Norman v1.0

(For Maya 6.0 or greater)

Created by:

Leif Jeffers
Morgan Loomis
Peter Starostin
Neal Thibodeaux

Table Of Contents

Installation Instructions:

1. Unzip file.
2. Open your maya directory
3. Place script mlAutoIKFK.mel into your scripts folder. (This is your FK/IK match script)
4. Place script mlPickWalk.mel into your scripts folder.
5. Place script mlCtrlPickWalk.mel into your scripts folder.
6. Place ikfk.bmp into your icon folder.

IK/FK MATCH

How it works: Select an Arm or leg, and click the IK/FK match script. It will automatically switch to the opposite of what it currently is and match it. For example if you select an FK arm it will automatically switch to a matched IK, and vice versa. Works on both legs and arms.

1. Add mlAutoIKFK.mel to a shelf or hotkey.
   - To add it to a shelf, simply type "mlAutoIKFK ;" (without the quotes) into the script editor, highlight the text, and middle mouse drag it to your shelf. Open your shelf editor, highlight "mlAutoIKFK ;" and click on change image. Browse and select ikfk.bmp from your icon folder. Click save all shelves.
   - To add it to a hotkey, simply go to Window>Settings/Preferences>Hotkeys. Scroll to the bottom category named "User" and select it. Click on "new" and name it "ikfkMatch". Type "mlAutoIKFK ;" into the command line (again remember, do not add the quotes to anything I have quoted.) Click on accept. Highlight ikfkMatch in the Commands column. Go to assign new hotkey, and select a hotkey you would like to use. After you have finished click on "Assign." Save and exit.

mlPickWalk

This script will allow you to navigate the controls using the arrow keys.

For example: Select one of the finger tweak controls. Hit the up arrow and you navigate up and down the finger. Hit left and right to select the corresponding joints of neighboring fingers. Hit ctrl+down the select all the joints in the finger. Hit ctrl+left to select all the parallel joints (for example, if you have the tip of one finger selected, ctrl+left with select all the finger tips.) ctrl+right selects the mirror, and ctrl+up selects all controls of the selected type.

2. Add mlPickWalk.mel commands to a hotkey (follow the instructions from the IK/FK match if you need help on how to do this.). **This one works a little differently. The commands that must be put into the hot keys are as follows:**
    - mlPickWalk u
    - mlPickWalk d
    - mlPickWalk l
    - mlPickWalk r
    - mlCtrlPickWalk u
    - mlCtrlPickWalk d
    - mlCtrlPickWalk l
    - mlCtrlPickWalk r

Set these hotkeys to overwrite your arrow key hotkeys as follows:

- mlPickWalk u = Up arrow
- mlPickWalk d = Down Arrow
- mlPickWalk l = Left Arrow
- mlPickWalk r = Right Arrow
- mlCtrlPickWalk u = Up arrow + Ctrl
- mlCtrlPickWalk d = Down arrow + Ctrl
- mlCtrlPickWalk l= Left arrow + Ctrl
- mlCtrlPickWalk r = Right arrow + Ctrl

You're ready to animate with Norman. Have fun!!!

Norman v1.0

Ok so I will try to cover the basics of Norman. When you first open the file you need to click on the green curve labeled “viz control” by his feet. This will open your control visibility options in the attribute editor. Most of them are self-explanatory but I will cover them just to be safe.

