Solid Modeling

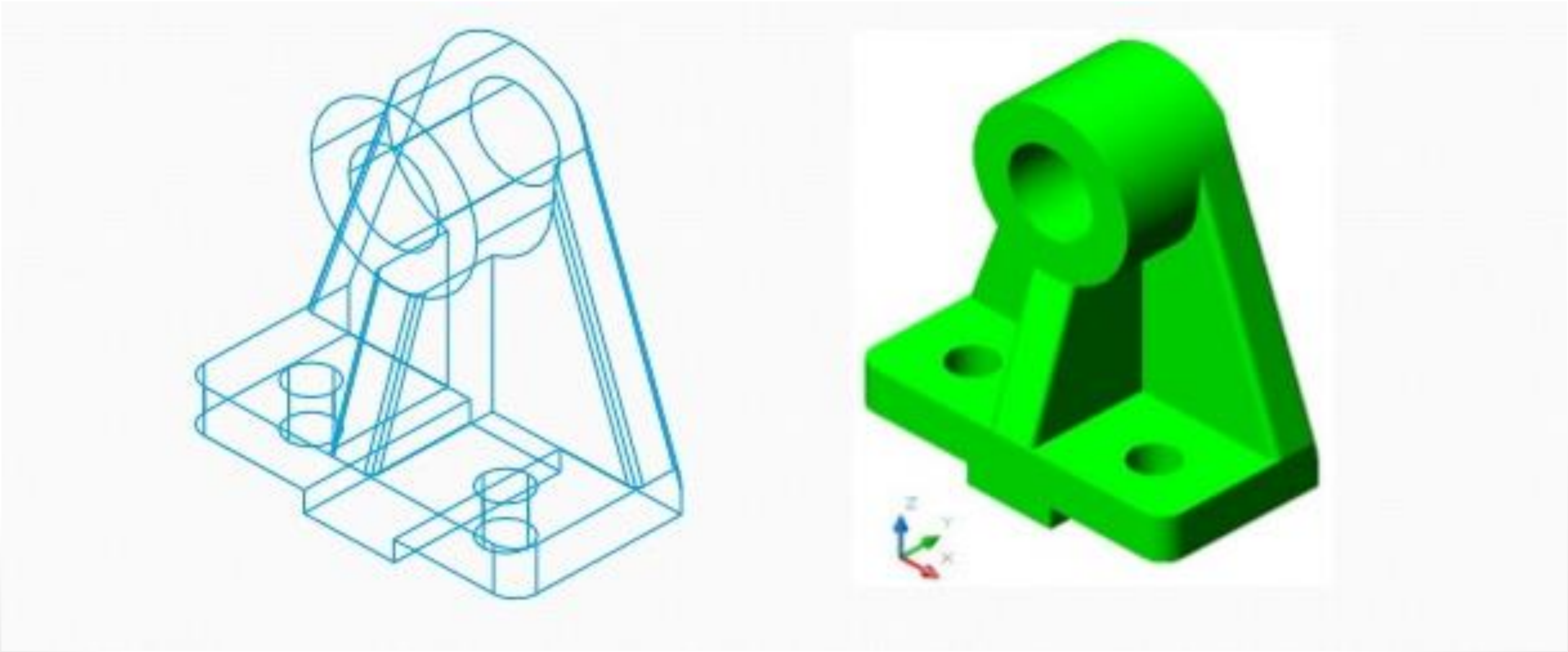
Unit 6

Solid Modeling

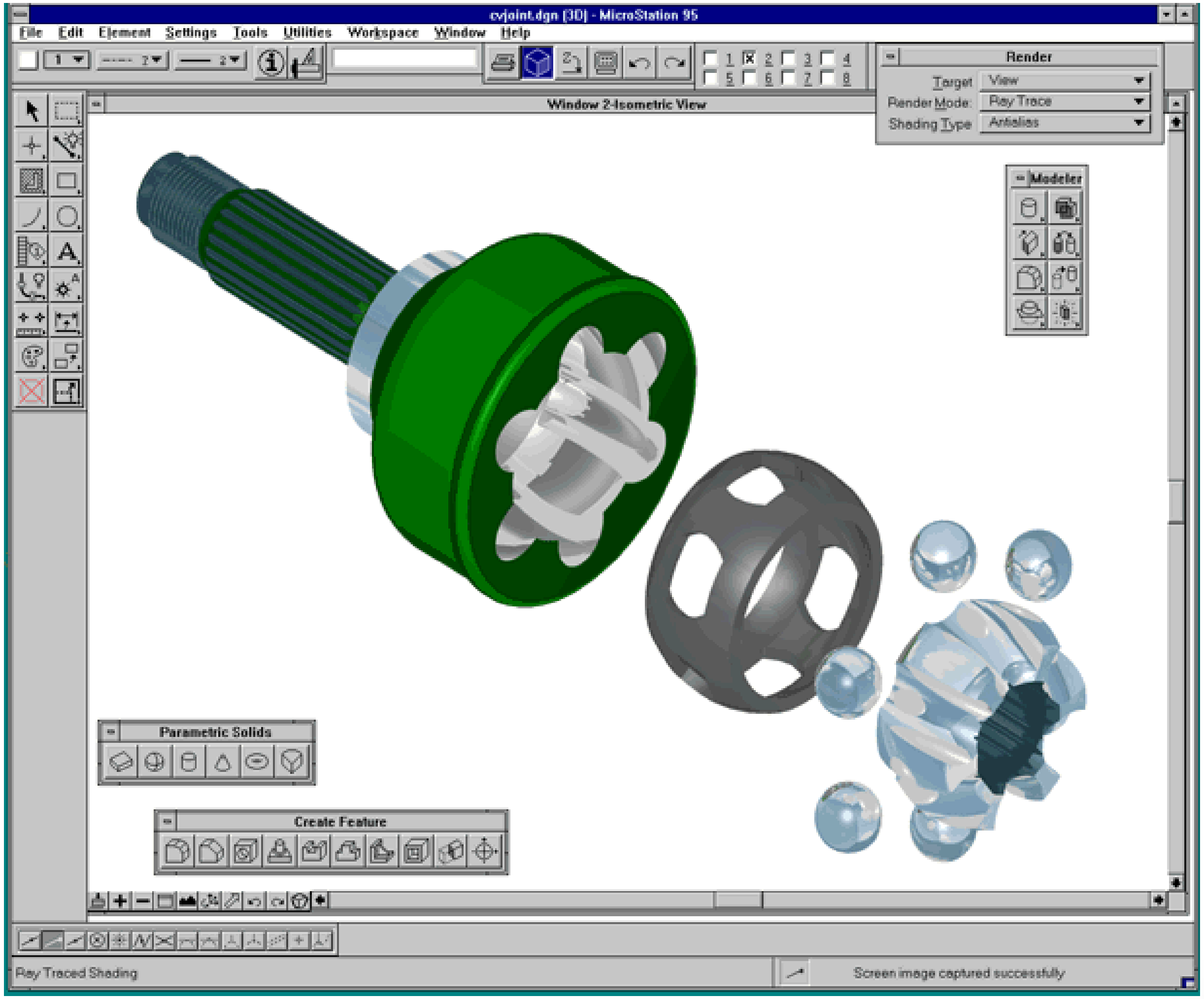
- **Solid modeling** (or **modeling**) is a consistent set of principles for mathematical and computer **modeling** of three-dimensional **solids. Solid modeling** is distinguished from related areas of geometric **modeling** and computer graphics by its emphasis on physical fidelity.
- representation of solid objects unambiguously (clearly). Represent only one object

Solid Modeling…

- In the solid Modeling, the solid definition include vertices(nodes), edges, surfaces, weight, and volume. The model is a complete and unambiguous representation of a precisely enclosed and filled volume



Solid Modeling



Basics of solid modeling theory

- Geometry , topology,
  - (x,y,z) coordinates of vertices : geometry
  - Connectivity matrix: topology
- Geometric closure,
- Set theory and Operations ,

A set is a collection of objects , in the context of geometric representation the basic element of a set is a point . two sets are commonly represented

- W universal set containing all the elements of all sets
- 0 null set containing no elements.
  - A B, A is a subset of B
  - Inequality equality
  - Regularized Boolean set operations:
  - Union difference intersection etc

Solid representation

