UNIVERSITÉ Concordia UNIVERSITY
FACULTY OF ENGINEERING AND COMPUTER SCIENCE
Department of Computer Science and Software Engineering

COMP371
COMPUTER GRAPHICS

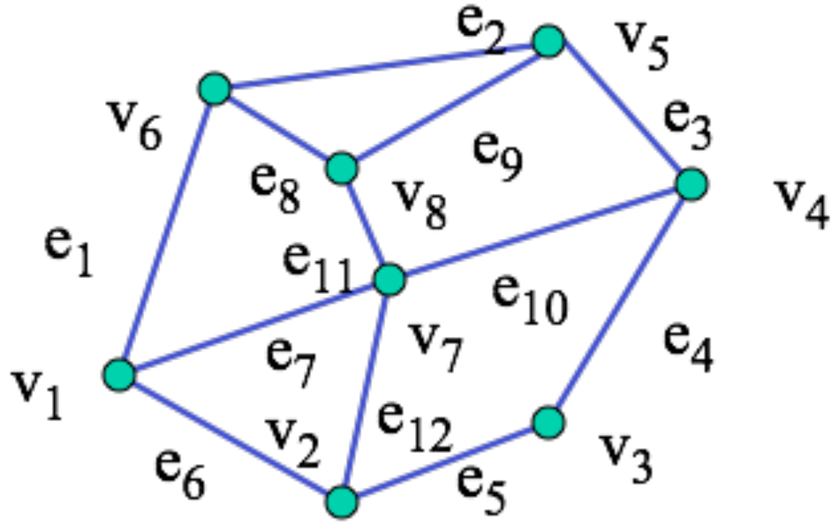
LECTURE 6
BUILDING MODELS

Lecture Overview

- Building models
  - Introduce simple data structures for building polygonal models

Representing a Mesh

- Consider a mesh



- There are 8 nodes and 12 edges
- 5 interior polygons
- 6 interior (shared) edges
- Each vertex has a location $v_i = (x_i\ y_i\ z_i)$

Simple Representation

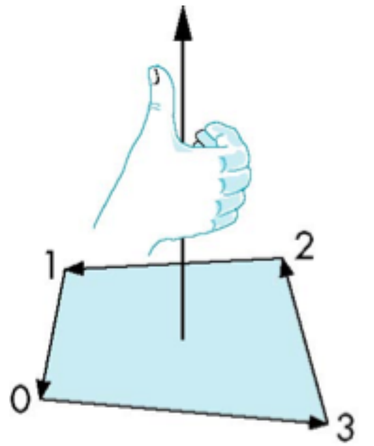
Define each polygon by the geometric locations of its vertices

```
glm::vec3 v1 = glm::vec3(x1, y1, z1);
glm::vec3 v2 = glm::vec3(x2, y2, z2);
…
```

- Inefficient and unstructured

Inward and Outward Facing Polygons

- The order $\{v_1, v_6, v_7\}$ and $\{v_6, v_7, v_1\}$ are equivalent in that the same polygon will be rendered by OpenGL but the order $\{v_1, v_7, v_6\}$ is different
- The first two describe *outwardly facing* polygons
- Use the *right-hand rule* = counter-clockwise encirclement of outward-pointing normal
- OpenGL can treat inward and outward facing polygons differently

