3D Software Toolsets

RIGGING

Intro to 3D Modelling | Gaming development

Rigging

Rigging is the process of creating a digital skeleton and control system that allows characters and objects to be animated. It acts as the bridge between modelling and animation, allowing animators to manipulate the model in a realistic and controllable way. The skeleton is a system of interconnected joints which can also be known as a rig. Each rig contains what we call bones and joints that have their own position, rotation and scale values which can determine the movements of the connected parts.

The skeleton also includes a set of controllers or controls that allows the animator to easily manipulate the model. These controllers are also known as handles, sliders and buttons. These controls are usually placed around the outside of the rig for easy access.

Why do we need to rig?

Rigging is the bridge between the modelling process and animating process. A rig allows animators to make characters and objects bend, stretch and twist using real world physics creating realism within games.

Specifically in relation to characters, being able to manipulate and add facial expression, gestures and body language allows animators to create complex emptions allowing developers to add a visual aspect to the narrative of the game.

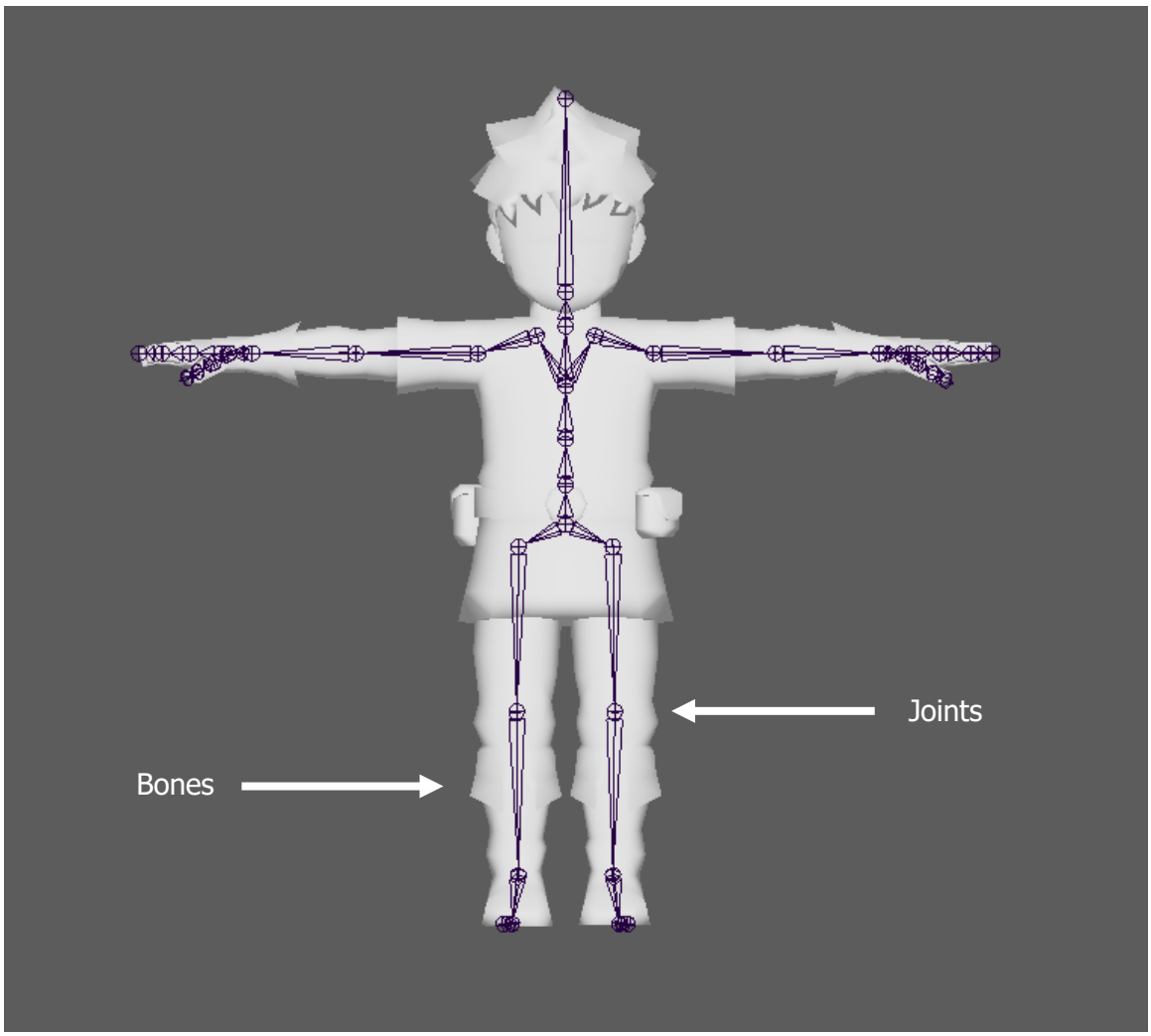
Rigging allows developers to create more engagement within the game.

Rigging Pipeline

Joint Placement:

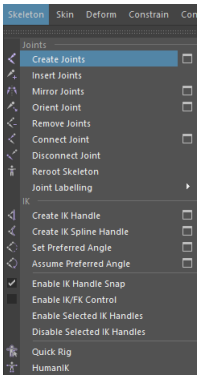
Creates a skeletal structure that defines movement and deformation points. Joints are the foundation of rigging. They function like real bones, defining how different parts of a model move relative to each other. They are crucial for establishing hierarchical relationships e.g., moving the upper arm affects the forearm, hand, and fingers. The placement of joints dictates how well the mesh deforms. Bad joint placement = bad deformation (e.g., if an elbow joint is misaligned, the arm won't bend correctly). Joints also drive deformation in a smooth, mathematically interpolated way, allowing for flexible bending and twisting of a mesh.

