

Autodesk®
Maya®
2012



Autodesk

Full Body IK (FBIK) Guide

Copyright Notice

Autodesk® Maya® 2011 Software

© **2010 Autodesk, Inc. All rights reserved.** Except as otherwise permitted by Autodesk, Inc., this publication, or parts thereof, may not be reproduced in any form, by any method, for any purpose.

Certain materials included in this publication are reprinted with the permission of the copyright holder.

The following are registered trademarks or trademarks of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and other countries:

3DEC (design/logo), 3December, 3December.com, 3ds Max, Algor, Alias, Alias (swirl design/logo), AliasStudio, Alias|Wavefront (design/logo), ATC, AUGI, AutoCAD, AutoCAD Learning Assistance, AutoCAD LT, AutoCAD Simulator, AutoCAD SQL Extension, AutoCAD SQL Interface, Autodesk, Autodesk Envision, Autodesk Intent, Autodesk Inventor, Autodesk Map, Autodesk MapGuide, Autodesk Streamline, AutoLISP, AutoSnap, AutoSketch, AutoTrack, Backburner, Backdraft, Built with ObjectARX (logo), Burn, Buzzsaw, CAiCE, Civil 3D, Cleaner, Cleaner Central, ClearScale, Colour Warper, Combustion, Communication Specification, Constructware, Content Explorer, Dancing Baby (image), DesignCenter, Design Doctor, Designer's Toolkit, DesignKids, DesignProf, DesignServer, DesignStudio, Design Web Format, Discreet, DWF, DWG, DWG (logo), DWG Extreme, DWG TrueConvert, DWG TrueView, DXF, Ecotect, Exposure, Extending the Design Team, Face Robot, FBX, Fempro, Fire, Flame, Flare, Flint, FMDesktop, Freewheel, GDX Driver, Green Building Studio, Heads-up Design, Heidi, HumanIK, IDEA Server, i-drop, ImageModeler, iMOUT, Incinerator, Inferno, Inventor, Inventor LT, Kaydara, Kaydara (design/logo), Kynapse, Kynogon, LandXplorer, Lustre, MatchMover, Maya, Mechanical Desktop, Moldflow, Moonbox, MotionBuilder, Movimento, MPA, MPA (design/logo), Moldflow Plastics Advisers, MPI, Moldflow Plastics Insight, MPX, MPX (design/logo), Moldflow Plastics Xpert, Mudbox, Multi-Master Editing, Navisworks, ObjectARX, ObjectDBX, Open Reality, Opticore, Opticore Opus, Pipeplus, PolarSnap, PortfolioWall, Powered with Autodesk Technology, Productstream, ProjectPoint, ProMaterials, RasterDWG, RealDWG, Real-time Roto, Recognize, Render Queue, Retimer, Reveal, Revit, Showcase, ShowMotion, SketchBook, Smoke, Softimage, Softimage|XSI (design/logo), Sparks, SteeringWheels, Stitcher, Stone, StudioTools, ToolClip, Topobase, Toxik, TrustedDWG, ViewCube, Visual, Visual LISP, Volo, Vtour, Wire, Wiretap, WiretapCentral, XSI, and XSI (design/logo).

ACE™, TAO™, CIAO™, and CoSMIC™ are copyrighted by Douglas C. Schmidt and his research group at Washington University, University of California, Irvine, and Vanderbilt University, Copyright © 1993-2009, all rights reserved.

Adobe, Illustrator and Photoshop are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Intel is a registered trademark or trademark of Intel Corporation or its subsidiaries in the United States and other countries.

mental ray is a registered trademark of mental images GmbH licensed for use by Autodesk, Inc.

OpenGL is a trademark of Silicon Graphics, Inc. in the United States and other countries. Python and the Python logo are trademarks or registered trademarks of the Python Software Foundation.

The Ravix logo is a trademark of Electric Rain, Inc.

All other brand names, product names or trademarks belong to their respective holders.

Disclaimer

THIS PUBLICATION AND THE INFORMATION CONTAINED HEREIN IS MADE AVAILABLE BY AUTODESK, INC. "AS IS." AUTODESK, INC. DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE REGARDING THESE MATERIALS.

Contents

Chapter 1	Full Body IK (FBIK)	1
	Restore Full body IK	1
	Full body IK overview	1
	Prepare your character for full body IK	2
	FBIK joint types	3
	FBIK joint labelling and naming conventions	4
	FBIK joint labelling and naming methods	5
	Joint labels and names	8
	Understanding the FBIK structure	20
	FBIK skeletons and effectors	21
	FBIK effectors	22
	FBIK Body Parts	22
	Character and subcharacter sets	23
	Character animation with FBIK	25
	IK and FK Reach	27
	FBIK Keys	28
	Create and use full body IK	30
	Full body IK workflow	30
	Create the full body IK system for your character	31
	Pin and unpin your character's effectors	35
	Example workflows: Full body IK	41
	Create auxiliary pivots	55
	Create auxiliary effectors	57
	Create floor contacts for your character's hands and feet	61

Create and use full body IK	66
Troubleshoot full body IK character setup	66
Animate > Set Full Body IK Keys	69
Skeleton > Full Body IK	71
Skeleton > Full Body IK > Add Full Body IK	71
Skeleton > Full Body IK > Add Auxiliary Effector	71
Skeleton > Full Body IK > Add Auxiliary Pivot	72
Skeleton > Full Body IK > Activate Auxiliary Pivot	72
Skeleton > Full Body IK > Change Auxiliary Pivot Placement	72
Skeleton > Full Body IK > Show FBlK FK Skeleton	73
Skeleton > Full Body IK > Hide FBlK FK Skeleton	73
Skeleton > Full Body IK > Add Floor Contact Plane	73
Skeleton > Full Body IK > Go to Stance Pose	73
Skeleton > Full Body IK > Add Missing Effectors	74
Skeleton > Full Body IK > Body Part Autoload	74
Edit > Keys > Delete FBlK Keys	74
hikEffector	75
hikFloorContactMarker	76
Index	81

Full Body IK (FBIK)



Restore Full body IK

In Maya 2012, full body IK has been replaced by HumanIK technology that provides additional full body rigging and animation capabilities. All FBIK menus and options have been removed from Maya, however you can still load your existing scenes with FBIK rigs.

To have the deprecated FBIK user interface elements display in Maya, you can enter the following in the Maya command line:

```
optionVar -intValue showFbikMenuItem 1;
```

Full body IK overview

Full body IK (FBIK) lets you create natural poses and animation for your characters. Instead of using single IK handles for each limb, full body IK lets you translate and rotate a character's body parts with FBIK effectors and make the rest of the body follow those movements.

For example, if you use FBIK to animate the arm of a biped character reaching down to pick up an object off the floor, your character will bend naturally at the waist and knees as you move your character's FBIK hand effectors downward. You can also use FBIK to animate character movements such as swinging from a tree branch, doing a handstand, a walk cycle, and so on.

Maya's FBIK is based on the Autodesk® HumanIK® solver used in Autodesk® MotionBuilder®. FBIK utilizes the full capabilities of the HumanIK solver so that you can use the various attributes to tweak the motions of your characters during interaction and playback when animating with FBIK. The Autodesk® FBX® plug-in and file interchange format allows you to transfer rigged characters

animated with FBIK between Maya and MotionBuilder without requiring the animation to be baked.

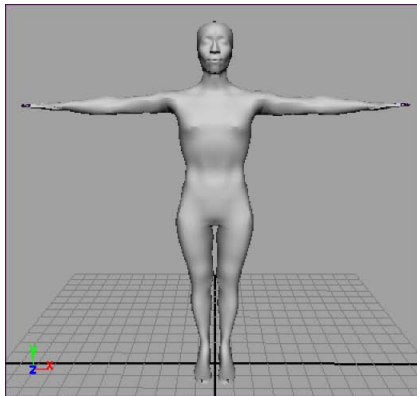
Related topics

- [Full body IK workflow](#) (page 30)

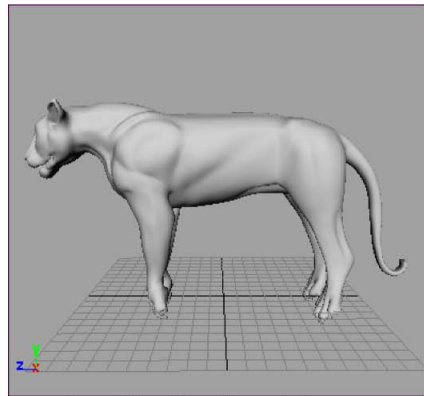
Prepare your character for full body IK

Before you create the FBIK effectors for your biped character, make sure your character is in a pose that will produce the best FBIK results.

- For a *biped*, position your character in a “T” pose (also known as the *T-stance*) facing down the +Z-axis. Position the character’s feet straight forward, hands flat with palms facing downward, and thumbs pointing down the +X-axis. The pose does not need to be an exact T-stance, but a close approximation.
- For a *quadruped*, position your character to face down the +Z-axis with its paws/hooves pointing downwards with all their toes fully-extended towards the floor (not flat on the ground). If your character has a longer neck (like that of a camel or giraffe), extend the neck (for example, as if the giraffe is reaching for a leaf) and point it down the +Z-axis.



Properly positioned biped character



Properly positioned quadruped character
Model by Ted Charlton

Related topics

- [Full body IK workflow](#) (page 30)

FBIK joint types

When animating characters with FBIK, the joint labelling and naming conventions provide names for the following types of joints:

- Base joints
- Roll joints
- Auxiliary joints
- Accessory and additional joints

Base joints

Base joints are the main joints in your character's skeleton that are *required* for FBIK to function. The base joints in your character's skeletons include: Hips (pelvis or *root*), Spine, Head (top of neck), UpLeg (hips), Leg (knees), Foot (ankles), Arm (shoulders), ForeArm (elbows), and Hand (wrists).

Roll joints

Roll joints are *optional* joints with no real anatomical equivalents that let you create realistic twisting motions in the limbs of your characters. When labelling or renaming roll joints, give them the same labels or names as their parent joints. The roll joints in your character's skeletons include: ArmRoll (between the shoulders and elbows), ForeArmRoll (between elbows and wrists), UpLegRoll (between hips and knees), and LegRoll (between knees and ankles).

Auxiliary joints

Auxiliary joints are *optional* joints that serve specific purposes that may not be needed for all characters. The auxiliary joints in your character's skeletons include: Neck, Shoulder (collar bone), fingers, toes, InHand/InFoot (metacarpal/metatarsal) and FingerBase/ToeBase (carpal/tarsal) joints.

FingerBase and InHand In your character's hands there are two specialized auxiliary joint types: FingerBase and InHand. FingerBase is most often used in quadruped hands to aid in paw/hoof placement. When you apply FBIK to your character, an effector named *FingerBaseEff* is created for your FingerBase joints. InHand joints are used to group each fingers' joints. If you use InHand, you must also use the FingerBase joint as an intermediary between your InHand joints and the LeftHand/RightHand (wrist) joints. FingerBase and InHand are not needed to animate your character's hands and fingers. However, when using hand floor contacts, FingerBase is required if you want your hands to bend at the bases of their fingers when they come in contact with the floor.

ToeBase and InFoot In your character's feet there are two specialized auxiliary joint types: ToeBase and InFoot. ToeBase is most often used in quadruped feet to aid in paw/hoof placement. When you apply FBlK to your character, an effector named *ToeBaseEff* is created for your ToeBase joints. For biped characters, you can use ToeBaseEff with Floor Contacts (similar to the way you can use auxiliary pivots) to create good bends at the base of your character's toes when animating foot rolls with FBlK. InFoot joints are used to group each toes' joints. If you use InFoot, you must also use the ToeBase joints as intermediaries between the InFoot and Foot joints in your character's feet. ToeBase and InFoot are not needed to animate your character's feet and toes. However, when using foot floor contacts, ToeBase is required if you want your feet to bend at the bases of their toes (balls of feet) when they come in contact with the floor.

HipsTranslation You can split the translation and rotation of your character's pelvis between two different joints by specifying a HipsTranslation joint in addition to the Hips joint. When you then apply FBlK to your character, an effector named HipsEff is created for your Hips joint, but an effector is not created for HipsTranslation. By default, the HipsTranslation joint will control the translation of your character's hips and the Hips joint and effector will control the rotation of your character's hips. The HipsTranslation joint must be placed above the Hips joint in your character's joint hierarchy and it must also be at the same position as the Hips joint in the scene view.

Accessory or additional joints

Accessory joints are the joints in your character's skeletons that you do not want to include in the FBlK solution because they are only used for skinning purposes, for example, chest or rib joints and the joint at the top of your character's head. Additional joints are the joints in your character's skeletons for which FBlK joint labels or names do not exist, for example, wing or tail joints. For these types of joints, you would use the None joint label or joint names that are not recognized by the hikSolver. For information on how to label or rename these joints, see [Other body parts](#) (page 20).

FBlK joint labelling and naming conventions

To use FBlK, you must *label* or *rename* each joint in your character's skeleton to follow the FBlK joint labelling and naming conventions. Any joints in a skeleton that are not labelled or renamed will not be included in the FBlK solution.

Each joint labelling and naming method provides the same level of control over character poses, with the exception of joint labelling, which cannot be used on skeletons that have InHand, InFoot, or HipsTranslation joints.

For information on joint labelling and naming methods, the types of joints that are included in each joint labelling and naming method, and a complete list of the available FBIK joint labels and names, see the following:

- [FBIK joint types](#) (page 3)

FBIK joint labelling and naming methods

There are three methods you can use to rename your joints so that they follow the FBIK joint labelling and naming conventions:

- Joint labelling
- Renaming your character's joints
- Specifying custom joint names with MEL.

NOTE

If you label or rename only the base joints in your character's skeleton, the joints in between the base joints *will not* be controlled by full body IK. If you label or rename all your character's joints, then the hikSolver will produce better results as its solution will include all your character's joints rather than just the base joints.

Joint labelling

You can use joint labels to tag your character's joints so that when you create your FBIK effectors, all the labeled joints are included in the full body IK solution. When using the *labelling* FBIK joint labelling and naming method, you can label only your character's base joints, or you can label all your character's base, roll, and many of the auxiliary joints.

Joint labelling produces a complete full body IK solution. However, joint labels do not exist for the specialized InHand, InFoot, or HipsTranslation auxiliary joints. If you want to use these auxiliary joints in your skeleton, you need to use joint naming or MEL instead of labelling. See [Accessory or additional joints](#) (page 4).

NOTE

- If you label FingerBase or ToeBase joints, their labels will appear as *Finger* and *Toe* in the scene view.
- For non-grasping hands/feet where their thumbs/big toes move with the rest of their digits (such as quadruped paws and hooves or human feet), name their thumb and big toe joints as *ExtraFingers* so that the FingerBaseEff and ToeBaseEff effectors affect the digits of the hands/feet.

For grasping hands/feet where their thumbs/big toes flex independent of the rest of their digits (such as human hands or primate hands and feet), name their thumb and big toe joints as *HandThumbs* and *FootThumbs* respectively so that the HandEff (wrist) and FootEff (ankle) effectors affect the digits of the hands/feet—rather than the BaseEff effectors.

Joint labels are useful when you do not want to rename the joints in your character's skeleton. For example, you can use joint labels if you have an existing character with custom joint names or if you are using referenced animation with namespaces.

For most characters, we recommend that you use joint labelling since it is the easiest and fastest FBIK joint labelling and naming method.

Labelling your joints for FBIK is similar to labelling your joints for animation *retargeting*. For steps on how to label your joints, see Label a skeleton's joints.

TIP

- If you do not see your joint labels in the scene view, select Skeleton > Joint Labelling > Show All Labels.
 - Label *rolljoints* the same as their parents are labelled. For example, you should label a left forearm roll joint as Elbow(L) since the forearm roll joints' parent joint is labelled *Elbow(L)*. Roll joints are supported for the elbow, shoulder, hip and knee.
 - You can automatically label entire limbs at a time, rather than labelling each individual joint in your character's skeleton. See Label a skeleton's joints.
-

Renaming your character's joints

As an alternative to joint labelling, you can *rename* your character's joints to produce a complete full body IK solution.

