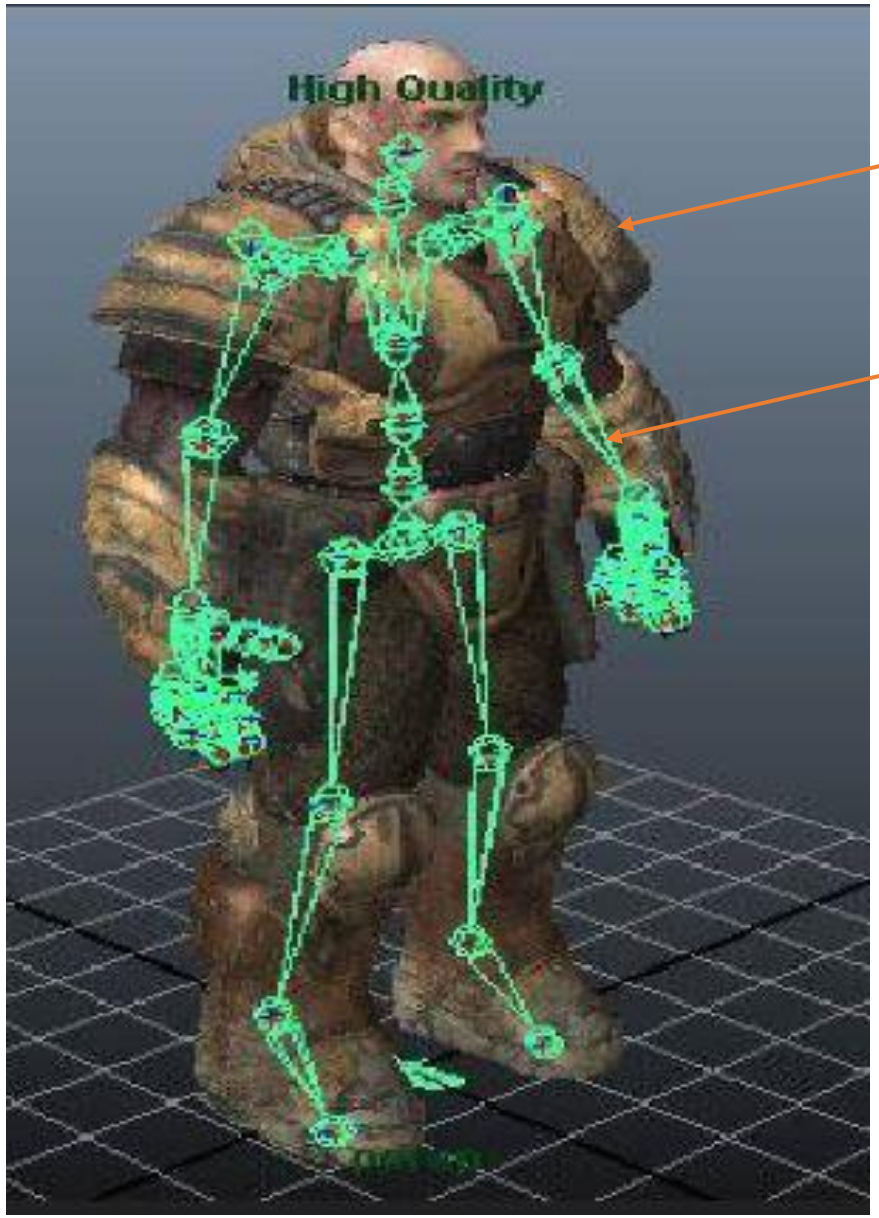
Skeletal Animation

What is Skeletal Animation

- *Skeletal animation or bone-based animation is a technique in computer animation in which the object being animated has two main components:*

    1. *A surface representation used to draw the object (skin, mesh, character)*
    2. *A set of interconnected bones used to animate the surface (skeleton, rig, set of bones).*

- Skeletal animation is based on building a simple skeleton that can fit the important parts of your animated object



A surface representation used to draw the object

A set of interconnected bones used to animate the surface

Terminology

- *Skin:*
  - *The model mesh or the character designed using 3D model tools like Maya, 3D max, Blender etc.*
- *Skeleton:*
  - Also called a rig, A skeleton is a set of bones that can be hierarchical organized.
- *Bone:*
  - *Unit that connects two joints together*
- *Key Frame:*
  - *That holds Next Movement of each joint of the skeleton*
  - A sequence of key frames defines a sequence of movements the viewer will see in the final animation.
- *Rigging:*
  - Defines a skeleton of bones underneath the mesh.