Think of joints as the puppet strings that players never see, but the character can't move without them.

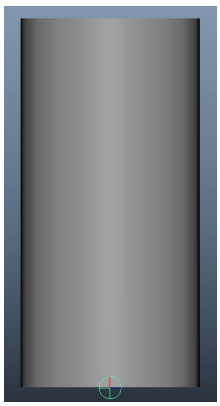


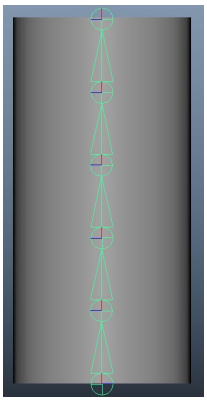
How to Create Joints

1. Go to Skeleton > Create Joints

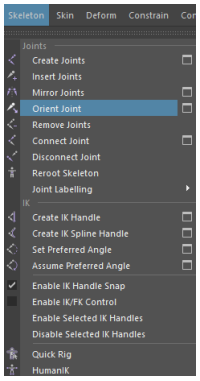


2. In the viewport, click to place joints sequentially
3. Use the orthographic views (Side & Front) for accurate placement





4. Once all joints are placed, press Enter to complete the joint chain.
5. Use the Outliner to Name your Joints according to the correct naming conventions.
6. Go to Skeleton > Orient Joint to ensure correct rotation axis, especially for animation.

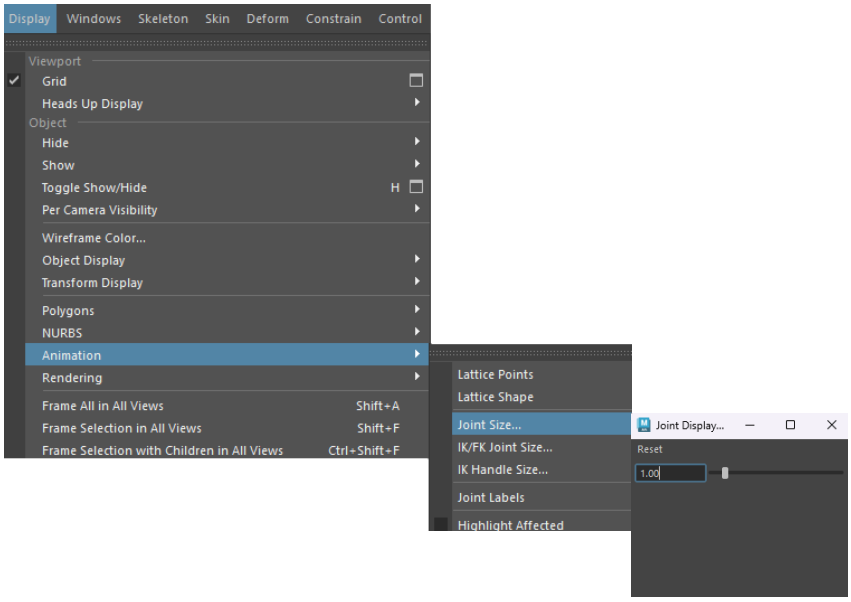


How to Change Joint Display Size

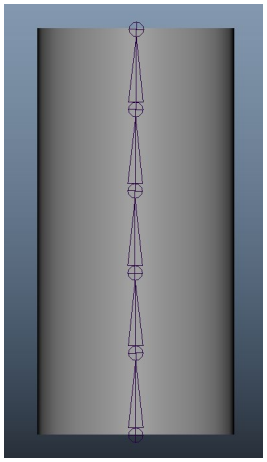
Never scale joints directly using the Scale Tool as it can break your rig, especially if it's skinned or constrained.

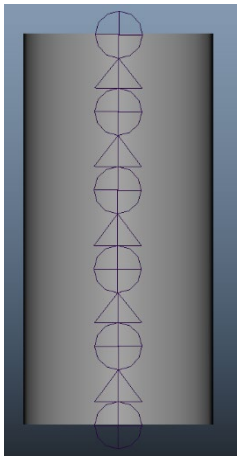
This does not affect animation or rigging, only how joints appear onscreen.

1. Go to Display > Animation > Joint Size



2. Or open the Attribute Editor with a joint selected:
    A. Find the "Display" section
    B. Adjust the "Joint Size" slider





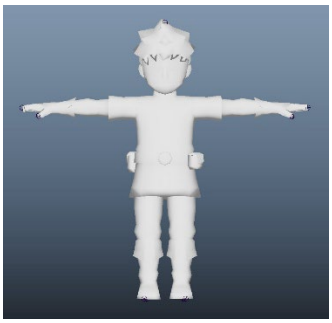
X-Ray Joint

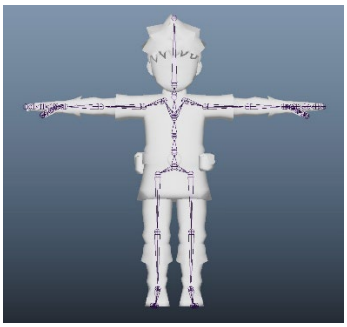
X-Ray Joints lets you see joints through geometry, making it easier

to place and adjust them inside the mesh, when joints are hidden

behind geometry

1. In the Viewport, click Shading (top menu of the panel).
2. Select X-Ray Joints from the dropdown.





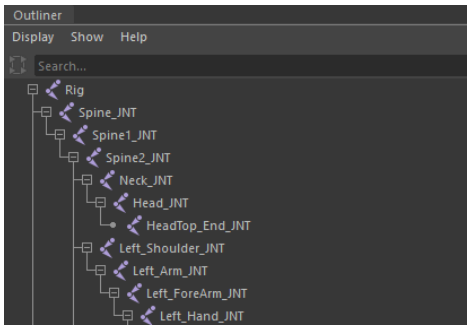
Parenting and Unparenting Joints

Parenting connects joints in a hierarchy, allowing motion to be passed down the chain.