When using the renaming FBIK joint labelling and naming method, you can rename only your character's base joints or you can rename *all* your character's base, roll, and auxiliary joints. See [FBIK joint types](#) (page 3).

For FBIK to work on your character, you are required to at least rename your character's base joints according to the FBIK joint labelling and naming conventions. See [Base joints](#) (page 3). Renaming only your character's base joints is useful for *simple* biped characters that do not have any custom or specialized joints, but require the ability of basic feet flexion. This joint naming method produces the simplest full body IK solution because the FBIK effectors that are created with this naming method will only use the main joints and *not* the rest of the joints in your character's skeleton (for example, roll joints or finger and toe joints).

If you want to include all your character's joints in your FBIK solution—including auxiliary joints like the InHand/InFoot or ToeBase/FingerBase joints, then you should rename *all* the joints in your character's skeleton.

Specify custom joint names with MEL

You can create a custom joint naming convention for your full body IK using MEL to meet the special needs of your production or game pipeline.

The `-sourceSkeleton (-sk)` flag on the command lets you specify the association between your character's current joint names and the FBIK joint names that adhere to the FBIK joint labelling and naming conventions. For each joint you want to include in the FBIK solution, you need to specify a partial path for the joint and the name of the full body joint it represents. These must then be separated by spaces, and multiple entries need to be separated by commas (.). For example:

```
characterize -sk "myhips hips,rightHip rightupleg,leftHip  
leftupleg,..."
```

This MEL string tells the `characterize` command to look for and use the alternate joint names (provided within the quotation marks of the string) during the full body IK solution.

NOTE

When using MEL to specify custom joints names for FBIK, you *cannot* have multiple joints in the scene with the same name. Instead, you need to provide the partial path name for the joints that are named the same.

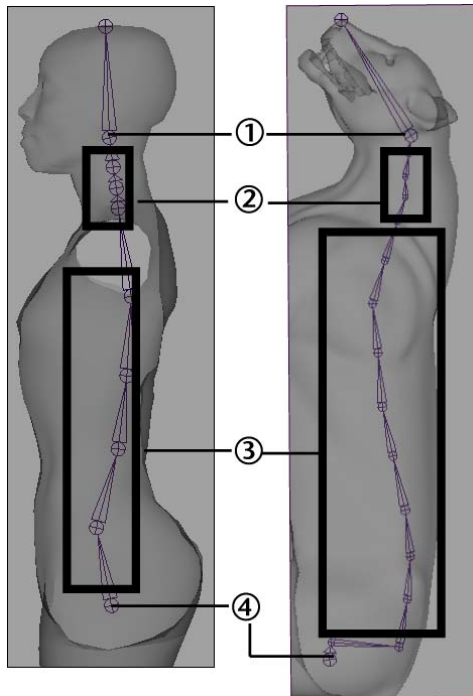
Joint labels and names

For information on joint types or joint labelling and renaming methods, see [FBIK joint types](#) (page 3) and [FBIK joint labelling and naming methods](#) (page 5).

NOTE

Joint naming is not case sensitive and you can use “l” or “r” instead of *left* or *right*.

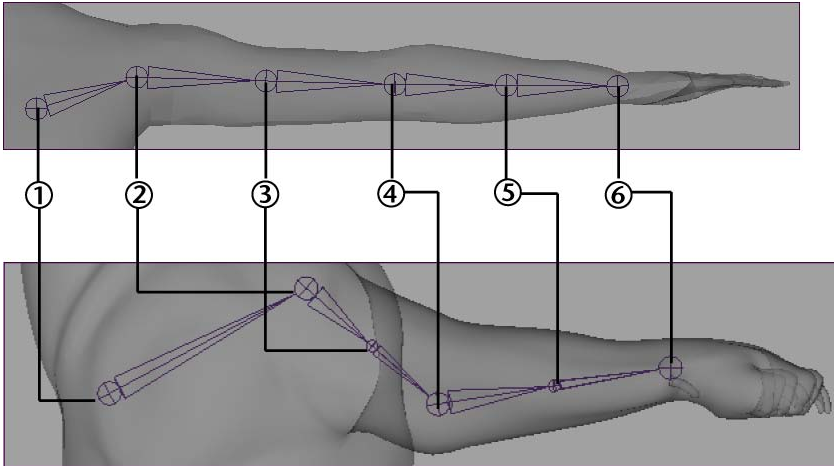
Hips, spine, neck, and head



	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
1	Joint at base of head (topmost neck joint)	Head	Head	Head
2	First joint in neck	Neck	-	Neck
	All other neck joints	Neck	-	Neck1-Neck9
3	First spine joint after the root	Spine	Spine	Spine

	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
	All other spine joints	Spine	Spine1-Spine9	Spine1-Spine9
4	Pelvis (root)	Root	Hips	Hips
		-	-	HipsTranslation

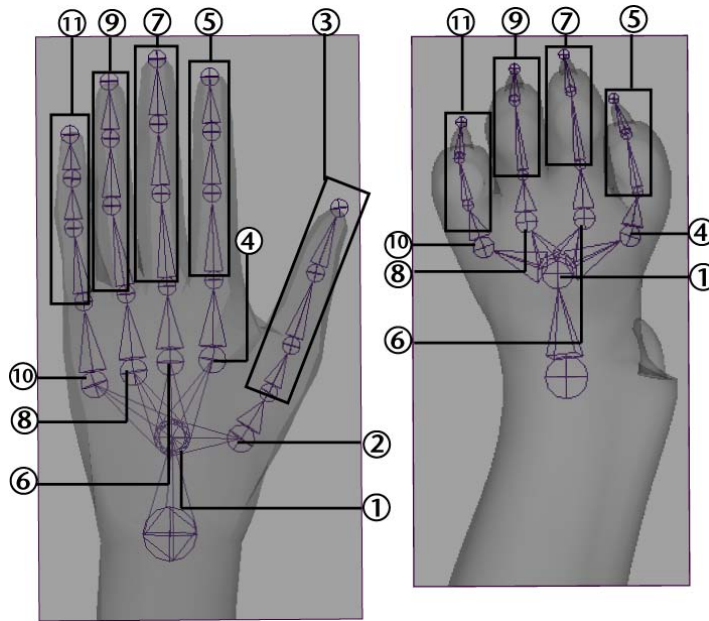
Left and right arms



	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
1	Collar bone	Collar (L)	-	LeftShoulder
		Collar (R)	-	RightShoulder
2	Shoulder	Shoulder (L)	LeftArm	LeftArm
		Shoulder (R)	RightArm	RightArm

	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
3	Upper arm roll joint	Shoulder (L)	-	LeftArmRoll
		Shoulder (R)	-	RightArmRoll
4	Elbow	Elbow (L)	LeftForeArm	LeftForeArm
		Elbow (R)	RightForeArm	RightForeArm
5	Lower arm roll joint	Elbow (L)	-	LeftForeArmRoll
		Elbow (R)	-	RightForeArmRoll
6	Wrist	Hand (L)	LeftHand	LeftHand
		Hand (R)	RightHand	RightHand

Left and right hands

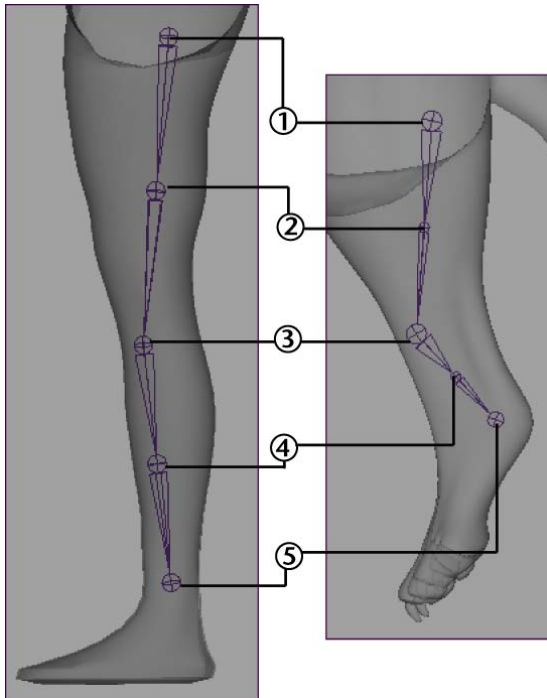


	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
1	Palm of hand between the wrist and fingers.	Finger Base (L)	-	LeftFingerBase
		Finger Base (R)	-	RightFingerBase
2	Before first joint of thumb	-	-	LeftInHandThumb
		-	-	RightInHandThumb
3	Thumb joints (from base to tip of thumb)	Thumb (L)	-	LeftHandThumb1- LeftHandThumb4

	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
		Thumb (R)	-	RightHandThumb1- RightHandThumb4
4	Before first joint of index finger	-	-	LeftInHandIndex
		-	-	RightInHandIndex
5	Index finger joints (from base to tip of finger)	Index (L)	-	LeftHandIndex1- LeftHandIndex4
		Index (R)	-	RightHandIndex1- RightHandIndex4
6	Before first joint of middle finger	-	-	LeftInHandMiddle
		-	-	RightInHandMiddle
7	Middle finger joints (from base to tip of finger)	Middle Finger (L)	-	LeftHandMiddle1- LeftHandMiddle4
		Middle Finger (R)	-	RightHandMiddle1- RightHandMiddle4
8	Before first joint of ring finger	-	-	LeftInHandRing
		-	-	RightInHandRing

	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
9	Ring finger joints (from base to tip of finger)	Ring Finger (L)	-	LeftHandRing1-LeftHandRing4
		Ring Finger (R)	-	RightHandRing1-RightHandRing4
10	Before first joint of pinky finger	-	-	LeftInHandPinky
		-	-	RightInHandPinky
11	Pinky finger joints (from base to tip of finger)	Pinky (L)	-	LeftHandPinky1-LeftHandPinky4
		Pinky (R)	-	RightHandPinky1-RightHandPinky4
	Before first joint of extra finger	-	-	LeftInHandExtraFinger
		-	-	RightInHandExtraFinger
	Extra finger joints (from base to tip of finger)	-	-	LeftHandExtraFinger1-LeftHandExtraFinger4
		-	-	RightHandExtraFinger1-RightHandExtraFinger4

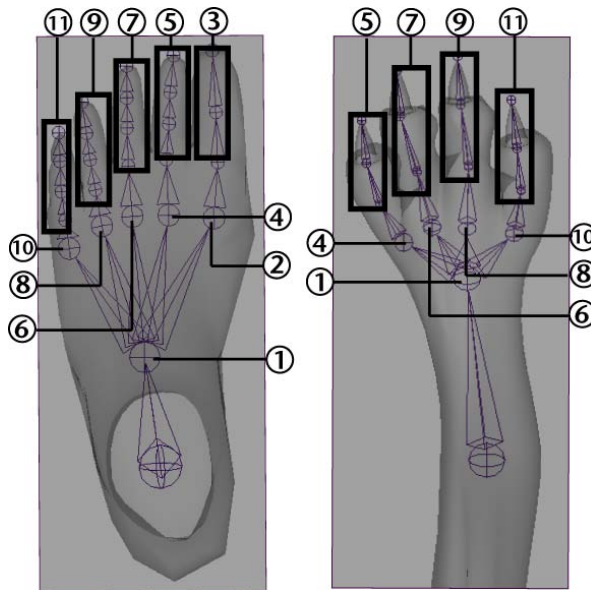
Left and right legs



	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
1	Hip	Hip (L)	LeftUpLeg	LeftUpLeg
		Hip (R)	RightUpLeg	RightUpLeg
2	Upper leg roll joint	Hip (L)	-	LeftUpLegRoll
		Hip (R)	-	RightUpLegRoll
3	Knee	Knee (L)	LeftLeg	LeftLeg

	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
		Knee (R)	RightLeg	RightLeg
4	Lower leg roll joint	Knee (L)	-	LeftLegRoll
		Knee (R)	-	RightLegRoll
5	Ankle	Foot (L)	LeftFoot	LeftFoot
		Foot (R)	RightFoot	RightFoot

Left and right feet



	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
1	Ball/base of foot between the ankle and toes	Toe Base (L)	-	LeftToeBase
		Toe Base (R)	-	RightToeBase
2	Before first joint of big toe	-	-	LeftInFootThumb
		-	-	RightInFoot-Thumb
3	Big toe joints (from base to tip of toe)	Big Toe (L)	-	LeftFootThumb1- LeftFootThumb4

	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
		Big Toe (R)	-	RightFoot-Thumb1-RightFootThumb4
4	Before first joint of second toe	-	-	LeftInFootIndex
		-	-	RightInFootIndex
5	Second toe joints (from base to tip of toe)	Index Toe (L)	-	LeftFootIndex1-LeftFootIndex4
		Index Toe (R)	-	RightFootIndex1-RightFootIndex4
6	Before first joint of middle toe	-	-	LeftInFootMiddle
		-	-	RightInFoot-Middle
7	Middle toe joints (from base to tip of toe)	Middle Toe (L)	-	LeftFootMiddle1-LeftFootMiddle4
		Middle Toe (R)	-	RightFoot-Middle1-RightFootMiddle4
8	Before first joint of fourth toe	-	-	LeftInFootRing
		-	-	RightInFootRing

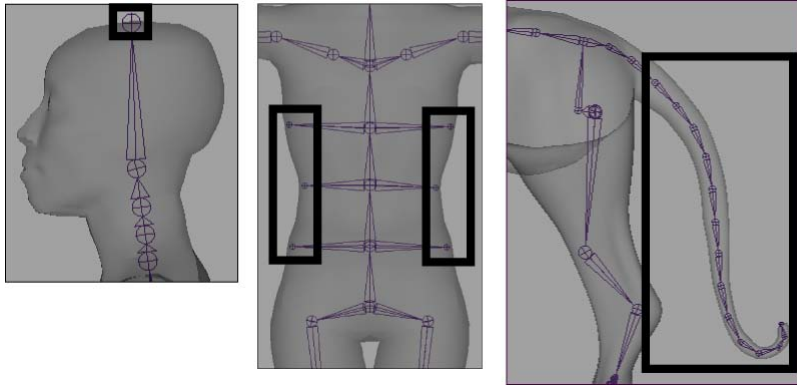
	Joint positions	FBIK joint labels	FBIK joint names (base only)	FBIK joint names (base and auxiliary)
9	Fourth toe joints (from base to tip of toe)	Ring Toe (L)	-	LeftFootRing1- LeftFootRing4
		Ring Toe (R)	-	RightFootRing1- RightFootRing4
10	Before first joint of baby toe	-	-	LeftInFootPinky
		-	-	RightInFootPinky
11	Baby toe joints (from base to tip of toe)	Pinky Toe (L)	-	LeftFootPinky1- LeftFootPinky4
		Pinky Toe (R)	-	RightFootPinky1- RightFootPinky4
	Before first joint of extra toe	-	-	LeftInFootEx- traFinger
		-	-	RightInFootEx- traFinger
	Extra toe joints (from base to tip of toe)	-	-	LeftFootEx- traFinger1-LeftFoot ExtraFinger4
		-	-	RightFootEx- traFinger1- RightFootEx- traFinger4

Other body parts

Use the **None** label or a joint name that does not adhere to the FBIK joint labelling and naming conventions for:

- Any joints for which FBIK joint labels or names do not exist
- Custom or accessory joints you do not want to include in the FBIK solution.

For example, chest or rib joints, the joint at the top of your character's head, wing or tail joints, and so on.



Understanding the FBIK structure

Full body IK uses multiple levels of hierarchy. These levels of hierarchy are used to group your effectors, joints, and even animation keys according to the anatomical structures of your character's body. When posing your characters with FBIK, you work with your character's FBIK effectors and skeletons. When animating your characters with FBIK, you work with your character's body parts and character sets.

When you add full body IK to your character, the following levels of hierarchy are created:

- A FK skeleton and a set of FBIK effectors are created for your character, and your character's original skeleton becomes its input skeleton. See [FBIK skeletons and effectors](#) (page 21).
- Each of your character's FBIK effectors and FK joints are made members of a *body part*. See [FBIK Body Parts](#) (page 22).
- A top-level *character set* is created for your character and a series of *subcharacter sets* are created for each of your character's body parts; all to

house your character's animation keys. See [Character and subcharacter sets](#) (page 23).