Visibility of Geometry: 0= off , 1=on

Visibility of Controls: 0= off , 1=on

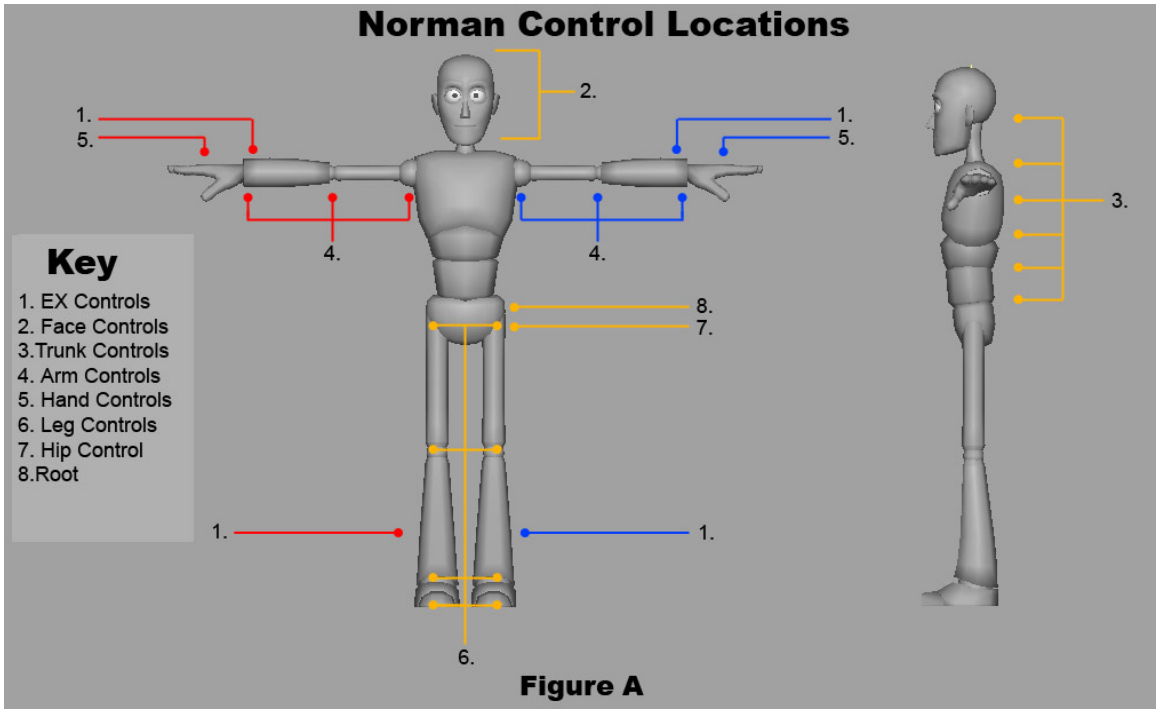


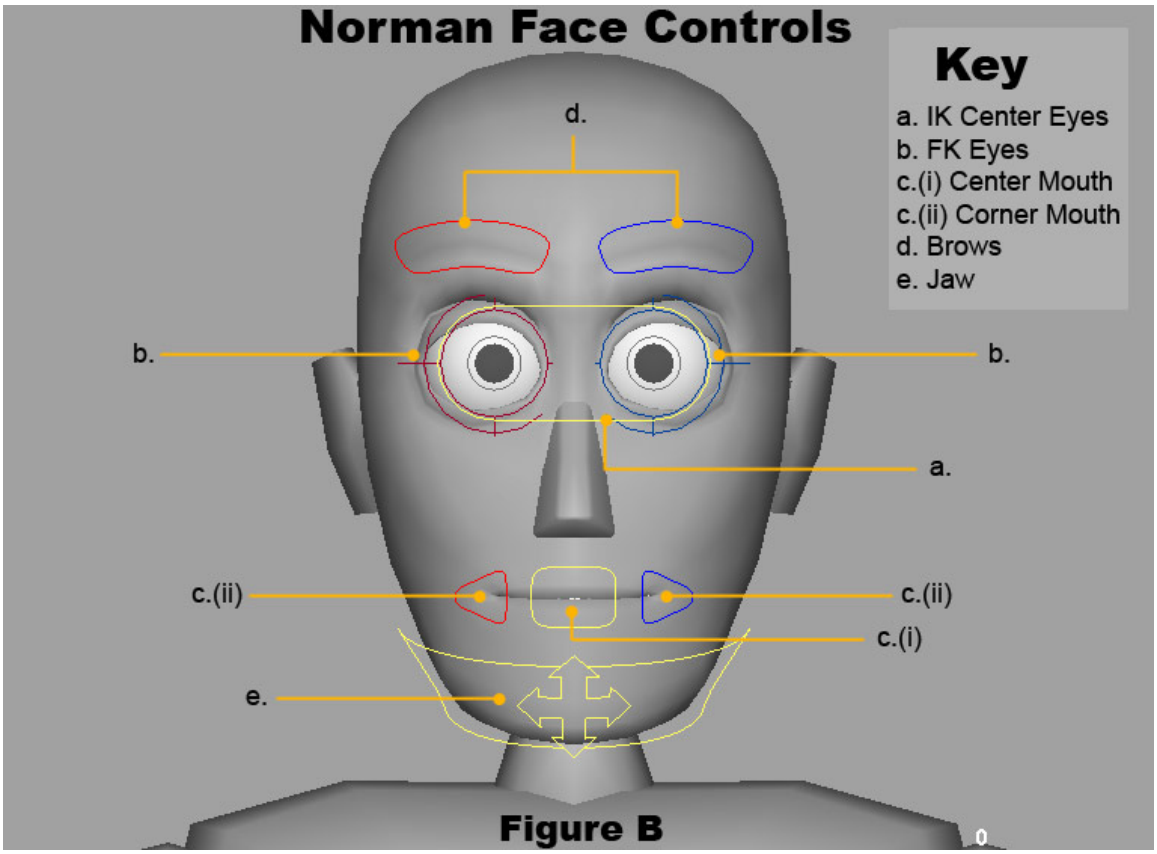
Figure A

1. **All Extra**: (only visible if the arm/leg control visibility is turned on.) It is a green curve that says "EX." It is located by the wrist and/or foot. There are four "EX" controls, one for each limb. This allows access to the following attributes:

   a. FK IK:
      i. This is your IK/FK switch control
   b. Stretchy IK:
      i. This allows you to stretch the limb while in IK mode. If you pull the IK away from the character it will stretch the limb beyond its normal proportions.
   c. Upper Stretch:
      i. Will stretch the upper limb in either IK or FK mode. If in IK mode it will stretch the upper limb, yet not change the position of the IK control itself.
   d. Lower Stretch:
      i. Will stretch the lower limb in either IK or FK mode. If in IK mode it will stretch the lower limb, yet not change the position of the IK control itself.

e. Upper Squash:
    i. Will uniformly scale the diameter of the upper limb.
f. Lower Squash:
    i. Will uniformly scale the diameter of the lower limb. In the case of the lower arm, **using this attribute will also let you shrink the lower arm's width to match the upper arms width.** Although the current wider lower arm design is the natural state of Norman, you are of course free to do what you see fit for the situation.

