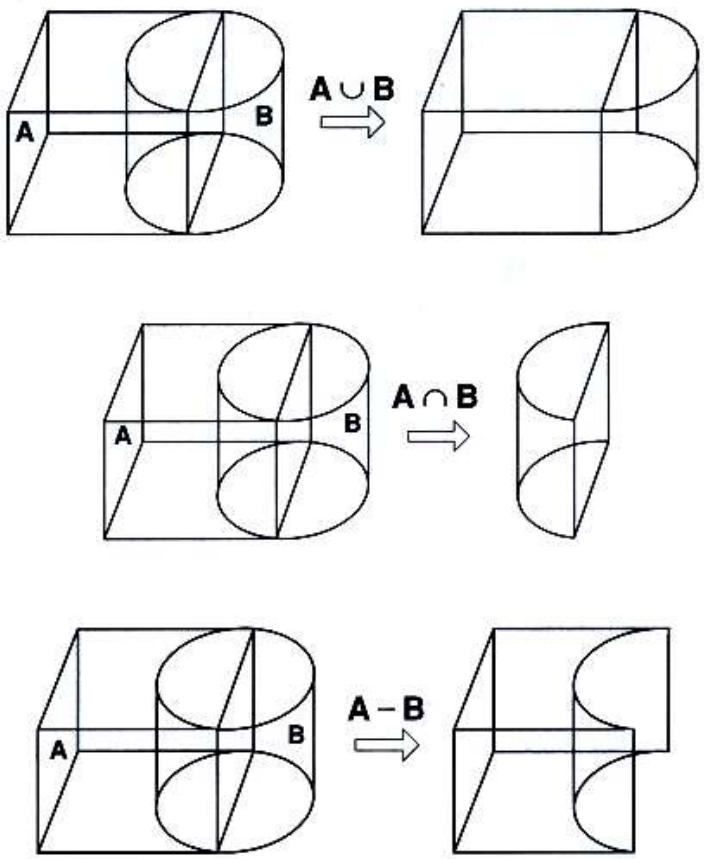



Figure 5: Examples of Boolean operations.

➢ **Sweeping and skinning functions:** *Sweeping* is a modeling function in which a planar closed domain is translated or revolved to form a solid.

➢ When the planar domain is translated, the modeling activity is called *translational sweeping*; when the planar region is revolved, it is called *swinging*, or *rotational sweeping*.

➢ Figure 6 illustrate the two types of sweeping operations, respectively.

➢ If the planar shape to be translated or revolved is not closed, the result after the sweeping is a surface instead of a solid.

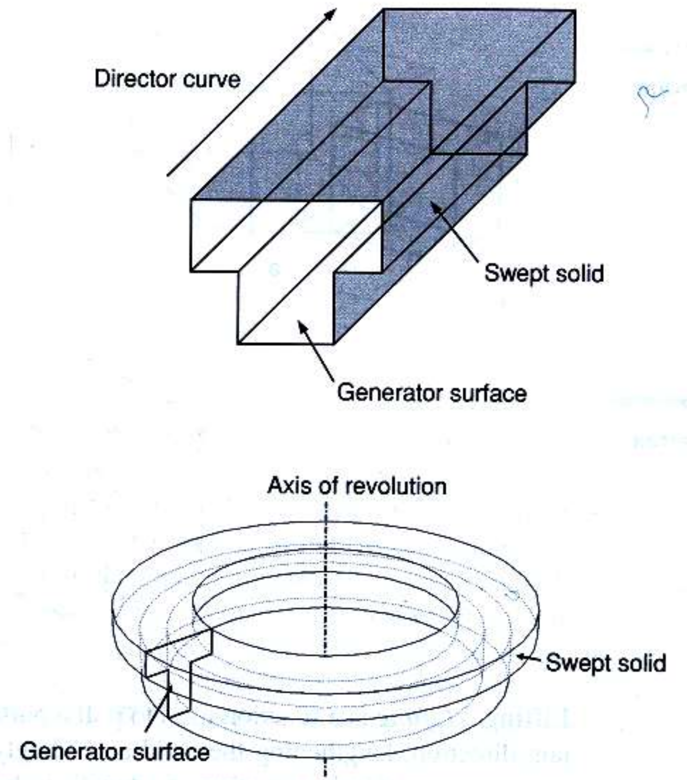



Figure 6: Examples of translational and rotational sweeping.

