Animations

- Need efficient representation of
  - Model geometry
  - Motion
  - Interactive rendering
Inherent relationship of parts

- Arm – Simple model
- Shoulder moves all the three parts
- Elbow moves everything below it
- Inherent hierarchical relationship
Inherent relationship of parts

Directed Acyclic Tree

Shoulder → Elbow → Wrist

Shoulder → Elbow → Wrist
Dependency

- Any transformation applied to the parent will be undergone by the children
  - Children must be placed appropriately with respect to the parent
- Children may have their own independent movement
  - Not transmitted to the parent
Representing Transformations

- Transformation with respect to the parent
- Transformation to place it appropriately with respect to the parent
Representing Transformations

- Transformation with respect to the parent
- Transformation to place it appropriately with respect to the parent
Assume each part is defined with origin at center
Representing Transformations

- Assume each part is defined with origin at center
  
  $R_w$
Representing Transformations

- Assume each part is defined with origin at center
  - $R_w$
  - $T_{we}$
Representing Transformations

- Assume each part is defined with origin at center
  - $R_w$ - Wrist
  - $T_{we}$ - Wrist
  - $R_E$ - Elbow and Wrist
Representing Transformations

- Assume each part is defined with origin at center
- \( R_w \) - Wrist
- \( T_{we} \) - Wrist
- \( R_e \) - Elbow and Wrist
- \( T_{es} \) - Elbow and Wrist
Representing Transformations

- Assume each part is defined with origin at center
- $R_w$ – Wrist
- $T_{we}$ - Wrist
- $R_e$ – Elbow and Wrist
- $T_{es}$ – Elbow and Wrist
- $R_s$ – Shoulder, elbow and wrist
Representing Transformations

- Assume each part is defined with origin at center
- $R_w$ – Wrist
- $T_{we}$ – Wrist
- $R_e$ – Elbow and Wrist
- $T_{es}$ – Elbow and Wrist
- $R_s$ – Shoulder, elbow and wrist

Wrist: $R_s T_{es} R_e T_{we} R_w$
Elbow: $R_s T_{es} R_e$
Shoulder: $R_s$
Data Structure

- Depth first traversal of the tree
- Push matrix when entering a node
- Pop matrix when leaving a node
- Render the node as you encounter it
- Example
Data Structure

Shoulder - Rs

Elbow - Re

Wrist - Rw

Torso - Rt

Rs Tes Re Tws Rw

Rs Tes Re

Rs
Data Structure

- **Shoulder** - $R_s$
  - $T_{es}$
  - $T_{ts}$
  - Torso - $R_t$

- **Elbow** - $R_e$
  - $T_{we}$

- **Wrist** - $R_w$

$$\begin{array}{c}
R_s \ T_{es} \ R_e \\
R_s
\end{array}$$
Data Structure

Shoulder - Rs

Torso - Rt

Elbow - Re

Wrist - Rw

Rs
Data Structure

Shoulder - $R_s$

Elbow - $R_e$

Wrist - $R_w$

Torso - $R_t$

$T_{es}$

$T_{ts}$

$T_{we}$
Representing Motion

- Keyframes
  - Generate the transformations for key postures
    - Done manually
  - Interpolate everything in between
    - Done automatically