2. **Face controls**: (includes eye, mouth, eyebrow, and jaw controls)



a. **IK Center Eye Controls** (yellow oval with two circles inside, located in front of the eyes.)(It is used to aim both eyes at once. The eyes will follow the center eye control. It can also be used in conjunction with FK Eye Controls)
    i. Translate x,y,z:
        1. Translates the eyes together.
    ii. Converge:
        1. Allows you to converge the pupils towards the center of the face. You may use any number between 0 and 1. 0 being completely converged, and 1 being normal.
    iii. Follow:
        1. This controls whether the eyes follow the Center Eye Control. **0=off 1=on**
    iv. Individual Ctrl:
        1. Allows for individual control of the eyes with the aim constraints. **0=off 1=on**
b. **FK Eye Controls** (circles around the eyeball itself)
    i. **Eyeball controls**
        1. Rotate x,y,z:

a. Self-explanatory. This lets you rotate the eyes individually for perfect positioning. Can be used in conjunction with the IK center eye control or independently from it.

2. IO:
    a. Allows you to move the position of the eye left or right.
3. UD:
    a. Allows you to move the position of the eye up or down.
4. FB:
    a. Allows you to move the position of the eye forward or back.
5. Twist:
    a. Allows you to twist the eyelid around the eyeball itself.
6. Width:
    a. Allows you to scale the width of the eye.
7. Height:
    a. Allows you to scale the height of the eye.
8. Depth:
    a. Allows you to scale the depth of the eye.

ii. **Eyelid Controls**

1. Up Lid UD:
    a. Allows you to open or close the upper lid.
2. Lo Lid UD:
    a. Allows you to open or close the lower lid.
3. Up Lid TW:
    a. Allows you to rotate the upper lid.
4. Lo Lid TW:
    a. Allows you to rotate the lower lid
5. Lid Follow:
    a. Will make the lids automatically follow the movement of the eyeball. Higher number will make the follow more extreme. Lower number will make the movement subtler. Zero means it is off.
6. Upper LidInner_UD:
    a. Allows up/down control of the inner portion of the upper eyelid.
7. Upper LidMid_UD:
    a. Allows up/down control of the middle portion of the upper eyelid.
8. Upper LidOuter_UD:
    a. Allows up/down control of the outer portion of the upper eyelid.

9. Lower LidInner_UD:
    a. Allows up/down control of the inner portion of the lower eyelid.
10. Lower LidMid_UD:
    a. Allows up/down control of the middle portion of the lower eyelid.
11. Lower LidOuter_UD:
    a. Allows up/down control of the outer portion of the lower eyelid.

c. **Mouth** (the three curves located in the mouth area.)

If not visible when you turn the face controls on; make sure that Face Blendshape is set to 1 in the Tweaks.

i. **ctrlFace_Cn_Lips**

1. Mouth_UD:
    a. Moves the entire mouth up or down.
2. Mouth_LR:
    a. Moves the entire mouth left or right
3. Mouth_Turn:
    a. Rotates the entire mouth clockwise or counter clockwise.
4. UpLip_UD:
    a. Raises and lowers the center upper lip.
5. DnLip_UD:
    a. Raises and lowers the center lower lip.
6. UpLip_CurlIO:
    a. Curls the upper lip in or out.
7. DnLip_CurlIO:
    a. Curls the lower lip in or out.
8. Mouth_PuffIO:
    a. Puffs the lips in or out.
9. Mouth_Clench:
    a. Clenches the upper and lower lips together.
10. Mouth_Pull:
    a. Will pull the upper and lower lips out away from the face.

ii. **ctrlFace_Rt/Lf_Lips**

1. UpLip_UD:
    a. Moves the outside upper lip up or down.
2. DnLip_UD:
    a. Moves the outside lower lip up or down.
3. Crnr_UD:
    a. Moves the corner of the mouth up or down.
4. Crnr_IO:
    a. Moves the corner of the mouth in or out.
5. Mouth_Puff IO:
    a. Puffs the cheek out or sucks the cheek in.

6. UpLip_Part:
    a. Controls whether the upper and lower lip remains closed together at the sides or separate from each other.

d. **Eyebrows** (the curves located over the brow region of the head.)
If not visible when you turn the face controls on; make sure that Face Blendshape is set to 1 in the Tweaks.

i. **ctrlFace_Rt/Lf_Brow**
1. BrowAll_UD:
    a. Moves entire brow up or down.
2. BrowAll_IO:
    a. Moves the entire brow in or out.
3. BrowInner_UD:
    a. Moves inner part of brow up or down.
4. BrowMid_UD:
    a. Moves middle part of brow up or down.
5. BrowOuter_UD:
    a. Moves outer part of brow up or down.

e. **Jaw** (the curve located at the chin region)

i. **CtrlFace_Cn_Jaw**
1. Translate x,y,z:
    a. Translate the jaw.
2. Rotate x,y,z:
    a. Rotate the jaw.
3. Scale x,y,z:
    a. Scale the jaw.

f. **Inner Mouth Controls** (Located to the right side of Normans Face) *If not visible when you turn the face controls on, make sure that inside mouth is set to 1 in the Tweaks*