NB: In rotational sweeping, revolutions of less than 360 degrees are allowed in most solid modeling systems.

- *Skinning* is a modeling function used to form a closed volume or a solid by creating a skin surface over prespecified cross-sectional planar surfaces, as illustrated in Figure 7.

- If the two end faces corresponding to the two end cross sections are not added to the skin surface, the resulting model would be a surface instead of a solid.

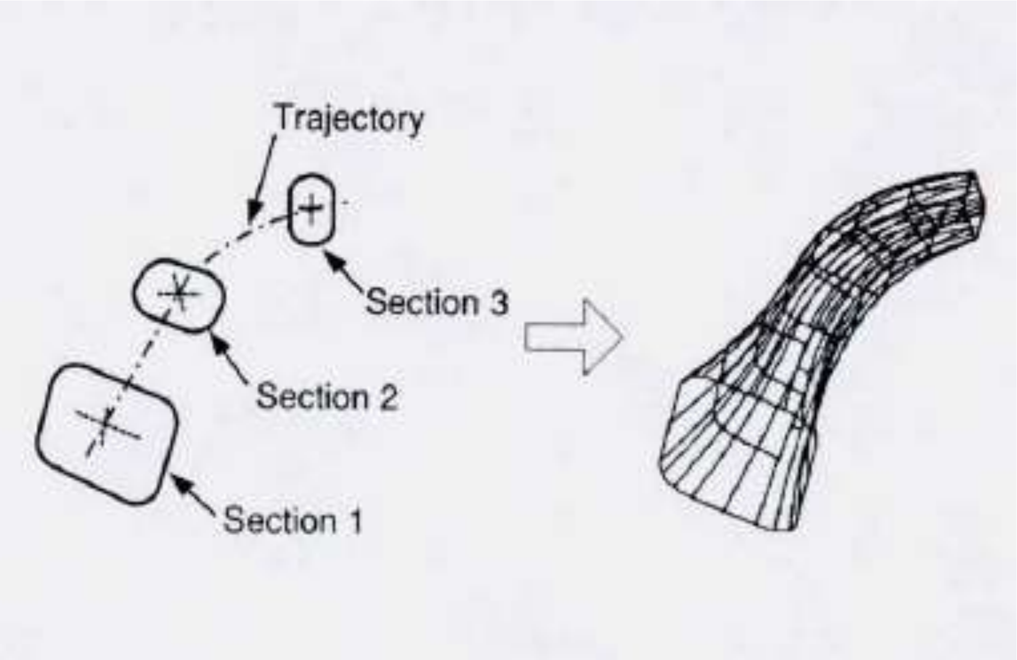



Figure 7: Example of creating a solid by skinning.

- **Rounding (or blending) and lifting functions**: *Rounding (or blending)* is a function used to modify an existing model so that a sharp edge or vertex is replaced with a smooth curved surface.

- Figure 8 (a) shows replacement of a sharp straight edge with a cylindrical surface.

- Filleting is a special case of rounding in which a rounding effect is obtained by addition, instead of elimination, of extra material, as illustrated in Figure 8(b).

- Figure 9 shows replacement of a sharp vertex with a spherical surface.