FBIK skeletons and effectors

When you add full body IK to your character, a FK skeleton and a set of FBIK effectors are created, and your character's original skeleton becomes its *input skeleton*. For FBIK to work, the hikSolver requires both the input and FK skeletons.

In the scene view, the input skeleton appears *brown* and the FK skeleton appears *dark blue*. To make the FK skeleton visible in the scene view, select [Skeleton > Full Body IK > Show FBIK FK Skeleton](#) (page 73) or **Hide FBIK FK Skeleton**.

During interaction, the hikSolver synchronizes your FK skeleton and the input skeleton driven by the FBIK effectors. Also, advanced FBIK attributes like **Pull** and Stiffness are not taken into account during interaction. See hikHandle.

During playback, your FK and input skeletons contribute to the resulting FBIK animation, but they are no longer synchronized.

Input skeleton

When posing with FBIK, you can transform the joints of your character's input skeleton to create joint-level poses. This is similar to posing with regular FK in Maya. See [Posing skeletons](#) .

When animating with FBIK, you can key the joints of the your character's input skeleton and their FBIK effectors to create FK animation. Any keys you set on the input skeleton are not actually placed there, but are instead placed on the FK skeleton.

When playing back your FBIK animation, the input skeleton shows the resulting FBIK animation in the scene view, similar to the blend skeleton when blending IK/FK animation. See [IK/FK blending](#).

The input skeleton is also used to bind your character's skin.

FK skeleton

When posing with FBIK, you do not transform the joints in the FK skeleton, but instead you manipulate the joints in the input skeleton to create FK-type poses.

When animating with FBIK, you do not key the joints of your FK skeleton, but instead you key the joints of your input skeleton and their FBIK effectors to create FK animation. However, the keys that you set on the input skeleton are placed on the FK skeleton.

When playing back your FBIK animation, you can show the FK skeleton to view the differences between your character's input and FK skeleton's poses.

FBIK effectors

When posing with FBIK, you can transform your FBIK effectors to create body part-level poses. This is similar to posing your characters with IK handles. See [Posing skeletons and IK handles](#).

When animating with FBIK, you can key your character's FBIK effectors and input joints to create IK animation. These keys are placed on the FBIK effectors and their FK joints, not the input skeleton's joints. You can also key your FBIK effectors' **Reach** values to set the *amount* of IK for your effectors. See [Reach Mode](#) (page 69).

WARNING

Freezing the transforms of your FBIK effectors will change the position of your character and its joints in world space.

FBIK Body Parts

Each of your character's FBIK effectors and FK joints are a member of a *body part* that corresponds to an existing limb or structure in your character's body. When animating with FBIK, you can then set keys on the FBIK effectors and input joints in a body part to animate your character's limbs.

Body part membership

Body parts	FK joints	FBIK effectors
Hips	Hips	HipsEff
Spine	Spine Spine1-Spine9	NeckEff SpineEff

Body parts	FK joints	FBIK effectors
Head	Neck Neck1 -Neck9 Head	HeadEff
Arms	Shoulder (clavicle) Arm (shoulder) UpArmRoll (upper arm roll) ForeArm (elbow) ForeArmRoll (lower arm roll) Hand (wrist) FingerBase	ArmEff ForeArmEff HandEff FingerBaseEff
Legs	UpLeg (hips) UpLegRoll (upper leg roll) Leg (knee) LegRoll (lower leg roll) Foot (ankle) ToeBase (ball of foot)	UpLegEff LegEff FootEff ToeBaseEff
Fingers	Each finger is a body part	Each finger tip has an effector
Toes	Each toe is a body part	Each toe tip has an effector

Character and subcharacter sets

When you add full body IK to your character, each of your character's body parts are assigned to a *subcharacter set*, and all those subcharacter sets are then placed in a top-level *character set*. When animating with FBIK, the keys that you set on your character's selected FBIK effectors, input joints, body parts, or the entire character are then grouped under their specific subcharacter sets.

The FBIK character set and subcharacter sets help make sure that you are keyframing the right things to keep the FBIK animation *synchronized*. FBIK works best when you have synchronized keys on your character's FBIK effectors and FK joints.

NOTE By default, when [Skeleton > Full Body IK > Body Part Autoload](#) (page 74) is on, your character's current **Key Mode** determines which of its keys display in the **Time Slider**, **Graph Editor**, and **Dope Sheet**.

For example, when in All Key Mode, all the FBIK keys (IK and FK) for the selected FBIK effector or FK joint's character display. When in Body Part Key Mode, the FBIK keys for all the FBIK effectors and FK joints for the selected effector or FK joint's body part display. And when in Selected Key Mode, only the FBIK keys on the selected FBIK effector or FK joint display.

Body parts	Character or subcharacter sets
Hips	fbikCharacter
Spine	_spine
Head	_head
Arms	_leftArm / _rightArm
Legs	_leftLeg / _rightLeg
Fingers	_leftFingerThumb / _rightFingerThumb _leftFingerIndex / _rightFingerIndex _leftFingerMiddle / _rightFingerMiddle _leftFingerRing / _rightFingerRing _leftFingerPinky / _rightFingerPinky _leftFingerExtra / _rightFingerExtra
Toes	_leftFootThumb / _rightFootThumb _leftFootIndex / _rightFootIndex _leftFootMiddle / _rightFootMiddle _leftFootRing / _rightFootRing _leftFootPinky / _rightFootPinky _leftFootExtra / _rightFootExtra

Character animation with FBIK

How do I animate my characters with full body IK?

To animate with full body IK, you need to do the following:

- Pose your character's body parts by translating and rotating their FBIK effectors or input joints
- Set your character's FBIK **Key Mode** (Selected, Body Part, or All) and **Reach Mode** (Simple, FK, or IK)
- Set keys on the FBIK effectors or input joints you posed with Animate > Set Key or the Set Key hotkey (**s** key).

WARNING

- For Animate > Set Key or the S hotkey to work with FBIK, turn on **Set FullBodyIK keys** in the **Set Key Options (Animate > Set Key >)**.
- You cannot use **Autokey** when animating with FBIK.

NOTE If you use Key Selected to set FBIK keys on your character, only the currently selected objects' (FBIK effectors or FK joints) keys are loaded in the Time Slider when Body Part Autoload is off. Therefore, you can easily desynchronize your FBIK Input and FK skeletons when editing FBIK keys set with Key Selected in the Time Slider. For FBIK to work properly, your Input and FK skeletons need to be always synchronized.

When editing FBIK keys set with Key Selected in the **Time Slider**, you need to select the FBIK effectors *and/or* their corresponding FK joints on which the keys reside so that you are editing *all* the appropriate FBIK keys in the **Time Slider**. This will keep your Input and FK skeletons synchronized.

Pose characters with full body IK

To pose your character with full body IK, you need to transform your character's FBIK effectors (for IK type behavior) and input joints (for FK type behavior) while pinning and unpinning your effectors when needed. See [Pin and unpin your character's effectors](#) (page 35).

You can also use auxiliary pivots and floor contacts for your characters to aid and enhance their FBIK poses. See [Create auxiliary pivots](#) (page 55) and [Create floor contacts for your character's hands and feet](#) (page 61).

NOTE

Key Mode: Body Part isolates posing of your character to individual limbs or body parts. For example, if **Body Part** is the current **Key Mode** for your biped character, and you translate the RightHandEff, then the new pose of your character's right arm does not affect or change the poses of any of its other limbs.

Key FBIK effectors and joints

Setting the **Key Mode** and **Reach Mode** for your FBIK keys determines where your FBIK keys are placed on your character and whether your character's FBIK animation is more FK or more IK at any given point during its animation.

When animating your character with FBIK, you can select the **Key** and **Reach** modes for your keys from the FBIK marking menu or the [Animate > Set Full Body IK Keys](#) (page 69) window.

WARNING

- For Animate > Set Key or the S hotkey to work with FBIK, turn on **Set FullBodyIK keys** in the **Set Key Options** (**Animate > Set Key >**).
 - You cannot use **Autokey** when animating with FBIK.
-

To set FBIK keys

- 1 Pose your character.
- 2 Select an appropriate **Reach Mode** for your character. See [Reach Mode](#) (page 69).
 - The IK mode sets the **Reach** channel values for your FBIK effectors to 1 and makes sure they are keyed when you set your FBIK keys. IK mode sets that your effectors are completely controlled by IK.
 - The FK mode sets the **Reach** channel values for your FBIK effectors to 0 and makes sure they are keyed when you set your FBIK keys. FK mode disables your effectors' IK; allowing the underlying FK skeleton's joints to drive your character's animation.
 - The **Simple Key** mode does not set the **Reach** channel values of the keyed FBIK effectors; the **Reach** values do not change and are not keyed.

- 3 Select the desired **Key Mode** for your character. See [Key Mode](#) (page 69).
 - The **Selected** mode sets keys on the input joints and FBIK effectors you select.
 - The **Body Part** mode sets keys on the FBIK effector's and input joints of your character's body parts.
 - The **All** mode sets keys on the FBIK effectors and input joints of your entire character.
- 4 Select the FBIK effectors or input joints you want to key and select Animate > Set Key or press s (the Set Key hotkey) to set keys on them.

NOTE

- FBIK works best when you have synchronized keys on the FK skeleton and FBIK effectors. You can *only* set synchronized keys when **Set FullBodyIK keys** is on in the **Set Key Options** (Animate > Set Key >).
- You can set advanced options in your FBIK solution to enhance and refine your character's poses and animation. See hikHandle.
- For Animate > Set Key or the S hotkey to work with FBIK, turn on **Set FullBodyIK keys** in the **Set Key Options**.

IK and FK Reach

You should use *IK* Reach on your character when:

- You need to hold the positions of your character's body parts during its animation's interpolation
- You need to ensure that its body parts do not deviate from their positions at which their keys were set.

For example, keying your character's foot placements when they contact the floor during a walk cycle. In this instance, setting keys on the FootEff effectors prevents undesirable foot sliding.

You should set *FK* Reach on your character when:

- You need to release its body parts from their held positions
- You need to key its positions, but the positions do not have to be exact during its animation's interpolation.

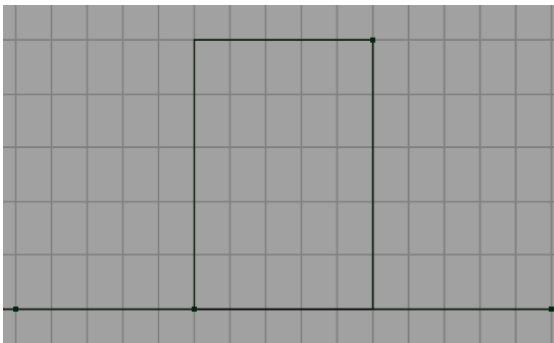
For example, keying your character's FootEff when its legs are off the floor swinging through the air during a walk cycle. In the instance, you need to key the movements of your character's legs, but your character's feet do not have stick to any one position.

FBIK Keys

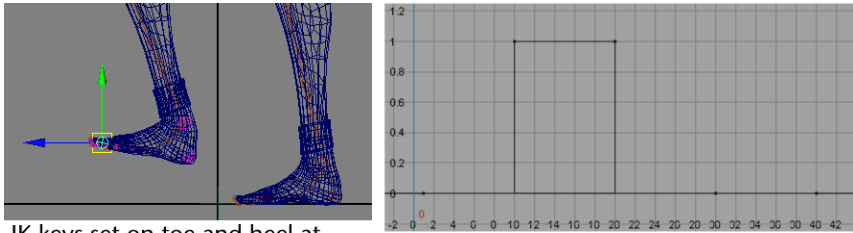
Stepped Next

FBIK animation keys use the special *stepped next* tangent type by default. This tangent type differs from regular stepped tangents in that its interpolation values jump immediately to that of the next key, rather than holding the value of the current key until the next key is reached. This tangent is useful because it lets you key your character's FBIK effectors as you manipulate them, so that you do not have to plan your animation ahead or think about what is the right effector to select and manipulate to get your character into its next pose. You do not have to actively manage your FBIK animation curve tangency, as the automatically done by the FBIK system.

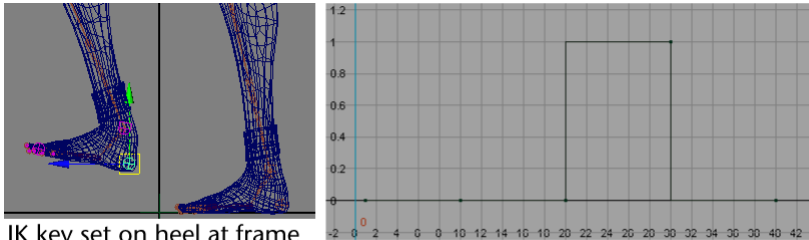
For example, if you used regular stepped tangents for a walk cycle that you are animating with FBIK, you would have to set keys on *each* joint and FBIK effector that translates or rotates as it moves from one pose to another (such as the rotations of the auxiliary pivots at the heels of your character's feet or the translations of the effectors at your character's ankles). But if you used stepped next tangents for the same FBIK animation keys, you would only have to set keys on your body parts once for each pose, not each individual FBIK effector or joint.



Stepped Next tangents



IK keys set on toe and heel at frame 30



IK key set on heel at frame 40

FBIK keys in the Time Slider, Graph Editor, and Dope Sheet

You can edit the positions and timing of your FBIK keys with the **Graph Editor** and **Dope Sheet**. However, when you need to view or delete your FBIK keys, use the **Time Slider**.

Deleting your character's FBIK keys from the **Time Slider**—not the **Graph Editor**—keeps your character's IK and FK skeletons *synchronized* during its animation interpolation. Keeping your IK and FK skeletons synchronized is essential to creating good, desirable FBIK animation results. You can delete your FBIK keys from the **Time Slider** with the **Delete FBIK Keys** menu in the **Time Slider** context sensitive menu. See Animation controls menu in the Animation guide.

When you select a keyed FBIK effector or input joint, only its keys display in the **Time Slider**, **Graph Editor**, and **Dope Sheet** by default. However, when **Skeleton > Full Body IK** (page 71) > **Body part Autoload** is *on*, the current Key Mode determines which keys display. For example, when in All Key Mode, all the FBIK keys (IK and FK) for the selected FBIK effector or input joint's character display. When in Body Part Key Mode, the FBIK keys for all the FBIK effectors and input joints for the selected effector or joint's body part display. And when in Selected Key Mode, only the FBIK keys on the selected FBIK effector or input joint display.

NOTE

When [Skeleton > Full Body IK](#) (page 71) > **Body part Autoload** is *on*, and Body Part or All Key Mode is on, then the selected FBIK effector or input joint's subcharacter set is automatically loaded as the **Current Character**.

Create and use full body IK

Full body IK workflow

Each of the following steps are mandatory when using FBIK in Maya unless otherwise specified.

- 1 [Prepare your character for full body IK](#) (page 2).
 - 2 [Label or rename your character's joints](#) (page 4).
 - 3 [Create the full body IK system for your character](#) (page 31).
 - 4 [Pose characters with full body IK](#) (page 25).
See also [Pin and unpin your character's effectors](#) (page 35).
 - 5 Key your character's movements with FBIK.
See [How do I animate my characters with full body IK?](#) (page 25) and [FBIK Keys](#) (page 28).
- (Optional) [Create auxiliary pivots](#) (page 55).
 - (Optional) [Create floor contacts for your character's hands and feet](#) (page 61).
 - (Optional) [Create auxiliary effectors](#) (page 57)

TIP

To help you get started with FBIK, we have provided an example biped and quadruped FBIK character. You can access them from [Skeleton > Full Body IK](#) (page 71) > **Get FBIK Example** or the **FBIK Examples** tab in the **Visor**.