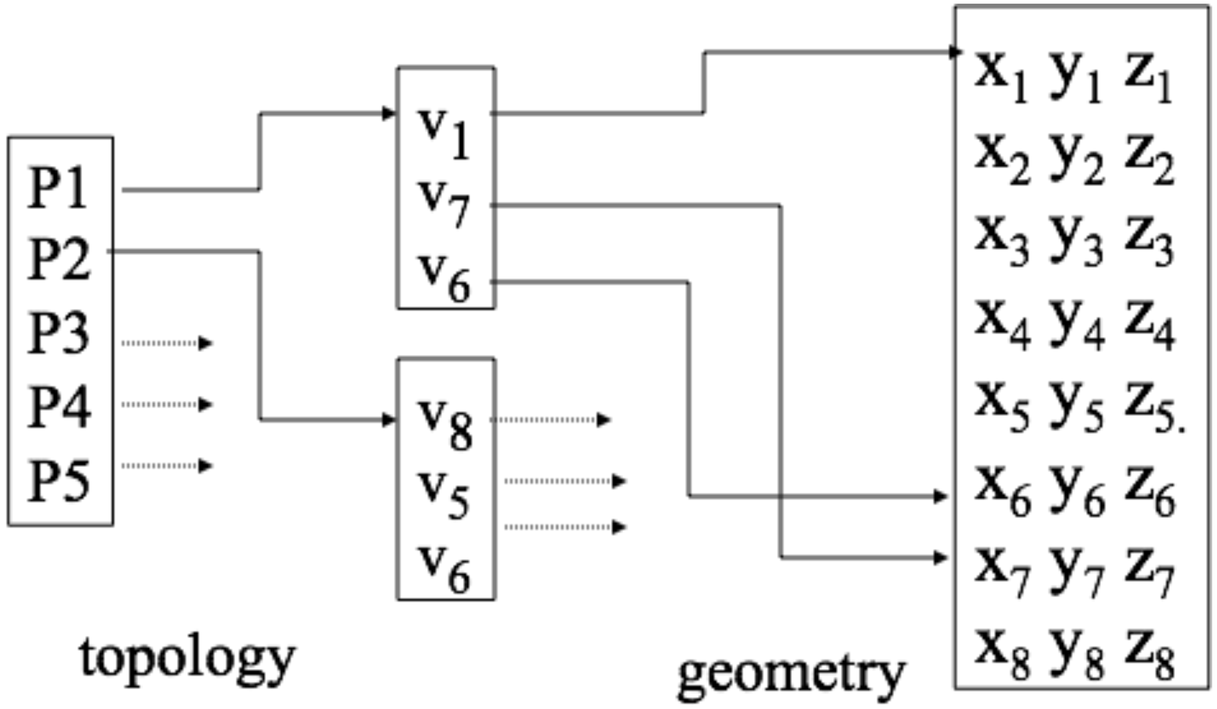


Geometry vs Topology

- Generally it is a good idea to look for data structures that separate the geometry from the topology
  - Geometry: locations of the vertices
  - Topology: organization of the vertices and edges
  - Example: a polygon is an ordered list of vertices with an edge connecting successive pairs of vertices and the last to the first
  - Topology holds even if geometry changes

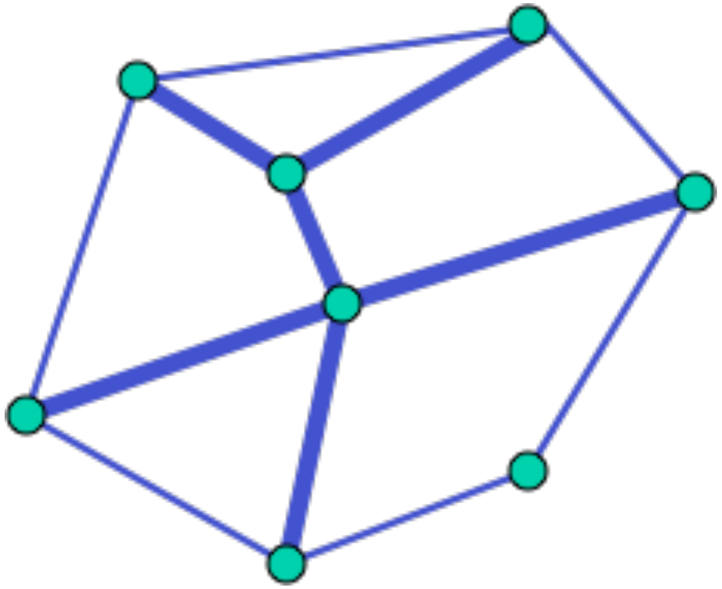
Vertex Lists

- Put the geometry in an array
- Use pointers from the vertices into this array
- Introduce a polygon list



