

Autodesk® MotionBuilder® 2011

Tutorials



© 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, AliasWavefront (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, SoftimageXSI (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).

Python and the Python logo are trademarks or registered trademarks of the Python Software Foundation.

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 (c) 1993-2009, all rights reserved.

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.

Published by: Autodesk, Inc.

111 McInnis Parkway

San Rafael, CA 94903, USA

Document Title: Autodesk MotionBuilder 2011 Tutorials

Date: 31 March 2010

Document Version: 2010.03.31.01

Build Version: 2010.03.31.01

Contents

- Chapter 1 **MotionBuilder Tutorials** 1
 - Installing the latest FBX Plug-ins 2
 - MotionBuilder workflow 2
- Chapter 2 **Loading and Characterizing Character Models** 7
 - Prepare the scene 7
 - Complete the character map 9
 - Characterize the character model 14
- Chapter 3 **Creating and Customizing a Control Rig** 17
 - Prepare the scene 17
 - Create a Control rig 18
 - Adjust the foot floor contact markers 21
 - Adjust the hand floor contact markers 24
 - Add Auxiliary pivots 26
- Chapter 4 **Creating a Character Extension** 31
 - Prepare the scene 31
 - Connect the extra limb to the character 33
 - Create a Character Extension 38

Chapter 5	Creating a Walk Cycle	43
	Prepare the scene	43
	Create poses	44
	Create animation with poses	51
	Mirror poses	54
	Play the animation	57
Chapter 6	Retargeting Character Animation	59
	Prepare the scene	59
	Save the character animation	61
	Create a scene	63
	Load character animation	65
	Play the animation	68
Chapter 7	Editing Character Animation	71
	Prepare the scene	71
	Modify the Character Extension animation	73
	Modify the head animation	78
	Plot the animation	82
	Play the resulting take	83
Chapter 8	Creating a Loop	85
	Prepare the scene	85
	Create a Character track	86
	Create poses	91
	Match clips	93
	Process the clips	96
	Test the walk cycle	98
Chapter 9	Manipulating Clips	101
	Prepare the scene	101
	Create a turn	103
	Blend two clips	107
	Add a clip	108
	Match clips	110
Chapter 10	Importing 3ds Max Files into MotionBuilder	113
	Preparation for this tutorial	114
	3ds Max skeletons	115
	3ds Max Biped	127
	3ds Max Characters	137
	Animating a 3ds Max Character in MotionBuilder	144

Exporting a Character and its animation back to 3ds Max	156
Glossary	163
Acronyms	163
Terms	164

MotionBuilder Tutorials

1

This book includes a set of nine Autodesk MotionBuilder tutorials that provide a common MotionBuilder workflow and demonstrate how to use the more powerful keyframe and character animation features.

You can find the tutorial assets in the MotionBuilder Asset browser's *Tutorials* folder as well as in the *Tutorials* folder located in the MotionBuilder directory on your system.

NOTE If the Asset Browser window is not available, from the MotionBuilder menu bar, select Window > Asset Browser.

NOTE If the *Tutorials* folder is not displayed in the Asset browser, you need to add a favorite path to display a directory in the Asset Browser.

To add a directory to be displayed in the Asset browser:

- Refer to the procedure “Adding a favorite path” in the MotionBuilder Help or User’s Guide “Interface” chapter, in the “Asset browser” topic.

Following are the tutorials:

- [Loading and Characterizing Character Models](#) on page 7
- [Creating and Customizing a Control Rig](#) on page 17
- [Creating a Character Extension](#) on page 31
- [Creating a Walk Cycle](#) on page 43
- [Retargeting Character Animation](#) on page 59
- [Editing Character Animation](#) on page 71
- [Creating a Loop](#) on page 85
- [Manipulating Clips](#) on page 101
- [Importing 3ds Max Files into MotionBuilder](#) on page 113

See also:

■ [MotionBuilder workflow](#) on page 2

NOTE You can download the tutorial assets (or support files) from:
<http://www.autodesk.com/motionbuilder2011-documentation>.

Installing the latest FBX Plug-ins

The Autodesk® FBX® technology is one of the most widely used and supported platform-independent 3D data interchange solutions around. Universal 3D asset exchange via Autodesk FBX helps to remove data compatibility barriers and gives you the freedom to build an efficient pipeline for your projects.

Autodesk FBX fosters interoperability between several Autodesk products. The MotionBuilder software product supports FBX natively, while the Autodesk® Maya® and Autodesk® 3ds Max® software products include FBX plug-ins. In addition, the Autodesk® Softimage® software product can read and write FBX through the Autodesk® Crosswalk software initiative. Autodesk® Mudbox™ 2010 software is the first Mudbox release to support FBX, streamlining common workflows between Mudbox, Maya, 3ds Max, and MotionBuilder.

You can download the latest FBX Plug-ins and FBX Plug-ins documentation from: <http://www.autodesk.com/fbx>

MotionBuilder workflow

This topic describes a common workflow that introduces the nine tutorials provided to help you familiarize yourself with the MotionBuilder software product. For any steps in the workflow that do not include a dedicated tutorial, you can find additional information in the MotionBuilder Help.

Although these tutorials assume you are using MotionBuilder for a character animation project, this workflow can be easily adapted to any animation project where MotionBuilder is used in conjunction with other 3D modeling or rendering software.

The basic workflow for using MotionBuilder can be summarized as follows:

- 1 Install the required FBX Plug-ins so you can transfer your work from other 3D software packages into and out of MotionBuilder.

For example, if you are using Maya or 3ds Max for character modeling, you need to install the appropriate Maya or 3ds Max FBX Plug-in to transfer your models into MotionBuilder. See [Installing the latest FBX Plug-ins](#) on page 2.

2 Create a character model in your 3D modeling software of choice.

Before you start your animation project using MotionBuilder, there are a few things you can do when modeling to facilitate your work in MotionBuilder.

Refer to “Guidelines for creating a character model”, “Bone naming conventions”, “Import and characterize a 3ds Max biped in MotionBuilder”, and “Choosing shapes to create” in the MotionBuilder Help.

3 Export the character model from your modeling software package.

When you export your work from a modeling software package, the FBX Plug-in you installed lets you save your character model in the *.fbx* file format. This format enables you to load your models in MotionBuilder.

4 Start MotionBuilder and load your character model.

Once you load a model into MotionBuilder, you can set it up to animate it using the MotionBuilder Character asset.

5 Add a Character asset for your character model and characterize it.

The Character asset helps you map the structure of your character model so that it can be animated in MotionBuilder. Once you complete this mapping process, you ‘activate’ the character model by characterizing it. Characterizing lets MotionBuilder know that this character model is ready to be animated.

All major character animation features in MotionBuilder, including Control rigs and animating in the Story window, require a characterized character.

The first tutorial shows you how to create a Character asset and use it to map out your character model’s structure.

See [Loading and Characterizing Character Models](#) on page 7.

6 Add a Control rig and customize it to fit your character animation needs.

Control rigs are an animation tool that make it easy to control and position your character model.

The second tutorial shows you how to customize a Control rig and add character animation features such as floor contacts and Auxiliary pivots.

See [Creating and Customizing a Control Rig](#) on page 17.

- 7 Add Character Extensions to support props or non-human body parts.
The third tutorial shows you how to augment your character with an extra limb, in this case a “Servo arm” with giant pincers attached to the character’s right shoulder.

See [Creating a Character Extension](#) on page 31.

- 8 Create your animation using different keyframing and character animation features.

- One efficient method of creating animation involves creating a set of poses that can be pasted onto your character at various points over time.

The fourth tutorial shows you how to use the Control rig and the Pose Controls to create a walk cycle.

See [Creating a Walk Cycle](#) on page 43.

- The seventh tutorial shows you an alternative method for creating a walk cycle using clips in the Story window.

See [Creating a Loop](#) on page 85.

- 9 Edit and refine your animation.

- The sixth tutorial shows you how to use layers to edit animation.

See [Editing Character Animation](#) on page 71.

- The eighth tutorial shows you how to combine animations using the Story window.

See [Manipulating Clips](#) on page 101.

- 10 Retarget your animation between Character models.

During animation projects, the Character model you use might change. Although not a required step for creating animation within MotionBuilder, instead of re-creating the animation on a new model, you can simply apply the same animation to the desired model(s).

The fifth tutorial shows you how to transfer animation and Character Extensions between character models.

See [Retargeting Character Animation](#) on page 59.

- 11 Plot your finished animation to your model’s skeleton.

Depending on the animation features you are using to create your character animation, plotting may consist of plotting from your Control

rig to your character model skeleton, or plotting the tracks in the Story window to a single take.

Whatever method you use to animate, the finished result must be plotted to the skeleton of your character model before you export it.

See “The plotting process” topic in the MotionBuilder Help “Plotting Animation” chapter.

12 Save your plotted model as an *.fbx* file.

Your finished animations can also be exported for rendering in the software of your choice using the appropriate FBX Plug-In.

You can download the latest FBX Plug-ins from:

<http://www.autodesk.com/fbx>.

13 If you want to animate 3ds Max characters in MotionBuilder, and then use that animation in 3ds Max, you need to import your 3ds Max scene into MotionBuilder, animate in MotionBuilder, then import your animation in 3ds Max.

This last tutorial shows you the major steps for importing animation into MotionBuilder, animating in MotionBuilder, and exporting the animation from MotionBuilder and importing it to 3ds Max.

See the following major sections: [3ds Max skeletons](#) on page 115, [3ds Max Bipeds](#) on page 127, [3ds Max Characters](#) on page 137, [Animating a 3ds Max Character in MotionBuilder](#) on page 144, and [Exporting a Character and its animation back to 3ds Max](#) on page 156.

Loading and Characterizing Character Models

2

This tutorial guides you through the procedures necessary to bring your character models into MotionBuilder and get them ready for animation.

Each character model brought into MotionBuilder has to be characterized before you can create a Control rig, create poses, and use other animation tools. To characterize a character model, you need to map its structure.

This tutorial shows you how to:

- [Prepare the scene](#) on page 7
- [Complete the character map](#) on page 9
- [Characterize the character model](#) on page 14

The following asset is required for this tutorial:

- *mia_blue.fbx*

NOTE The tutorial assets can be found in the *Tutorials* folder in the Asset Browser and in the *Tutorials* folder in the MotionBuilder directory on your system.