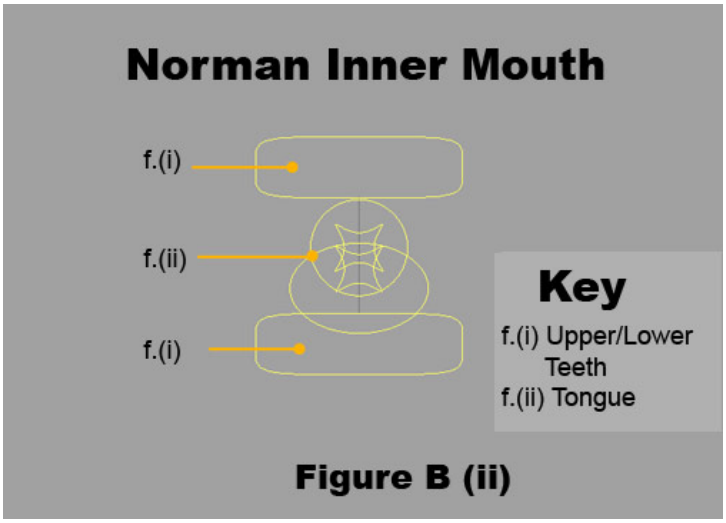


Figure B (ii)

i. **ctrlFace_Cn_TeethUpper/Lower** (Allows you to have individual control over the top and bottom teeth for better mouth shapes.)
    1. Translate x,y,z:
        a. Translate the teeth.
    2. Rotate x,y,z:
        a. Rotate the teeth.
    3. Scale x,y,z:
        a. Scale the teeth.

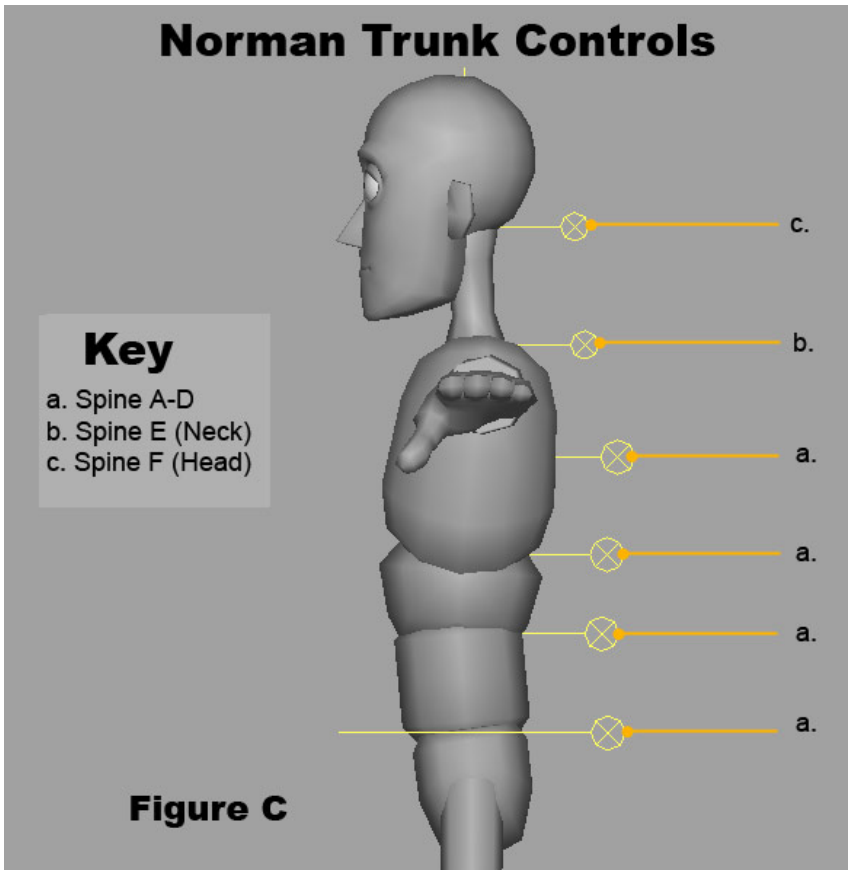
ii. Tongue

1. ctrlFaceFK_All_TongueA (base)
    a. Translate x,y,z:
        i. Translate the tongue.
    b. Rotate x,y,z:
        i. Rotate the tongue.
    c. Width:
        i. Will scale the width of the tongue.
    d. Height:
        i. Will scale the height of the tongue.

2. ctrlFaceFK_Cn_TongueB (middle)
    a. Translate x,y,z:
        i. Translate the tongue.
    b. Rotate x,y,z:
        i. Rotate the tongue.

3. ctrlFaceFK_Cn_TongueC (tip)
    a. Translate x,y,z:
        i. Translate the tongue.
    b. Rotate x,y,z:
        i. Rotate the tongue.

3. **Trunk Controls** (located on the back, neck, clavicle, and head)



a. **ctrlFK_Cn_SpineA-D (Back Controls):** (the curve shaped like a circle with an x in it located on the back of the character) (used for rotational control over the corresponding region of the back.)
    i. Rotate x,y,z:
        1. Rotate the corresponding region of the back.
b. **ctrlFK_Cn_SpineE (Neck Controls):** (the curve shaped like a circle with an x in it located on at the base of the neck)
    i. **General Controls**
        1. Translate x,y,z:
            a. Translate the neck
        2. Rotate x,y,z:
            a. Rotate the neck.
        3. Scale x,y,z:
            a. Scale the neck
    ii. SS Controls (0=off 1=on)
        1. Parent:
            a. Allows the neck to follow the rotations of the torso and root.
        2. World:

a. Allows the neck to remain in world space at all times. In other words if you rotate the body, the neck does not rotate. It remains oriented how it was posed.

