Computer Graphics

Lecture 20
Hierarchical Transformations
Scene Graphs
Announcements
Goals

- Know how to structure a collection of objects in a scene graph where transformations are applied hierarchically.
Transformation Hierarchies
AKA Scene Graphs

• Represent a drawing (“scene”) as a list of objects
• Transform for each object
  – can use minimal primitives: ellipse is transformed circle
  – transform applies to points of object
Example

- Can represent drawing with flat list
  - but editing operations require updating many transforms
Groups of objects

• Treat a set of objects as one

• Introduce new object type: group
  – contains list of references to member objects

• This makes the model into a tree
  – interior nodes = groups
  – leaf nodes = objects
  – edges = membership of object in group
Demo: Drawing in Keynote
Example

- Add group as a new object type
  - lets the data structure reflect the drawing structure
  - enables high-level editing by changing just one node
The Scene Graph (tree)

- A name given to various kinds of graph structures (nodes connected together) used to represent scenes
- Simplest form: tree
  - just saw this
  - every node has one parent
  - leaf nodes are identified with objects in the scene
  - alternate design: interior nodes can also have objects
Instances

• Simple idea: allow an object to be a member of more than one group at once
  – transform different in each case
  – leads to linked copies
  – single editing operation changes all instances
Questions?
Questions?

• That wraps up our discussion of transformations.

• We have an (almost) fully-featured wireframe rendering framework.
  • We haven't implemented clipping yet for geometry outside the view volume.

• Next up:
  • more realism: occlusion, shading
  • speed: using hardware