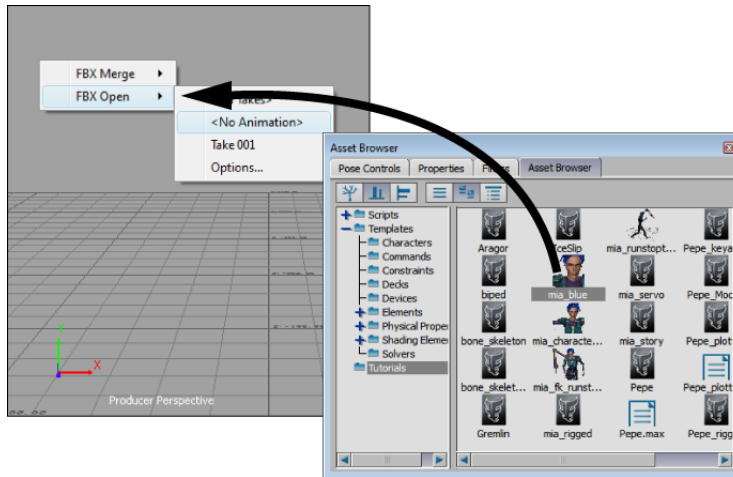
Prepare the scene

In the following procedure, you prepare the MotionBuilder scene and open the file needed to start this tutorial.

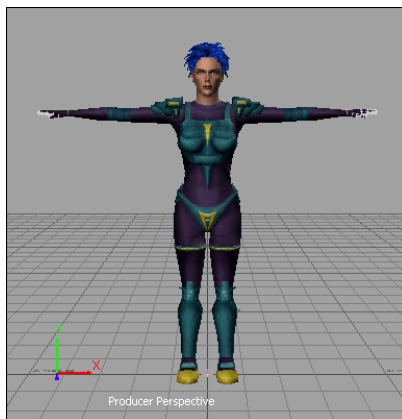
- 1 From the menu bar, select File > New, then select Layout > Editing (or press *Ctrl-Shift-3*).

MotionBuilder displays a new 3D scene using the Editing layout. This layout displays all the windows you need for your work in this tutorial.

- 2 Click the *Tutorials* folder in the Asset browser.
- 3 Drag the *mia_blue* asset (*mia_blue.fbx* file) from the Asset browser into the Viewer window, then select **FBX Open > No Animation** as shown in the following figure.



In the Viewer window, a model named Mia appears in the T-stance.



Mia in the T-stance

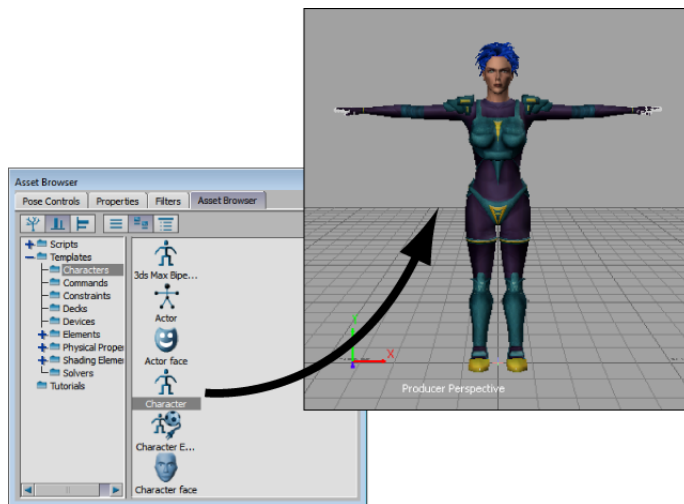
NOTE This model was created in Maya, and the bones were named according to the naming conventions in the MotionBuilder Mapping list.

Complete the character map

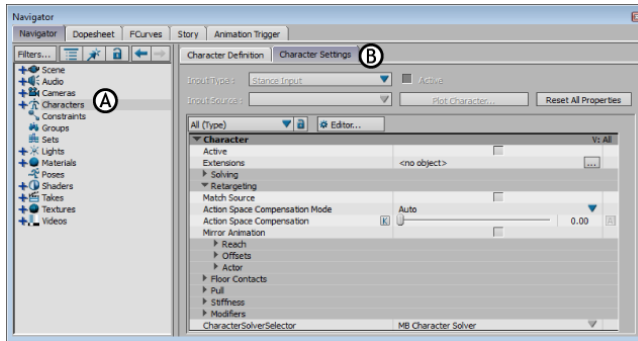
In the following procedure, you define the structure of your character model for MotionBuilder by mapping the required nodes in the Mapping list. Character mapping describes the character model for MotionBuilder, indicating what are the legs, arms, and so on.

Although you can automatically map and characterize a character by dragging the Character asset directly onto a character model, for the purpose of this tutorial, you manually map out Mia's structure.

- 1 From the *Templates > Characters* folder of the Asset browser, drag the *Character* asset into an empty area of the scene.

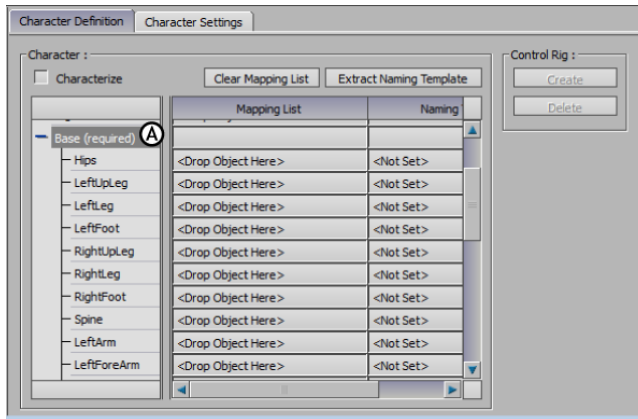


A Character asset is added in the Scene browser (A) and the Character Settings are displayed in the Navigator window (B).



Navigator window: A. Character in the Scene browser B. Character Settings

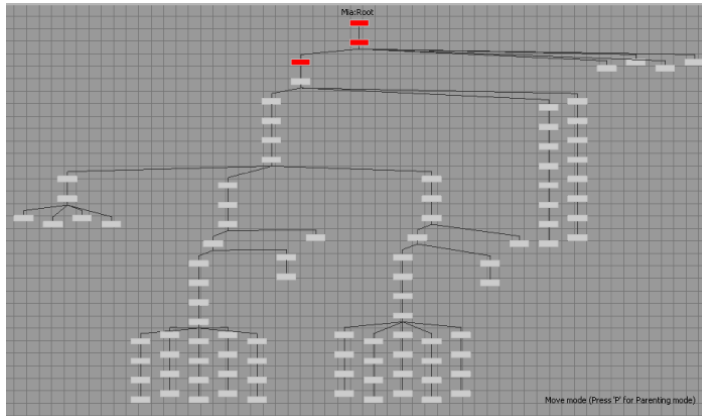
- 2 Switch to the Character Definition pane in the Character Settings and expand the *Base (required)* group of nodes in the Mapping list (A).



Character Definition pane: A. Base nodes

This group of nodes is required for MotionBuilder to recognize the structure of your character model. If you had automatically characterized this character, the Mapping list would be populated with the character's bone names.

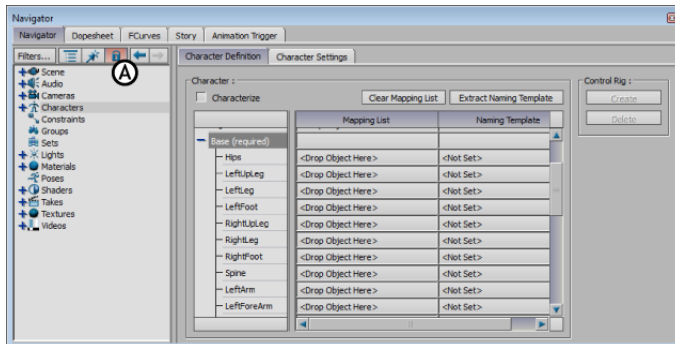
- 3 In the Viewer window, switch to the Schematic view (*Ctrl-W*) and press *A* to frame the hierarchy.



Schematic view of Mia's structure

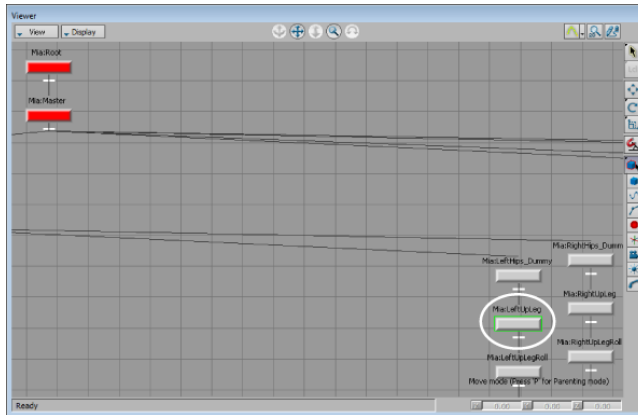
The Schematic view makes it easier to select bones from the model's hierarchy because each bone is represented as a rectangular node.

- 4 In the Scene browser, activate the Lock option to lock the view of the Character Definition pane.



Navigator window: A. Scene browser Lock option activated

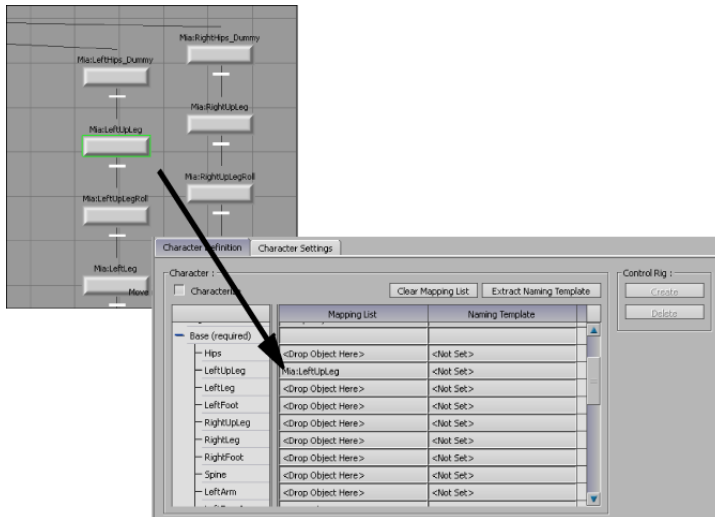
- 5 In the Schematic view, zoom in (*Ctrl*-drag) and select the *Mia:LeftUpLeg* node.



