Computer Science 598d

Advanced Topics in Computer Science

Geometric Modeling for Computer Graphics

Thomas A. Funkhouser

Today's Lecture

- **General Information**
- **Interactive 3D Graphics**
  - Motivating examples
  - Object representations
  - Run–time operations
- **3D Geometric Modeling**
  - Interactive tools
  - Procedural generation
  - Data capture
  - Geometric manipulation
- **Course Overview**
  - Goals
  - Syllabus
  - Coursework

General Information

- **Lectures**
  - TTh 1:30–2:50PM
  - CS 105 (Small Auditorium)
- **Prerequisites**
  - COS426 or equivalent
- **Office Hours**
  - TTh 2:50–3:50PM
  - Room 422, CS Bldg
  - funk@cs.princeton.edu

3D Interactive Graphics

- **Goals**
  - Fast frame rates
  - Realism and detail
  - Animation and interaction
- **Applications**
  - Training
  - Communication
  - Entertainment
  - Visualization
  - Computer–aided design

Examples

- **Soda Hall**
- **Light Field**
  - Levoy & Hanrahan
- **Prophet of Doom**
  - Naylor
- **The Campanile**
  - Debevec
- **Hang Glider**
  - Evans & Sutherland

Geometric Representations

- **How is the graphics database represented?**
  - Set of polygons
  - Mesh
  - Curved surfaces
  - Scene graph
  - Voxels
  - BSP tree
  - Image samples

Geometric Operations

- ♦ **How are run–time operations implemented?**

  - **Image Generation**
    - Polygonization
    - Texturing
    - Culling
    - Detail elision
    - Hidden surface removal

  - **Intersection**
    - Collision detection
    - Picking

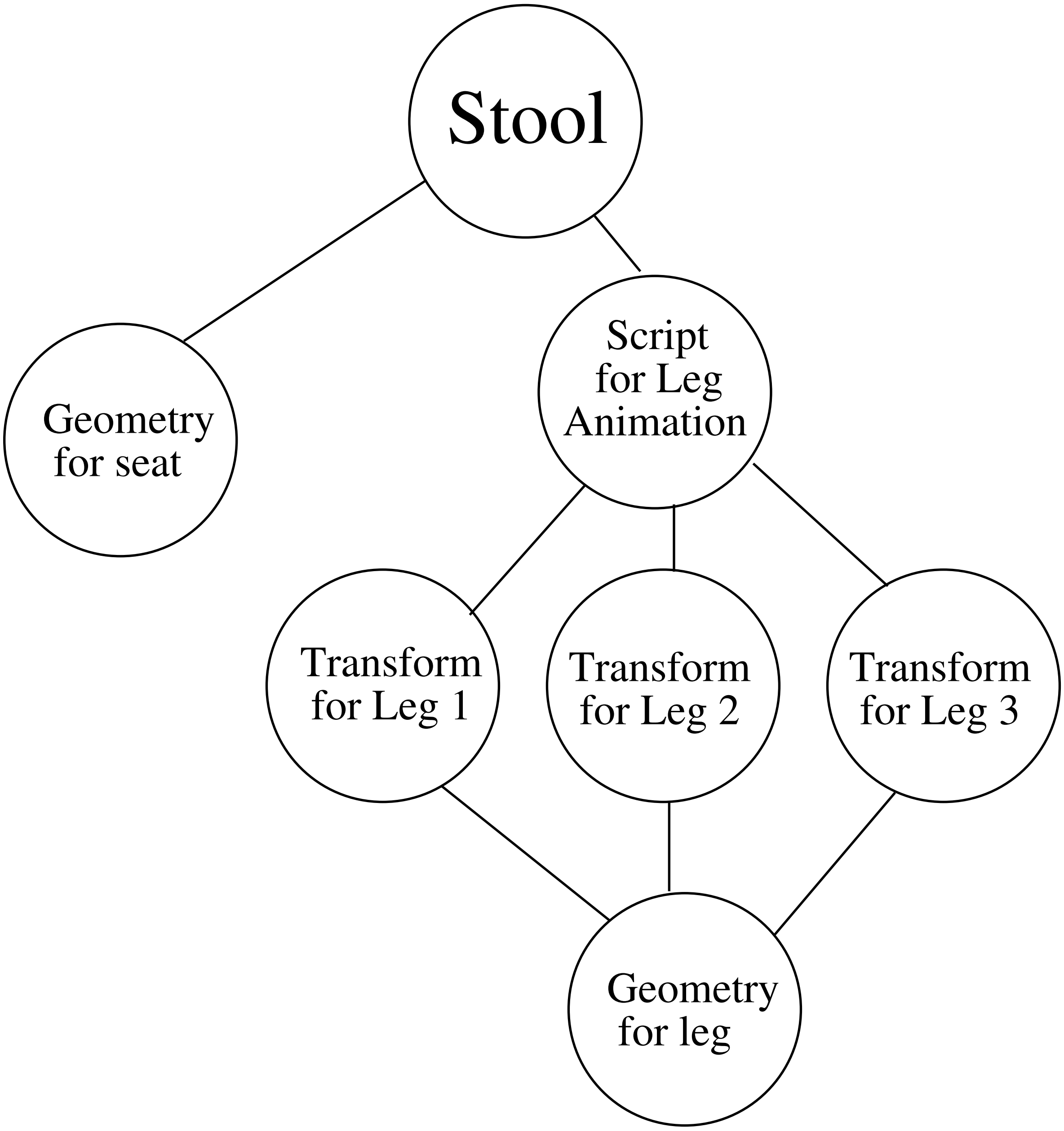
  - **Simulation**
    - Animation, dynamics, lighting, acoustics, haptics