- Divide Euclidean space into two regions interior and exterior to it separated from each other by the boundary of the object
- Idea: physical object is enclosed by a set of faces, which themselves belong to closed and orientable surfaces
- Information on both geometric and topological elements is stored in the ***B- rep*** database
- Euler's law gives a quantitative relationship among faces edges vertices faces etc.

Properties of Solid models :

- Bounded boundary limits and contain the interior of the solid
- Homogenously 3D no dangling edges or faces presented boundary is always in contact with the interior of the solid
- Finite solid is not infinite in size can be described by a limited amount of information

Why Solid Modeling

Solid Modeling Support

- Using volume information
  - weight or volume calculation, centroids, moments of inertia calculation,
  - stress analysis (finite elements analysis), heat conduction calculations, dynamic analysis,
  - system dynamics analysis
- Using volume and boundary information
  - generation of CNC codes, and robotic and assembly simulation

Boundary Representation

Euler's Formula

For any polyhedron *that doesn't intersect itself,* the

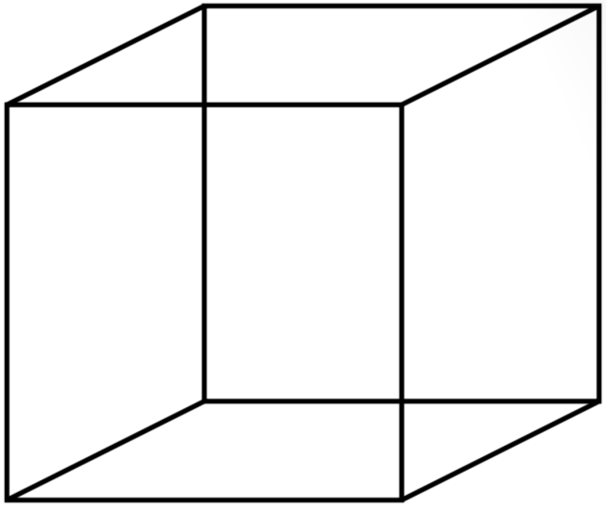
- **Number of Faces**
- plus the **Number of Vertices** (corner points)
- minus the **Number of Edges**

always equals 2

This can be written: **$F + V - E = 2$**

Boundary Representation

Try it on the cube:

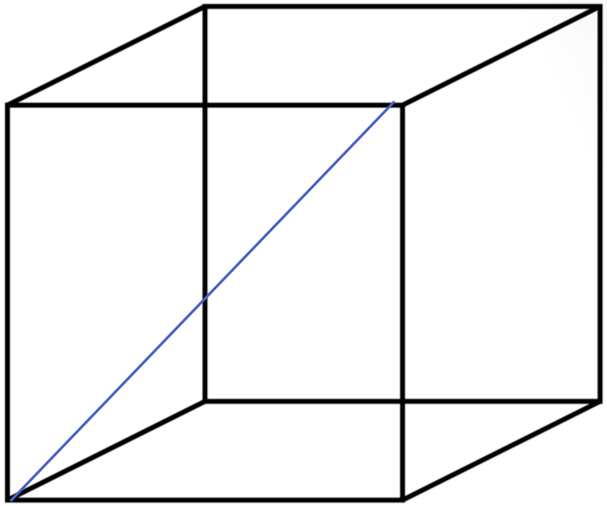


A cube has **6 F**aces, **8 V**ertices, and **12 E**dges, so:

$$6 + 8 - 12 = \mathbf{2}$$

Boundary Representation

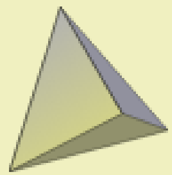
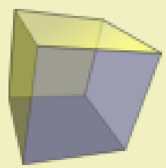
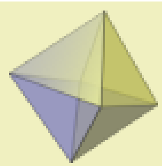
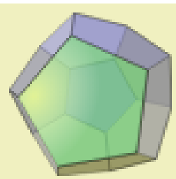
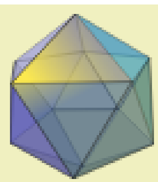
Try it on the cube:



A cube has **7 F**aces, **8 V**ertices, and **13 E**dges, so:

$$7 + 8 - 13 = \mathbf{2}$$

Boundary Representation

Name		Faces	Vertices	Edges	F+V-E
Tetrahedron		4	4	6	**2**
Cube		6	8	12	**2**
Octahedron		8	6	12	**2**
Dodecahedron		12	20	30	**2**
Icosahedron		20	12	30	**2**

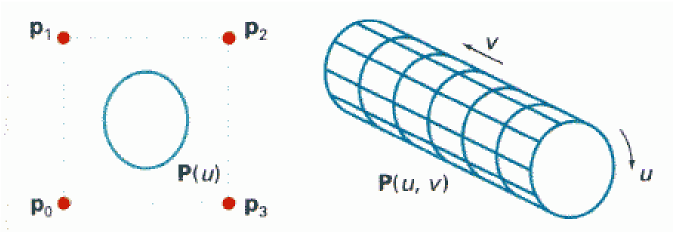
Sweep representation

- The former are known as extruded solids and are created via linear or translational **sweep**; the latter are solids of revolution which can be created via rotational **sweep**. **Sweeping** is based on the notion of moving a point, curve, or a surface along a given path.

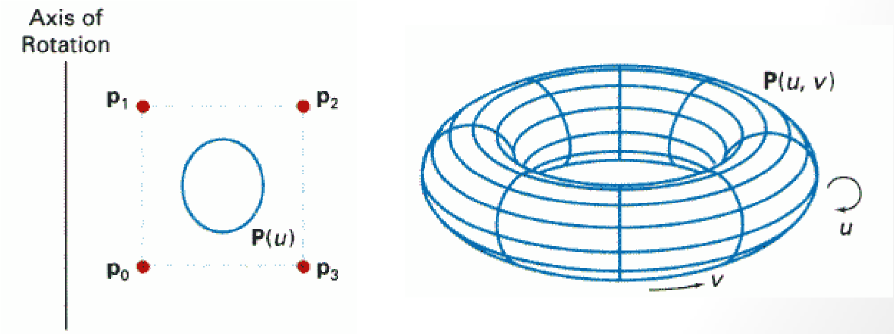
Sweep representation

- Sweep representations mean sweeping a 2D surface in 3D space to create an object. However, the objects created by this method are usually converted into polygon meshes and/or parametric surfaces before storing.

A Translational Sweep:



A Rotational Sweep:



Sweep representation

Other variations:

- We can specify a special path for the sweep as some curve function.
- We can vary the shape or size of the cross section along the sweep path.
- We can also vary the orientation of the cross section relative to the sweep path.