Mia:LeftUpLeg node selected

NOTE When you know the name of the node you are looking for, you can press *Shift-N* to open the Find Model by Name dialog box and do a quick search.

- 6 Alt-drag the *Mia:LeftUpLeg* node into the *Base (required) LeftUpLeg* Mapping List slot.



Mia's LeftUpLeg node mapped to MotionBuilder *LeftUpLeg*

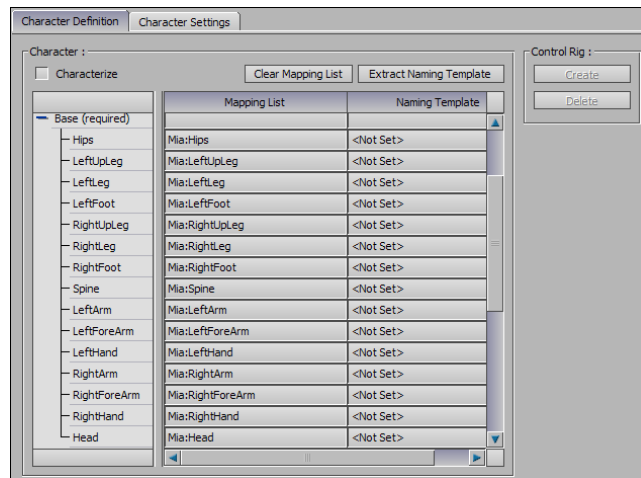
When you characterize this character, MotionBuilder recognizes that for this skeleton the *LeftUpLeg* node is called *Mia:LeftUpLeg*.

- 7 Use the following checklist and figure as guides to map the rest of Mia's bones to the *Base (required)* nodes in the Mapping list.

NOTE Although Mia has many bones, you are only required to map the Base group of fifteen for MotionBuilder characterization.

Bone	Slot	Mapped
Mia:Hips	Hips	
Mia:LeftUpLeg	LeftUpLeg	x
Mia:LeftLeg	LeftLeg	
Mia:LeftFoot	LeftFoot	
Mia:RightUpLeg	RightUpLeg	
Mia:RightLeg	RightLeg	
Mia:RightFoot	RightFoot	
Mia:Spine	Spine	
Mia:LeftArm	LeftArm	
Mia:LeftForeArm	LeftForeArm	
Mia:LeftHand	LeftHand	
Mia:RightArm	RightArm	
Mia:RightFore-Arm	RightForeArm	
Mia:RightHand	RightHand	
Mia:Head	Head	

After completing the character mapping process for the Base group, the Mapping list resembles the Mapping List shown in the following figure.



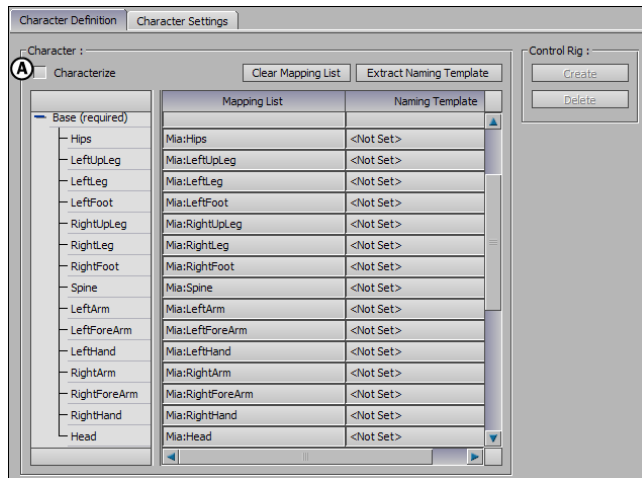
Mia's base bones mapped to the Base nodes in the Mapping list

The Character mapping is now complete.

Characterize the character model

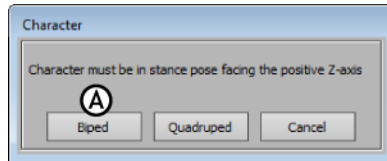
The following procedure shows you how to characterize a model. The moment you characterize a character model, MotionBuilder reads the structure you have outlined in the Mapping list, taking the model's current pose as the base for all future poses and movement.

- 1 In the Character Definition pane, activate the Characterize option (A).



Character Definition pane: A. Characterize option

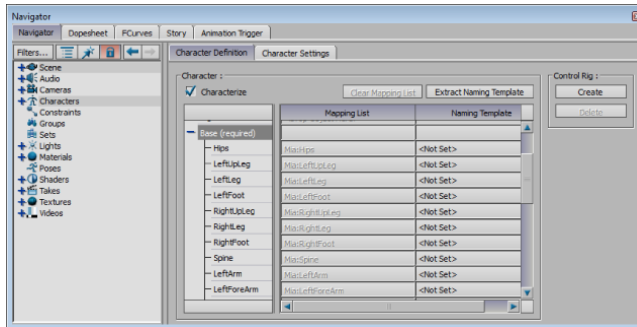
- 2 In the Character dialog box that appears, click Biped (A), since the Mia skeleton stands on two legs and makes contact with the floor using only the feet.



Character dialog box: A. Biped option

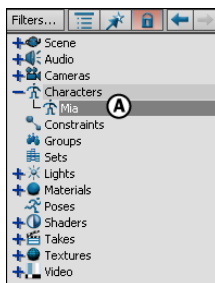
Generic offsets are calculated so that the character is compatible with any source, the character is characterized, and MotionBuilder recognizes its structure.

The nodes in the Mapping list are gray and cannot be edited.



NOTE If you want to add more bones or edit the Mapping list later, you can temporarily disable the Characterize option when your character is in the T-stance.

- 3 In the Scene browser, expand the *Characters* branch, right-click the Character asset, select Rename from the contextual menu, and name the character “Mia” (A).



Scene browser: A. Character asset named Mia

Your character is now fully characterized and ready to be animated.

Summary

In this tutorial, you loaded a character model, mapped out its structure, and characterized it.

In the next tutorial, ([Creating and Customizing a Control Rig](#) on page 17), you create and customize a Control rig for your characterized character.

Creating and Customizing a Control Rig

3

This tutorial guides you through the procedures necessary to create a Control rig and customize the Control rig to create animation in subsequent tutorials.

Control rigs are an animation tool that make it easy to control and position your character model. You can re-purpose Control rigs for other models.

This tutorial shows you how to:

- [Prepare the scene](#) on page 17
- [Create a Control rig](#) on page 18
- [Adjust the foot floor contact markers](#) on page 21
- [Adjust the hand floor contact markers](#) on page 24
- [Add Auxiliary pivots](#) on page 26

The following asset is required for this tutorial:

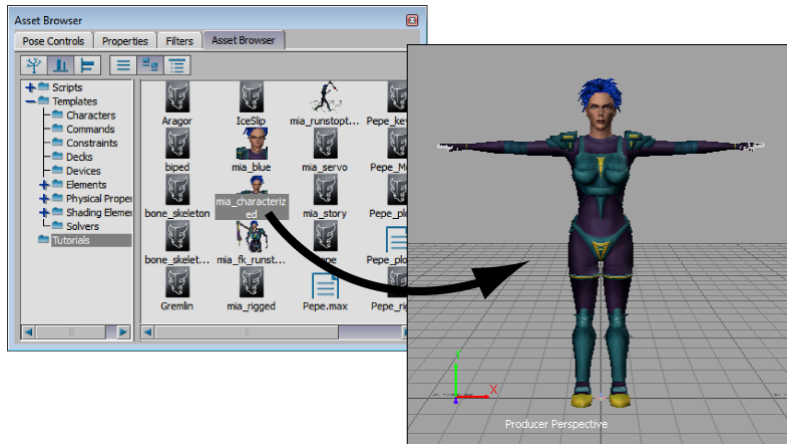
- *mia_characterized.fbx*

NOTE The tutorial assets can be found in the *Tutorials* folder in the Asset Browser and in the *Tutorials* folder in the MotionBuilder directory on your system.

Prepare the scene

In the following procedure, you prepare the MotionBuilder scene and open the file needed to start this tutorial.

- 1 From the menu bar, select File > New, then select Layout > Editing (or press *Ctrl-Shift-3*).
MotionBuilder displays a new 3D scene using the Editing layout. This layout displays all the windows you need for your work in this tutorial.
- 2 Click the *Tutorials* folder in the Asset browser.
- 3 Drag the *mia_characterized* asset (*mia_characterized.fbx* file) from the Asset browser into the Viewer window as shown in the following figure, then select FBX Open > No Animation.



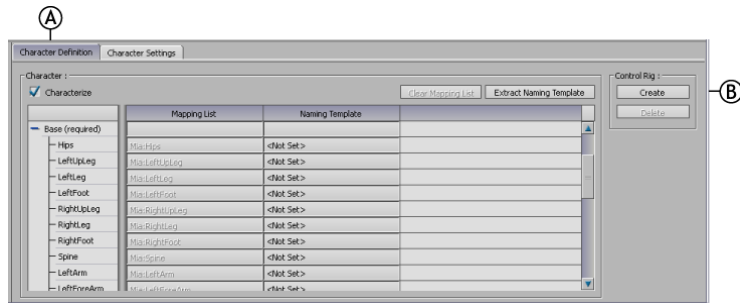
Mia shown in the T-stance

A model named Mia appears in the Viewer window, in the T-stance.

Create a Control rig

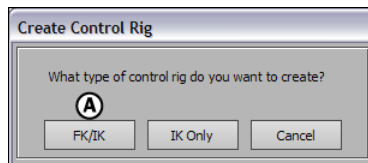
In the following procedure, you create and prepare a Control rig for the Mia character.