- *Bind Pose Space:* The bind pose is the pose in which the character mesh is modeled (usually a T pose). The Skeleton class manages the bind pose transformations of all bones.

- *Bind Pose Transformation:* A bind pose transformation defines the position, orientation and scale of a bone relative to its parent bone or relative to model space. The Skeleton class manages the bind pose transformations of all bones.

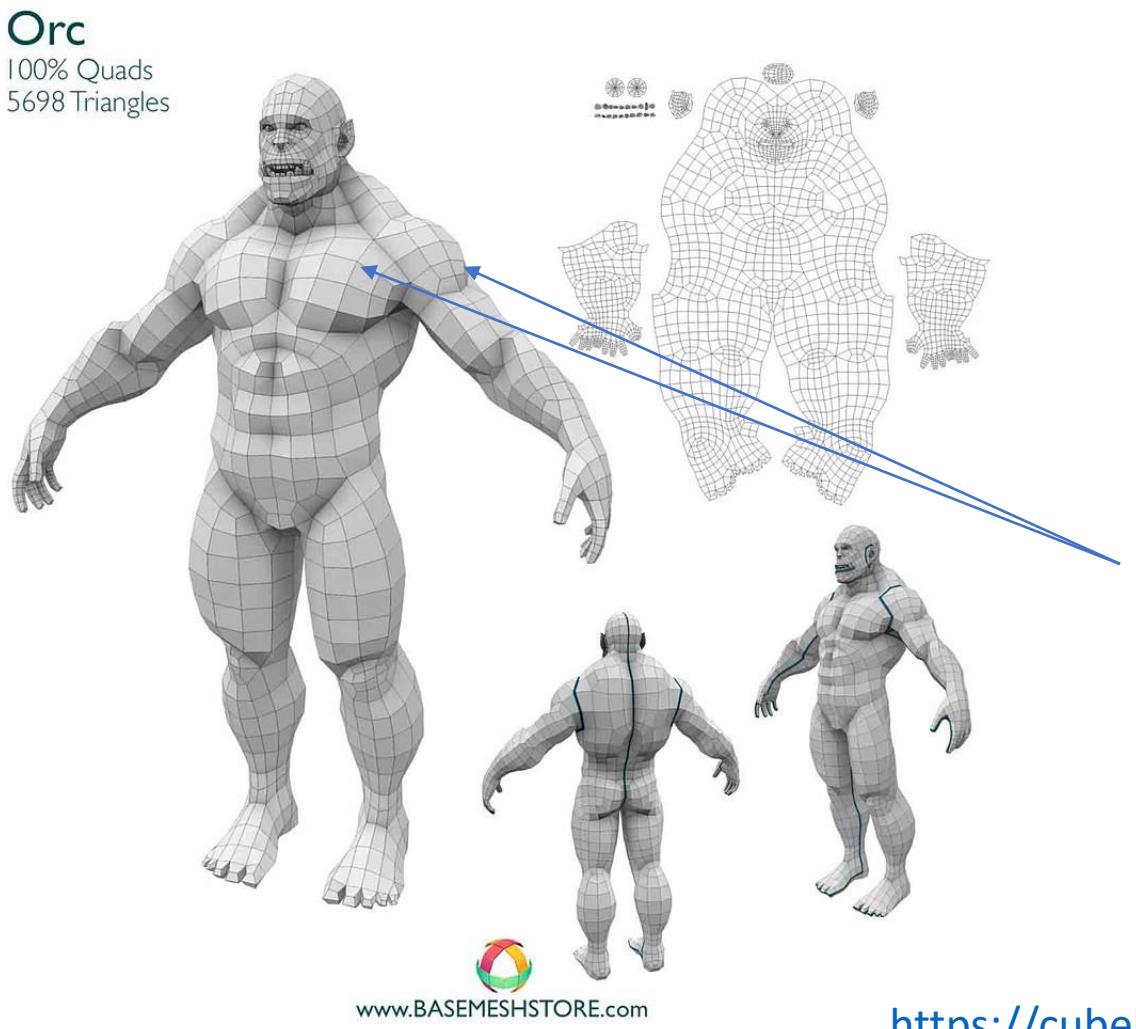
- *Bone Space:* The bone space is the local coordinate space that is fixed on a bone. The origin of this space is the origin of the bone. Sometimes, one coordinate axis points into the direction of the bone. Bone Transforms are always relative to bone space.