Sweep representation

- Used for creating solids with uniform thickness in a particular direction and axisymmetric solids by translational and rotational sweeping
- Sweeping requires
- A surface to be moved and a trajectory (along which the movement should occur)
- Recent Advancements In solid modeling
- NURBS (Non Uniform Rational BSplines)
- Solid modeler for handling free form surface definitions and not just polyhedral and quadric models

Sweep Representation

- It is difficult to apply regularized Boolean set operations.
- Even simple sweeps are not close under regularized Boolean set operation

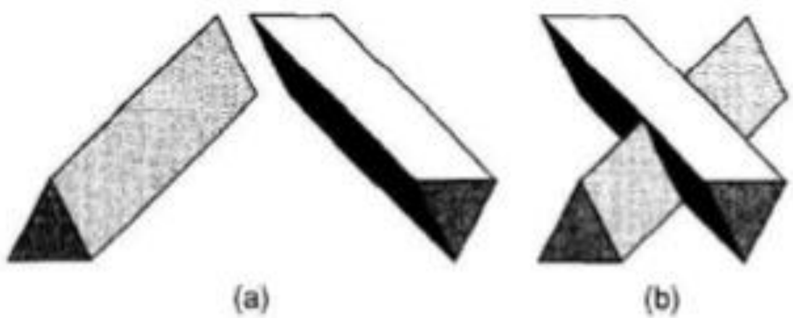
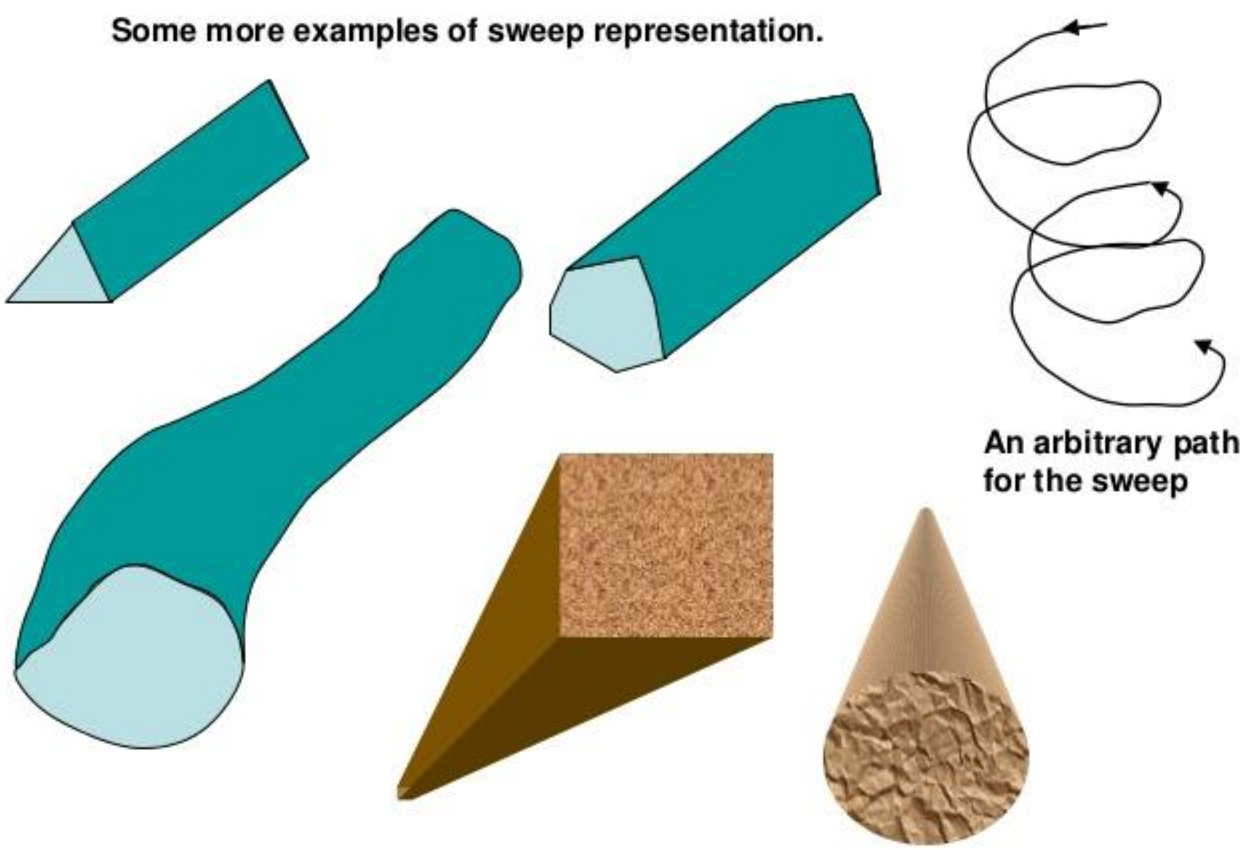
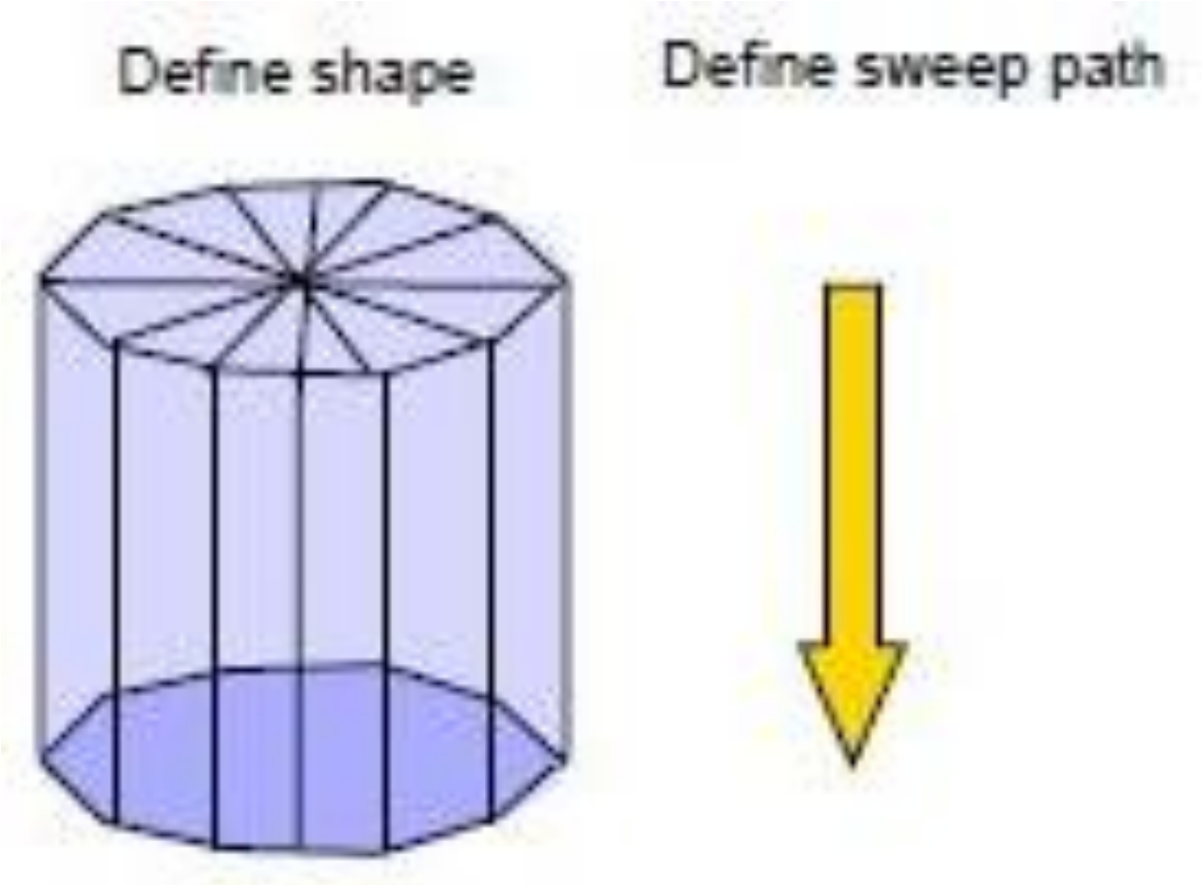