- 1 In the Scene browser, expand the *Characters* branch and double-click the *Mia* character.
- 2 Click the Character Definition pane (A) and click Create in the Control Rig area (B).



Character Definition: A. Settings B. Control Rig area

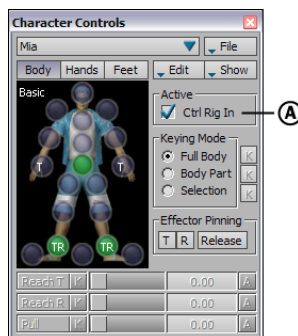
- 3 In the Create Control Rig dialog box that appears, select FK/IK (A).



Create Control Rig dialog box: A. FK/IK option

An FK/IK Control rig is created for the Mia Character.

- 4 In the Character Controls window, click Character Controls and activate the Ctrl Rig In option. This makes the Control rig the active motion source for the Mia character.



Character Controls: A. Ctrl Rig In option activated

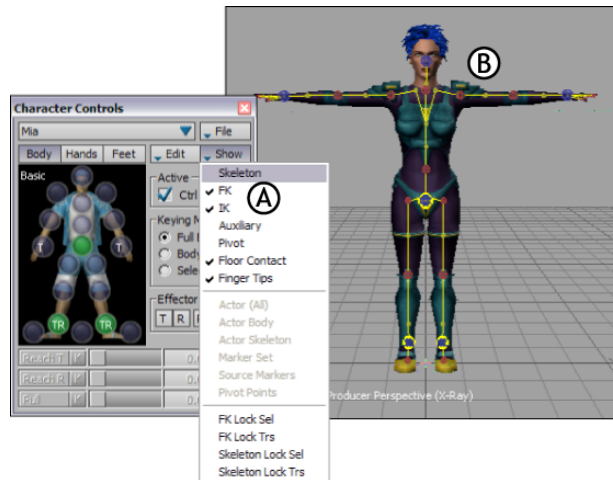
- 5 Click in the Viewer window, then press *Ctrl-A* until you are in X-Ray display mode.

NOTE The display mode is shown at the bottom left of the Viewer window.

In X-Ray display mode, you can see the FK and IK effectors that make up the Control rig. The blue and red IK effectors let you intuitively manipulate the character using a setup that simulates how the human body moves. The yellow FK effectors let you selectively fine-tune individual body parts. If you plan to do any fine-tuning with your characters, create Control rigs with both FK and IK effectors.

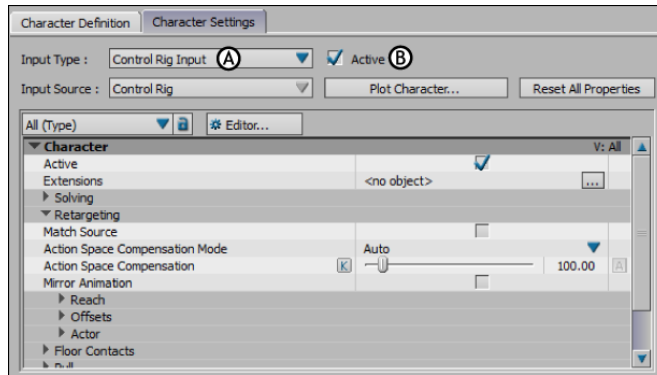
- 6 In the Character Controls window, open the Show menu (A) and disable the Skeleton option.

This hides the character's skeleton so you can clearly see the FK and IK Control rig effectors in the Viewer window (B).



A. Skeleton option disabled B. Control rig displays on the Mia character

- 7 In the Navigator window, switch back to view the Character Settings pane (B).



Character Settings pane: A. Input Type menu B. Active option shown enabled

The Control rig is also shown as the active motion source by the Input Type menu and the Active option in the Character Settings pane (A and B).

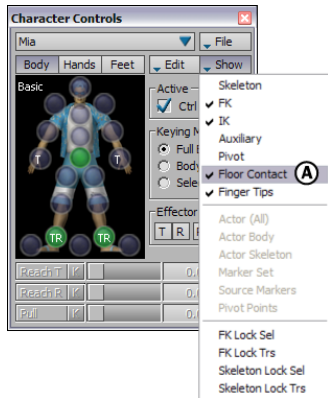
Adjust the foot floor contact markers

In the following procedure, you adjust how the character's feet touch the floor using the floor contact markers.

The floor contact markers are the blue and green markers that appear around the character's hands and feet when you characterize your character model. These markers create an invisible grid that determines where the character's feet come in contact with the floor.

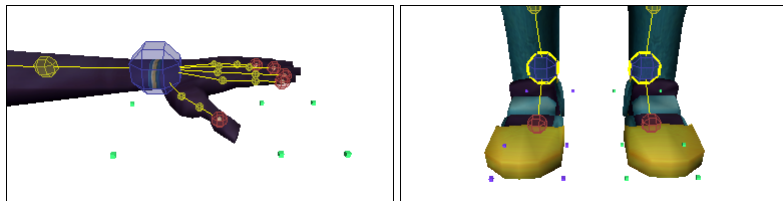
When no floor object is defined in the Mapping list of the Character Definition pane, the MotionBuilder grid is used as the floor. In this tutorial, the floor is not defined.

- 1 In the Character Controls window, make sure Floor Contact is active in the Show menu (A).



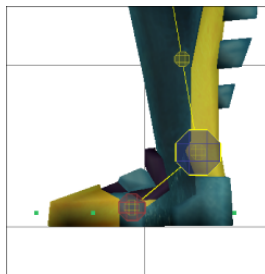
Character Controls: A. Floor Contact enabled

The green and blue floor contact markers display around Mia's hands and feet.

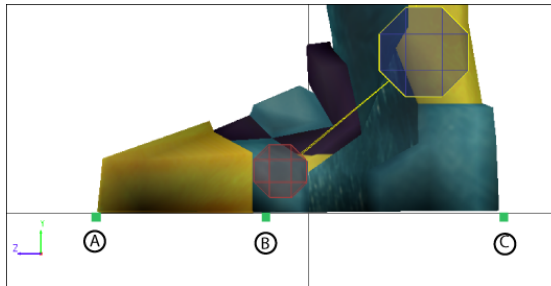


Green and blue floor contact markers displayed around Mia's hands and feet

- 2 In the Viewer window, click the View menu and select Orthographic > Producer Right (or click in the Viewer and press *Ctrl-R*) to switch to Producer Right camera view.
- 3 Zoom in on Mia's feet as shown in the following figure.



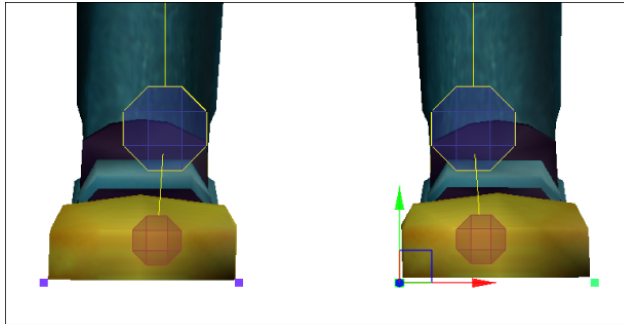
- 4 Click one of the floor contact markers underneath Mia's feet.
- 5 Click the Translate button in the Viewer toolbar (or press *T*) to activate the Translate mode.
- 6 Translate the floor contact marker using the following guidelines and figure for the marker placement:
 - Align the middle marker where the toe bone starts (B).
 - Align the front marker with the toe of the model (A).
 - Align the rear marker with the heel of the model (C).



Mia's foot floor contact markers: A. Front marker B. Middle marker C. Rear marker

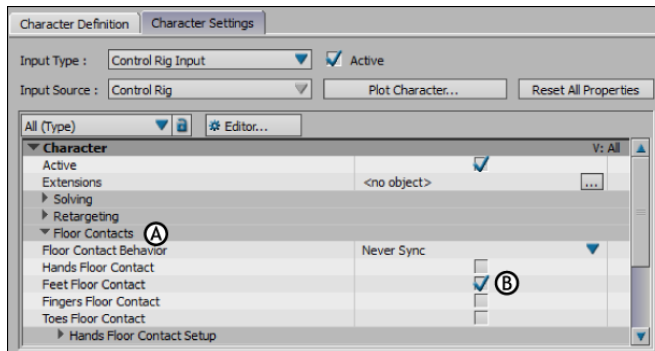
NOTE Moving one foot marker adjusts the other markers accordingly so that as you adjust the green markers on Mia's left foot, the blue markers on the right foot also get adjusted.

- 7 Switch to Orthographic Producer Front view (*Ctrl-F*), zoom in on the feet (*Ctrl-drag*), and translate the foot markers right or left to position them at the edges of the feet as shown in the following figure.



Left and right position of the markers

- 8 In the Character Settings pane, expand *Floor Contacts* (A) and activate the *Feet Floor Contact* option (B).

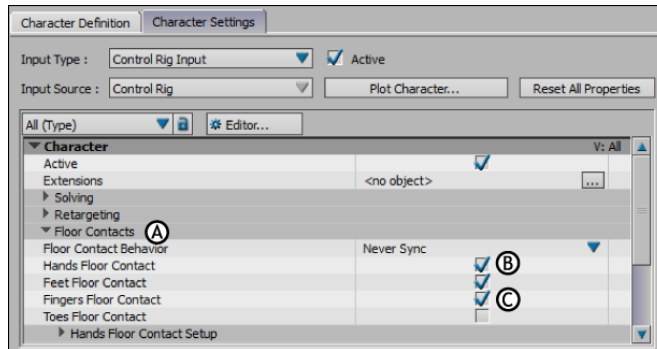


Character Settings pane A. Floor Contacts group of properties B. Feet Floor Contact option

Adjust the hand floor contact markers

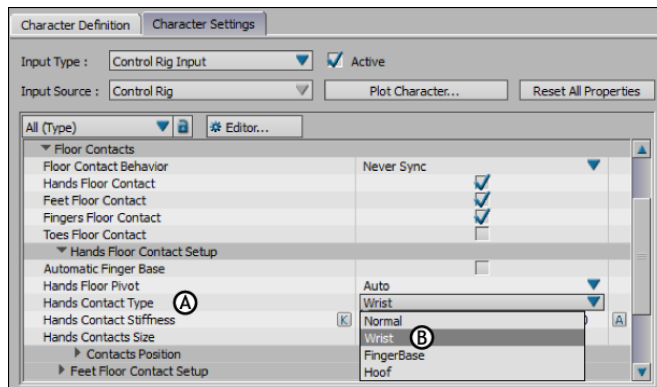
In the following procedure, you define the floor contact for the hands so that when Mia's hands touch the floor, it produces a realistic result.