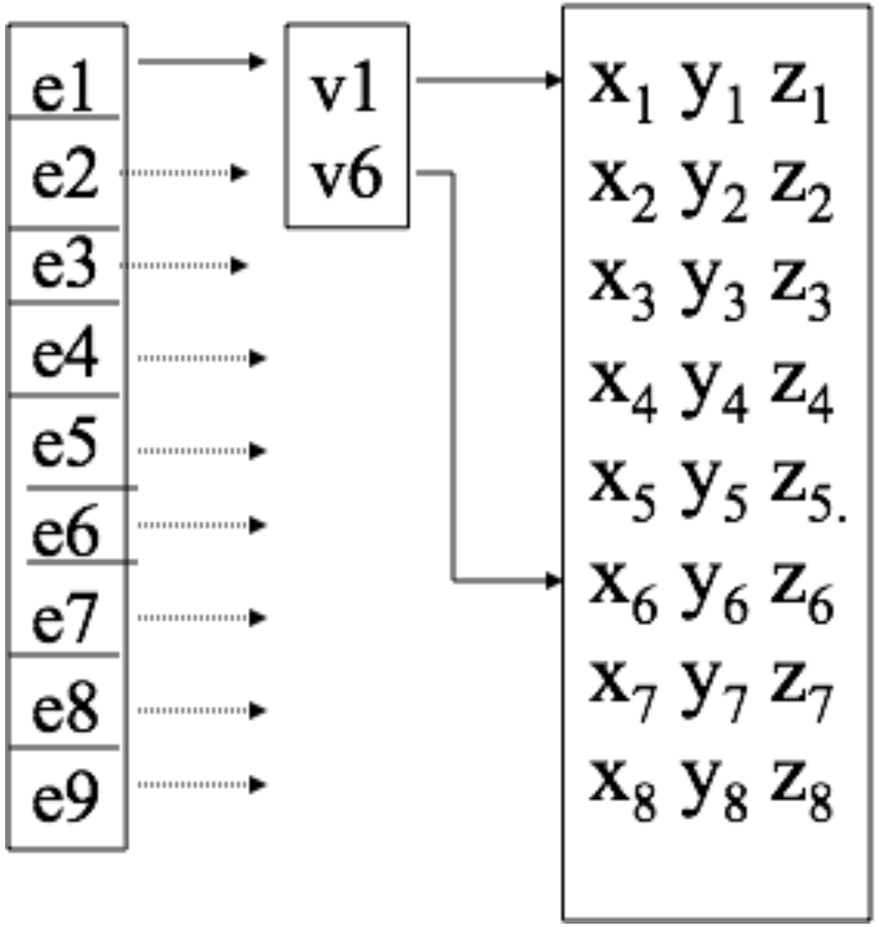
Shared Edges

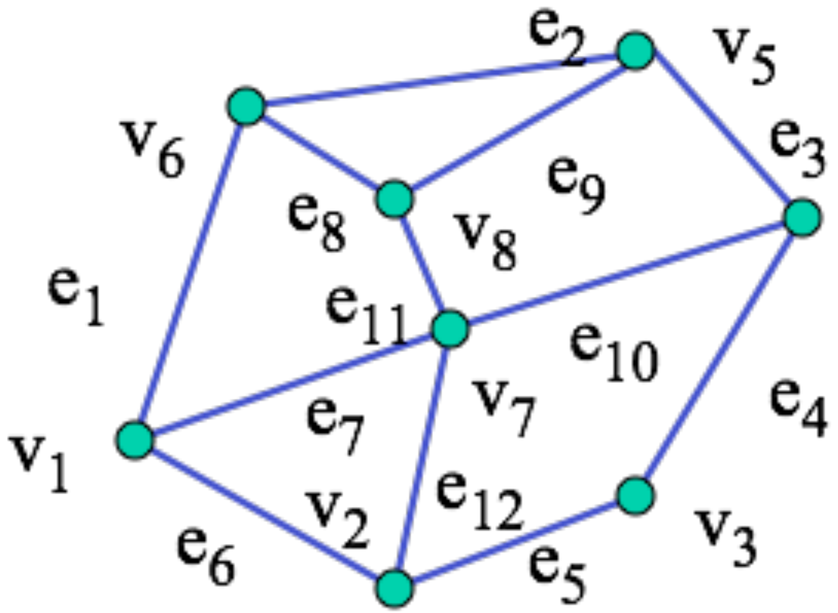
- Vertex lists will draw filled polygons correctly but if we draw the polygon by its edges, shared edges are drawn twice



- Can store mesh by *edge list*

Edge List





Note polygons are not represented

Vertex Arrays

- OpenGL provides a facility called *vertex arrays* that allows us to store array data in the implementation
  - attributes:
    - position, color, texture coordinates, indices, normals

Example

Review

- Geometry and Transformations
  - Introduce the elements of geometry
    - Scalars, Vectors, Points
- Coordinate Systems
- Homogeneous Coordinates

Next Lecture

- Building models

CONCORDIA.CA

Studying Support Slides