- *Parent Bone Space:* Each bone except the root bone of the model has a unique parent bone. The parent bone space is the bone space of the parent bone. The parent bone space of the root bone is the model space.

- *Model Space:* The model space is the local coordinate space of the whole model. The origin of this space is usually at the feet of the character.
- *World Space:* The world space is the coordinate space that contains all game objects. The origin of this space is usually the center of a game level.
- *Bone Transform:* A bone transform is a local transformation in bone space that is applied to a bone. Bones are animated by changing the bone transforms of a skeleton pose.
- *Bone Pose:* A bone pose transformation matrix (or short: bone pose) defines the resulting pose of bone after the bone transforms are applied. A bone pose describes the bone's position, orientation and scale relative to another coordinate space.
  - A relative bone pose describes the pose of a bone relative to the parent bone.
  - An absolute bone pose describes the pose of a bone relative to model space.

- *Skinning Matrix:* Skinning matrices are used to transform the vertices of the character mesh. A skinning matrix transform a vertex position of the bind pose mesh to a vertex position (in model space) of the deformed mesh

Skin (Mesh)



A 'skin' or mesh that represents what the object looks like

Higher the number of polygons leads to a improve the quality of the image

Vertices are the points defined the model mesh in polygon level

Vertices

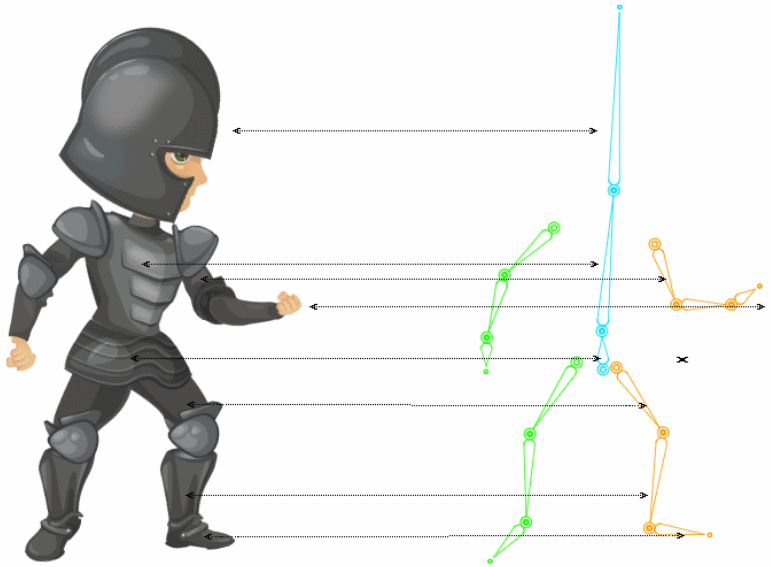
https://cubebrush.co/basemeshstore/products/l8yhiq/orc-base-mesh

Skeleton

- Usually an object has some important joints (vertexes).
- For instance, for a character the most important joints are the hip, the knee, the neck, etc.
- The bone controls the final Movement