- 1 In the Scene browser, double-click the *Mia* character and click the Character Settings pane to display the Character Settings.
- 2 In the Character Settings pane, expand *Floor Contacts* (A) if it is not already expanded, and activate the *Hands Floor Contact* (B) and the *Fingers Floor Contact* (C) options to activate the floor contact for the hands and fingers.



Character Settings pane: A. Floor Contacts B. Hands Floor Contact C. Fingers Floor Contact

- 3 Expand the *Hands Floor Contact Setup* option, click the *Hands Contact Type* menu (A) and select *Wrist*.

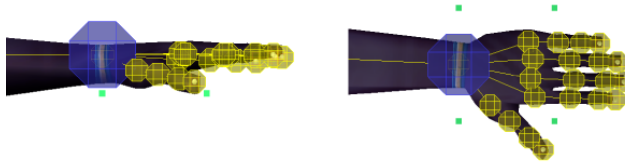


Character Settings pane: A. Hands Contact Type B. Wrist Hands Contact Type

By default, the *Hands Contact Type* is set to *Normal*, which gives Mia six hand floor contact markers. Changing this option to *Wrist* gives each hand four floor contact markers for basic control.

- 4 Zoom in on one of Mia's hands in the Viewer window using various camera views.
- 5 Align the rear hand markers with the wrist and the front markers with the base of the fingers (not including the thumb). Also translate the rear

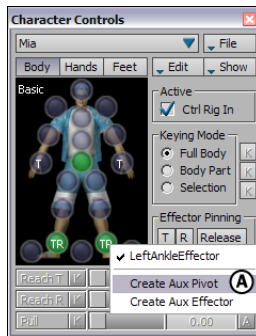
markers lower on the Y-axis to align them with the base of the palm as shown in the following figure.



Add Auxiliary pivots

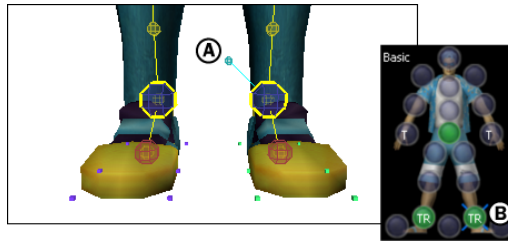
In the following procedure, you create two Auxiliary pivots for additional control over the IK system on Mia's Control rig. These Auxiliary objects can be used to create realistic rotation on Mia's feet as she walks. They also make it easier to rotate Mia's feet while creating keyframe animation.

- 1 In the Character Controls window, right-click the Left Ankle cell and select *Create Aux Pivot* from the contextual menu (A).



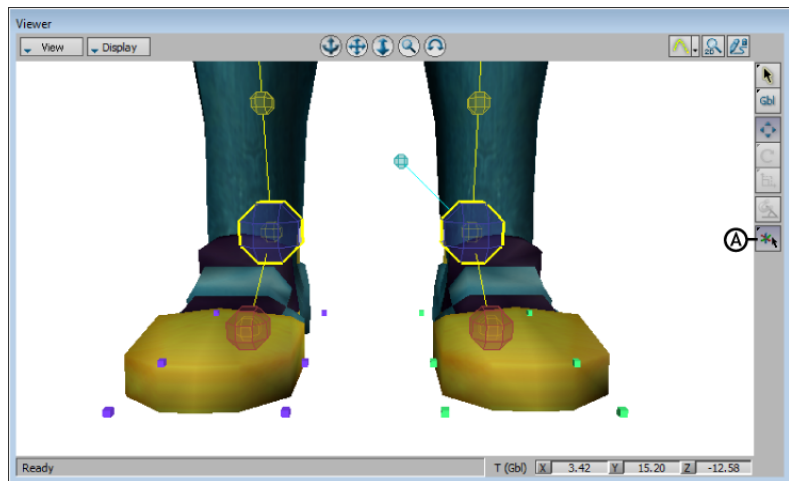
Character Controls A. Left Ankle effector cell contextual menu options

An Auxiliary pivot is created for the left ankle IK effector (A). The Auxiliary pivot displays on the left ankle cell in the Character Controls window as an X (B).



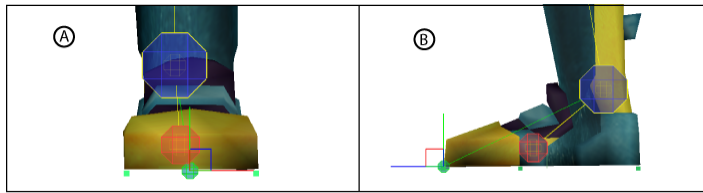
A. Auxiliary pivot displayed in the Viewer window B. X represents the Auxiliary pivot

By default, the foot effector is deselected when you create the Auxiliary pivot, and the Auxiliary pivot is selected. In the Viewer window, the Pivot Selection mode is automatically selected.



Viewer window A. Pivot Selection mode

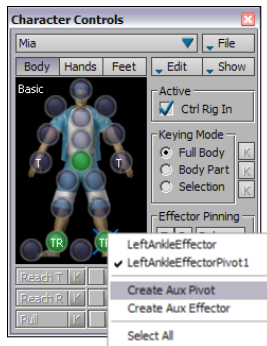
- 2 Select the Auxiliary pivot you created, if it is not already selected, and translate it until it is placed at the tip of Mia's toes, as shown in the following figure.



Left Ankle Auxiliary pivot A. Front view B. Side view

NOTE You can use the Show menu in the Character Controls to hide the Control rig effectors and floor contact markers as you place the Auxiliary pivots.

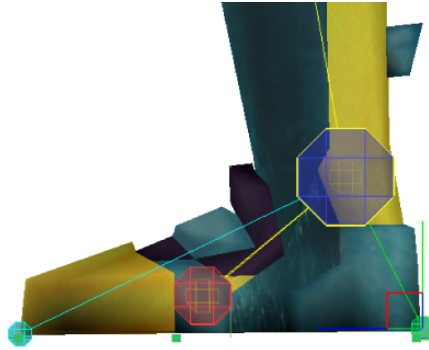
- 3 Right-click the Left Ankle cell again and select *Create Aux Pivot* from the contextual menu.



Create Aux Pivot on the left ankle

A second Auxiliary pivot displays in the Viewer window.

- 4 Translate the second Auxiliary pivot to display at the heel of the foot, as shown in the following figure.



Summary

In this tutorial, you created a Control rig, arranged the floor contact markers on the character's feet, then created two Auxiliary pivots to control the rotation of the foot.

In the next tutorial, ([Creating a Character Extension](#) on page 31), you add a Character Extension to the Mia character.

Creating a Character Extension

4

This tutorial guides you through the procedures necessary to create a Character Extension that enables you to control extra appendages for a character.

In this tutorial, you load a limb for the Mia character, attach it to Mia using a Character Extension, and define its animation in relation to Mia's body.

This tutorial shows you how to:

- [Prepare the scene](#) on page 31
- [Connect the extra limb to the character](#) on page 33
- [Create a Character Extension](#) on page 38

The following assets are required for this tutorial:

- *mia_rigged.fbx*
- *servo.fbx*

NOTE The tutorial assets can be found in the *Tutorials* folder in the Asset Browser and in the *Tutorials* folder in the *MotionBuilder* directory on your system.

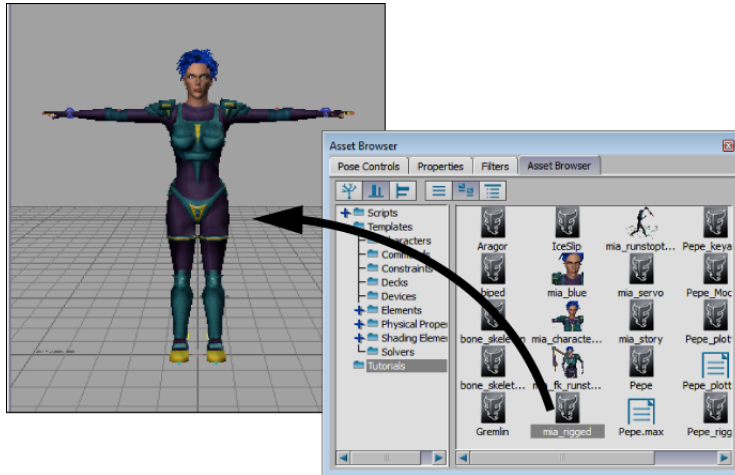
Prepare the scene

In the following procedure, you prepare the MotionBuilder scene and open the files needed to start this tutorial.

- 1 From the menu bar, select File > New, then select Layout > Editing (or press *Ctrl-Shift-3*).

MotionBuilder displays a new scene using the Editing layout. This layout displays all the windows you need for your work in this tutorial.

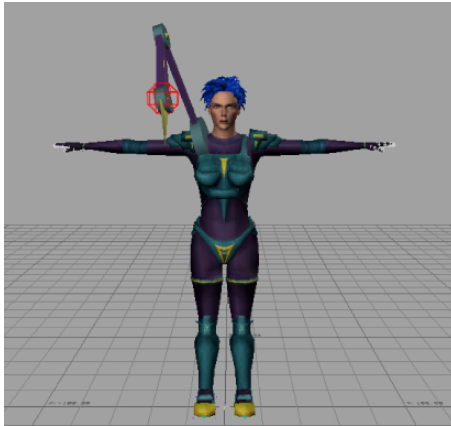
- 2 Click the *Tutorials* folder in the Asset browser.
- 3 Drag the *mia_rigged* asset (*mia_rigged.fbx* file) from the Asset browser into the Viewer window, then select **FBX Open > No Animation** from the contextual menu as shown in the following figure.



Characterized Mia appears in the Viewer window

A characterized character named Mia appears in the Viewer window in the T-stance.