Fig. 12.10 (a) Two simple sweeps of 2D objects (triangles). (b) The union of the sweeps shown in (a) is not itself a simple sweep of a 2D object.

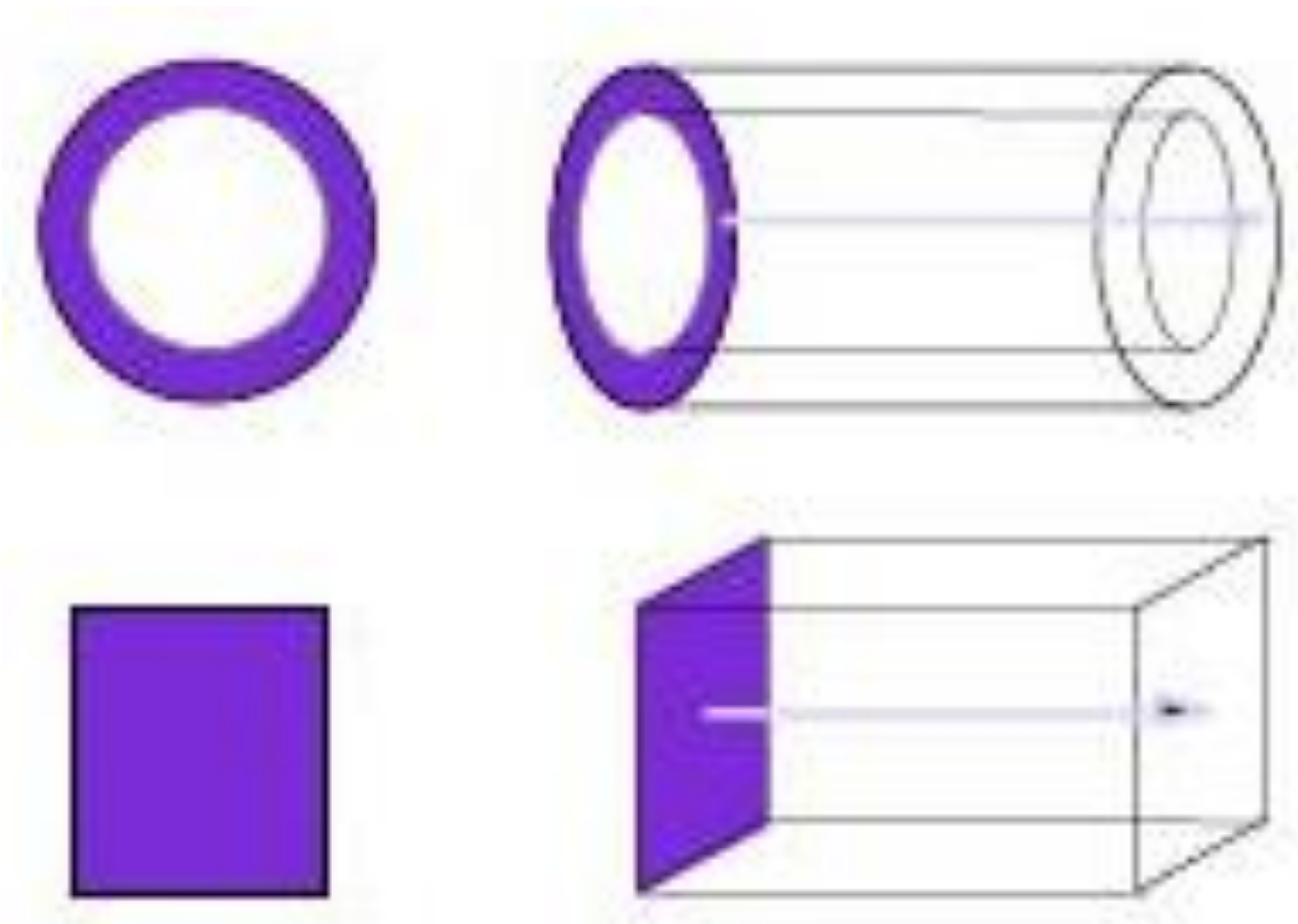
Sweep Representation



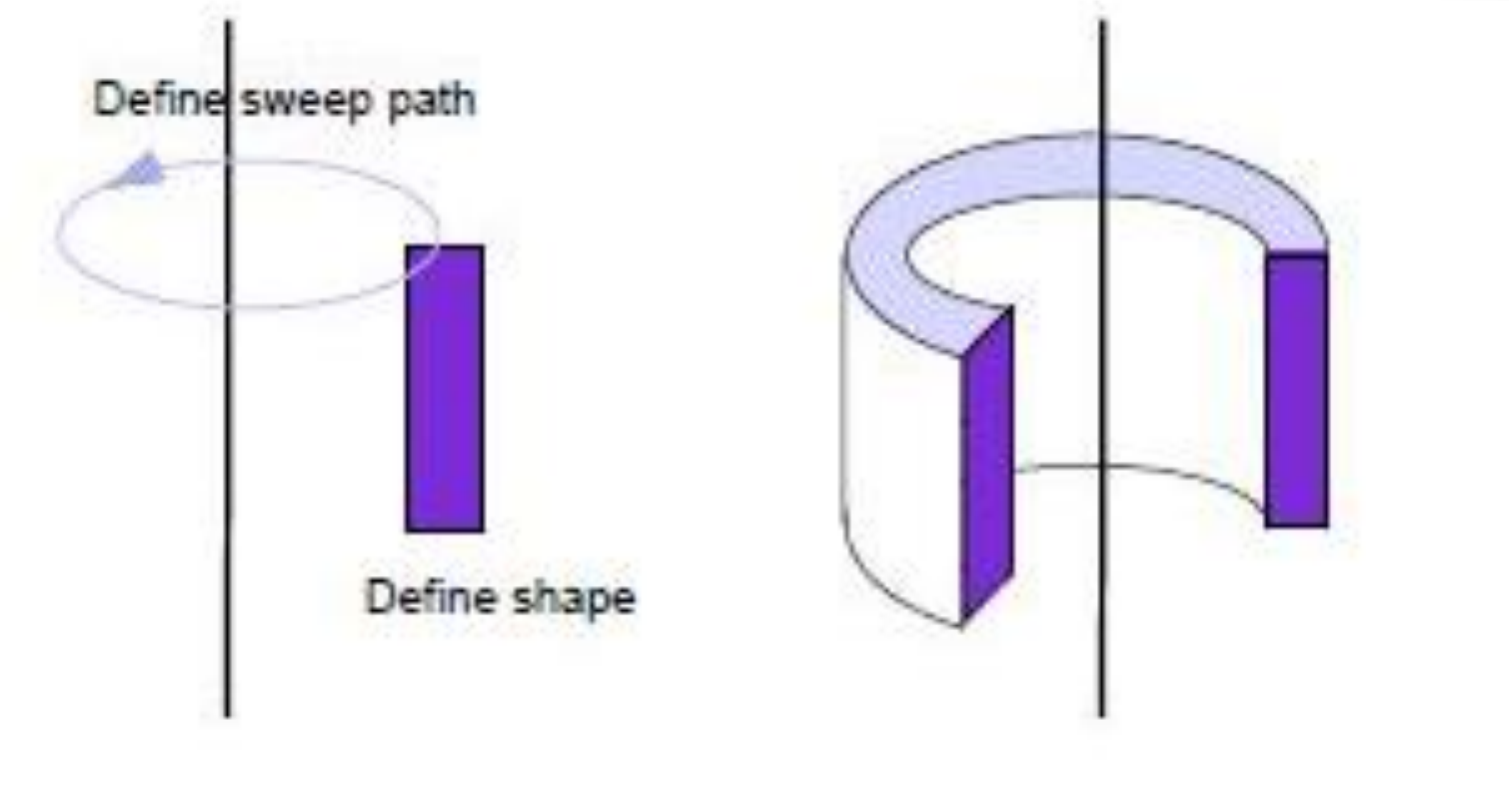
Sweep Representation



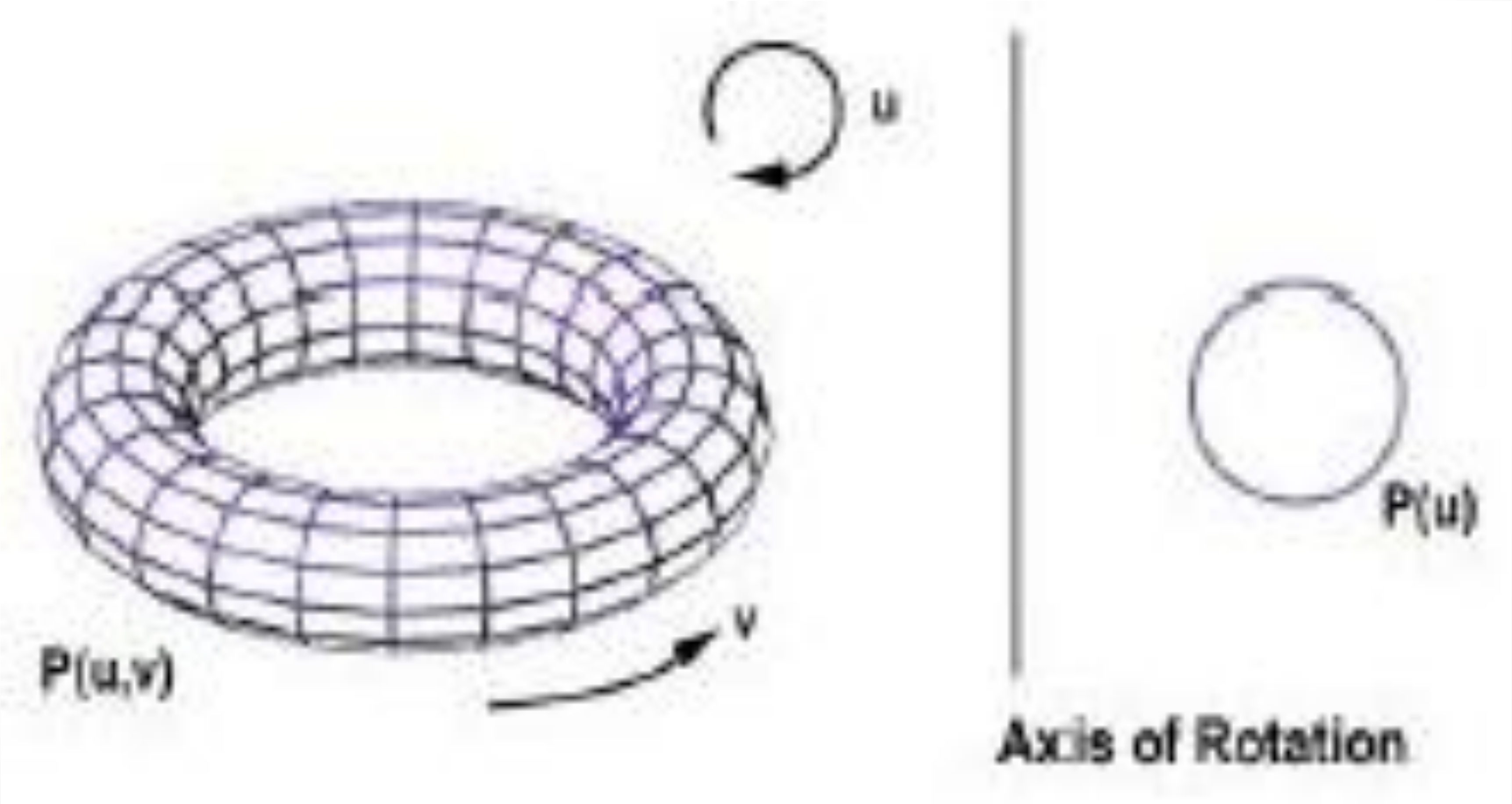
Sweep Representation



Sweep Representation



Sweep Representation



Octree Representation: (Solid-object representation)

- This is the ***space-partitioning*** method for 3D solid object representation. This is hierarchical tree structures (octree) used to represent solid object in some graphical system. Medical imaging and other applications that require displays of object cross section commonly use this method. E.g.: CT-scan
- It provides a convenient representation for storing information about object interiors.
- An octree encoding scheme divides region of 3D space into octants and stores 8 data elements in each node of the tree. Individual elements are called volume element or voxels. When all voxels in an octant are of same type, this type value is stored in corresponding data elements. Any heterogeneous octants are subdivided into octants again.

Binary Space Partitioning (BSP)

- Binary space partitioning is a 3-D graphics programming technique of dividing a scene into two recursively using hyperplanes.
- In other words, a 3-D scene is split in two using a 2-D plane, then that scene is divided in two using a 2-D plane, and so on. The resulting data structure is a binary tree, or a tree where every node has two branches.
- The technique is widely used to speed up rendering of 3-D scenes, especially in games.