1. Select the child joint first, then Shift + select the parent joint
2. Press P on the keyboard – or – use Edit > Parent

Unparenting disconnects the joints in a hierarchy, allowing you to edit
or reorganise the chain.

1. Select the joint to remove (If you want to disconnect the parent joint, select the child joint)
2. Press Shift + P - or - use Edit > Unparent



Controllers:

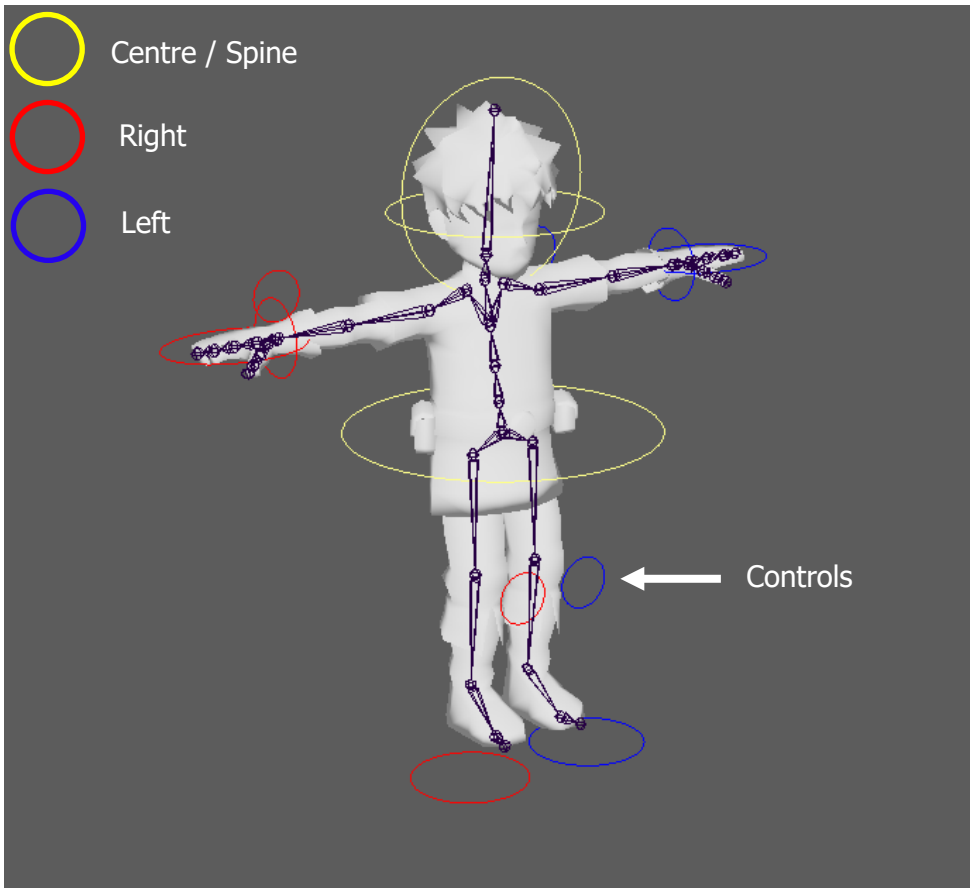
Provides a clean, intuitive, and animator-friendly interface to manipulate the character. Controllers abstract away the complexity of directly animating joints or mesh. They act like "remote controls" for the underlying skeleton. They are visually distinct and easy to select in the viewport often color-coded and shaped to represent the body part they control (e.g., circle for wrist, arrow for direction).

Controllers also offer custom attributes (like sliders or switches) that give animators fast access to features like:

- Finger curls
- Facial expressions
- IK/FK blending

Without controllers, animators would be forced to animate raw joints, which is not only tedious but increases the chance of breaking the rig.

Controllers are like the steering wheel and pedals in a car, animators don't touch the engine (joints), they just drive the character.

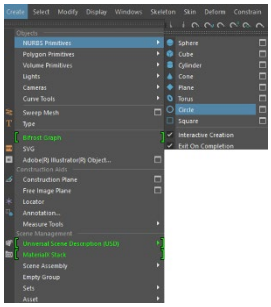


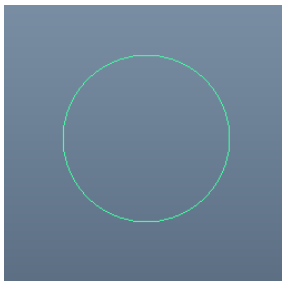
How to Create a Controller

Controllers are typically used to allow animators to control the movement of joints, bones, or other parts of a character model, without directly manipulating the joints themselves.

These controllers are often custom objects, like NURBS curves or custom shapes, that are easy to grab and manipulate.