https://marionettestudio.com/skeletal-animation/

Rigging

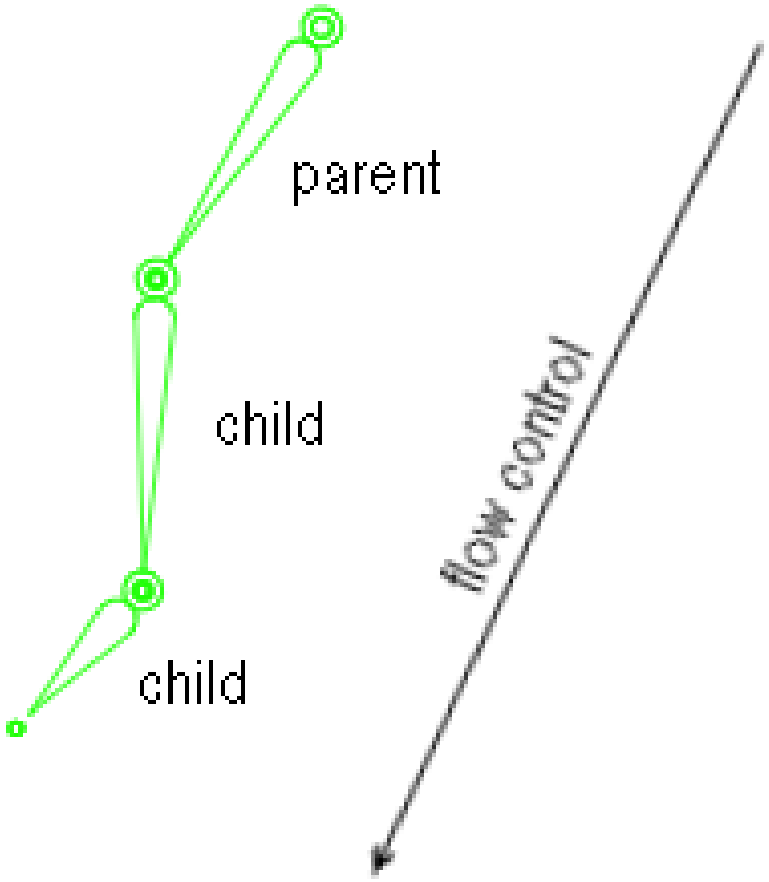


The mesh represents the skin of the object
(can be human, monster or whatever) and the bones are used to move the mesh in a way that would mimic actual movement in the real world.

This is done by assigning each vertex to one or more bones.

https://marionettestudio.com/skeletal-animation/

Bone Hierarchy



- A bone belongs to a parent, and each bone can have a child
- The first joint is called the ***root bone***.
- Every subsequent bone will be connected to the root bone either directly, or indirectly through another

Animation Cycle

- A sequence of keyframes defines a series of movements the viewer will see in the final animation

- An animation cycle contains all important keyframes that are needed in order to get a smooth transition between the last and the first keyframes.

How it Works

- Skeletal animation works by creating 'transformation matrices' which dictate how each bone in the skeleton moves depending on the frame of animation.

- Each vertex on the mesh is assigned a 'weighted' value per bone

  ex: *x number of bones in a skeleton would mean X weighted values per vertex.*

vertex structure

The bone IDs are indices into an array of bone transformations.

Usually, we would interpolate between animation key frames and update the array of bone transformations in every frame.

Each weight signifies how much the vertex is affected by changes in the corresponding bone.

Ex: A vertex in a leg might have a weighted value of .8 for a leg bone, .3 for a hip bone, and a weighted value of 0 for a neck bone, since the neck moves independently from the leg.

Applying Transformation

- Steps

1. Move the vertex to bone space
2. Apply bones transformation Matrix
3. Scale the transformation effect how

    Ex: how much of a bone movement influenced the vertex (weight measure)

4. Repeat the process for all the bones adding up all the values

Calculation

- $X_i(0)$ - The initial vertex position loaded from obj file.
- $M_j(0)$ - The bind matrix of j^{th} bone
- $M_i(k)$ - The animation transformation matrix of the j^{th} bone at frame number k
- w_{ij} - The skinning weight of the vertex j^{th} bone
- $x_i(k)$ - The resulting position of the skinned vertex

$$x_i(k) = \sum_{j=0}^{17} w_{ij} M_j(k)(M_j(0))^{-1} x_i(0),$$

Timer

- A single animation object represents a sequence of animation frames such as "walk", "run", "shoot", etc.
- By interpolating between the frames we get the desired visual effect which matches the name of the animation.
- An animation has a duration in ticks and the number of ticks per second (e.g 100 ticks and 25 ticks per second represent a 4 second animation) which help us time the progression so that the animation will look the same on every hardware.

Advantages

1. Bone-based animations are simpler when you want to make multiple animations for one character.

2. Skeletal animation requires a fewer image and less memory.

3. Skeletal animation allows you to create a natural movement with the help of inverse or forward kinematics.

Disadvantages

1. Skeletal animations are more complex.
2. Skeletal animations require more processor time.
3. Skeletal animations have the impediment of bone rigidity.

Examples and Sample code

- https://threejs.org/examples/webgl_animation_skinning_blending.html
- https://www.3dgep.com/gpu-skinning-of-md5-models-in-opengl-and-cg/
- https://developer.nvidia.com/gpugems/GPUGems/gpugems_ch04.html
- Research