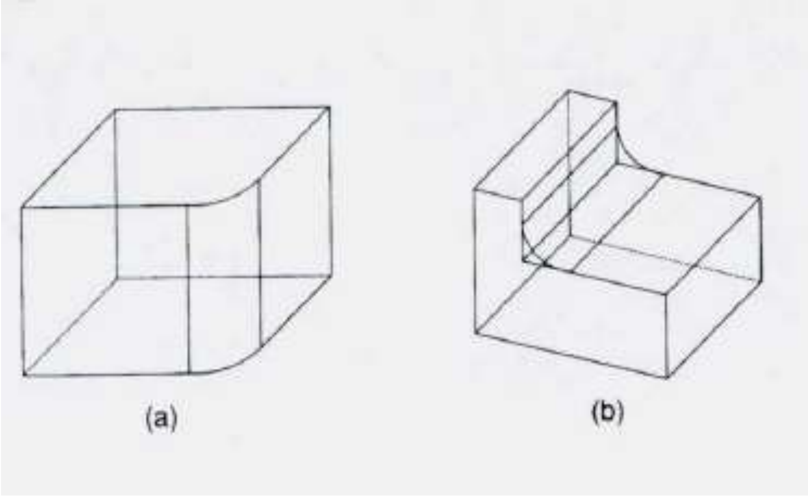



Figure 8: Examples of edge rounding.

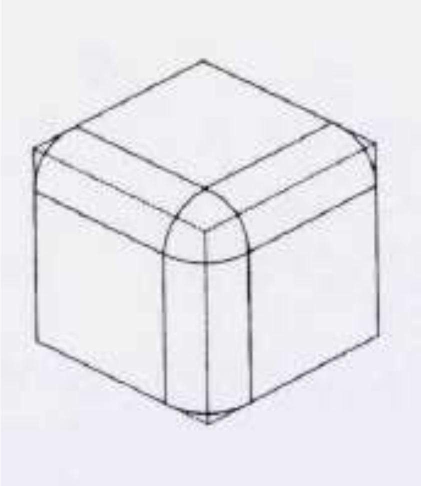


Figure 9: Example of vertex rounding.

- *Lifting* is a function used to pull a portion or entire face of a solid in a certain direction, lengthening the solid accordingly.

- Figure 10(a) illustrates the lifting operation. If only a portion of a face is to be lifted, as shown in Figure 10(b), the face should be split beforehand.

- When using the lifting function, you must specify properly the lifting direction or distance to avoid interference between the extended portion and the original solid, as illustrated in Figure 11.

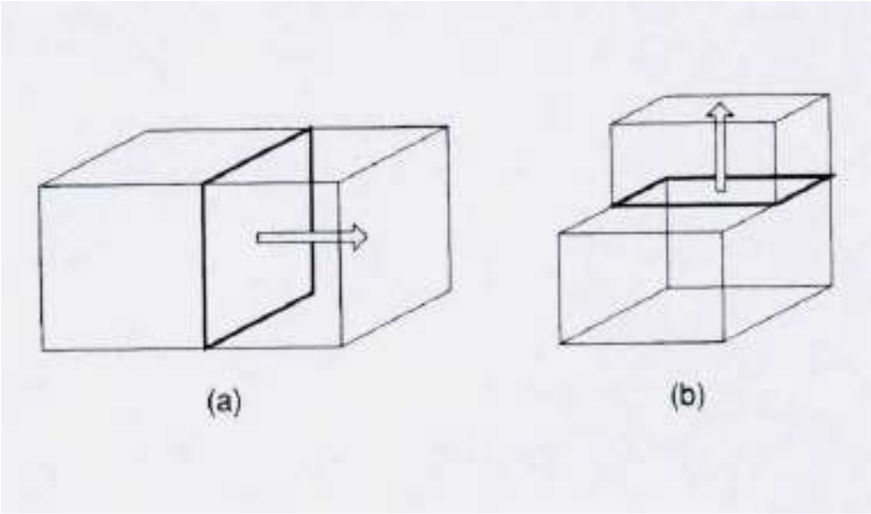



Figure 10: Examples of the lifting operation.

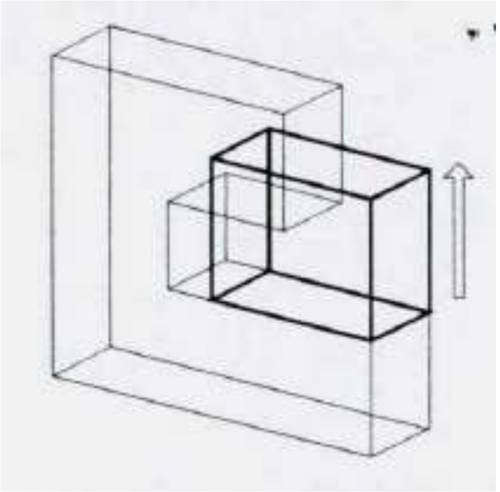


Figure 11: Self-interference caused by lifting.