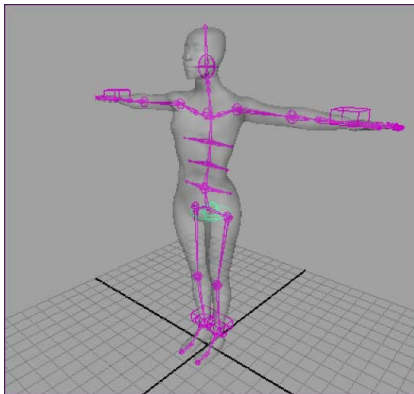
Create the full body IK system for your character

Before you can pose your character with full body IK, you first need to *characterize* it. When you characterize a character, a *full body IK system* is created. The full body IK system includes: the hikSolver, hikHandle, and a set of FBIK effectors. The hikSolver controls the FBIK solution, the hikHandle drives the joints and bones in your character's skeleton, and the FBIK effectors let you pose your character's limbs.

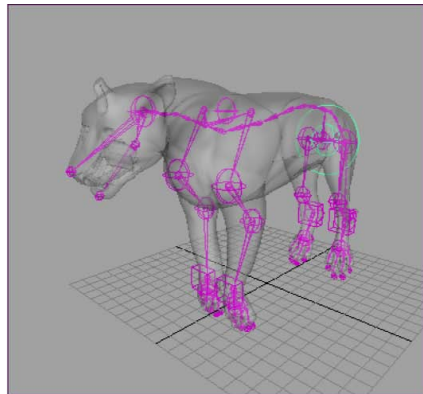
To characterize your character

- 1 Select the root joint of your character's skeleton.
- 2 Do one of the following:
 - Select **Skeleton > Full Body IK > Add Full Body IK** (page 71).
 - In the **Script Editor** or **Command Line** type `characterize;` and then press on the number pad of your keyboard.

A set of *FBIK effectors* for your character appears in the scene view. These effectors are manipulators that you can transform and key to pose and animate your characters with full body IK in Maya.



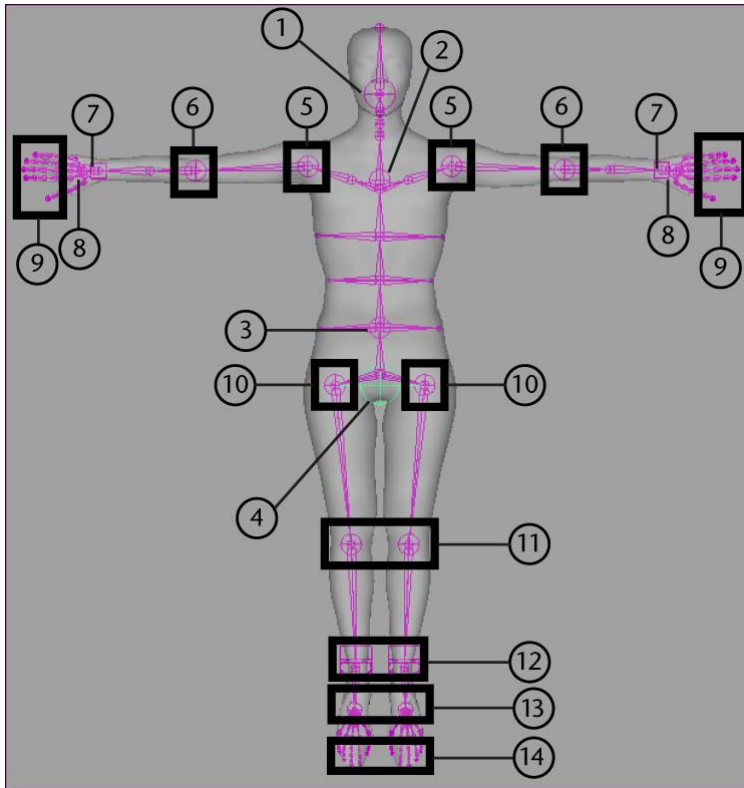
Biped FBIK effectors



Quadruped FBIK effectors

Each FBIK effector is positioned over the joint it drives. FBIK effectors appear as spheres around your character's joints. By default, your character's wrist (LeftHandEff/RightHandEff) effectors are pinned in translation, and their ankle (LeftFootEff/RightFootEff) effectors are pinned in translation and rotation. See [Pin and unpin your character's effectors](#) (page 35).

To change the size of a FBIK effector, select the effector and in the **Attribute Editor** set its **Radius** attribute located in the **HIK Effector Attributes** section of its *JointNameEff* tab.



	Joint labels	Joint names	FBIK Effector names
1	Head (last joint in neck at the base of head)	Head	HeadEff
2	Spine (last spine joint before the neck)	Spine#	NeckEff
3	Spine	Spine	SpineEff

	Joint labels	Joint names	FBIK Effector names
4	Root	Hips	HipsEff
	-	HipsTranslation	HipsTranslationEff
5	Shoulder (L)	LeftArm	LeftArmEff
	Shoulder (R)	RightArm	RightArmEff
6	Elbow (L)	LeftForeArm	LeftForeArmEff
	Elbow (R)	RightForeArm	RightForeArmEff
7	Hand (L)	LeftHand	LeftHandEff
	Hand (R)	RightHand	RightHandEff
8	Finger Base (L)	LeftFingerBase	LeftFingerBaseEff
	Finger Base (R)	RightFingerBase	RightFingerBaseEff
9	Thumb (L)	LeftHandThumb#	LeftHandThumb1Eff
	Thumb (R)	RightHandThumb#	RightHandThumb1Eff
	Index (L)	LeftHandIndex#	LeftHandIndex1Eff
	Index (R)	RightHandIndex#	RightHandIndex1Eff
	Middle Finger (L)	LeftHandMiddle#	LeftHandMiddle1Eff
	Middle Finger (R)	RightHandMiddle#	RightHandMiddle1Eff
	Ring Finger (L)	LeftHandRing#	LeftHandRing1Eff

	Joint labels	Joint names	FBIK Effector names
	Ring Finger (R)	RightHandRing#	RightHandRing1 Eff
	Pinky (L)	LeftHandPinky#	LeftHandPinky1 Eff
	Pinky (R)	RightHandPinky#	RightHandPinky1 Eff
	-	LeftHandExtraFinger#	LeftHandExtra1 Eff
	-	RightHandExtraFinger#	RightHandExtra1 Eff
10	Hip (L)	LeftUpLeg	LeftUpLegEff
	Hip (R)	RightUpLeg	RightUpLegEff
11	Knee (L)	LeftLeg	LeftLegEff
	Knee (R)	RightLeg	RightLegEff
12	Foot (L)	LeftFoot	LeftFootEff
	Foot (R)	RightFoot	RightFootEff
13	Toe Base (L)	LeftToeBase	LeftToeBaseEff
	Toe Base (R)	RightToeBase	RightToeBaseEff
14	Big Toe (L)	LeftFootThumb#	LeftFootThumb1 Eff
	Big Toe (R)	RightFootThumb#	RightFootThumb1 Eff
	Index Toe (L)	LeftFootIndex#	LeftFootIndex1 Eff
	Index Toe (R)	RightFootIndex#	RightFootIndex1 Eff

	Joint labels	Joint names	FBIK Effector names
	Middle Toe (L)	LeftFootMiddle#	LeftFootMiddle1 Eff
	Middle Toe (R)	RightFootMiddle#	RightFootMiddle1 Eff
	Ring Toe (L)	LeftFootRing#	LeftFootRing1 Eff
	Ring Toe (R)	RightFootRing#	RightFootRing1 Eff
	Pinky Toe (L)	LeftFootPinky#	LeftFootPinky1 Eff
	Pinky Toe (R)	RightFootPinky#	RightFootPinky1 Eff
	-	LeftFootExtraToe#	LeftFootExtra1 Eff
	-	RightFootExtraToe#	RightFootExtra1 Eff

Pin and unpin your character's effectors

You can use *pinning* to hold or limit the movements of your character's body parts when posing with FBIK. Pinning allows you to lock the translation or rotation channels of specific effectors; so that when you move unpinned body parts, the pinned body parts do not follow. For example, you can pin a character's foot effectors in translation and rotation when posing a walk cycle to prevent foot sliding.

When you first create the FBIK effectors for your character, by default, your character's feet are pinned in translation and rotation and your character's hands are pinned in translation. This prevents your character's limbs from being moved or dragged when you move the rest of the body.

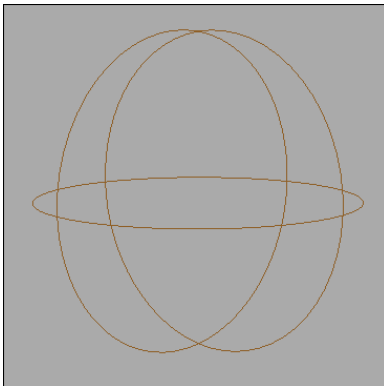
NOTE

- Pinning is not a constraint. Pinning only restricts the movements of your effectors during *interaction*. If you want an a body part to stay in place during *playback*, then you need to set an IK key on that body part's FBIK effector and FK joints.
- Do not pin your finger or toe effectors, as pinning these effectors has no effect on the FBIK solution.

You can set the **Pinning** attribute on each FBIK effector to the following states: **unpinned**, **pinTranslate**, **pinRotate**, or **pinAll**. You can set the pinning states of your FBIK effectors from the **Channel Box**, the effector's marking menus, or by using the +w (pin translation), +e (pin rotation), or +s (pin translation and rotation) hotkeys.

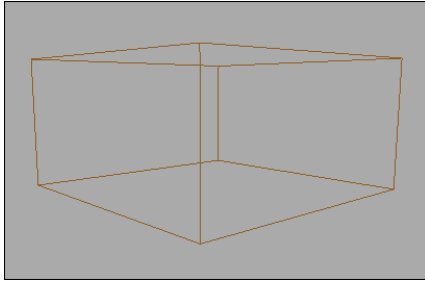
Pinning states

unpinned



When a FBIK effector is *unpinned*, movement in other areas of the body can affect or change the translation and rotation of the unpinned effector.

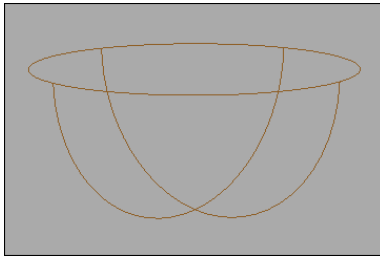
pinTranslate



When a FBIK effector is pinned in *translation*, movement in other areas of the body can only affect or change the effector's rotations.

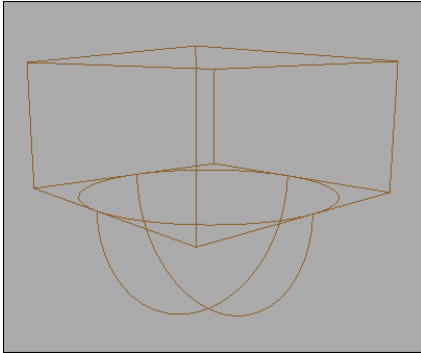
By default, the *LeftHandEff* and *RightHandEff* are pinned in translation.

pinRotate



When a FBIK effector is pinned in *rotation*, movement in other areas of the body can only affect or change the translation of the effector.

pinAll



When a FBIK effector is pinned in *translation* and *rotation*, movement in other areas of the body can *not* affect or change the effector's transformations.

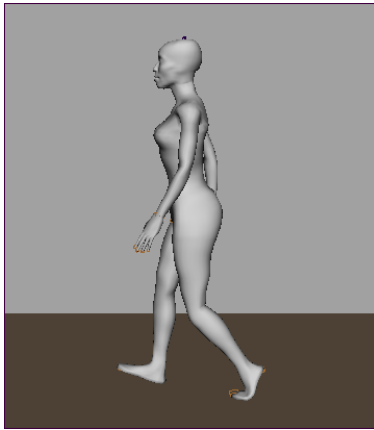
By default, the *LeftFootEff* and *RightFootEff* are pinned in both translation and rotation.

How do I use pinning when animating with FBIK?

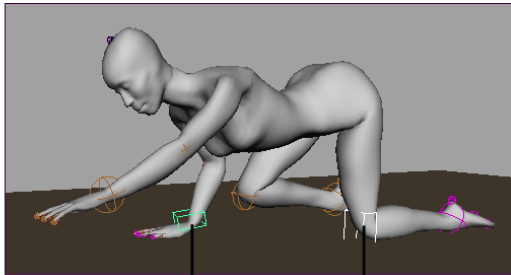
When animating with FBIK, you should use pinning anytime you want to keep part of your character still while posing or when you want to isolate the orientation and position of one or more body parts while repositioning the rest of the character.

Example: Pinning your character's effectors

- Pin the feet of your character during a walk cycle.



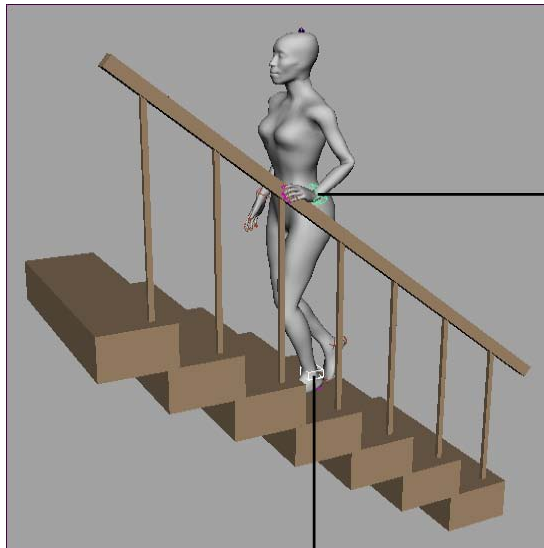
- Pin the knees and hands of a character as it crawls on the floor.



RightHandEff pinned in translation

LeftLegEff pinned in translation

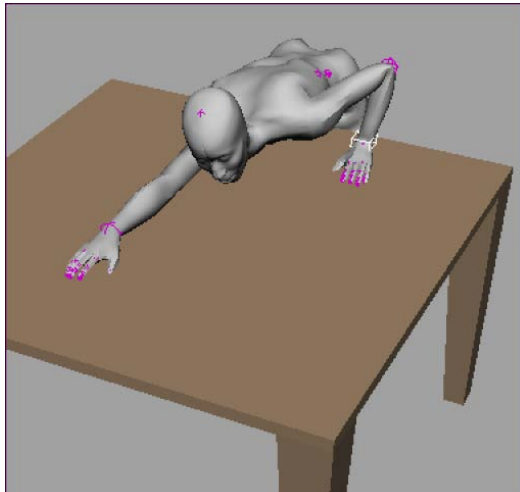
- Pin the feet and hands of a character as it is holding a railing while climbing stairs.



LeftHandEff pinned in translation and rotation

LeftFootEff pinned in translation

- Pin the hips, wrists and feet (when needed) of a character as it reaches over a desk.



HipsEff, LeftHandEff, LeftFootEff and RightFootEff pinned in translation

Example workflows: Full body IK

The following examples demonstrate just a few ways you can use full body IK to quickly and easily pose and animate your characters.

- [Example 1: Bending over to touch the floor](#) (page 42).
- [Example 2: Crawling on hands and knees across the floor](#) (page 46).
- [Example 3: Walking across the floor](#) (page 50).

TIP

- If you want your character's FB IK solution to be influenced by a FB IK effector during playback, you need to set the effector's **Reach Mode** to **IK** and then key it. Typically, this technique is used to keep certain parts of a character's body still during playback. See [Reach Mode](#) (page 69).
- If you want more control when posing your characters, rotate their Input skeletons' joints rather than just manipulating their FB IK effectors.
- When keying your poses, select **Key Mode: All** to key your character's whole body and select **Key Mode: Body Part** to set keys on only the body part(s) you have selected (such as individual limbs).
- At times your FB IK effectors may not stay with your character during playback. Do not be concerned because this is normal FB IK behavior and it does not affect your FB IK animation. To return the FB IK effectors to their proper positions, go to a frame in your character's animation where the effectors move away from the character, select a FB IK effector or an Input skeleton joint, and move it slightly. This should cause the effectors that have moved away from the character to pop back to their proper positions.