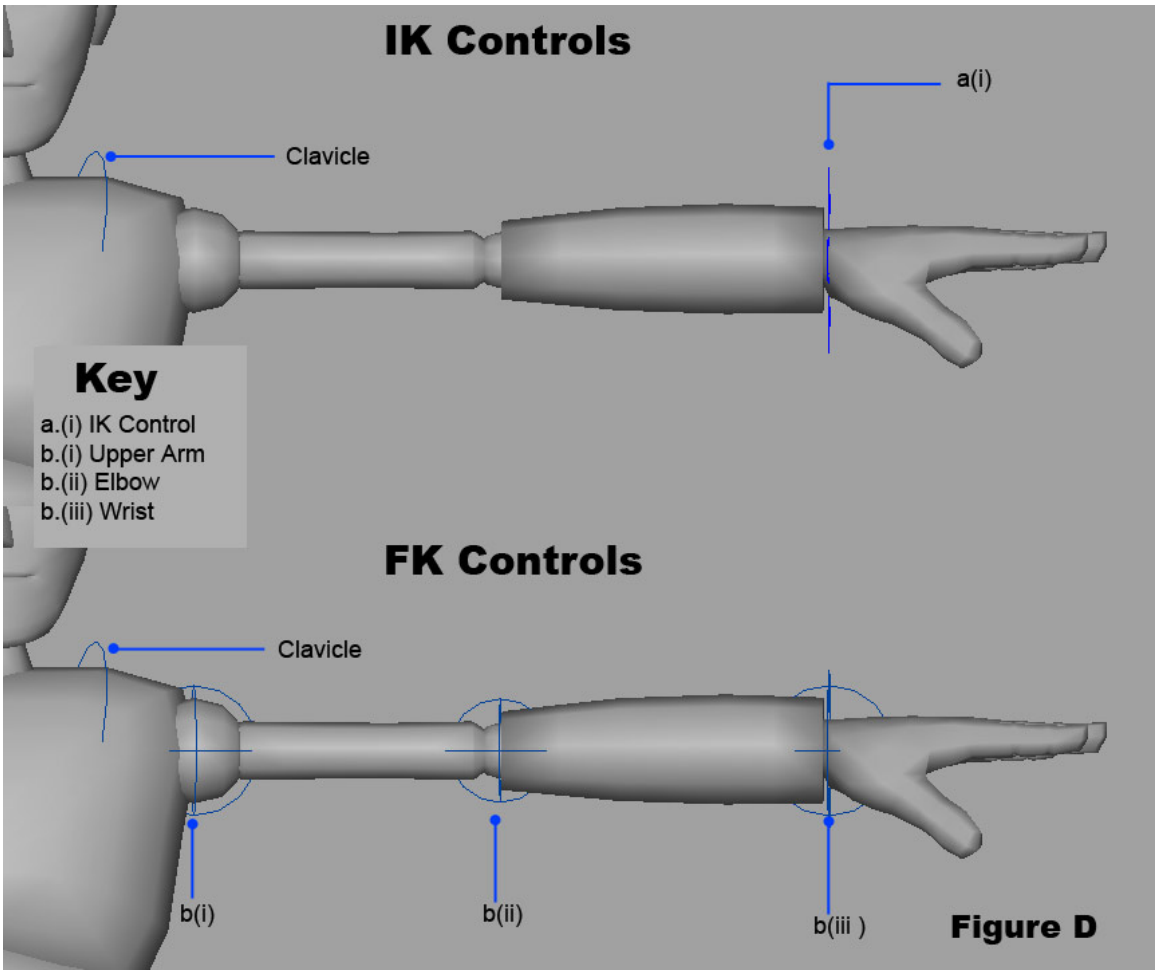
3. Root:
    a. Allows for the neck to remain in world space, yet remain influenced by the root. In other words, if you rotate the spine, the neck will remain oriented how it was posed. If you rotate the root, the neck will follow the rotation.

c. **ctrlFK_Cn_SpineF (Head Controls):** (the curve shaped like a circle with an x in it located on at the base of the head)
    i. **General Controls**
        1. Translate x,y,z:
            a. Translate the head
        2. Rotate x,y,z:
            a. Rotate the head.
        3. Scale x,y,z:
            a. Scale the head
    ii. SS Controls (0=off 1=on)
        1. Parent:
            a. Allows the head to follow the rotations of the neck, torso, and root.
        2. World:
            a. Allows the head to remain in world space at all times. In other words if you rotate the body or neck, the head does not rotate. It remains oriented how it was posed.
        3. Root:
            a. Allows for the head to remain in world space, yet remain influenced by the root. In other words, if you rotate the spine or neck, the head will remain oriented how it was posed. If you rotate the root, the head will follow the rotation.
        4. Chest:
            a. Chest puts it in chest space, so the base of the neck doesn't affect the orientation of the head.

d. **ctrlFK_Rt/Lf_Clavicle:** (The curves located between the upper arm joint and the neck, in the shoulder region.) (Arm visibility must also be turned on to see the curve.) (See Figure D)
    i. Translate x,y,z:
        1. Translate the shoulder region.
    ii. Rotate x,y,z:
        1. Rotate the shoulder region
    iii. Scale x,y,z:

1. Scale the shoulder region.

4. **Rt/Lf Arm Controls** (include both FK and IK controls for the arms)



(Message from Morgan Loomis about the IK Arm control)

"Hi. You may have noticed that when you zero out rotation of the IK hands, they don't match up to what you thought was the default pose. Good grief! Before you go swamping Leif with emails about bugs, there's a logical explanation.