- **Boundary modeling**: *Boundary modeling functions* are used to modify a shape of an existing solid model by adding, deleting, or modifying the lower entities of a solid, such as vertices, edges, and faces.

- A vertex can be moved to a new location accompanying modifications of its related edges and faces, as illustrated in Figure 12.

- A straight edge can be replaced by a curved edge, resulting in modification of the related faces and vertices, as illustrated in Figure 13.

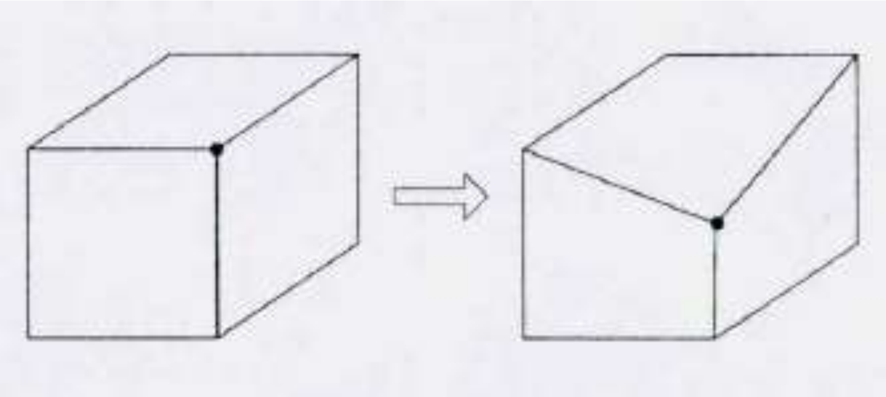


Figure 12: Modification by vertex moving.

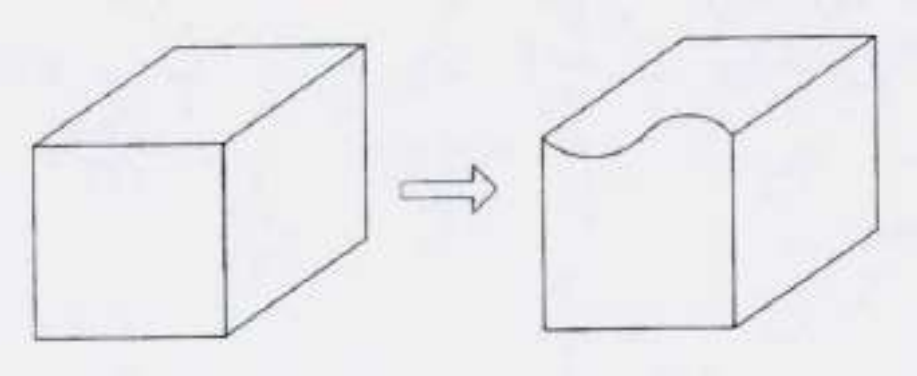


Figure 13: Modification by edge replacement.

- A planar surface can be replaced with a new curved surface accompanying modification of the related edges and vertices, as illustrated in Figure 14.

- These modeling functions are called *tweaking functions*.

- **Feature-based modeling**: *Feature-based modeling* enables the designer to model solids by using familiar shape units called *features*. Popular features supported by most feature-based modeling systems are manufacturing features such as chamfer, hole, fillet, slot, pocket, and so on.

- Figure 15 illustrates the use of various features in modeling.

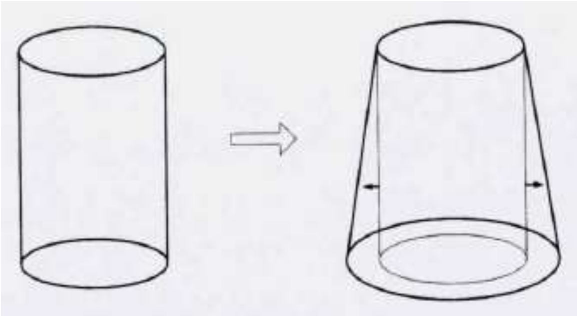


Figure 14: Modification by surface replacement.