Binary Space Partitioning (BSP)

- **binary space partitioning** (**BSP**) is a method for recursively subdividing a space into convex sets by hyperplanes. This subdivision gives rise to a representation of objects within the space by means of a tree data structure known as a **BSP tree**.
- Binary space partitioning was developed in the context of 3D computer graphics, where the structure of a BSP tree allows spatial information about the objects in a scene that is useful in rendering, such as their ordering from front-to-back with respect to a viewer at a given location, to be accessed rapidly. Other applications include performing geometrical operations with shapes (constructive solid geometry) in CAD, collision detection in robotics and 3D video games, ray tracing and other computer applications that involve handling of complex spatial scenes.

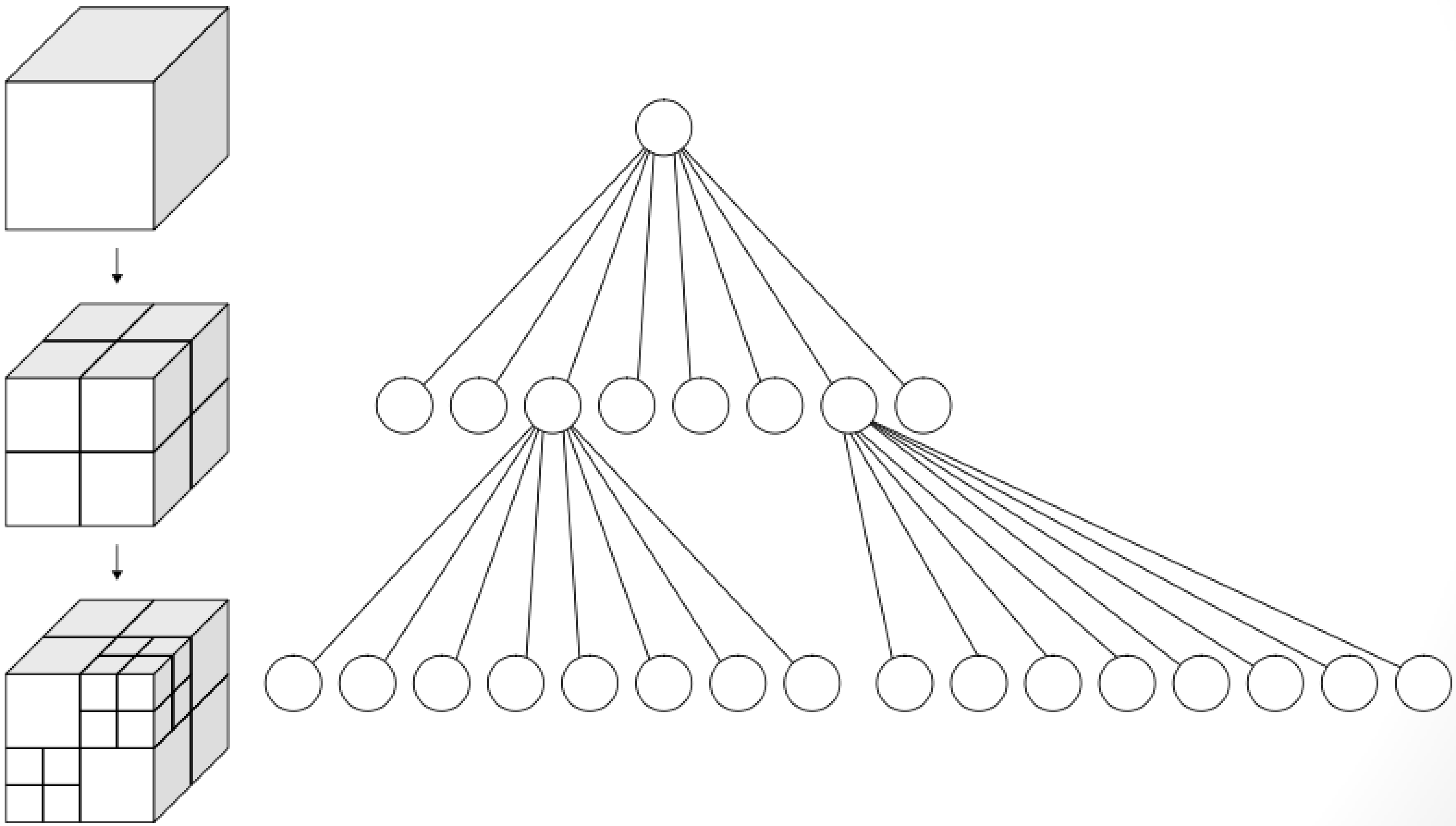
Binary Space Partitioning (BSP)

- Previous list priority algorithms fail in a number of cases non of them is completely general
- BSP tree is a general solution, but with its own problems
  - Tree size
  - Tree accuracy

octree

- **octree** is a tree data structure in which each internal node has exactly eight children. Octrees are most often used to partition a three-dimensional spaceby recursively subdividing it into eight octants.
- Octrees are the three-dimensional analog of quadtrees. The name is formed from *oct* + *tree*, but note that it is normally written "*octree*" with only one "t". Octrees are often used in 3D graphics and 3D game engines.

octree



Chapter 6 Finished