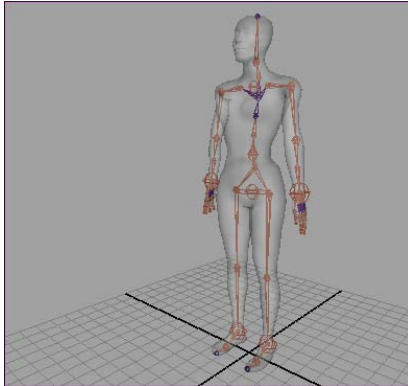
WARNING

- For **Animate > Set Key** or the **Set Key** hotkey (**s** key) to work with FB IK, *Set FullBodyIK keys* needs to be *on* in **Animate > Set Key >** .
- FB IK works the best when you have synchronized keys on the FK skeleton and FB IK effectors. You can *only* set synchronized keys with [Animate > Set Full Body IK Keys](#) (page 69) or when **Animate > Set Key > > Set FullBodyIK keys** is *on*.
- You cannot use **Autokey** when animating with FB IK.

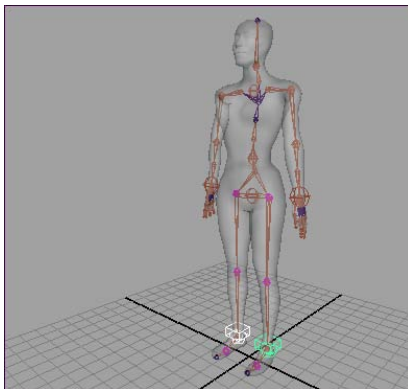
Example 1: Bending over to touch the floor

To pose and animate a character bending over to touch the floor

- 1 Pose your character so that it is standing at its initial pose.



- 2 Pin the LeftFootEff and RightFootEff effectors in translation and rotation.



- 3 Set your character's **Reach Mode** to **IK** and **Key Mode** to **All**. You can set reach and key settings from [Animate > Set Full Body IK Keys](#) (page 69) > or the FBIK marking menu.

This sets all your character's **Reach** values to 1 and puts the character into All keying mode.

- 4 Select [Animate > Set Key](#) > .

The **Set Key Options** window appears.

- 5 Turn *on* **Set FullBodyIK keys**, click **Apply**, and then click **Close**.

6 Press s (Set Key hotkey).

This sets keys on all the FIK effectors and FK joints in your character.

Your character's foot effectors now influence your character's FIK solution during playback. Also, setting IK keys like this keeps your character's feet perfectly still during playback. This is useful because you do not want your character's feet to move during its animation. If you had instead set FK keys, your character's feet would slide around during playback rather than stay still.

Throughout this example you will be repeating this step frequently to make sure that your character's feet stay still during each segment of the animation.

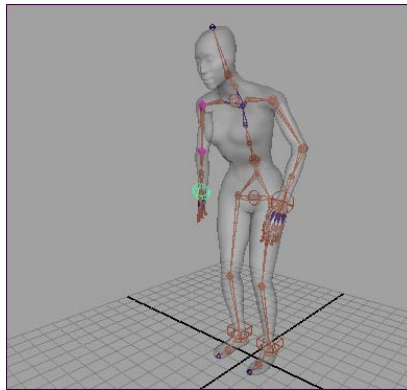
7 Move forward in time.

- Manipulate the character's back so that it is slightly hunched.

You can create this pose by rotating the SpineEff and NeckEff effectors or by rotating the spine joints of your character's Input skeleton directly.

- Translate the RightHandEff effector slightly downward so that it is reaching towards the floor.

8 Press s (Set Key hotkey).



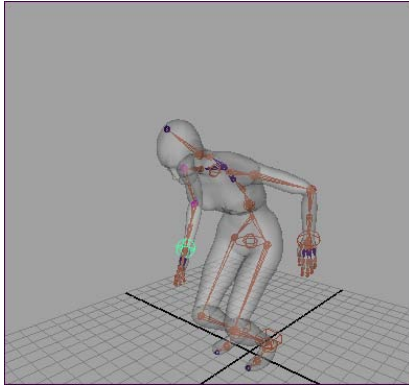
9 Move forward in time.

- Translate the HipsEff effector downward.

Your character's knees are now bending.

- Manipulate the character's back so that it is more hunched.
- Translate the RightHandEff effector downward some more.

10 Press s (Set Key hotkey).



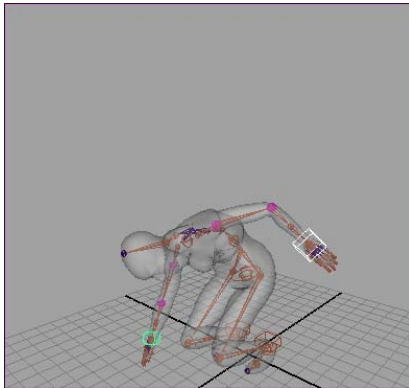
11 Move forward in time.

- Pin the LeftHandEff effector in translation.

This prevents the LeftHandEff from swinging backward as the RightHandEff continues to move downward.

- Translate the RightHandEff effector downward until it reaches the floor.

12 Press s (Set Key hotkey).



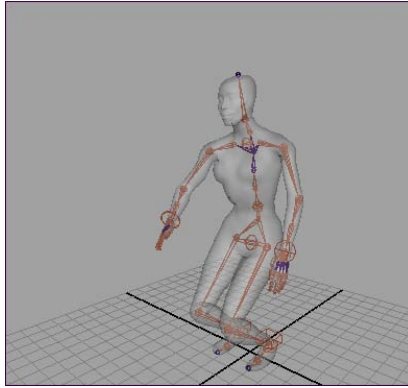
13 Move forward in time.

- Unpin the LeftHandEff effector.
- Translate the HipsEff effector upward.

Your character's knees are now less bent.

- Translate the RightHandEff effector upward.
- Manipulate your character's back so that it is less hunched.

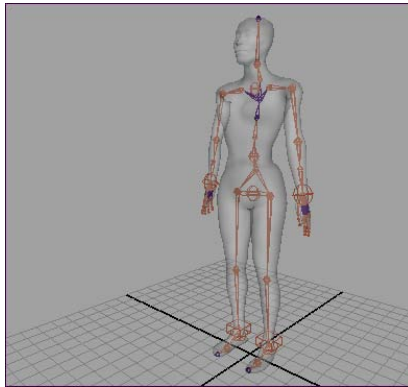
14 Press s (Set Key hotkey).



15 Move forward in time.

- Manipulate the character's back so that it is no longer hunched.
- Translate the HipsEff effector upward until your character's knees are no longer bent.
- Translate the LeftHandEff and RightHandEff effectors until both the arms are back at the sides of the character.

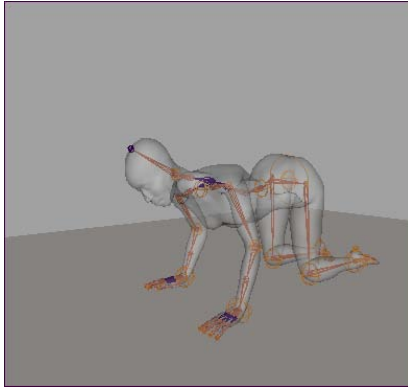
16 Press s (Set Key hotkey).



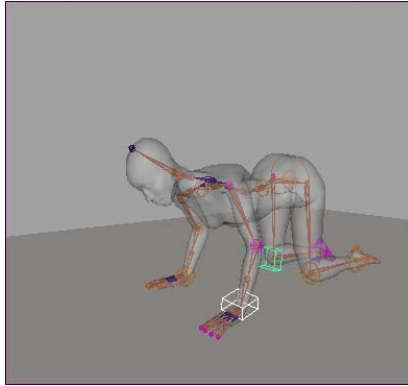
Example 2: Crawling on hands and knees across the floor

To pose and animate a character crawling on its hands and knees across the floor

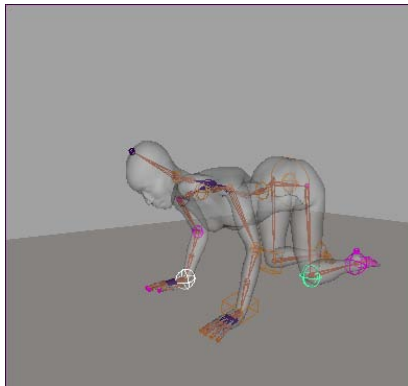
- 1 Pose the character so that it is on its hands and knees at its initial crawl pose.



- 2 Select **Animate > Set Key >** .
The **Set Key Options** window appears.
- 3 Turn on **Set FullBodyIK keys**, click **Apply**, and then click **Close**.
- 4 Pin the **LeftHandEff** and **RightLegEff** FBIK effectors in translation.
LeftHandEff and RightLegEff effectors will now not move when you manipulate the RightHandEff or LeftLegEff effectors.
Throughout this example you will be repeating this step frequently to make sure that your character's left hand and right knee stay still during the segments of animation when the right hand and left knee are moving.

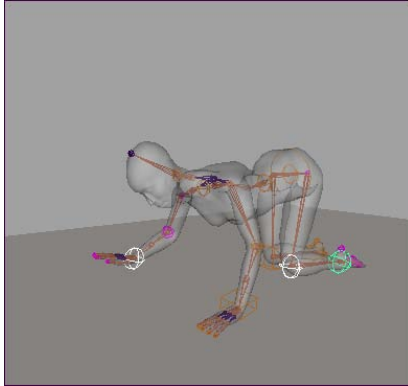


- 5 Set your character's **Reach Mode** to **IK** and **Key Mode** to **All**. You can set reach and key settings from [Animate > Set Full Body IK Keys](#) (page 69) > or the FBIK marking menu. This sets all your character's **Reach** values to 1 and puts the character into All keying mode.

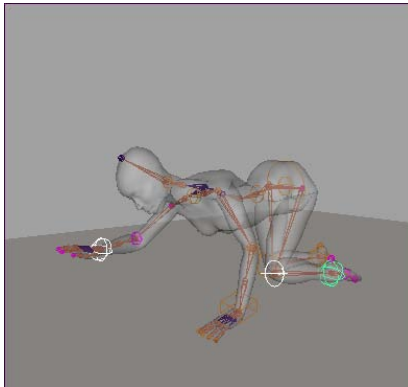


- 6 Press s (Set Key hotkey). Your character's left hand and right knee effectors now influence your character's FBIK solution during playback. Also, setting IK keys like this keeps your character's left hand and right knee still during playback. This is useful because you do not want your character's left hand and right knee effectors to move when its right hand and left knee effectors are moving. If you had instead set FK keys, your character's left hand and right knee would shift around during playback rather than stay still.

- 7 Move forward in time.
 - Translate the RightHandEff, LeftLegEff and LeftFootEff effectors slightly upward and forward.
- 8 Press s (Set Key hotkey).



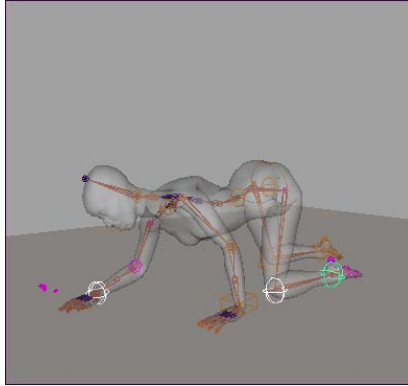
- 9 Move forward in time.
 - Translate the HipsEff effector forward.
 - Translate the RightHandEff, LeftLegEff and LeftFootEff effectors upward and forward some more.
- 10 Press s (Set Key hotkey).



- 11 Move forward in time.
 - Translate the HipsEff effector forward.
 - Translate the RightHandEff, LeftLegEff and LeftFootEff effectors forward and then down to the floor.

12 Press s (Set Key hotkey).

This step is important because keying all these effectors keeps the left hand and right knee still, and prepares your character for the next frame where you want to keep the right hand and left knee still while the left hand and right knee begin their segment of the crawl animation.



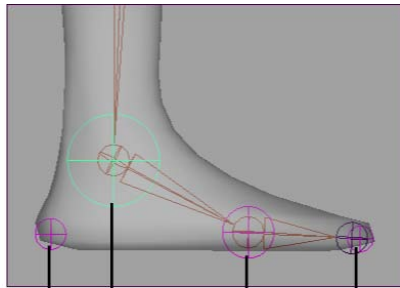
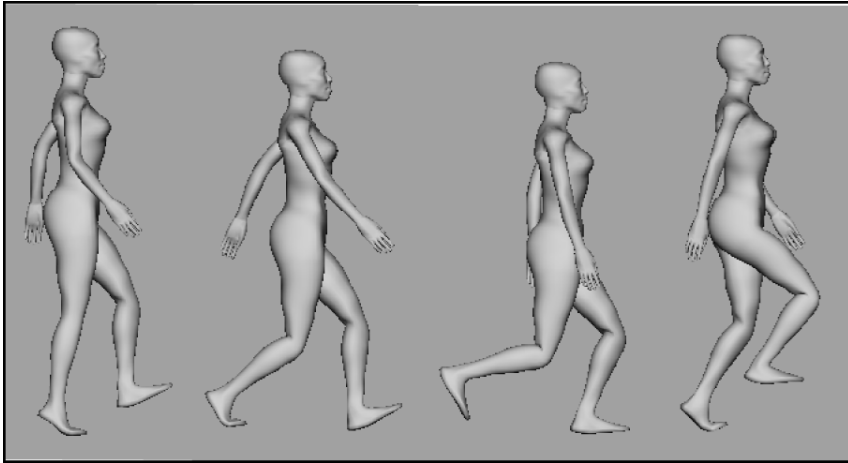
13 Move forward in time.

- Repeat steps 7-12 with the LeftHandEff, RightLegEff and RightFootEff effectors to complete the crawl animation. This will complete your crawl animation.

TIP

- You can use Simple Keys to tweak or fine tune the positions of your character's body parts between keyframes. For example, If the placement of your character's right hand is not quite right during certain segments of its animation, you can use **Reach Mode: Simple** with **Key Mode: Body Part** to correct the positioning of the right hand without changing your character's **Reach** settings.
- If your character's body twists or moves undesirably when you translate its Left or Right LegEff effectors, you can prevent this twisting by pinning your character's hips in translation and rotation.

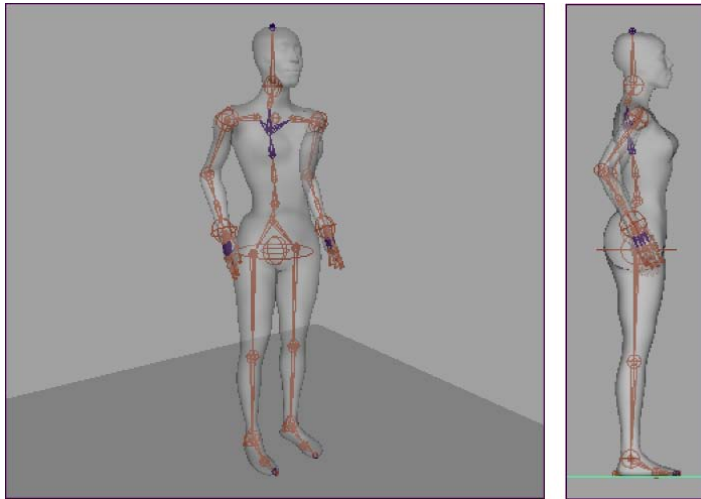
Example 3: Walking across the floor



FootEffPivot1 FootEff ToeBase FootEffPivot
Character foot setup for this example

To pose and animate a character walking

- 1 Pose the character so that it is at its initial pose.



- 2 Set your character's **Reach Mode** to **IK** and **Key Mode** to **All**. You can set reach and key settings from [Animate > Set Full Body IK Keys](#) (page 69) > or the FBIK marking menu.

This sets all your character's **Reach** values to 1 and puts the character into All keying mode.

- 3 Select Animate > Set Key > .

The **Set Key Options window** appears.

- 4 Turn *on* **Set FullBodyIK keys**, click **Apply**, and then click **Close**.