- 4 From the Asset browser drag the *servo* asset (*servo.fbx* file) into the scene and select **FBX Merge > No animation** from the contextual menu.
A Servo arm is loaded into the scene, positioned over Mia's shoulder. In the next procedure, you attach this arm to Mia as another limb.



Mia and Servo arm loaded into the scene

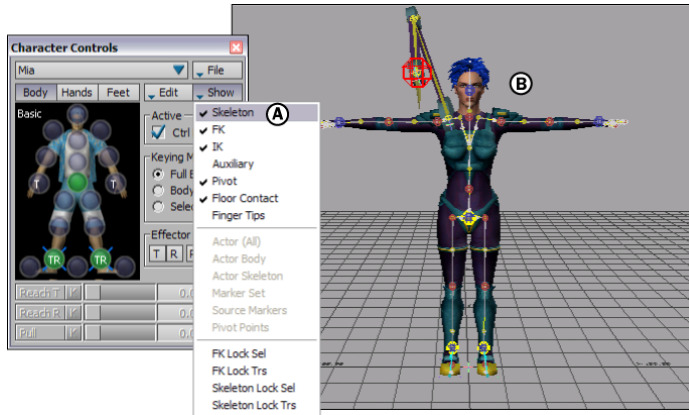
Connect the extra limb to the character

In the following procedure, you create a Parent-Child relationship between the Servo arm and Mia's shoulder.

- 1 Switch to X-Ray display mode (*Ctrl-A*) in the Viewer window.

NOTE Toggle between Normal mode, Models Only mode, and X-Ray mode using the keyboard shortcuts *Ctrl-A*.

- 2 In the Character Controls window, make sure Mia is selected as the current character and activate the Skeleton option in the Show menu (A). The FK and IK options should also be selected in the Character Controls Show menu (A) if they are not already.

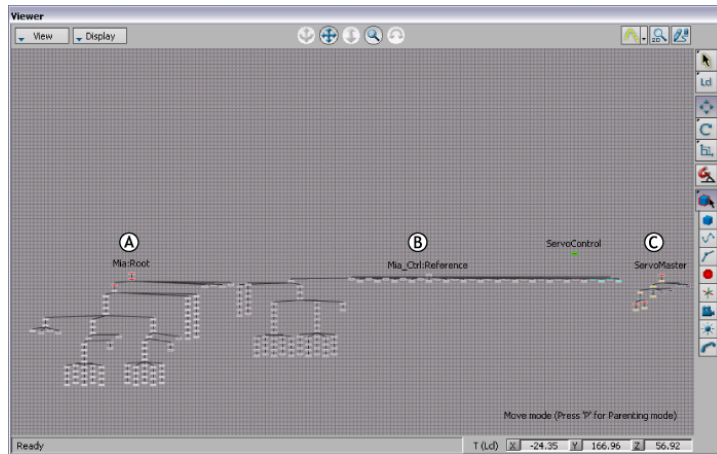


A. Skeleton activated B. Skeleton displayed on the Mia character

The Skeleton display makes it easier to view and select Mia's shoulder bone.

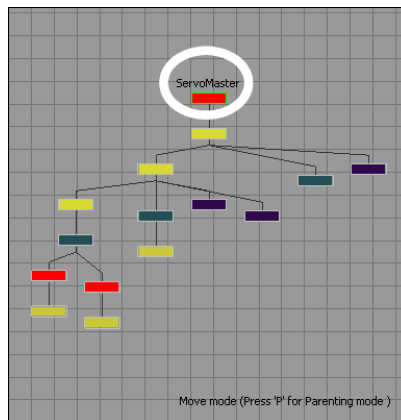
3 Click in the Viewer window then do the following:

- Press **Ctrl-W** to switch to the Schematic view.
- Right-click in the Schematic view and select **Auto-Arrange** and then **Arrange-All** from the contextual menu.
- Press **A** to see all the nodes in the Schematic view.
The Schematic view displays a hierarchy for Mia's skeleton (A), Mia's Control rig (B), and a third hierarchy for the Servo arm ("ServoMaster", C).



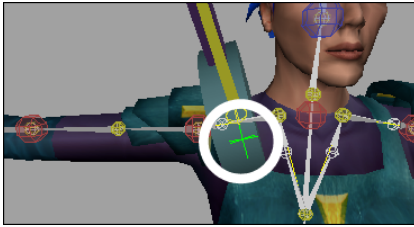
Schematic view of hierarchies in the scene: A. Mia's skeleton B. Mia's Control rig C. The Servo arm

- 4 Zoom in on the Servo arm (ServoMaster) hierarchy at the right of the view (C) and select the ServoMaster node.



ServoMaster node selected

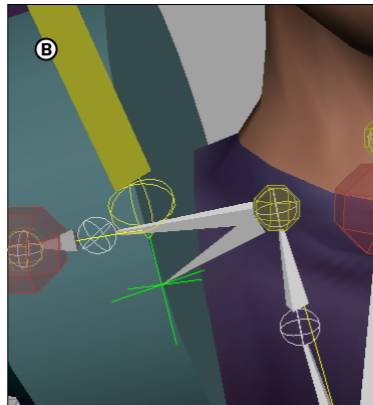
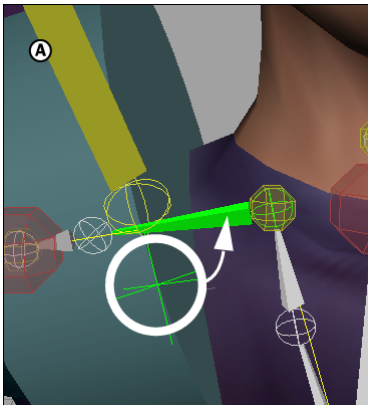
- 5 Switch back to the Producer camera view (*Ctrl-W*) and zoom in on Mia's right shoulder. The ServoMaster node null is still selected.



The ServoMaster null is selected in the Viewer window

- 6 Press P to activate Parenting mode, then drag the ServoMaster null to the Mia:RightShoulder bone (A).

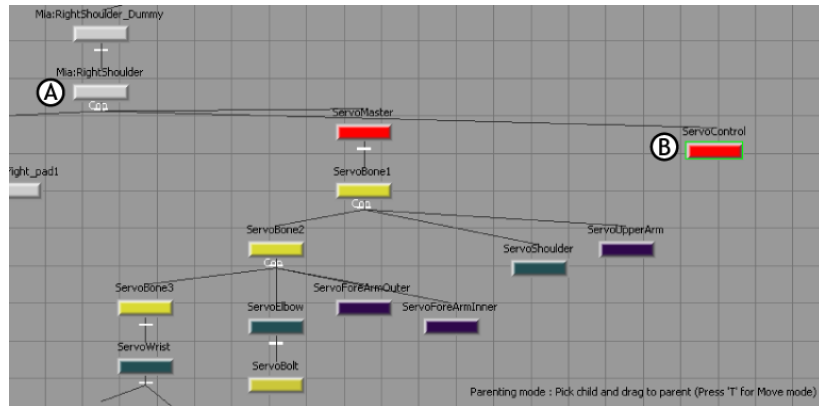
The bone is highlighted green as you parent the Servo arm. This parents the Servo arm to the right shoulder bone (B).



A. Parenting the ServoMaster null to Mia's right shoulder bone B. After parenting

- 7 Switch to the Schematic view to verify that the Servo arm is a child of the Mia:RightShoulder bone.

- 10** Switch to the Schematic view to verify that the ServoControl effector is a child of the Mia:RightShoulder bone.

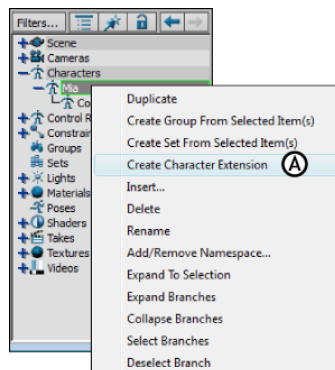


Schematic view showing parenting structure A. Mia:RightShoulder bone B. ServoControl effector node

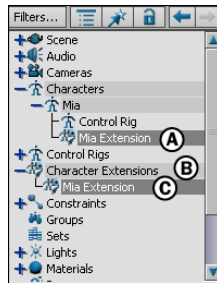
Create a Character Extension

In the following procedure, you create a Character Extension to connect the Servo arm to the Mia character so that they can be controlled and keyframed together.

- 1 In the Scene browser, expand Characters, right-click Mia, and select Create Character Extension from the contextual menu (A).

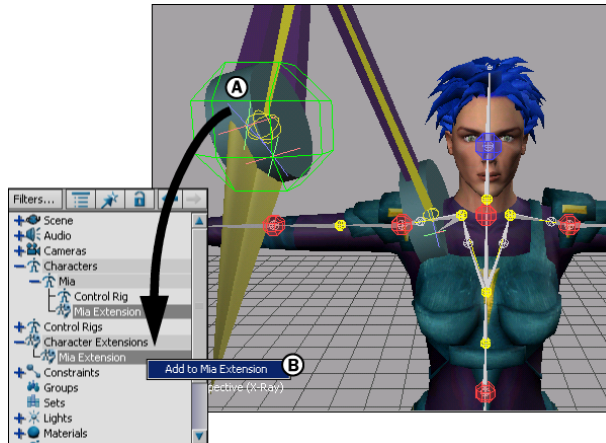


- 2 Expand the Character Extensions folder in the Scene browser to see the Mia Extension (C).



Scene browser: A. Character Extension added to Mia character B. Character Extensions heading added to the Scene browser C. Character Extension named for the Mia character

- 3 Switch back to the Producer Perspective view.
- 4 *Alt*-drag the ServoControl effector from the Viewer window onto the *Mia Extension* and select *Add to Mia Extension* from the contextual menu as shown in the following figure.