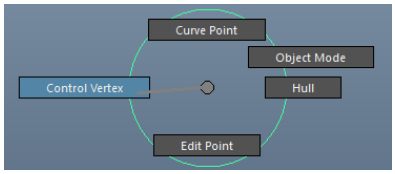
1. Go to Create > NURBS Primitive > Circle
2. Draw a circle in the shape of the controller you want.



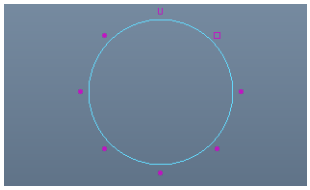


How to Edit the Shape if a Controller in Maya

1. Use the Scale Tool (shortcut R) to adjust the size of the controller.
2. To get more control over the shape, you can edit the curve's CVs (Control Vertices).
    A. Right-click the circle and select Control Vertex from the marking menu.

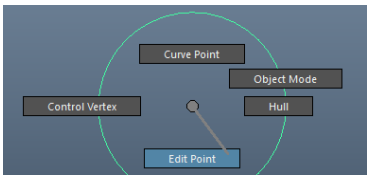


- This will display the CVs as small points along the curve.

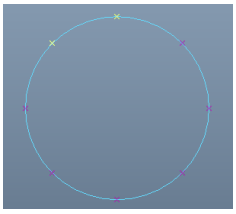


    B. Use the Move Tool (W) to adjust individual CVs, altering the shape of the NURBS circle.

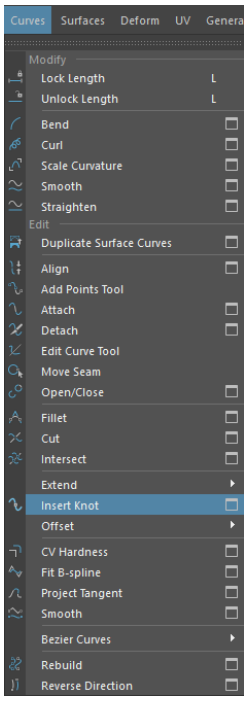
C. You can add more CVs if needed by selecting the circle, then going to edit point.

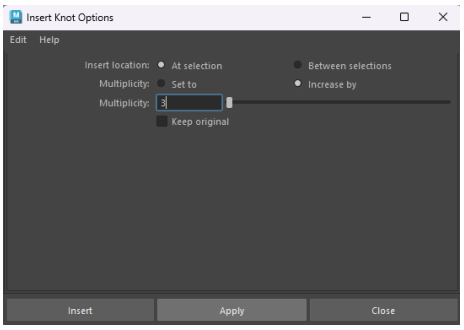


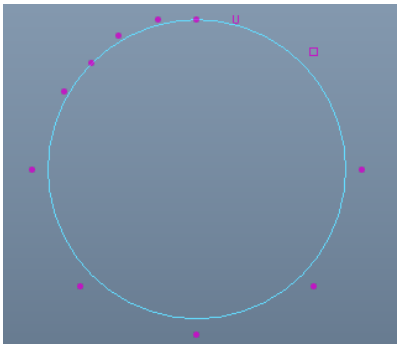
D. Select the points at which you want to insert new points between



E. Curves > Insert Knot Options Box.
- Insert location: Between Selections
- # Knots to insert: and change the slider to the desired amount

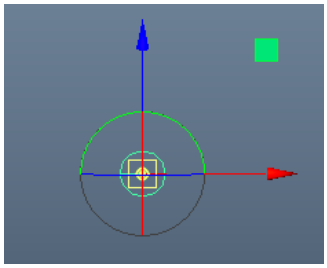




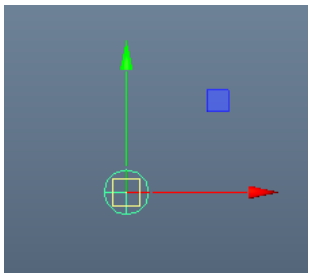


How to Edit the Pivot Point

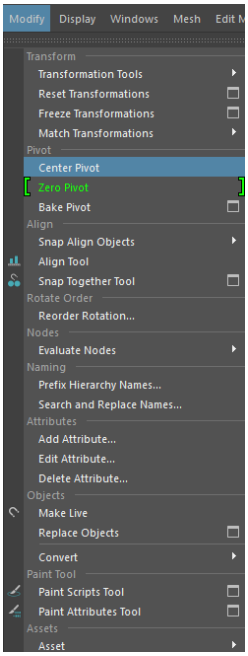
1. Press and hold the D key on your keyboard to toggle on and off the Pivot Editing Mode.
2. You can then move the pivot point around freely.



3. Make sure to toggle Pivot Editing Mode off when done.

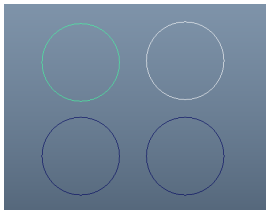


4. To reset a pivot point to the center of an object.
5. Go to Modify > Center Pivot.
6. This will move the pivot point back to the center of the object's bounding box.

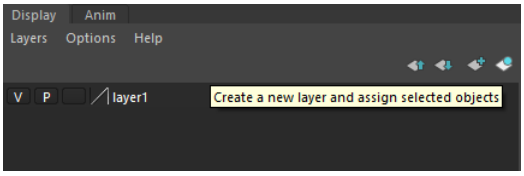


How to Colour Controllers

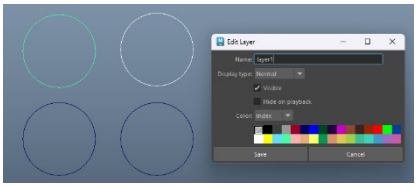
1. Select your controllers.