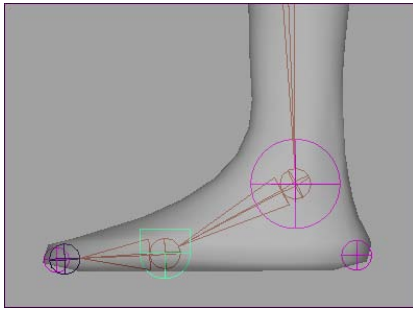
- 5 Press s (Set Key hotkey).

Keys are set on all the FBIK effectors and FK joints in your character.

- 6 Pin the LeftToeBaseEff FBIK effector in translation and rotation.

The LeftToeBaseEff effector now will not shift or slide when you pose your character's right foot.

Throughout this example you will be repeating this step frequently to make sure that when one of your character's feet is on the floor, it does not slide around when the other foot is moving.



7 Move forward in time.

- Translate the HipsEff effector forward.
- Translate the LeftFootEff effector upward.

This creates a heel lift motion for the left foot.

- Translate the RightFootEff upward and forward.

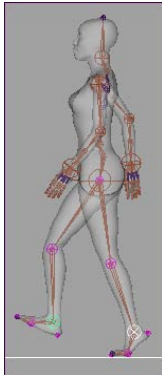
This creates a lifting motion, and the beginning of a swinging motion, for the right foot.

- Rotate the RightFootEff so that the toe of the right foot is pointing slightly upward.

This prepares the right foot for its heel strike motion.

- In the Input skeleton, rotate the RightArm (left shoulder) joint so that the right arm swings slightly backward and rotate the LeftArm (left shoulder) joint so that the left arm swings slightly forward.

This begins the arm swinging motion for the walk animation.



8 Press s (Set Key hotkey).

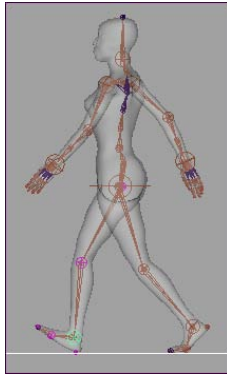
Keys are set on all the FBIK effectors and FK joints in your character.

9 Move forward in time.

- Translate the HipsEff effector forward.
- Translate the RightFootEff effector downward so that the heel touches the floor.

This creates the heel strike motion for the right foot.

- Rotate the LeftArm joint so that the left arm swings forward some more and rotate the RightArm joint so that the right arm swings backward some more.

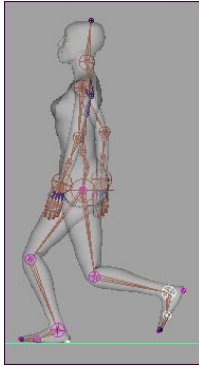


10 Press s (Set Key hotkey).

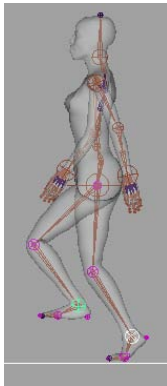
Keys are set on all the FBIK effectors and FK joints in your character.

11 Move forward in time.

- Translate the HipsEff effector forward.
- Rotate the RightFootEffPivot1 effector so that the right foot is now flat on the floor.
- Unpin the LeftToeBaseEff effector, translate the LeftFootEff effector upward and rotate the LeftToeBaseEff to straighten out the left foot.
- Rotate the LeftArm joint so that the left arm swings backward slightly and rotate the RightArm joint so that the right arm swings forward slightly.



- 12 Press s (Set Key hotkey).
Keys are set on all the FBlK effectors and FK joints in your character.
- 13 Move forward in time.
 - Repeat steps 4-6 so that the right leg swings forward, the left foot lifts at its heel and rolls from its ball, and so on. This will complete your walk animation.



TIP

You can use Simple Keys to tweak or fine tune the positions of your character's body parts between keyframes. For example, you can use **Reach Mode: Simple** with **Key Mode: Body Part** to add more detail to your character's walk cycle—such as translating the hips along the Y-axis to add more bounce to its step—without changing your character's Reach settings.

Create auxiliary pivots

You can add additional pivot points to your FBK effectors using *auxiliary pivots*. Auxiliary pivots allow you to transform your FBK effectors from different points in 3D space to create specialized motions such as a foot roll during a walk cycle, the ball of a foot putting out a cigarette, a body spinning head over heels while flying through the air, and so on.

NOTE

Pinning auxiliary pivots has the same effect as pinning their parent effectors. For example, if you pin a FootEffPivot, the FootEff actually gets pinned—not the FootEffPivot.

To create auxiliary pivots

- 1 Select the HipsEff effector and select [Skeleton > Full Body IK > Go to Stance Pose](#) (page 73).
- 2 Select the effector for which you want to add a secondary pivot and then select [Skeleton > Full Body IK > Add Auxiliary Pivot](#) (page 72).

A auxiliary pivot is created for the foot FBK effector. The name of the auxiliary pivot is the same as its parent effector except for the suffix *Pivot*. Also, each subsequent effector is given a number. For example, the first auxiliary pivot for LeftFootEff would be named *LeftFootEffPivot* and the second auxiliary pivot would be named *LeftFootEffPivot1*.

- 3 Select the auxiliary pivot, scale it as desired by setting its **Radius** value in the **Channel Box**, and then translate it to the position from which you want its parent effector to rotate.
- 4 With the auxiliary pivot selected, select [Skeleton > Full Body IK > Activate Auxiliary Pivot](#) (page 72).

Now when you transform the auxiliary pivot, its parent effector will rotate from the position of its active auxiliary pivot.

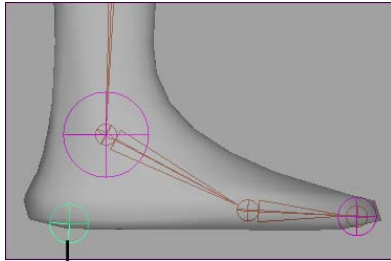
Example: Biped foot rolls

To create good foot roll behavior with auxiliary pivots when animating a walk cycle for your biped character, you need to create two auxiliary pivots for your LeftFootEff and RightFootEff effectors: one auxiliary pivot for the heel and the other for the ball of the foot.

- 1 Select your LeftFootEff FBK effector and then select [Skeleton > Full Body IK > Add Auxiliary Pivot](#) (page 72).

A auxiliary pivot called LeftFootEffPivot appears at the position of LeftFootEff.

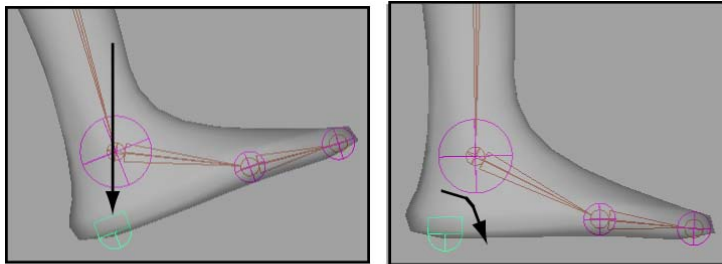
- 2 Select LeftFootEffPivot, scale it as desired by setting its Radius value in the Channel Box, and translate it to the heel of your character's left foot.



LeftFootEffPivot

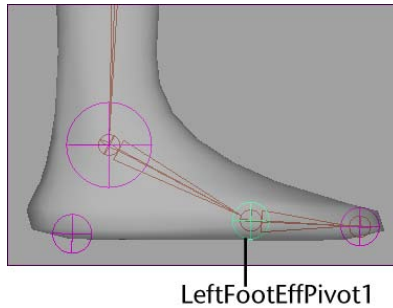
- 3 With LeftFootEffPivot still selected, select [Skeleton > Full Body IK > Activate Auxiliary Pivot](#) (page 72).

The LeftFootEff will now rotate from the position of LeftFootEffPivot (heel), producing a heel strike and downwards foot roll motion from the ankle.



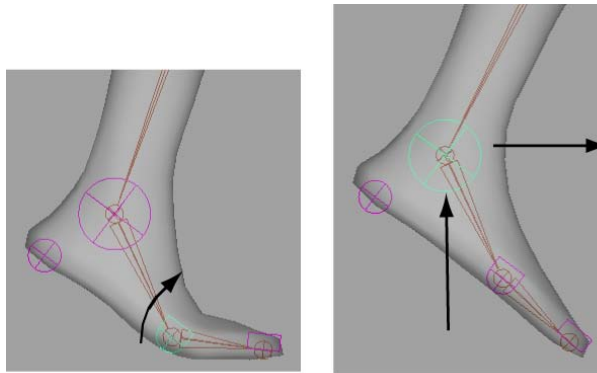
When the character's foot meets the ground, it first strikes at the heel and then rolls from the heel.

- 4 Select LeftFootEff and create another auxiliary pivot.
A auxiliary pivot called LeftFootEffPivot1 appears at the position of LeftFootEff.
- 5 Select LeftFootEffPivot1 and translate it to the ball of your character's left foot.



- 6 With LeftFootEffPivot1 still selected, select **Skeleton > Full Body IK > Activate Auxiliary Pivot** (page 72).

The LeftFootEff will now also rotate from the position of LeftFootEffPivot1 (ball of the foot), bending the foot at the ball and producing a pushing off motion in the foot.



After the heel motion the foot bends/rotates at the ball and then pushes off the ground to propel the character forward.

- 7 Repeat steps 1-6 for the RightFootEff.

Create auxiliary effectors

Auxiliary effectors are supplementary effectors in a FBIK system that correspond to existing FBIK effectors. Every FBIK effector in a FBIK system can have multiple auxiliary effectors. You can translate and rotate an auxiliary effector to the position you want its corresponding FBIK body part to reach, then control the amount of reach using the auxiliary effector's **Reach** channel

values. Auxiliary effectors are useful in situations where you want an additional level of FBIK control.

To create auxiliary effectors

- 1 In the scene view, select the full body IK effector for which you want to create an auxiliary effector.

- 2 Select **Skeleton > Full Body IK > Add Auxiliary Effector** (page 71).

The auxiliary effector appears at the same Translate and Rotate XYZ coordinates as its corresponding FBIK effector.

Auxiliary effectors are named after their target FBIK effectors with the suffix *Aux* plus a sequential number. For example, if you create three auxiliary effectors for the RightFootEff effector, the first auxiliary effector is named RightFootEffAux, the second is named RightFootEffAux1, and the third is named RightFootEffAux2.

- 3 Translate the auxiliary effector to the position in XYZ space you would like its corresponding FBIK body part to reach.

- 4 In the **Channel Box**, change the auxiliary effector's **Reach Translation** and **Reach Rotation** channel values to 0 and its corresponding FBIK effector's **Reach Translation** and **Reach Rotation** channel values to 0.

Key the **Reach Translation** and **Reach Rotation** channels for the auxiliary effector and its corresponding FBIK effector.

- 5 Move the Time Slider forward in time to the point where you want the FBIK body part to reach the position of its auxiliary effector.

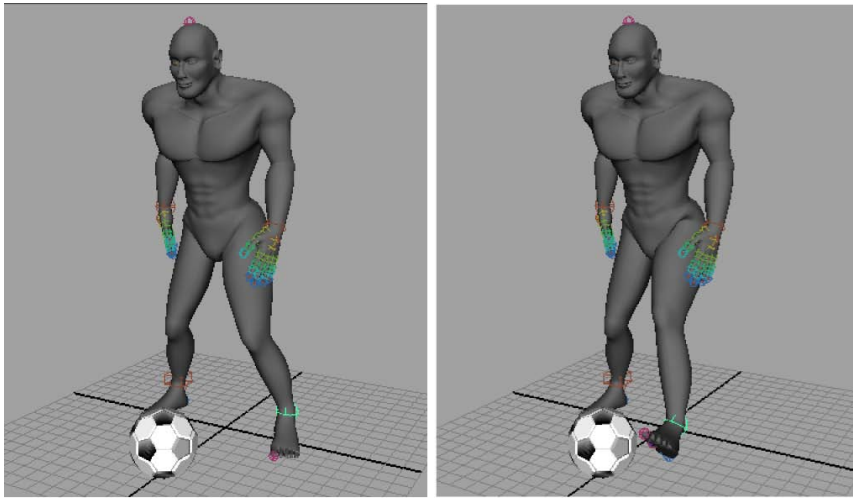
- 6 In the **Channel Box**, change the auxiliary effector's **Reach Translation** and **Reach Rotation** channel values to 1.

Key the **Reach Translation** and **Reach Rotation** channels for the auxiliary effector and its corresponding FBIK effector.

The FBIK effector will now move to the position of its auxiliary effector during its FBIK animation and its corresponding body part will reach towards the auxiliary effector.

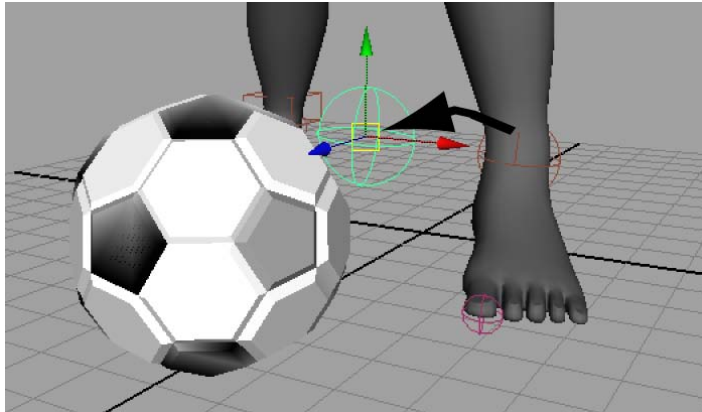
Example: Foot moving towards soccer ball

To place an auxiliary effector in a soccer ball and link it to the RightFootEff of your character and then animate its **Reach** channel values so that its right foot meets the soccer ball for a kick.



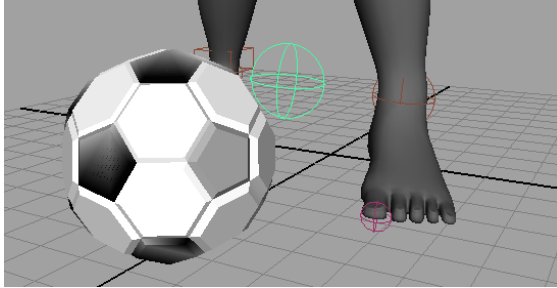
Before auxiliary effector

- 1 In the scene view, select your RightFootEff FBIK effector and then select [Skeleton > Full Body IK > Add Auxiliary Effector](#) (page 71).
An auxiliary effector named *RightFootEffAux* appears at the same Translate and Rotate XYZ coordinates as the RightFootEff effector.
- 2 Make sure that the RightFootEffAux effector's **Reach Translation** and **Reach Rotation** channel values are 0 and then translate it to the position of the soccer ball object.

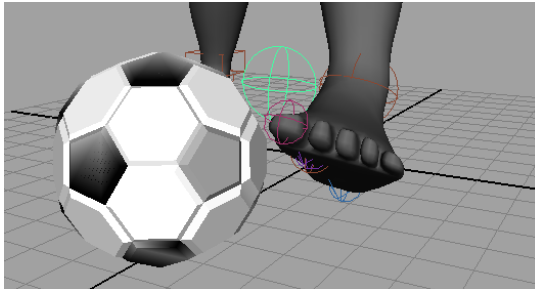


- 3 Select RightFootEffAux, then -click the soccer ball, and parent RightFootEffAux to the soccer ball object (hotkey: **p** key).

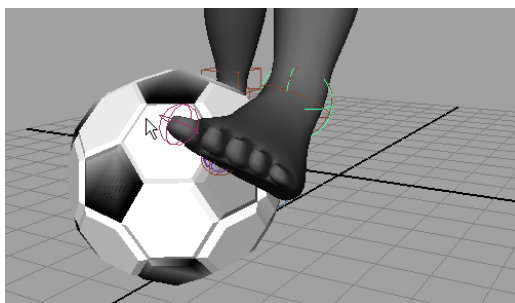
- 4 Move the Time Slider to the beginning of the soccer player's FBIK animation.
- 5 Select the RightFootEffAux effector. In the Channel Box, change its **Reach Translation** and **Reach Rotation** channel values to 0 and then key the channels.