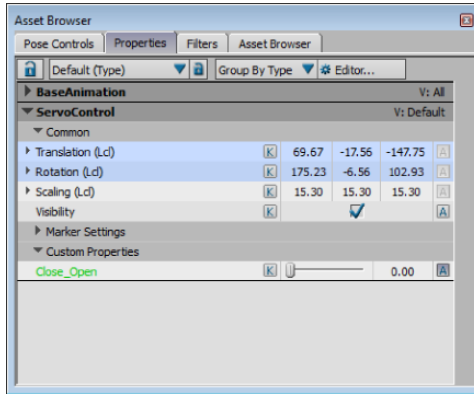


A. ServoControl effector B. ServoControl effector added to Mia Extension

The Servo arm is defined as a Character Extension of Mia, and is considered as a new “body part” of the Mia character.

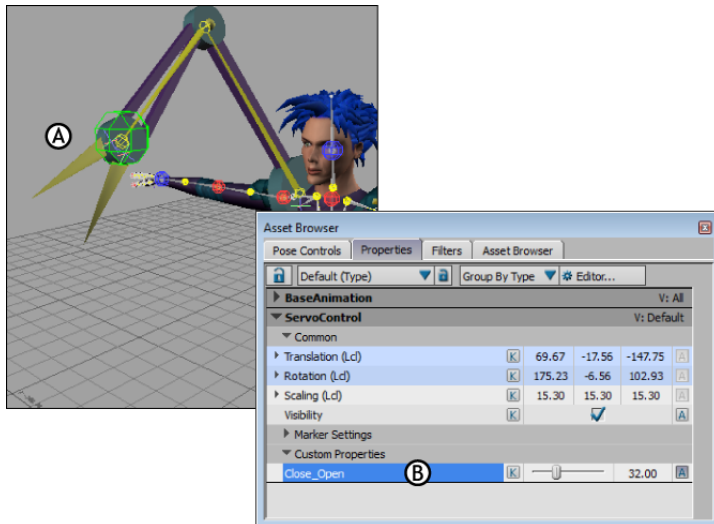
- 5 With the ServoControl effector still selected, open the Properties window.

NOTE The Properties window is on the right side of the interface, in one of the Asset browser tabs.



Asset browser Properties window

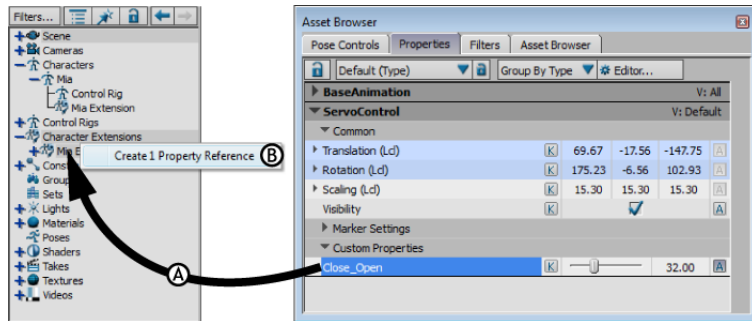
- 6 Select the green custom property *Close_Open* and drag its slider left and right as shown in the following figure.



A. Servo arm opens B. Custom property *Close_Open*

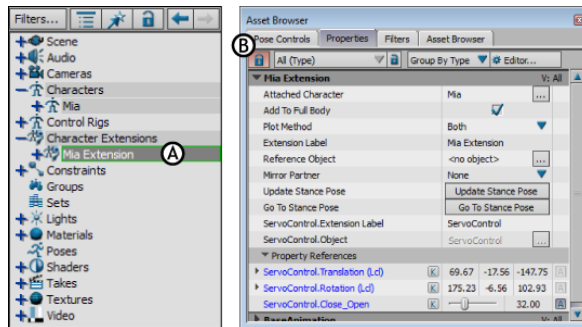
In the Viewer window, the pincer moves on the Servo arm.

- 7 *Alt*-drag the *Close_Open* property over the *Mia Extension* (A) and select *Create 1 Property Reference* from the contextual menu (B).



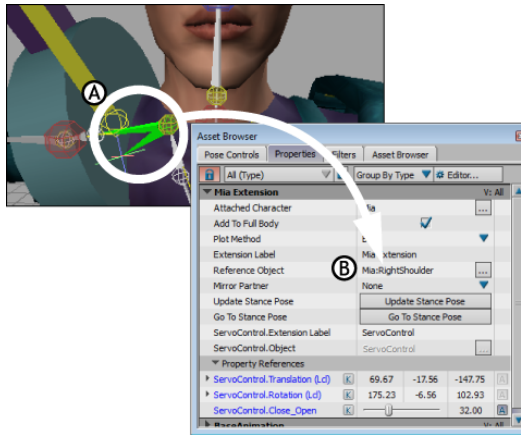
A. *Close_Open* property dragged to *Mia Extension* B. Select *Create 1 Property Reference*

- 8 Select the *Mia Extension* in the Scene browser (A), then activate the *Lock* option in the Properties window (B) so that the *Mia Extension* properties stay open no matter what you select.



A. *Mia Extension* selected B. *Lock* option activated in the Properties window

- 9 Define *Mia*'s right shoulder bone as the Reference object for the Character Extension by *Alt*-dragging the *Mia:RightShoulder* bone (A) into the Reference Object field in the Properties window (B).



A. Mia:RightShoulder B. Mia:RightShoulder Reference Object

- 10 Click OK in the Reference Object Change dialog box that appears.
The Reference object for your Character Extension is used to calculate all future positioning of the Extension, for example when the Character Extension is included in a pose.

NOTE You can also use the Include Part In Full Body option to define whether you want the Character Extension to be keyed when you set keys in Full Body Keying mode.

Summary

In this tutorial you added a limb to the Mia character by creating a Character Extension.

In the next tutorial, ([Creating a Walk Cycle](#) on page 43), you animate the character and the Character Extension using the Pose Controls.

Creating a Walk Cycle

5

This tutorial guides you through the procedures necessary for using poses to create a walk cycle.

This tutorial shows you how to:

- [Prepare the scene](#) on page 43
- [Create poses](#) on page 44
- [Create animation with poses](#) on page 51
- [Mirror poses](#) on page 54
- [Play the animation](#) on page 57

The following asset is required for this tutorial:

- *mia_servo.fbx*

NOTE The tutorial assets can be found in the *Tutorials* folder in the Asset Browser and in the *Tutorials* folder in the MotionBuilder directory on your system.

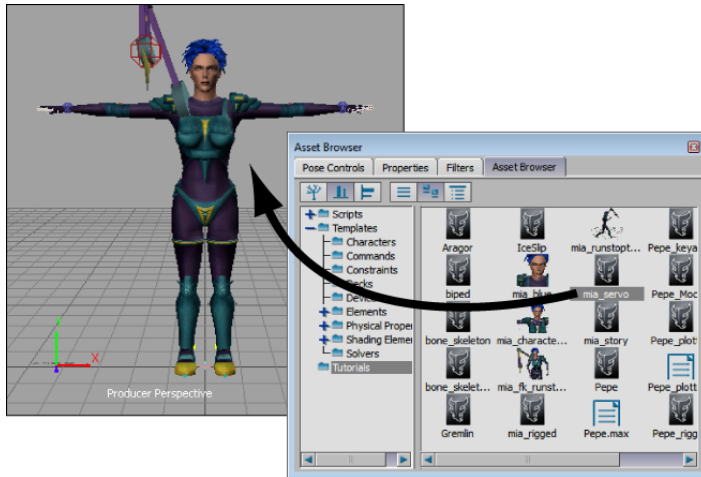
Prepare the scene

In the following procedure, you prepare the MotionBuilder scene and open the files needed to start this tutorial.

- 1 From the menu bar, select File > New, then select Layout > Editing (or press *Ctrl-Shift-3*).

MotionBuilder displays a new 3D scene using the Editing layout. This layout displays all the windows you need for your work in this tutorial.

- 2 Click the *Tutorials* folder in the Asset browser.
- 3 Drag the *mia_servo* asset (*mia_servo.fbx* file) into the Viewer window, then select **FBX Open > No Animation** from the contextual menu that appears. A model named Mia appears in the Viewer window as shown in the following figure.



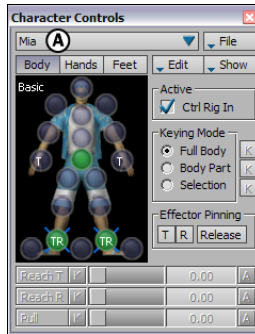
mia_servo appears in the scene

This character includes a “Servo arm” that is parented to the right shoulder bone and added as a Character Extension.

Create poses

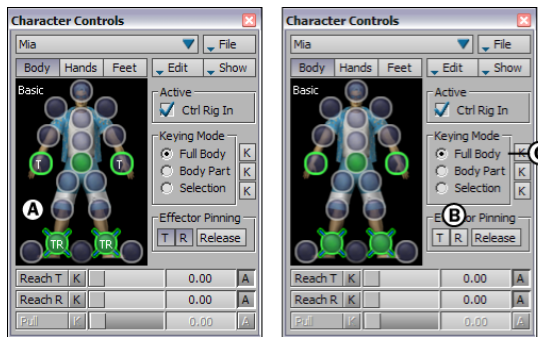
In the following procedure, you create several full body poses on your character, including the Character Extension, to create a walk cycle.

- 1 In the Character Controls window, click **Character Controls** and select Mia in the **Current Character** menu (A), if she is not selected already.



Character Controls window: A. Mia selected from the Current Character menu

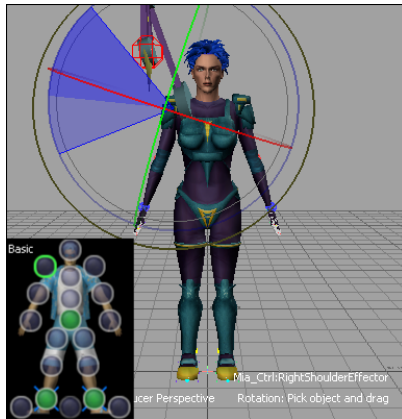
- 2 *Ctrl*-click to select the wrist and ankle effectors (A) and turn off all effector pinning by disabling the T and R options in the Effector Pinning area (B).



Character Controls window: A. Wrist and Ankle effector cells shown selected B. T and R pinning shown disabled C. Full Body Keying mode shown active

In the following step, you use the default Full Body keying mode (C). In Full Body Keying mode, pasted poses are placed onto the character's entire body, and keyframes are placed on all effectors.