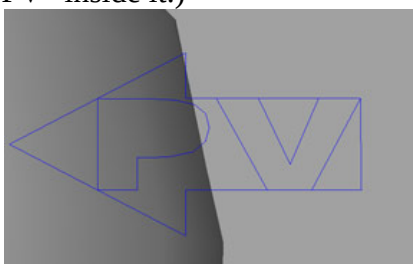
The character's default pose is arbitrary. It's a way for the rig to get back to where the model was skinned, and also a starting point. But usually this pose doesn't jive with the correct orientation of certain world space controls. In the default pose, the hands are pointed in opposite directions; the left is pointed toward positive X, and the right is pointed toward negative X. For the sake of the consistency of the rig, it is ideal for both hands to be oriented the same way, just like the feet. This ensures predictable results when you're adjusting attributes or curves, and also makes all the world space controls oriented the same way: hands and feet will all be pointing in positive X when they're zeroed out. When operating in world space, it makes it possible to copy animation predictably from one hand to the other as well, without having to account for a 180 degree

difference. “

a. **IK Controls**
   i. **ctrlIK_Rt/Lf_ArmIK** (These are your IK controls, and are located at the wrist when IK is turned on.)

1. **General Controls**
   a. Translate x,y,z:
      i. Translate the IK.
   b. Rotate x,y,z:
      i. Rotate the IK
2. **SS Controls**
   a. World:
      i. Allows the arm to remain in world space at all times. In other words if you rotate the body, the arm does not rotate. It remains oriented how it was posed.
   b. Root:
      i. Constrains the IK control to the root.
   c. Hips:
      i. Constrains the IK control to the hips.
   d. Chest:
      i. Constrains the IK control to the chest.
   e. Head:
      i. Constrains the IK control to the head.
   f. Pin:
      i. Constrains the IK to a locator. The locator will appear in the middle of the wrist when this option is turned on. You can then constrain/parent objects to the locator, or constrain/parent the locator to objects.

ii. **ctrlIK_Rt/Lf_ArmPV** (Pole vector/arm flap for the arm IK. Located behind the shoulder. It is a curve in the shape of an arrow with the letters “PV” inside it.)



1. **General Controls**
   a. Translate x,y,z:
      i. Translate the Pole vector.
   b. Rotate x,y,z:

i. Rotate the Pole vector

2. SS Controls.

a. Local World:

i. This controls whether the pole vector moves with the body, or stays in its current location. When set to 0 the pole vector moves with the body (local space) and is behind the shoulder joint, when set to 1 it acts independently from the body (world space) and is in front of the shoulder joint.

b. FK Controls

i. **ctrlFK_Rt/Lf_ArmA** (Upper Arm control)

1. **General Controls**

a. Rotate x,y,z:

i. Rotate the upper arm.

2. SS Controls

a. Parent:

i. Allows the arm to follow the rotations of the neck, torso, and root.

b. World:

i. Allows the arm to remain in world space at all times. In other words if you rotate the body, the arm does not rotate. It remains oriented how it was posed.

c. Root:

i. Allows for the arm to remain in world space, yet remain influenced by the root. In other words, if you rotate the spine, the arm will remain oriented how it was posed. If you rotate the root, the arm will follow the rotation.

d. Chest:

i. Allows you to orient the arm to the chest so that if you move the clavicle it will not directly affect your arm pose.

ii. **ctrlFK_Rt/Lf_ArmB** (Elbow control)

