Modelling

- Modelling and Rendering
- Transformations

Moving this model?
- Change the transformations over time.

WHY?!!

Modelling

- Modelling and Rendering
- Transformations

The object we are modelling is constrained but the model does not know that.

- We need:
  - To represent the structure of the model.
  - A handle on parameters so that we can move only through valid poses.
  - So we structure our transformations into a hierarchy.
Modelling

- Modelling a two-link arm
  - Rigid links
  - Joints:
    - Upper arm link B has two joints p and q (shoulder and elbow)
    - Lower arm link A has one joint r
  - Parameters to control:
    - Shoulder position T
    - Shoulder angle θ (A and B rotate about p)
    - Elbow angle φ (A rotates about r, and stays attached to B at q)

- Attach point q on B to r on A.

Modelling

- Modelling a two-link arm
  - Start with A and B in their original positions
  - Apply only to A:
    - Translate by -r
    - Rotate by φ about the origin.
    - Translate by q, bringing r and q together.
  - We can now consider q as the origin of the lower arm link, and regard A as being in this coordinate system.

Modelling

- Modelling a two-link arm
  - Now the transformations apply to both A and B:
    - Translate by -p
    - Rotate by θ about the origin.
    - Translate by T to place the two-link arm at the proper position.

Hierarchical Modelling

- Store the modelling sequence in a hierarchy
  - Leave the geometry.
  - Internal nodes have transformations.
  - Transformations apply to everything under them – start at the bottom and work your way up.

- Parameters
- Structural constraints
- Geometric Primitives

Hierarchical Modelling

- Another view
  - The shoulder coordinate transform moves everything below it w.r.t. the shoulder:
    - B and its transformation.
  - The elbow coordinate transform moves A with respect to the shoulder coordinate transform.

Parameters

Structural constraints

Geometric Primitives
Hierarchical Modelling

- Articulated Figures
  - Each node represents the geometry, rotation parameters and structural transformations.
  - Root can be anywhere – here it is at the hip.
  - A realistic human is much more complex
  - Difficult to control so many DoF’s (later problem)

- Directed Acyclic Graph
  - Not necessarily a tree, as geometry can be transformed instances of each other

Hierarchical Modelling

- We can model a lot of things this way

- Articulated Figures
  - Character Rigging and skinning

- Wall-E, PIXAR Animation Studios, 2008

- Doing this in OpenGL 2.x and earlier
  - Use the Matrix Stack
  - Current matrix is automatically product of everything already on the stack
  - This is the matrix on top of the stack

- Recursive algorithm
  - Load identity matrix
  - For each internal node
    - Push new matrix into stack
    - Concatenate transformations onto current matrix
    - Recursively descend tree
  - For leaf node
    - Draw the geometry using the current transformation
Hierarchical Modelling

- Doing this in OpenGL
- Using VAO, VBO and shaders

http://www.gamedev.net/articles/article1267.asp