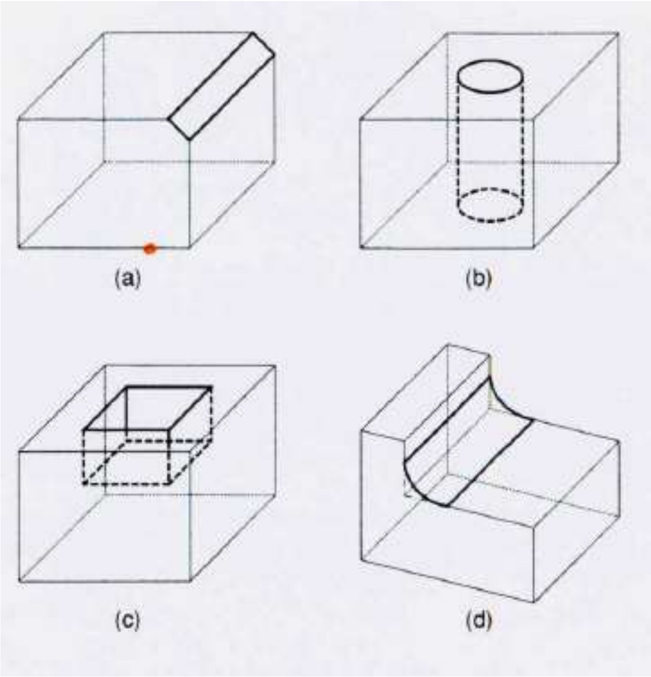



Figure 15: Examples of modeling using other features: (a) chamfering, (b) hole, (c) pocket, and (d) fillet.

- **Parametric modeling**: In *parametric modeling* the designer models a shape by using geometric constraints (for example, two faces are parallel, two edges lie in a plane, a curved edge is tangent to a neighboring straight edge), dimension data and/or dimensional relation on its elements.

- In parametric modeling, a shape is usually constructed in the following manner.

1. Input a two-dimensional shape as a rough sketch.
2. Input geometric constraints and dimensional data interactively.
3. Reconstruct the two-dimensional shape for the given geometric constraints and dimensional data.
4. Repeat steps 2 and 3, modifying the geometric constraints and/or dimensional data until the desired model is obtained. This step is illustrated in Figure 16.
5. Create a three-dimensional shape by sweeping the two-dimensional shape.

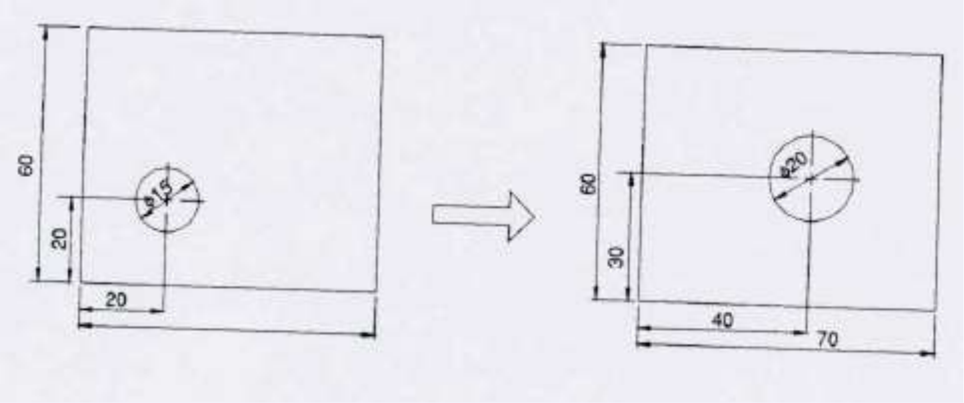



Figure 16: Modifying a shape by changing constraints.

## Data Structure:

- When a solid model is created using the modeling functions provided in solid modeling systems, mathematical descriptions of these models are stored in various data structures.

- The data structure needed to describe a solid can be classified into two types generally according to the entities stored.
  - Constructive solid geometry (CSG) tree structure
  - Boundary representation (B-Rep) data structure

- **CSG tree structure**: This structure stores in a tree the history of applying Boolean operations (see Figure 17) on the primitives. This history is called *constructive solid geometry (CSG) representation*.