1. Rotate x,y,z:

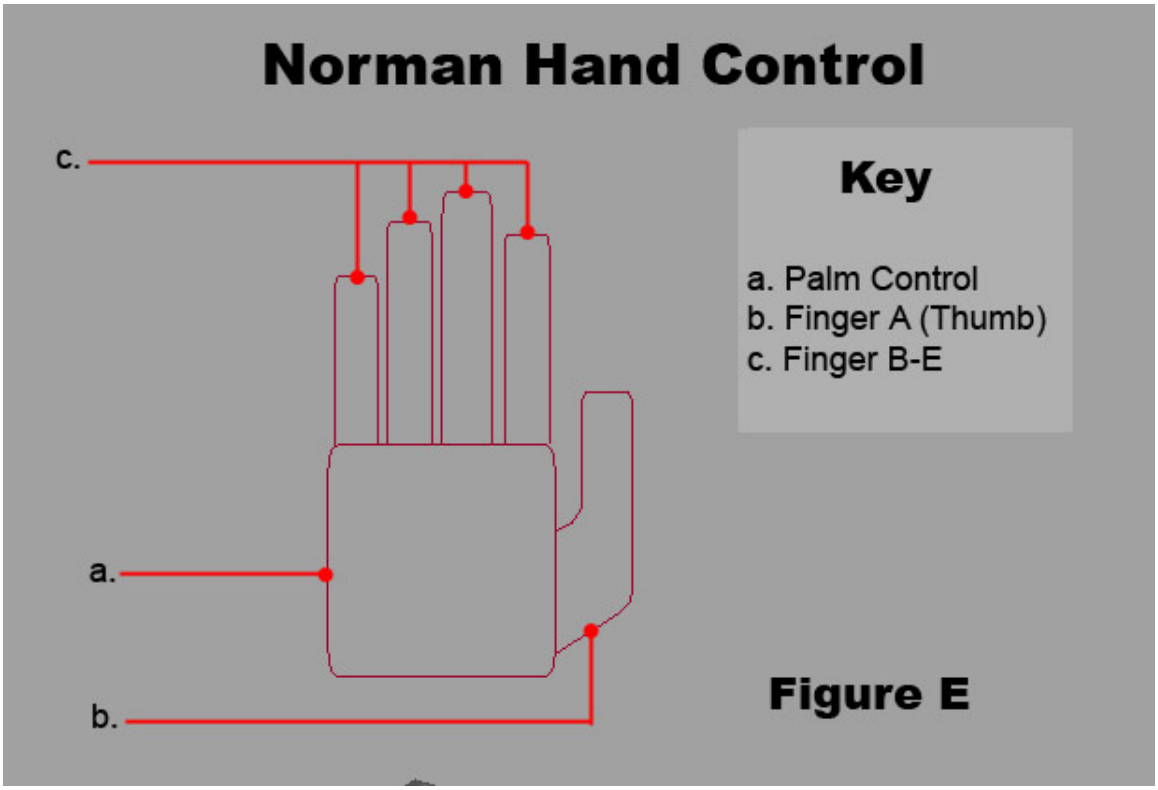
a. Rotate the elbow

iii. **ctrlFK_Rt/Lf_ArmC** (Wrist control)

1. Rotate x,y,z:

a. Rotate the wrist.

5. **Rt/Lf Hand Controls** (Located above the wrists. It is a curve in the shape of a hand. Clicking on individual parts of the curve will bring up attributes for the corresponding location on the hand)



   a. **ctrlDriv_R/tLf_Hand (Palm Control):** (The square palm on the curve shaped like a hand.)
      i. Palm Twist:
         1. Rotates the palm in a clockwise or counterclockwise motion
      ii. Palm Turn:
         1. Rotates the palm left or right
      iii. Palm Bend:
         1. Folds the palm up or down
      iv. Palm Cup:
         1. Cups the palm.
      v. Hand Slide:
         1. Translates the hand left or right.
      vi. Hand Scale:
         1. Scales the size of the hand. 1=default size

b. **ctrlDriv_R/tLf_fingerA (Thumb Control):** (The thumb on the curve shaped like a hand.)
    i. Base:
        1. Controls rotation of the base of the thumb.
    ii. Mid:
        1. Controls rotation of the middle of the thumb.
    iii. Tip:
        1. Controls rotation of the tip of the thumb.
    iv. Twist:
        1. Twists the entire thumb.
    v. Spread:
        1. Spreads the thumb towards or away from the other fingers.
    vi. Stretch:
        1. Allows you to stretch the length of the thumb.
c. **ctrlDriv_R/tLf_fingerB-E (Finger Controls):** (the Fingers on the curve shaped like a hand)
    i. Base:
        1. Controls rotation of the base of the finger.
    ii. Mid:
        1. Controls rotation of the middle of the finger.
    iii. Tip:
        1. Controls rotation of the tip of the finger.
    iv. Twist:
        1. Twists the entire finger.
    v. Spread:
        1. Spreads the finger towards or away from the other fingers.
    vi. Stretch:
        1. Allows you to stretch the length of the finger.

6. **Rt/Lf Leg Controls** (include both FK and IK controls for the legs)