Geometric Modeling

- **How is the graphics database created?**
  - Interactive modeling
  - Procedural generation
  - Capture from sensors
  - Derivation from other data

Example: VRML

- Representation = Scene Graph



Example: VRML

- **Operation = Scene graph traversal**
  - Image Generation
    - Culling
    - Detail elision
  - Intersection
    - Collision detection
    - Picking
  - Simulation
    - Animation

Example: VRML

- **Creation = Interactive Modeling**

  - Interactive tools
    - CosmoWorlds, Alias, etc.

  - Basic modeling operations
    - Create face sets
    - Assign attributes
    - Specify transformations
    - Insert/remove nodes
    - etc.

  - More modeling operations
    - CSG operations
    - Curves and surfaces
    - etc.

Scene Graph Requirements

- **Hierarchical structure**
  - Bounding volumes
  - Few transformations
  - Few graphics state changes
- **Levels of Detail (LODs)**
  - Multiresolution
  - Detail elision
- **Textures**
  - Replace surface detail
  - Replace complex lighting
  - Replace distant geometry
- **Simulations**
  - Precomputed

Interactive 3D Modeling

- **Problems**
  - Tedious
  - Error prone
  - Imprecise
- **Common Errors**
  - Cracks between polygons
  - Intersecting polygons
  - Overlapping polygons
  - Unoriented polygons
  - Missing polygons
  - T–junctions
- **Other Difficulties**
  - High polygon counts
  - No levels of detail
  - Poor hierarchical structure

Case Studies

- **Soda Hall**
  - 14K objects
  - 2M polygons
  - 2 person years
- **VirtualWorks**
  - 500 objects
  - 80K polygons
  - 1 person year
- **Evans & Sutherland**
  - handful of databases per year
  - 80 people in database group

Objective of This Course

Investigate **automatic** methods for creating and fixing 3D models suitable for interactive graphics.

- **Model Generation**
  - Procedural methods
  - Capture from sensors
  - Derivation from other data
- **Model Manipulation**
  - Fixup
  - Simplification
  - Optimization
- **Model Simulation**
  - Lighting
  - Appearance
  - Dynamics

Syllabus

♦ Object Representations

- **Polygonal Representations**
  - Mesh representations
  - Procedural generation of mesh representations
  - Mesh capture and reconstruction
  - Fixup of mesh representations
  - Mesh simplification
  - Multiresolution mesh representations
- **Scene Graph Representations**
  - Optimization of scene graphs
- **Curve and Surface Representations**
  - Real–time surface rendering
- **Solid Representations**
  - BSP Trees
- **Image–Based Representations**
  - View interpolation, light field, and lumigraph
  - Hybrid geometry– and image–based methods

♦ Simulation

- Lighting
- Appearance
- Acoustics
- Dynamics

Coursework

- **Thought Exercises**
  - Short answers to open questions (once per week or so)
- **Programming Assignments**
  - Implement geometric algorithm (three or four during semester)
- **Oral Presentations**
  - Lecture on research topic (once)
  - Present project proposal (once)
  - Present project results (once)
- **Course Project**
  - Attack research problem

Laboratory

- **Meca**
  -

First Assignment

- **Background Questionaire**
  - Fill–out and email to me