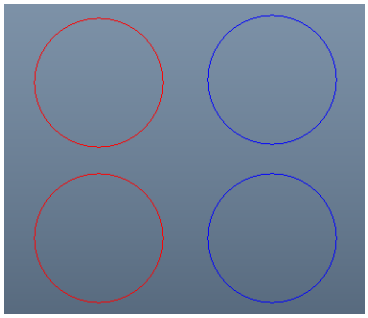
2. In the Channel Box, click the Layers tab.
3. Click Create a new layer and assign selected objects.

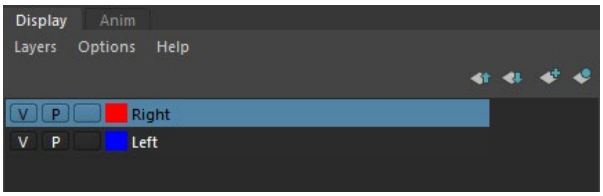


4. Double-click the new layer to open its settings.



5. Select a colour to represent the type of controller.
6. Red for Right
7. Blue for Left
8. Yellow or Green for Center





Inverse Kinematics - IK

Inverse Kinematics (IK) is a rigging system where you move the end joint, and the joints above it in the chain automatically adjust to follow.

1. Go to Skeleton > Create IK Handle Tool.
    A. Click the start joint (e.g., hip), then the end joint (e.g., ankle).
    B. Maya creates an IK handle that controls the entire joint chain.
2. Create a Control Curve for the IK Handle.
    A. Freeze Transformations on Controllers.
3. Parent the IK Handle to the Control.
4. Create another controller.
    A. Freeze Transformations on Controllers.
    B. Position for the middle joint.
5. Add a Pole Vector for direction
    A. Select the pole vector controller, then the IK handle.
    B. Go to Constrain > Pole Vector to add a Pole Vector for controlling the middle joints direction.

Forward Kinematics - FK

Forward Kinematics (FK) is a method of animating where you rotate each joint from the top of a hierarchy downward. e.g., shoulder → elbow → wrist to pose a limb.
It gives animators full control over the rotation of each joint, making it's ideal for natural arcs and motion like swinging arms or tails.

1. Create NURBS Controllers.
2. Freeze Transformations on Controllers.
3. Parent Controllers Hierarchically.
4. Constrain Joints to Controllers.

IK Spline

IK Spline is a special type of Inverse Kinematics used for animating flexible, joint-based objects like spines, tails, necks, snakes, or ropes. It uses a curve to control the bend and twist of a joint chain, offering smooth and natural deformation.

1. Create the IK Spline Handle.
2. Go to Skeleton > Create IK Handle Tool.
3. Click the start joint and then the end joint.
4. Maya will automatically generate:
    A. An IK Handle
    B. A curve that controls the bending
5. Hide the joints and IK in the scene view to access the Spline curve.
    A. Create Controllers for the Curve.
    B. Select a CV or group of CVs.
    C. Go to Deform > Cluster.
    D. Then parent the cluster to a controller.

Constraints:

Creates automated relationships between controllers and joints, or between different parts of the rig.

Constraints allow for non-destructive, flexible connections. Instead of parenting (which locks down hierarchy), constraints let you define how much influence something has (e.g., 50% from one control, 50% from another).

They also allow for automation:

- Eyes that always aim at a look target (Aim Constraint).
- Hands that follow a moving object (Parent Constraint).
- Elbows that stay pointed correctly during IK movement (Pole Vector Constraint).

Constraints are often key to rig modularity you can plug and unplug pieces of the rig more easily with them than with parenting alone.

Think of constraints like invisible wires that connect systems together and automate behaviour behind the scenes.

Parent Constraint

A Parent Constraint connects one object's position and rotation to another, like making a controller drive a joint. Unlike regular parenting, a constraint can be turned on/off, blended, or removed without affecting the hierarchy.

1. Select the Driver First (Parent)
    A. This is the object that will control the other (e.g., a controller curve).
2. Select the Driven Second (Child)
    A. This is the object that will follow (e.g., a joint or another object).
3. Apply the Constraint
    A. Go to Animation Menu Set > Constrain > Parent.
    B. In the Options Box:
        - Maintain Offset ON: Keeps the driven object in its current position.
        - Maintain Offset OFF: Snaps it to the driver's position and rotation.
    C. Click Apply.

Point Constraint

A Point Constraint makes one object follow the position (translation) of another, while ignoring rotation and scale. This is useful when you want something to move with another object but not rotate with it.

1. Select the Driver First
    A. This is the object you want to follow (e.g., a controller).
2. Select the Driven Object Second
    A. This is the object that should follow the position (e.g., a joint, mesh, locator).
3. Apply the Constraint
4. Go to Animation Menu Set > Constrain > Point.
    A. In the Options Box:
        - Maintain Offset ON: Keeps the driven object in its current position.
        - Maintain Offset OFF: Snaps the driven object to the driver's position.
    B. Click Apply.

Orient Constraint

An Orient Constraint makes one object follow the rotation of another, without affecting its position or scale. It's perfect when you want an object to rotate with another, but stay in its own location, such as a joint rotating with a control.

1. Select the Driver First
    A. This is the object with the rotation you want to follow (e.g., a controller).
2. Select the Driven Object Second
    A. This is the object that will copy the rotation (e.g., a joint or mesh).
3. Apply the Constraint
4. Go to Animation Menu Set > Constrain > Orient.
    A. In the Options Box:
        - Maintain Offset ON: Keeps the driven object's current orientation.
        - Maintain Offset OFF: Snaps its rotation to match the driver.
    B. Click Apply.