- The Boolean operations of the solid model shown in Figure 18 can be envisioned to be a binary tree. The terminal nodes of that tree are primitives and the non-terminal nodes are Boolean operators.

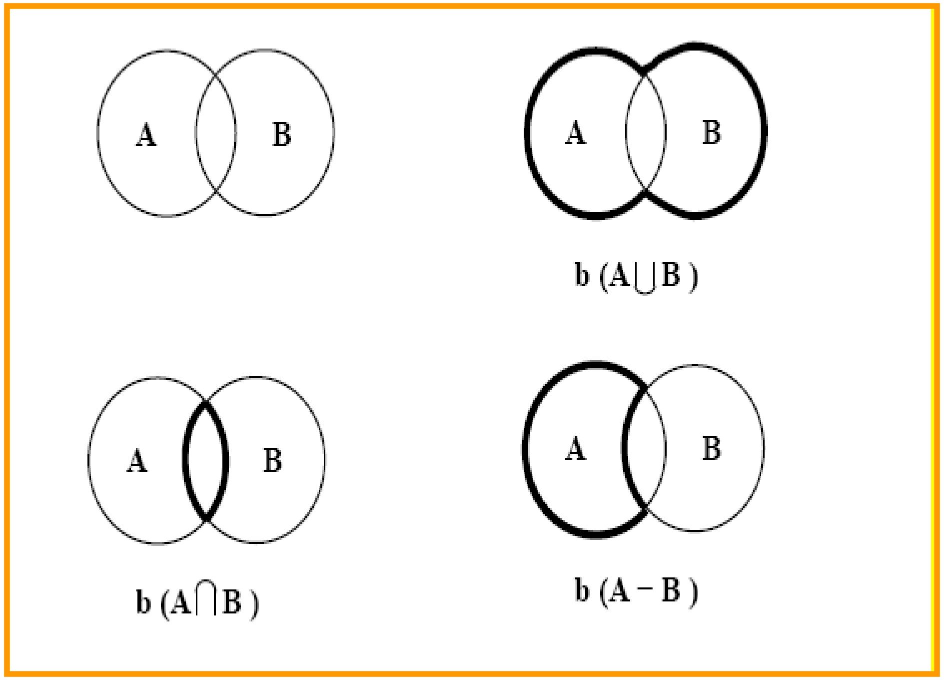



Figure 17: Boolean operators.

