Chapter 6 - The Scene Graph

- Why a scene graph?
- What is stored in the scene graph?
  - objects
  - appearance
  - camera
  - lights
- Rendering with a scene graph
- Practical example
The 3D Rendering Pipeline (our version for this class)

3D models in model coordinates → 3D models in world coordinates → 2D Polygons in camera coordinates → Pixels in image coordinates

- Scene graph
- Camera
- Animation, Interaction
- Rasterization
- Lights
Why a Scene Graph?

- Naive approach:
  - for each object in the scene, set its transformation by a single matrix (i.e., a tree 1 level deep and N nodes wide)
    - advantage: very fast for rendering
    - disadvantage: if several objects move, all of their transforms change

- Observation: Things in the world are made from parts

- Approach: define an object hierarchy along the *part-of* relation
  - transform all parts only relative to the whole group
  - transform group as a whole with another transform
  - parts can be groups again

http://www.boby-online.de/Veritherm/Explosionszeichnung.jpg
Chapter 6 - The Scene Graph

• Why a scene graph?

• What is stored in the scene graph?
  – objects
  – appearance
  – camera
  – lights

• Rendering with a scene graph

• Practical example
Geometry in the Scene Graph

• Leaves are basic 3D objects (polygon meshes, primitives, ...)

• Non-leaf nodes (groups) contain a transformation
  – can have one or several children
  – transformation is given by a homogeneous Matrix

• Root is the entire world

• Nodes can be the child of several groups
  – not a tree, but a directed acyclic graph (DAG)
  – effective reuse of geometry
Appearance in the Scene Graph

• Scene graph also contains appearances
  – Appearance: E.g. Color, reflection, transparency, texture
    Details see next lecture
  – can be reused similarly to geometry

• Appearance can be only partially specified
  – unspecified values are inherited
Lights in the Scene Graph

- Light sources also need a position and/or direction
  - Just include them into the scene graph
  - Can be animated just like geometry

- Lights can be in local coordinate systems of geometry groups
  - move with them
  - example: headlamps on a car
The Camera in the Scene Graph

- Camera also needs a position and direction
  - Just include it into the scene graph
  - Can be animated just like geometry

- Camera can be in local coordinate systems of geometry groups
  - move with them
  - example: driver’s view from a car
Chapter 6 - The Scene Graph

• Why a scene graph?
• What is stored in the scene graph?
  – objects
  – appearance
  – camera
  – lights

• Rendering with a scene graph
• Practical example
Scene graph traversal for rendering

- set $T_{act}$ to $T_{Auto}$
- push state
- set $T_{act}$ to $T_{act} \times T_{Karosserie}$
- push state
- set $T_{act}$ to $T_{act} \times T_{Chassis}$
- render Quader1
- pop state
- set $T_{act}$ to $T_{act} \times T_{Kabine}$
- render Quader2
- pop state
- pop state
- set $T_{act}$ to $T_{act} \times T_{Räder}$
- ...
Scene Graph Libraries

- Scene graphs exist on a more abstract layer than OpenGL!
- VRML/X3D
  - descriptive text format, ISO standard
- OpenInventor
  - based on C++ and OpenGL
  - originally Silicon Graphics, 1988
  - now supported by VSG3d.com
- Java3D
  - provides 3D data structures in Java
  - not supported anymore
- Open Scene Graph (OSG)
- Various Game Engines
  - e.g. JMonkey 3 (scene graph based game engine for Java)
Scene Graphs in Practice

- Creation of scene graphs and objects
  - Specific authoring software (e.g. Blender, Maya, 3DS Max)
- Assets (models, objects) exported to exchange formats
  - E.g. (X3D,) Wavefront OBJ (.obj), 3ds Max (.3ds), Ogre XML (.mesh)
- Objects typically are tesselated
  - Polygon meshes
  - No primitive geometric objects visible/readable anymore
- Example:
  - JME Scene
Chapter 6 - The Scene Graph

• Why a scene graph?
• What is stored in the scene graph?
  – objects
  – appearance
  – camera
  – lights
• Rendering with a scene graph
• Practical example
Example of a scene graph

- Graph to be drawn together in the lecture
- VRML world linked from the class page