Aim Constraint

An Aim Constraint forces an object to always rotate to "look at" another object, like a camera tracking a target, or a character's eyes always following something. It adjusts rotation only, based on a specified aim axis.

1. Select the Target Object First
    A. This is the object the constrained object will point at (e.g., a controller or locator).
2. Select the Object to Aim Second
    A. This is the object that will rotate to face the target (e.g., a joint, camera, eye control).
3. Apply the Constraint

A. Go to Animation Menu Set > Constrain > Aim.
B. In the Options Box, configure:
    - Aim Vector: Axis that should point toward the target (e.g., X = 1, 0, 0 if X should aim).
    - Up Vector: Axis used to keep the object from twisting incorrectly (typically Y or Z).
    - World Up Type: Determines how the up direction is calculated:
    - Common option: Object Up (use a second object to control up orientation).
C. Click Apply.

Pole Vector Constraint

A Pole Vector Constraint is used with an IK handle (especially for limbs like arms or legs) to control the elbow or knee direction. It ensures the IK joint chain bends in a predictable, controllable direction.

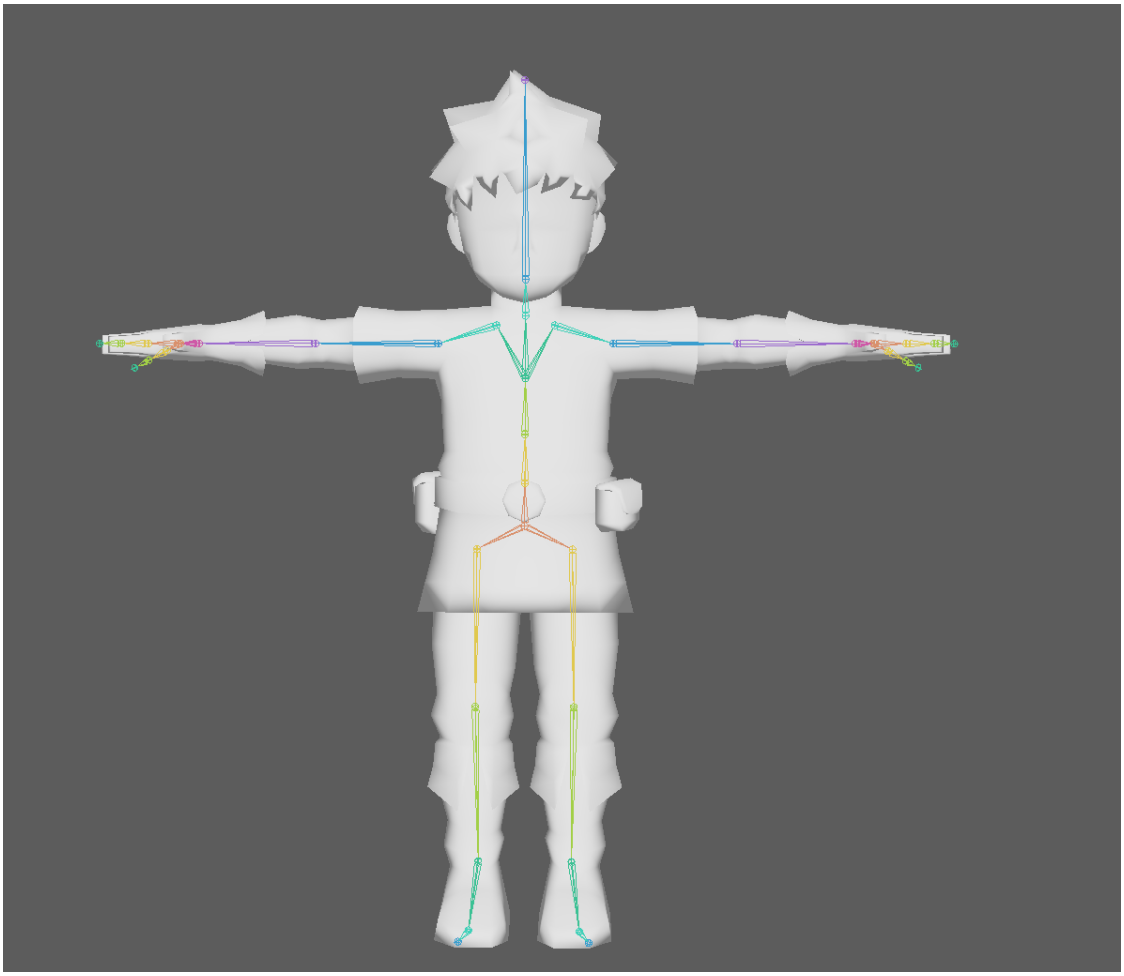
1. Create a Pole Vector Control
    A. Place it in front of the elbow (in the direction the elbow or knee should point).
2. Select the Pole Vector Control First
    A. This is the driver or parent, e.g. the NURBS circle.
3. Then Select the IK Handle
    A. This is the driven or child, e.g. ikHandle1 for the arm or leg.
4. Apply the Pole Vector Constraint
    A. Go to Animation Menu Set > Constrain > Pole Vector.

Skinning:

Attach's the 3D mesh (the character's visible body) to the skeleton so it deforms with joint movement.

Skinning tells Maya how the skeleton affects the mesh. Without skinning, the mesh doesn't move with the joints at all. When a joint moves, it pulls the mesh around it based on its assigned weight. The binding step starts this process, creating a connection between the skin (geometry) and the skeleton (joints). At this point, your rig "comes alive" for the first time: moving a joint now moves the character's body.