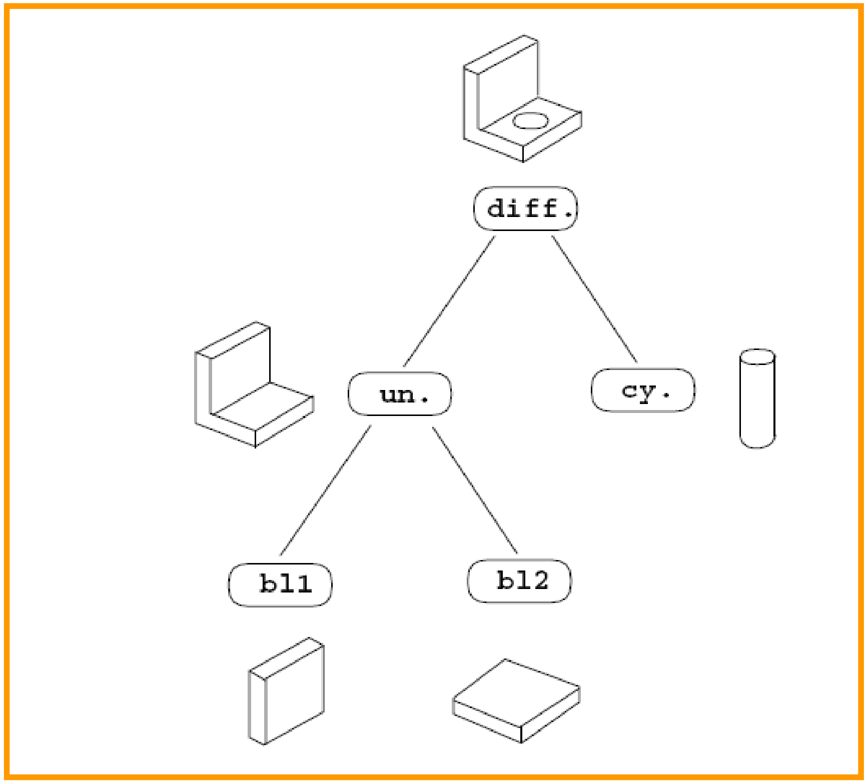



Figure 18: Example of a CSG tree.

- Advantages of CSG tree
  - Validity: the solid stored in a CSG tree is always valid,
  - Conciseness: the CSG tree is simple and stores compact data,
  - Computational ease: the management of data is easy,
  - Unambiguity (clarity): every CSG tree unambiguously models a rigid solid.
- Disadvantages of CSG tree
  - Non-uniqueness: a solid could have more than one CSG representation,
  - Limit on primitives: only simple primitives are used,
  - Redundancy of CSG tree: it may have redundant primitives that do not contribute to final solid.

- **B-Rep data structure**: This structure stores the boundary information of a solid (i.e., vertices, edges, and faces together with the information on how they are connected). This way of describing a solid is called *boundary representation* (*B-Rep*), and its data structure is called a *B-Rep data structure*.

- An example of the B-Rep data structure is given in Table 1. The data structure represents the solid shown in Figure 19.
  - The face table stores the list of bounding edges for each face. The sequence of edges for each face is given by traversing it counterclockwise when the solid is viewed from the outside.
  - Each row of the edge table stores the vertices at the ends of each edge.
  - The vertex table stores the x, y, and z coordinates of the vertices.

- The B-Rep data structure looks very simple and compact. However, it is basically designed for storing planar faces.

Table 1.

**Three tables for storing B-Rep**

| Face Table | | Edge Table | | Vertex Table | |
|---|---|---|---|---|---|
| *Face* | *Edges* | *Edge* | *Vertices* | *Vertex* | *Coordinates* |
| $F_1$ | $E_1, E_5, E_6$ | $E_1$ | $V_1, V_2$ | $V_1$ | $x_1, y_1, z_1$ |
| $F_2$ | $E_2, E_6, E_7$ | $E_2$ | $V_2, V_3$ | $V_2$ | $x_2, y_2, z_2$ |
| $F_3$ | $E_3, E_7, E_8$ | $E_3$ | $V_3, V_4$ | $V_3$ | $x_3, y_3, z_3$ |
| $F_4$ | $E_4, E_8, E_5$ | $E_4$ | $V_4, V_1$ | $V_4$ | $x_4, y_4, z_4$ |
| $F_5$ | $E_1, E_2, E_3, E_4$ | $E_5$ | $V_1, V_5$ | $V_5$ | $x_5, y_5, z_5$ |
| | | $E_6$ | $V_2, V_5$ | | |
| | | $E_7$ | $V_3, V_5$ | | |
| | | $E_8$ | $V_4, V_5$ | | |

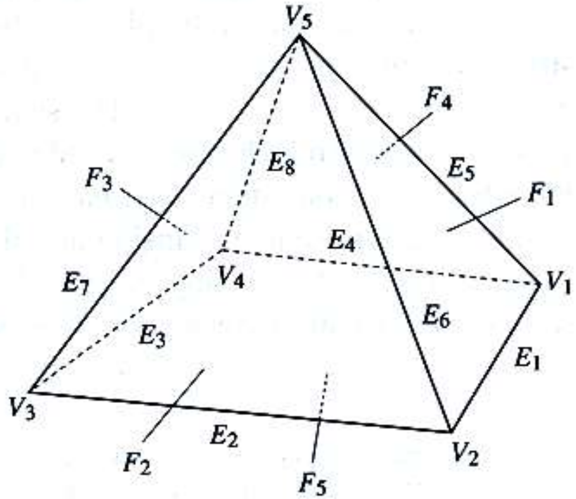



Figure 19: Example of a solid for which data are to be stored.