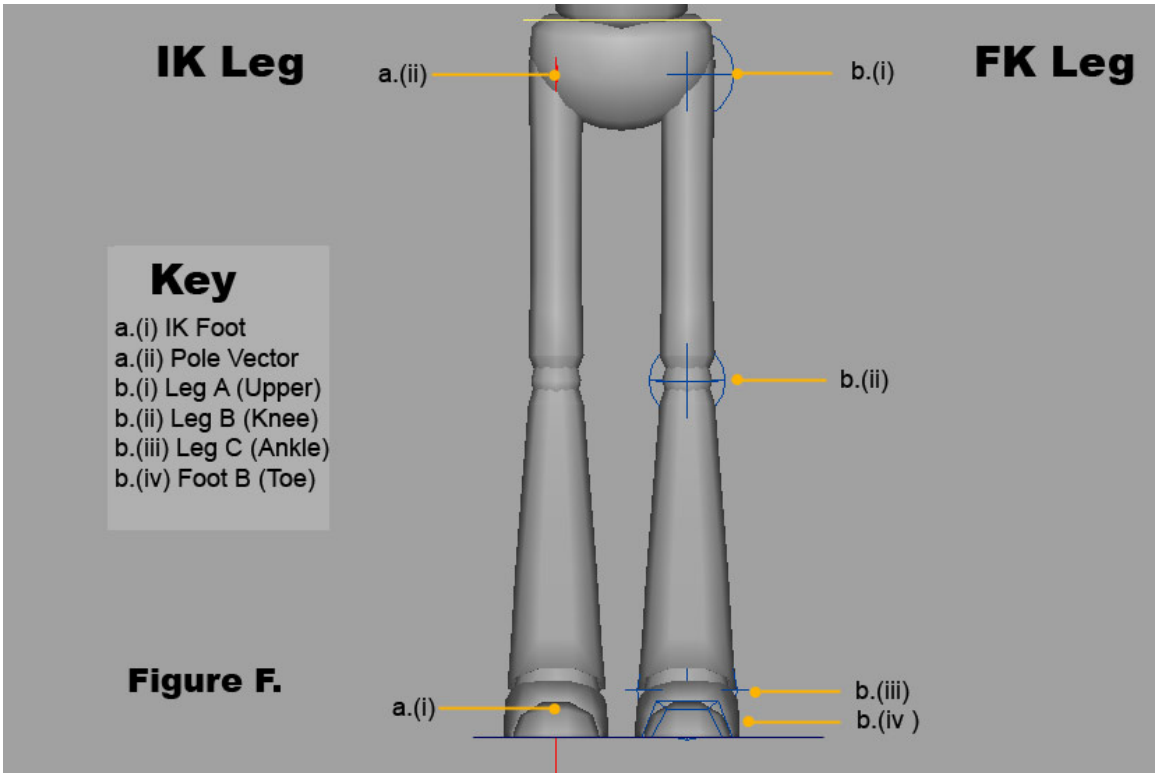


Figure F.

a. **IK Controls**
    i. **ctrlIK_Lf_FootIK** (These are your IK controls, and are located at the foot when IK is turned on.)
        1. **General Controls**
            a. Translate x,y,z:
                i. Translate the IK.
            b. Rotate x,y,z:
                i. Rotate the IK
            c. Scale x,y,z:
                i. Scale the IK
            d. Heel Roll:
                i. Rolls the foot from the heel.
            e. Ball Roll:
                i. Rolls the foot from the ball of the foot.
            f. Toe Roll:
                i. Rolls the foot from the toe.
            g. Ball Twist:

i. Twists foot from the ball of the foot.

h. Foot Rock:

i. Rocks the foot from side to side.

2. **SS Controls**

a. World: (Default IK foot setting)

i. This will keep the foot control in world space and affected by no other part of the body.

b. Root:

i. This will constrain the IK foot to the root joint, allowing you to rotate and translate the root and have the foot follow.

c. Hips:

i. This will constrain the IK foot to the hip joint, allowing you to rotate and translate the root or hip and have the foot follow.

d. Pin:

i. This will constrain the IK foot to a locator. The locator will appear in the middle of the foot when this option is turned on. You can then constrain/parent objects to the locator, or constrain/parent the locator to objects.

ii. ctrlIK_Rt/Lf_LegPV (Pole vector/leg flap for the leg IK. Located infront of the hip joint. It is a curve in the shape of an arrow with the letters "PV" inside it.)

1. **General Controls**

a. Translate x,y,z:

i. Translate the Pole vector.

b. Rotate x,y,z:

i. Rotate the Pole vector

2. SS Controls.

a. Local World:

i. This controls whether the pole vector moves with the body, or stays in its current location. When set to 0 the pole vector moves with the body (local space), when set to 1 it acts independently from the body (world space).

b. FK Controls

i. **ctrlFK_Rt/Lf_LegA** (Upper Leg control)

1. **General Controls**

a. Rotate x,y,z:

i. Rotate the upper leg.

2. SS Controls

a. Parent:

i. Allows the Leg to follow the rotations of the hip and root.

b. World:

i. Allows the leg to remain in world space at all times. In other words if you rotate the body, the leg does not rotate. It remains oriented how it was posed.

c. Root:

i. Allows for the leg to remain in world space, yet remain influenced by the root. In other words, if you rotate the hips, the leg will remain oriented how it was posed. If you rotate the root, the leg will follow the rotation.

ii. **ctrlFK_Rt/Lf_LegB** (Knee control)

1. Rotate x,y,z:

a. Rotate the knee.

iii. **ctrlFK_Rt/Lf_LegC** (Ankle control)

1. Rotate x,y,z:

a. Rotate the ankl.

iv. **CtrlFK_Rt/Lf_FootB** (Toe Control)

1. Rotate x,y,z:

a. Rotate the toe.

7. Tweaks

a. Allows you greater control over various parts of the body. (I strongly urge you to explore them. They can be very useful.)

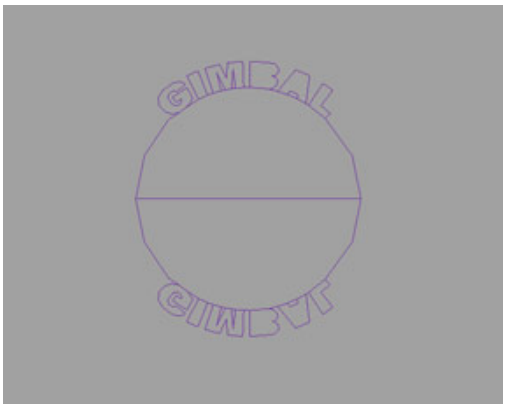
8. BendBo's (When turned on, the curves are circles located in the middle of the upper and lower limbs of the body.)

a. Bendable bones that will allow you to create curves to your upper and lower limbs. Intended to aid in making your poses more appealing.

i. Translate x,y,z:

1. Translates the BendBo.

9. Gimbal Controls (When turned on, the curves are located on top of the appropriate joint.)



a. Gimbal Controls are there to aid you in avoiding Gimbal Lock. It allows you the ability to split your rotations and translates over two controls. It also allows you an extra control if you want to add a secondary motion on top of your main action. (IE: A shake on top of the main motion. The shake can be on the Gimbal Control, and allow you to keep cleaner curves.)

Please report any bugs or issues to:

fixnorman@gmail.com