Binding is like sewing the puppet's fabric skin onto the wooden skeleton, it makes the mesh move and bend.



How to Bind

Skin Binding connects your 3D model (geometry) to a skeleton (joints), allowing the mesh to deform when the joints move, essential for character rigging and animation.

1. Select the mesh of your asset and then shift select the root joint.
2. Go to your tool bar > Skin > Bind Skin > Option Box
    A. Set the Bind Options
    B. Bind to:
    C. Joint Hierarchy (usually)
    D. Bind method:
        - Closest Distance: Assigns joints to vertices based on distance.
        - Closest in Hierarchy: Assigns joints to vertices based on both distance and joint hierarchy.
        - Heat Map: Assigns joints to vertices based on a heat diffusion algorithm.
        - Geodesic Voxel: Assigns joints to vertices based on voxel-based analysis.
    E. Skinning Method:
        - Classic Linear: The position of each vertex is calculated as a weighted average of the positions of the influencing joints.
        - Dual quaternion: Uses quaternion math instead of linear math to rotate vertices, preserving the original shape better.
        - Weight blended: Automatically chooses which method to use per vertex based on the motion and rotation.
    F. Normalize Weights:
        - Interactive: Easier for manual painting.
        - Post: Harder to control manually.
    G. Max Influences: Number of joints affecting a vertex (e.g., 4)
        - Maintain Max Influences: Keep it consistent
3. Click "Bind Skin"

Weight Painting:

Refines how much influence each joint has on the surrounding mesh to ensure natural, clean deformation.

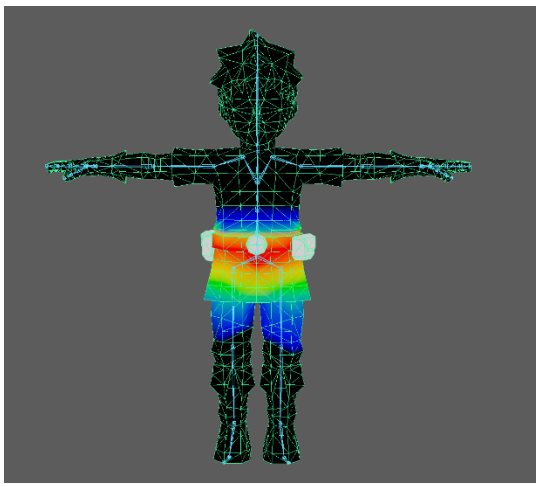
After skinning, the mesh often deforms poorly, especially at joints with complex motion-like shoulders or hips. Weight painting fixes this by letting the rigger define how strongly each joint affects each vertex.

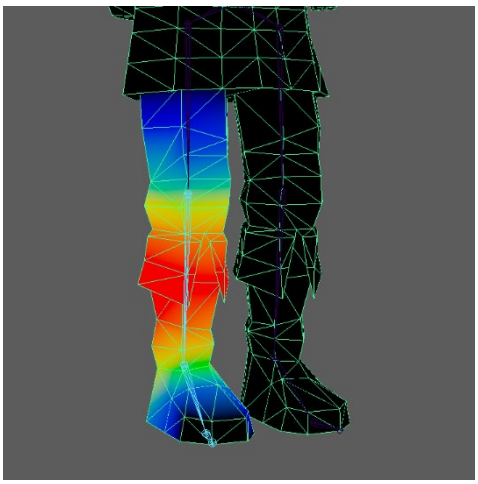
For example:

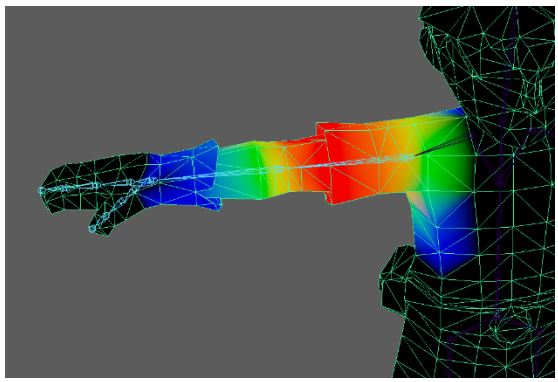
- Elbow joint should mostly affect the middle of the arm.
- Shoulder joint should gradually blend into the chest and upper arm.

Without proper weight painting, you get ugly creases, stretching, or collapsing during animation. You can visualize this as a gradient of influence black (no influence) to white (full control) or this gradient can be changed to a rainbow gradient.

Weight painting is like fine-tuning the strings on a puppet ensuring that every part bends and moves just right.