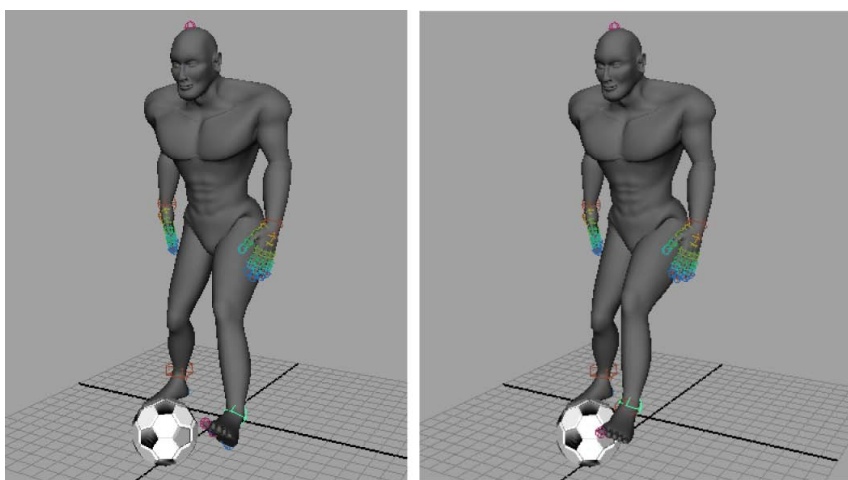
- 6 Select the RightFootEff effector. In the **Channel Box**, change its **Reach Translation** and **Reach Rotation** channel values to 0 and then key the channels.
- 7 Move forward to the point in time where you want RightFootEff to reach the position of RightFootEffAux for the kick.



- 8 Select the RightFootEffAux effector. In the **Channel Box**, change its **Reach Translation** and **Reach Rotation** channel values to 1 and then key the channels.



Now when you playback the soccer player's animation, its right leg and foot will move to the soccer ball for the kick.



After auxiliary effector

Create floor contacts for your character's hands and feet

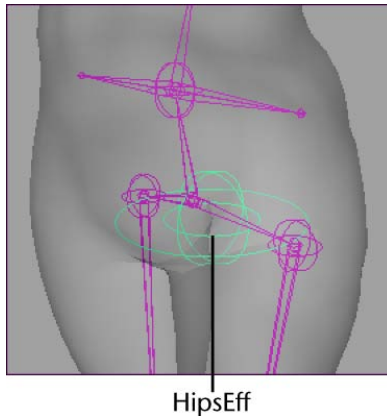
You can create *floor contacts* for your character's hands and feet to make them collide with—rather than pass through—the floor surfaces you defined in your scene. Floor contacts allow your character's hands and feet to react to the floor by stopping and bending the wrist, ankles, toes and fingers. By default, the scene grid is the floor surface, but you can also use *floor contact planes* to define new floor heights for your character's hands or feet.

NOTE

When using hand or foot floor contacts, your characters must have FingerBase and ToeBase joints in their hands and feet for them to bend at the bases of their fingers and toes when they come in contact with the floor. See [Auxiliary joints](#) (page 3), [Left and right hands](#) (page 12), and [Left and right feet](#) (page 17).

To create and adjust your floor contacts

- 1 In the scene view, select the HipsEff effector.



- 2 Do one of the following:

- In the **Channel Box**, select hikFloorContactMarker.

All the floor contact attributes for your FBIK system appear. You can use these attributes to adjust all your character's floor contact settings.

Turn on the appropriate floor contacts by typing *on* or *1* in the **Hands** or **Feet Contact** fields and then press .

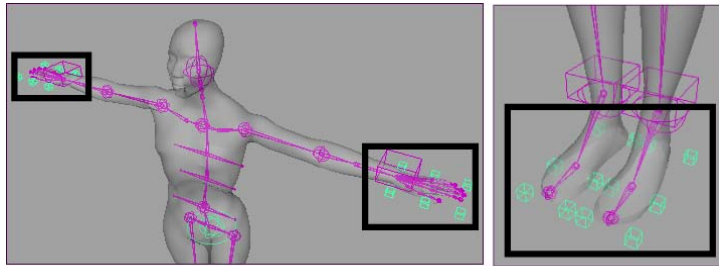
- In the **Attribute Editor**, select the hikFloorContactMarker tab.

In the **Feet to Floor Contact Attributes** or the **Hand to Floor Contact Attributes** sections, turn on the **Hands** or **Feet Contact** attributes.

For example, if you are creating floor contacts for your character's feet, then you would turn on the **Feet Contact** attribute.

A set of hand or foot *floor contact markers* appear for the hands or feet of your character. By default, each floor contact marker set contains 12

markers (six markers for each hand or foot) for bipeds and 8 markers for quadrupeds (four markers for each paw or hoof).



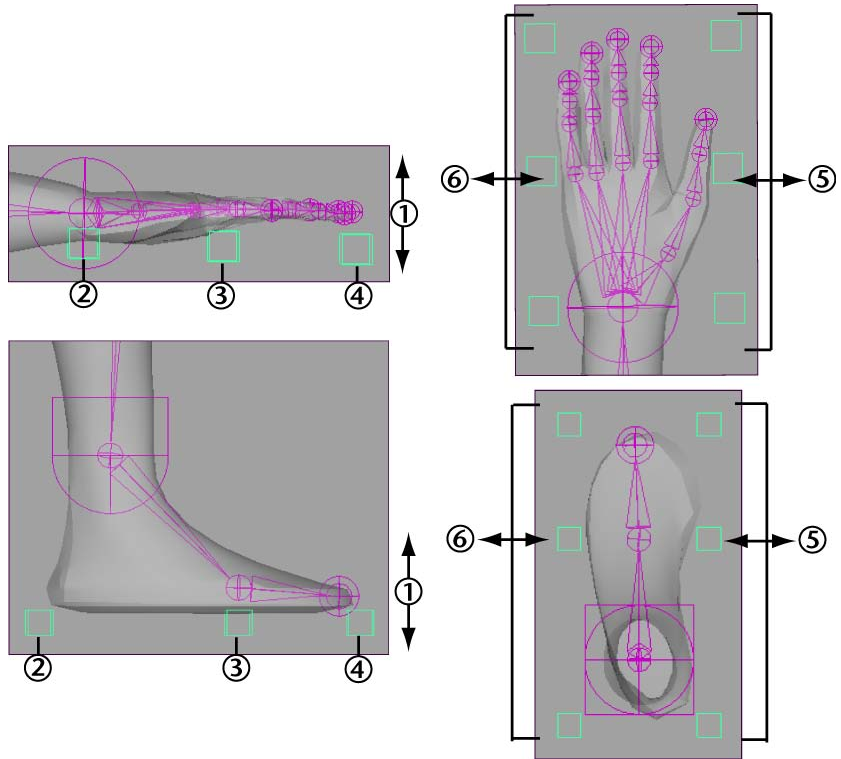
Hand floor contact markers

Foot floor contact markers

NOTE

If your character was properly positioned when you created the FBIK effectors, then the floor contact markers will be correctly oriented in your scene. See [Prepare your character for full body IK](#) (page 2).

- 3 Adjust the size of your individual floor contact markers with the **Marker Size** attribute.
- 4 Adjust the positions of your floor contact markers to best fit your character's hands or feet with the hand and foot floor contact attributes. See [hikFloorContactMarker](#) (page 76).
 - Adjust the **Hand** or **FootHeight** attribute so that all the markers are at the bottom of the hands or feet.
 - Adjust the **Hand** or **FootBack** attributes so that the back markers are at the base of the hands or behind the heels of the feet.
 - Adjust the **Hand** or **FootMiddle** attributes so that the middle markers are positioned at the base of the fingers or toes.
 - Adjust the **Hand** or **FootFront** attributes so that the markers are at the ends of the hands or feet.
 - Adjust the **Hand** or **FootIn Side** attributes so that the markers are at the inner sides of the hands or feet.
 - Adjust the **Hand** or **FootOut Side** attributes so that the markers are at the outer sides of the hands or feet.



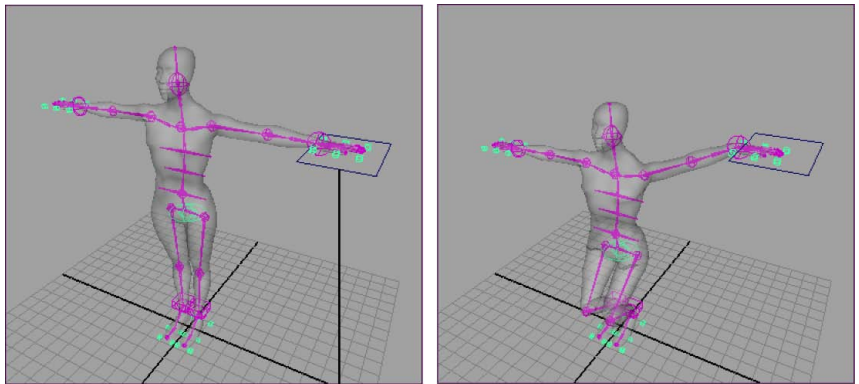
Properly positioned floor contact markers	Attributes that control marker placement
1	Hand/Foot Height
2	Hand/Foot Back
3	Hand/Foot Middle
4	Hand/Foot Front
5	Hand/Foot In Side
6	Hand/Foot Out Side

TIP

- You can customize the orientation and number of floor contact markers for your character with the **Hands** and **Feet Contact Type** attributes.
- You can control the visibility of your hand and foot floor contact markers in your scene view with the **Draw Hand Contact** or **Draw Feet Contact** attributes.
- You can also turn on floor contact markers for your character's *fingers* and *toes*. If you decide to use finger and toe floor contacts, do not also use hand and foot floor contacts.

See [hikFloorContactMarker](#) (page 76).

- 5 (Optional) Create *floor contact planes* for your character's hands or feet. Select the ankle or wrist effector for which you want to create a *floor contact plane*, and then select **Skeleton > Full Body IK > Add Floor Contact Plane** (page 73).



Floor contact plane

A *floor contact plane* for the selected effector appears. This plane determines a position in your scene beyond which the hands or feet affected by the floor contact markers cannot pass. By default, the floor height is 0 in the Y-axis and the scene *grid* is the hand and foot floor contact plane.

Floor contact planes are useful because they make it easy to animate your character walking across uneven terrain like stairs or a rocky surface. A surface can also be rotated and the character reacts appropriately.

NOTE

You can change the floor height without having to create a floor contact plane by going to the node and adjusting the attributes related to floor contact planes. See [hikFloorContactMarker](#) (page 76).

- 6 If you created a floor contact plane, translate the floor contact plane to the height at which you want your floor. You can also adjust the rotation of the plane to represent the incline or decline of the floor you want your model to stand on.

Create and use full body IK

Troubleshoot full body IK character setup

Exploded skin geometry or joints

Problem

Some of your character's joints or segments of geometry appear out-of-scale with their parent joints or other geometry. This can occur if you have the **Segment Scale Compensate** attribute for a joint turned on. When **Segment Scale Compensate** is on, scaling parent joints also scales lower joints in the chain, a process that conflicts with the HumanIK and FBlk solvers.

Solution

If you experience out-of-scale geometry or joints, make sure that you have **Segment Scale Compensate** turned off. The **Segment Scale Compensate** attribute is automatically turned off when you load an FBlk rig, or if you map the joint in a HumanIK character definition. (See also Create a HumanIK character definition.)

See Segment Scale Compensate for more information on this attribute.

Error when creating full body IK effectors for animated character

Problem

You cannot add full body IK effectors to a character that is already animated with traditional FK, IK, or motion capture data. You can only create FBIK effectors for characters with no previous animation.

Solution

- 1 Duplicate your character's skeleton without the keys.
- 2 Create the full body IK effectors for the *duplicate* skeleton.
See [Prepare your character for full body IK](#) (page 2), [Label or rename your character's joints](#) (page 4), and [Create the full body IK system for your character](#) (page 31).
- 3 Turn on the display of the FBIK FK skeleton.
Select [Skeleton > Full Body IK > Show FBIK FK Skeleton](#) (page 73).
- 4 Alternatively, set up the original skeleton as an HIK character and retarget the animation to another HIK character.
See [Retargeting workflow](#).

Characterization fails for characters with HipsTranslation joints

Problem

Characterization may fail and an error may occur when your character's HipsTranslation joint is not properly positioned in the scene view or in your character's joint hierarchy.

Solution

- Place the HipsTranslation joint above the Hips joint in your character's joint hierarchy
- In the scene view, move the HipsTranslation joint to the same Translate X, Y, and Z coordinates as the Hips joint.

FBIK effectors corrupt skinning of characters with HipsTranslation joints

Problem

Creating FBIK effectors for a character that has a HipsTranslation joint may corrupt the character's skinning if your character's HipsTranslation joint is not properly positioned in the scene view or in your character's joint hierarchy.

Solution

- Place the HipsTranslation joint above the Hips joint in your character's joint hierarchy.
- In the scene view, move the HipsTranslation joint to the same Translate X, Y, and Z coordinates as the Hips joint.

Feet slide around when they are pinned

Problem

Your pinned FBIK effectors translate and rotate during playback. This is because pinning only affects your FBIK effectors during interaction—not during playback.

Solution

To constrain a FBIK effector in rotation or translation during playback, set an IK key on the FBIK effector. The IK key will anchor your effector and prevent it from being transformed by movements in other parts of the body.

Constrained FBIK effectors break FBIK solution

Problem

If you constrain your FBIK effectors to other objects (for example, control objects), your character's FBIK IK animation (input skeleton) will not follow its FBIK effectors. This is because during playback, FBIK is only activated for effectors whose Reach values are set to 1.0 (pure IK), and constraining your effectors causes your Reach values to become values other than 1.0.

Solution

Set the constrained FBIK effectors' Reach values to 1.0 (pure IK).

Animate > Set Full Body IK Keys

Animate > Set Full Body IK Keys >

The **Set Full Body IK Keys Options** window lets you do the following:

- Specify which of your character's FBIK effectors and input joints are keyed
- Set your character's FBIK effector's **Reach** channel values
- Place keys on your character's FBIK effectors, FK joints, and **Reach** channels

NOTE

When animating with FBIK, you can pose and key your character's FBIK effectors and input joints to create IK animation. These keys are placed on the FBIK effectors and their FK joints, not the input skeleton's joints.

Key Mode

Determines the parts of your character that are keyed when you click **Apply**.

All

When on, keys are placed on all of the current character's FBIK effectors and FK joints. Also, the **Reach** channel values of all the body parts that are not selected are keyed. You can also set the keying mode to **All** with the +d hotkey or through the FBIK effector marking menu.

Body Part

When on, keys are placed only on the FBIK effectors and FK joints in the selected body part. You can also set the keying mode to **Body Part** with the +x hotkey or through the FBIK effector marking menu.

Selected

When on, keys are placed only on the selected FBIK effectors and their FK joints. You can also set the keying mode to **Selected** with the +4 hotkey or through the FBIK effector marking menu.

Reach Mode

Specifies the **Reach** channel value of the selected FBIK effector and determines whether or not **Reach** is keyed when you click **Apply**.

Reach determines the amount of IK that is set for an effector. For example, a Reach of 1.0 determines that the effector is completely controlled by IK and a Reach of 0.0 disables the effector's IK; allowing you to then animate the underlying FK skeleton's joints.

IK Key (Reach = 1)

When on, the **Reach** channel values of the selected FBIK effectors are set to 1 and are keyed when you click Apply. Also, the **Reach** channel values of all other effectors within the selected body part are set to 0 and the keys on the body part of the selected effectors are given **Stepped Next** tangents. You can also set the **Reach Mode** to **IK Key** with the +3 hotkey or through the FBIK effector marking menu.

FK Key (Reach = 0)

When on, the **Reach** channel values of the selected FBIK effectors are set to 0 and are keyed when you click apply. Also, the **Reach** channel values of all other effectors within the selected body part are set to 0 and the keys set on **Reach** are given regular **Stepped** tangents.