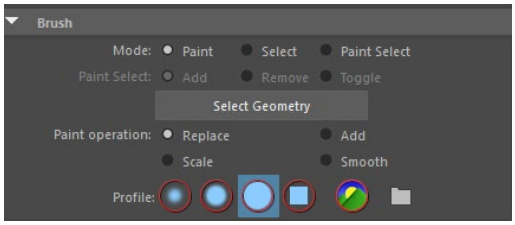


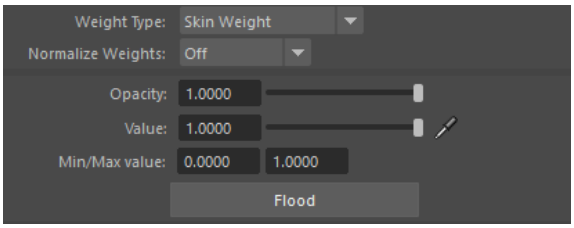


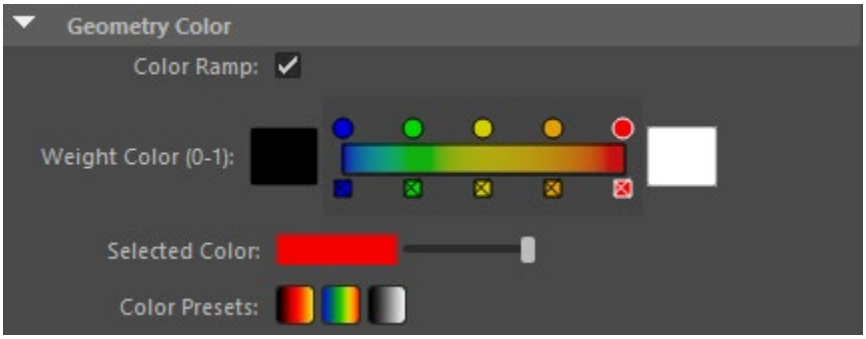
How to Weight Paint

Weight Painting is the process of adjusting how much influence each joint has over each vertex of a skinned mesh. It's essential for clean deformations in animation.

1. Select your skinned mesh.
2. Go to Skin > Paint Skin Weights Tool > ❐ (Options).
    A. Painting Modes
        - Add: Adds influence (weight) to selected joint
        - Replace: Replaces current weights with a set value
        - Smooth: Blends weight between neighboring joints
    B. Common Settings
        - Value: The strength (0–1) to paint with
        - Opacity: How strongly to apply the value
        - Flood: Applies the current brush action to the full mesh
    C. Geometry Colour
        - Color Ramp: Gives you visual feedback about how much a joint is influencing different parts of your model.
        - Weight Color(0-1): These colors help you see how much influence each joint has on specific parts of the mesh.
        - Color Presets: Maya includes preset colour ramps to change how skin weights look visually.







Animating:

Uses the rig to create believable motion, performance, and storytelling through keyframes and posing. Once the rig is complete, animators can now manipulate the controllers to bring the character to life. Animation is about posing over time placing a character in key positions and letting Maya interpolate the movement between.

A good rig makes animating:

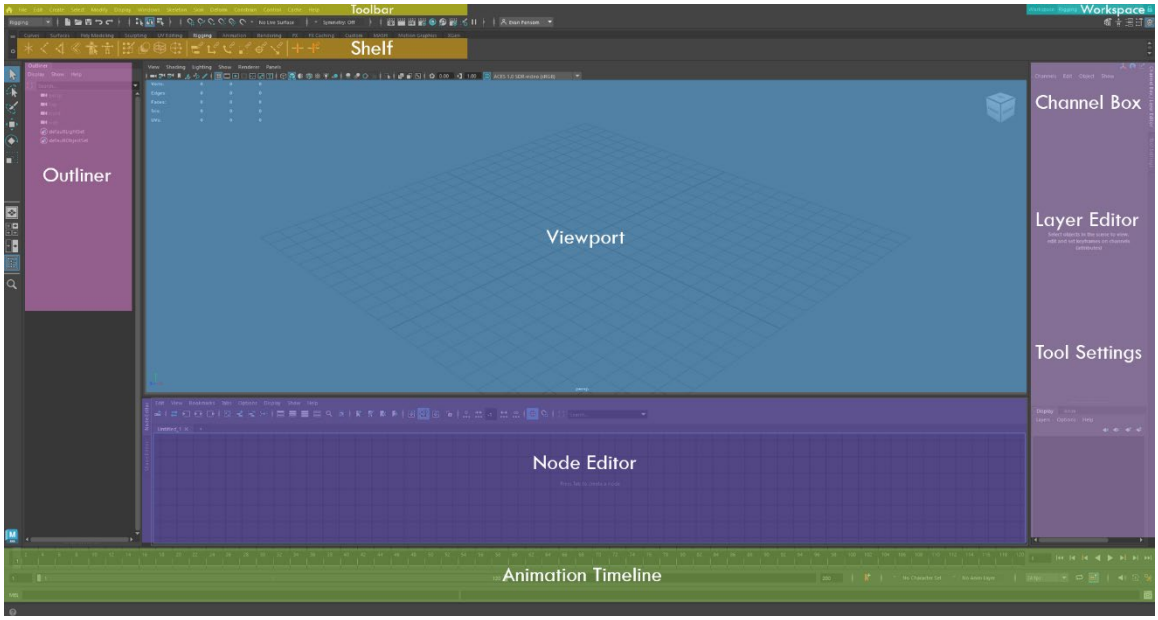
- Faster – fewer clicks, more control.
- Cleaner – better deformation and control = fewer visual issues.
- More expressive – with facial rigs, hand controls, squash/stretch, etc.

Without a rig, animators would be forced to move raw geometry or joints, which would be unmanageable, error-prone, and time-consuming.

Animation is the goal of the rigging pipeline, everything builds up to this creative act of bringing characters to life.

- Timeline: The bar at the bottom showing frames.
- Keyframes: Markers that store values at a certain time.
- Auto Key: Automatically sets keyframes when you move an object.
- Set Key Tool (S Key): Manually adds a keyframe for the selected object/attribute.
- Graph Editor: Fine-tune animation curves for smoother and more realistic motion.

The Rigging Workspace



Activities

Activity 1: Rig Basic Objects

Practice creating rigs on simple objects.

END OF WORKBOOK