You can also set the **Reach Mode** to **FK Key** with the +1 hotkey or through the FBIK effector marking menu.

Simple Key (Reach not keyed)

When on, the **Reach** channel values of your FBIK effectors are not changed or keyed when you click Apply. Simple key uses the global default tangency. Simple keys are most often used to rough out your character's animation before keying the finer details of your FBIK animation with IK and FK keys. You can also set the **Reach mode** to **Simple Key** with the +2 hotkey or through the FBIK effector marking menu.

Apply and Close

Sets a key on the current character's FBIK effectors and input joints according to the current **Key Mode** and then closes the **Set Full Body IK Keys Options** window.

Apply

Sets a key on the current character's FBIK effectors and joints according to the current Key Mode.

Close

Cancels the operation and closes the **Set Full Body IK Keys Options** window.

Related topics

- [Pose and animate your character with full body IK](#) (page 20)
- [Examples: Working with full body IK](#) (page 41)

Skeleton > Full Body IK

Skeleton > Full Body IK > Add Full Body IK

Skeleton > Full Body IK > Add Full Body IK >

These options let you specify the kind of character you want to create the full body IK effectors for and the FBIK joint labelling and naming method you used for your characters.

Identify Joints

Select one of the following options.

By Label When on, labels are used to create the set of full body IK effectors. See [Joint labelling](#) (page 5).

By Name When on, joint names are used to create the set of full body IK effectors. See [Renaming your character's joints](#) (page 6).

Posture

Select one of the following options.

Biped When on, the character you want to create the full body IK effectors for is a biped.

Quadruped When on, the character you want to create the full body IK effectors for is a quadruped.

Related topics

- [FBIK joint labelling and naming conventions](#) (page 4)

Skeleton > Full Body IK > Add Auxiliary Effector

Creates an auxiliary effector for the selected FBIK effector. Auxiliary effectors let you direct the positions of their corresponding FBIK effectors during playback.

Related topics

- [Create auxiliary effectors](#) (page 57)

Skeleton > Full Body IK > Add Auxiliary Pivot

Creates a auxiliary pivot for the selected FBIK effector. Auxiliary pivots let you create and position an additional pivot point for an FBIK effector.

Related topics

- [Create auxiliary pivots](#) (page 55)

Skeleton > Full Body IK > Activate Auxiliary Pivot

Turns on the selected FBIK auxiliary pivot. When a auxiliary pivot is active, it acts like an additional pivot point for its parent effector during the FBIK solution.

If you want to reposition a auxiliary pivot, deactivate it by selecting [Skeleton > Full Body IK > Change Auxiliary Pivot Placement](#) (page 72).

Related topics

- [Create auxiliary pivots](#) (page 55)

Skeleton > Full Body IK > Change Auxiliary Pivot Placement

Turns off the selected FBIK auxiliary pivot, allowing you to translate the effector to a new position. When a auxiliary pivot is inactive, it is not included in the FBIK solution.

Once you have repositioned the selected auxiliary pivot, you need to reactivate it by selecting [Skeleton > Full Body IK > Activate Auxiliary Pivot](#) (page 72).

Related topics

- [Create auxiliary pivots](#) (page 55)

Skeleton > Full Body IK > Show FBIK FK Skeleton

Turns on the visibility of the full body IK *FK* skeleton. The FK FBIK skeleton appears dark blue in your scene view. The FK skeleton stores all the FK keys for the FBIK solution. See [FK skeleton](#) (page 21).

Skeleton > Full Body IK > Hide FBIK FK Skeleton

Turns off the visibility of the full body IK *FK* skeleton. See [FK skeleton](#) (page 21).

Skeleton > Full Body IK > Add Floor Contact Plane

Creates a plane for the hand or foot floor contacts that represents the floor which the floor contact markers must respect. By default, the scene grid is the floor.

Related topics

- [Create floor contacts for your character's hands and feet](#) (page 61)

Skeleton > Full Body IK > Go to Stance Pose

Moves and rotates your full body effectors to their initial positions on creation.

Skeleton > Full Body IK > Add Missing Effectors

Adds new effectors to a character that has an existing set of FBIK effectors and adds back any effectors you have deleted. For example, if you have a character that is already animated using full body IK, you can add fingers to its skeleton and then use **Add Missing Effectors** to create FBIK effectors for the fingers.

Add Missing Effectors is useful because it allows you to add new or deleted effectors to your character without having to recreate its entire set of FBIK effectors, which would cause it to lose all its existing FBIK animation.

By Label Recreates your deleted effectors using label names.

By Name Recreates your deleted effectors using joint names.

Skeleton > Full Body IK > Body Part Autoload

When on, your character's current **Key Mode** determines which of its keys display in the Time Slider, Graph Editor, and Dope Sheet. For example, when in Selected Key Mode, only the FBIK keys on the selected FBIK effector or FK joint display. When in Body Part Key Mode, the FBIK keys for all the FBIK effectors and FK joints for the selected effector or FK joint's body part display. And when in All Key Mode, all the FBIK keys (IK and FK) for the selected FBIK effector or FK joint's character display.

When off, only the selected FBIK effector or FK joint's keys are displayed in the Time Slider, Graph Editor, and Dope Sheet.

Edit > Keys > Delete FBIK Keys

Deletes all FBIK keys, body part keys, or selected keys. These items are also accessible from the context-sensitive menu that appears by right-clicking in the Time Slider.

Edit > Keys > Delete FBIK Keys > Delete All FBIK Keys

For the current character, deletes all the keys at the **Current Time Indicator** from its FBIK effectors and FK joints.

Edit > Keys > Delete FBIK Keys > Delete Body Part Keys

For the current body part, deletes all the keys at the **Current Time Indicator** from its FBIK effectors and FK joints.

Edit > Keys > Delete FBIK Keys > Delete Selected FBIK Keys

For the selected FBIK effectors, deletes all their keys at the **Current Time Indicator** as well as those on their FK joints.

hikEffector

HIK Effector Attributes

Pinning

When a FBIK effector's channel is pinned, it will not move or rotate when you pose other parts of your character's body.

NOTE

Pinning is not a constraint. Pinning only restricts the movements of your effectors during *interaction*. If you want an effector to stay in place during *playback*, then you need to set an IK key on that effector.

unpinned The current effector is not pinned during interaction.

pinTranslate Pins the current effector in translation during interaction.

pinRotate Pins the current effector in rotation during interaction.

pinAll Pins the current effector in translation and rotation during interaction.

Reach Translation Specifies the amount of IK for the FBIK effector and controls how much your character's joints want to reach their associated effector's translation.

For example, a **Reach Translation** of 1 sets that the effector is completely controlled by IK and its joints will try to reach the effector's translation. A **Reach Translation** of 0 disables the effector's IK (allowing you to then animate the underlying FK skeleton joints) and its joints ignore the effector's translation. The default **Reach Translation** values for your hip, hand, and foot FBIK effectors are: 1.0 Foot Translate, 1.0 Hips Translate, and 1.0 Hands Translate.

Reach Rotation Specifies the amount of IK for the FBIK effector and controls how much your character's joints want to reach their associated effector's

rotation. For example, a **Reach Rotation** of 1 determines that the effector's rotation is completely controlled by IK and a **Reach Rotation** of 0 disables the effector's IK (allowing you to then animate the underlying FK skeleton joints) on its rotation channels. The default **Reach Rotation** values for your hip, hand, and foot FBIK effectors are: 1.0 Foot Rotate, 1.0 Hips Rotate, and 0.0 Hands Rotate.

Pivot Offset Sets the amount (in local coordinates) a secondary effector is offset from its parent effector. All non-secondary FBIK effectors also have **Pivot Offsets**, which are 0,0,0 by default. This should not be manually modified.

Radius Sets the size of the current FBIK effector in the scene view. **Radius** is measured in the current user set working units.

hikFloorContactMarker

Floor Contact Marker Attributes

See also [Create floor contacts for your character's hands and feet](#) (page 61).

Marker Size Sets the size of the individual floor contact markers. **Marker Size** does not affect how your character's hands and feet contact the floor. **Marker Size** is measured in the current user set working units.

Feet to Floor Contact Attributes

Feet Contact When on, the current character's foot floor contacts are active.

Draw Feet Contact When on, your foot floor contact markers are visible in the scene view.

Feet Floor Pivot

Specifies which parts of your feet take priority when they make contact with the floor.

auto When selected, the priority is averaged between the ankle and toes. When the foot then makes contact with the floor, the ankle is translated backwards and the toes are pushed forwards. **Auto** is selected by default.

ankle When selected, the ankles are given priority. When the foot then makes contact with the floor, the toes are translated forward to maintain the ankle's trajectory towards the floor.

toes When selected, the toes are given priority. When the foot then comes in contact with the floor, the ankle is translated backwards to keep the toes firmly planted on the floor.

Feet Contact Type

Specifies the number and orientation of the markers used to define the floor contacts for your character's feet.

normal When selected, six markers define each foot's floor contact. Of the six markers, the middle markers define the ball of each foot or the point where the toes begin.

ankle When selected, four markers define each foot's floor contact and they are positioned around the base of each foot.

toe base When selected, four markers define each foot's floor contact and they are positioned around the toes of each foot.

hoof When selected, four markers define each foot's floor contact. The markers are oriented at a 90 degree angle to allow for 180 degree movement. **Hoof** is the best feet contact type for animals with hooves such as horses.

Feet Contact Stiffness Sets how much your character's hands resist going through the floor when any part of the feet contact the floor. A **Feet Contact Stiffness** of 1.0 produces no ankle rotations and maintains your foot transformations. A **Feet Contact Stiffness** of 0.0 rotates your character's ankles to minimize foot-floor penetration. With any **Feet Contact Stiffness** value, your character's ankles are translated for any final corrections.

Foot Height Sets the positions of your character's foot floor contact markers along the Y-axis.

Foot Back Sets the positions of the back most foot floor contacts along the Z-axis. The positions of the back markers determine where the heels of your character's feet contact the floor.

Foot Middle Sets the positions of the middle foot floor contact markers along the Z-axis. The positions of the middle markers determine where your character's feet will bend when they contact the floor.

Foot Front Sets the positions of the front most foot floor contact markers along the Z-axis. The positions of the front most markers determine where the ends of your character's toes will contact the floor.

Foot In Side Sets the positions of the inside (the markers closer to the middle of your character) floor contact markers along the X-axis.

Foot Out Side Sets the positions of the outside (the markers farthest from the middle of your character) floor contact markers along the X-axis.

Hands to Floor Contact Attributes

Hands Contact When on, the current character's hand floor contacts are active.

Draw Hand Contact When on, your hand floor contact markers are visible in the scene view.

Hands Floor Pivot

Specifies which parts of your hands take priority when they make contact with the floor.

auto When selected, the priority is averaged between the wrist and fingers. When the hand then makes contact with the floor, the wrist is translated backwards and the fingers are pushed forwards. **Auto** is selected by default.

wrist When selected, the wrists are given priority. When the hand then makes contact with the floor, the fingers are translated forward to maintain the wrist's trajectory towards the floor.

fingers When selected, the fingers are given priority. When the hand then comes in contact with the floor, the wrist is translated backwards to keep the fingers firmly planted on the floor.

Hands Contact Type

Specifies the number and orientation of the markers used to define the floor contacts for your character's hands.

normal When selected, six markers define each hand's floor contact. Of the six markers, the middle markers define where the fingers begin.

wrist When selected, four markers define each hand's floor contact and they are positioned around each hand's palm.

finger base When selected, four markers define each hand's floor contact and they are positioned around the fingers of each hand.

hoof When selected, four markers define each hand's floor contact. The markers are oriented at a 90 degree angle to allow for 180 degree movement.

Hoof is the best hand contact type for animals with hooves such as horses.

Hands Contact Stiffness Sets how much your character's hands resist going through the floor when any part of the hands contact the floor. A **Hands Contact Stiffness** of 1.0 produces no wrist rotations and maintains your hand transformations. A **Hands Contact Stiffness** of 0.0 rotates your

character's wrists to minimize hand-floor penetration. With any **Feet Contact Stiffness** value, your character's wrists are translated for any final corrections.

Hand Height Sets the positions of the hand floor contact markers along the Y-axis.

Hand Back Sets the positions of the back most hand floor contacts along the Z-axis. The positions of the back markers determine where the heels of your character's hands contact the floor.

Hand Middle Sets the positions of the middle hand floor contact markers along the Z-axis. The positions of the middle markers determine where your character's hands bend when they contact the floor.

Hand Front Sets the positions of the front most hand floor contact markers along the Z-axis. The positions of the front most markers determine where the ends of your character's fingers contact the floor.

Hand In Side Sets the positions of the front most hand floor contact markers along the Z-axis. The positions of the front most markers determine where the ends of your character's fingers contact the floor.

Hand Out Side Sets the positions of the outside (the markers farthest from the middle of your character) hand contact markers along the X-axis.

Fingers/Toes to Floor Contact Attributes

Fingers Contact When on, the current character's finger floor contacts are active and the fingers will be taken into account by the **hikSolver**.

NOTE

When using **Fingers Contact**, do not also use **Hands Contact** as it will override your finger floor contact.

Fingers Contact Type

Controls the behavior of your character's fingers when they contact the floor.

sticky When selected, each finger sticks to the floor exactly where contact is made.

spread When selected, the fingers spread as they make contact with the floor, attempting to keep the position of the fingers intact. Rotation is applied to the root of each finger.

sticky-spread When selected, the fingers' behavior is an average between **Sticky** and **Spread**.

Fingers Contact Roll Stiffness Corrects unnatural finger rotation when finger floor contact markers touch the floor. A **Fingers Contact Roll Stiffness** of 1.0 attempts to prevent the fingers from rotating in unnatural directions.

Toes Contact When on, the current character's toe floor contacts are active and the toes will be taken into account by the **hikSolver**.

NOTE

When using **Toes Contact**, do not also use **Feet Contact** as it will override your toes floor contact.

Toes Contact Type

Controls the behavior of your character's toes when they contact the floor.

sticky When selected, each toe sticks to the floor exactly where contact is made.

spread When selected, the toes spread as they make contact with the floor, attempting to keep the position of the toes intact. Rotation is applied to the root of each toe.

sticky-spread When selected, the toes' behavior is an average between **Sticky** and **Spread**.

Toes Contact Roll Stiffness Corrects unnatural toe rotation when toe floor contact markers touch the floor. A **Toes Contact Roll Stiffness** of 1.0 attempts to prevent the toes from rotating in unnatural directions.

Index

A

- accessory joints 4
- animate
 - character's with FBIK 20
 - FBIK 25
- auxiliary joints 3

B

- base joints 3
- biped
 - FBIK preparation 2

C

- create
 - FBIK floor contacts 61

E

- edit
 - FBIK floor contacts 61
- effectors
 - create FBIK 31
 - FBIK node 75
 - pinning 35
 - secondary FBIK 55
- examples
 - FBIK pinning 38

F

- FBIK
 - animate 25
 - animate characters 20
 - biped preparation 2
 - create effectors 31
 - custom joint names 7
 - floor contacts 61

- joint labelling 5
- joint renaming 6
- joint types 3
- labelling joints 4
- manage keys 28
- MEL 7
- pinning 35, 38
- pinning examples 38
- posing 25
- preparation 2
- quadruped preparation 2
- renaming joints 4
- secondary effectors 55
- workflow 30
- floor contacts 61
 - marker node 76

H

- hikEffector 75

J

- joints
 - accessory 4
 - auxiliary 3
 - base 3
 - custom FBIK names 7
 - labelling for FBIK 4
 - naming conventions 8
 - renaming 6
 - renaming for FBIK 4
 - roll 3

K

- keys
 - manage FBIK 28

L

label

 FBIK joints 4, 5

M

manage

 FBIK keys 28

MEL

 FBIK 7

N

nodes

 floor contact marker 76

 hikEffector 75

P

pinning 35

 how to 38

posing

 FBIK 25

prepare

 character for FBIK 2

Q

quadruped

 FBIK preparation 2

R

rename

 FBIK joints 4

 joints 6

roll joints 3

S

secondary effectors 55

W

workflow

 FBIK 20, 30