- 3 Select the Left Shoulder effector, then press R and rotate the effector until Mia's left arm is in a more natural position at her side. Repeat for the Right Shoulder and right arm.



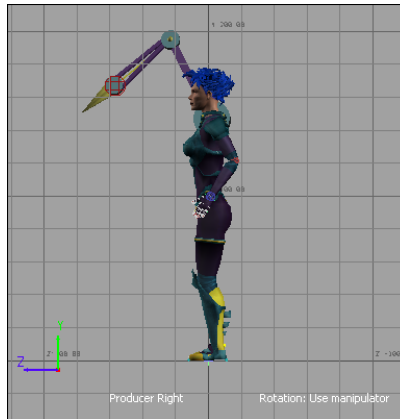
Mia's arms rotated downward in Z-axis using the shoulder effectors

- 4 Select both wrist effectors and translate them upward on the Y-axis to give the elbows a natural bend.



Arms translated upward in Y-axis using the wrist effectors

- 5 Choose a camera view that lets you see a side view of the character. For example you can press *Ctrl-R* to switch to the Producer Right camera view.



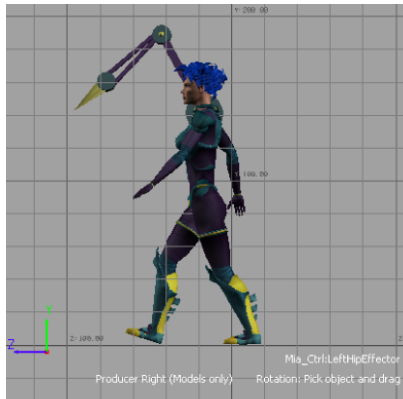
Producer Right camera view

NOTE You can switch the camera view at any time during the tutorial to get a better view.

Create the first pose for the walk cycle:

- 1 In the Character Controls, select the Right Hip effector then rotate (press *R*) the right leg forward on the Z-axis, as if Mia is stepping forward.
- 2 Select the Left Hip effector and rotate the left leg slightly backward on the Z-axis.
- 3 Select the Right Shoulder effector and rotate the right arm slightly backward, then select the Left Shoulder rotate the left arm forward as if Mia is naturally swinging her arms.
- 4 Select the Auxiliary pivots (*AnkleEffectorPivots*) on Mia's feet and rotate them until her feet are positioned naturally.

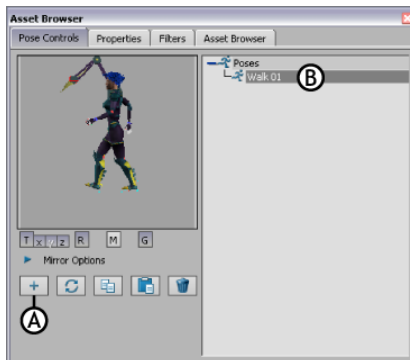
This pose should have Mia with her right leg beginning the forward motion of a step, as shown in the following figure.



First pose ready to create

NOTE If your transformations cause Mia to float above the floor, select the Hips effector and translate Mia downward at any time. The default floor contact makes Mia’s feet interact naturally with the default floor.

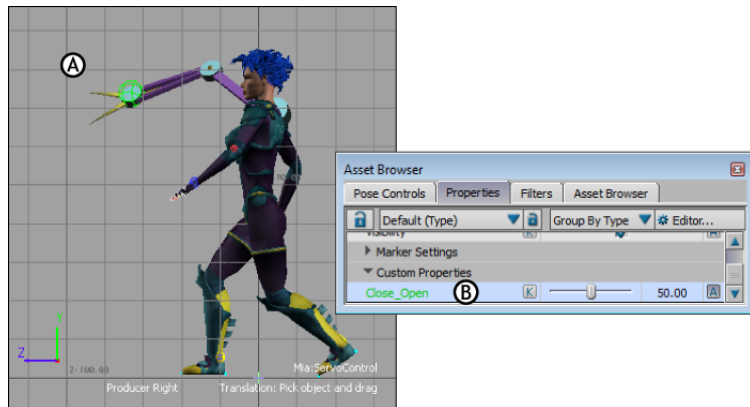
- 5 In the Pose Controls window, click the Pose Controls tab and click Create (A) to add this pose to the Pose browser.
- The position of the Character Extension is included with the position of Mia’s body in this pose.
- 6 Expand the *Poses* folder to see the pose.
- 7 Right-click the pose created and rename the pose “Walk 01” (B).



Pose Controls: A. Create pose B. Pose renamed to “Walk 01”

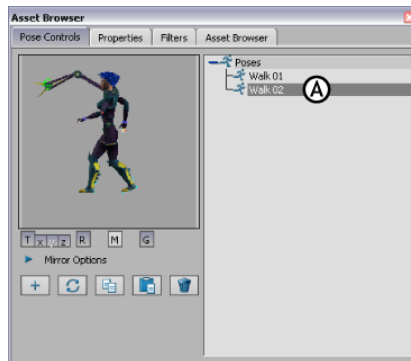
Create the second pose for the walk cycle:

- 1 Position Mia's legs and arms so that she looks similar to the following figure. The right foot is forward and on the ground, and the left foot is back to provide momentum.
- 2 Select the Mia:ServoControl effector, and translate the Servo arm so it reaches in front of Mia.
- 3 With the Mia:ServoControl effector still selected, click the Asset browser Properties tab and use the Close_Open property (B) to open the pincers about half-way as shown in the following figure.



A. Second pose for the walk cycle B. Close_Open property to control the pincers

- 4 In the Pose Controls, click Create and rename this pose as "Walk 02" (A).

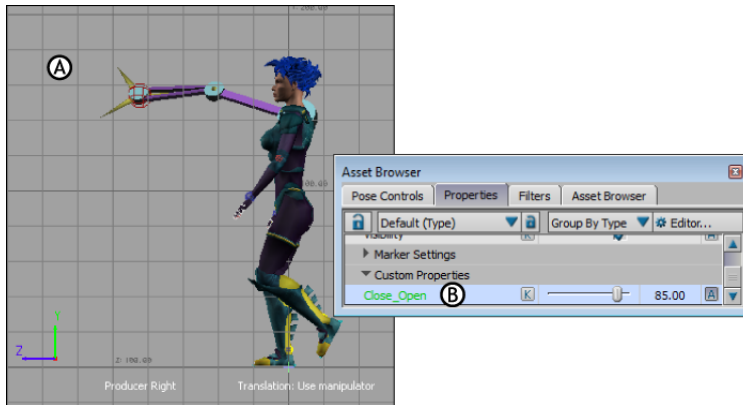


Pose Controls: A. Second pose renamed "Walk 02"

Create the third and final pose for the walk cycle:

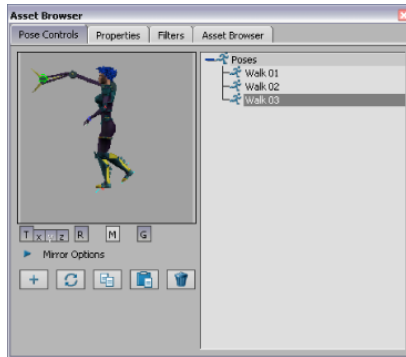
- 1 Position Mia's legs and arms so that her step appears similar to the following figure.
- 2 Select the Mia:ServoControl effector and extend the Servo arm to reach even further in front of Mia.
- 3 With the Mia:ServoControl still selected, open the pincers further using the Close_Open property in the Properties window (B).

In this pose, the left leg goes back, and the Servo arm goes forward, completing one step for the first half of the walk cycle.



A. Third pose for the walk cycle B. The Close_Open property

- 4 In the Pose Controls, click Create and then rename this pose "Walk 03". You now have three poses in the scene. The three poses are listed in the Pose browser.

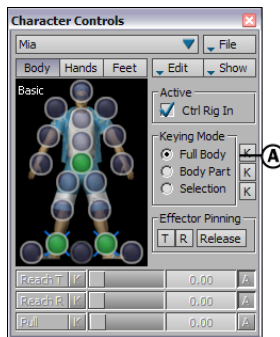


Three poses created for the walk cycle

Create animation with poses

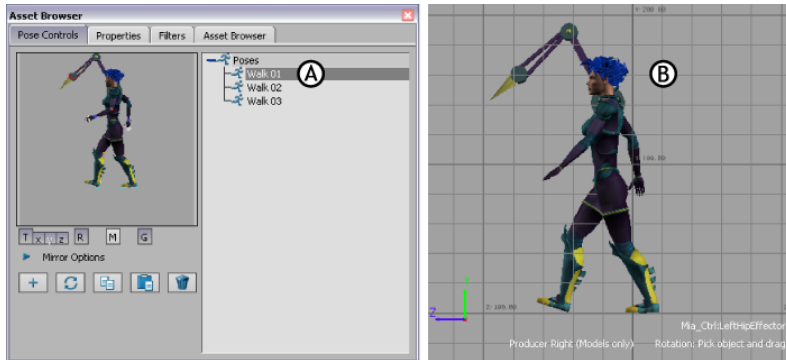
In the following procedure, you use the three poses you created to create one half of a walk cycle. By keyframing these poses at different frames, you create a short walking animation.

- 1 In the Character Controls window, deselect any effectors that may be still selected and ensure that Full Body keying mode is selected (A).



Character Controls: A. Full Body Keying mode selected

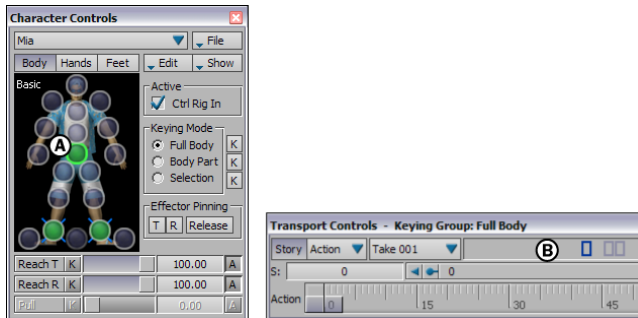
- 2 In the Pose browser, double-click the “Walk 01” pose to paste it on Mia.



Pose Controls: A. First pose for the walk cycle B. "Walk 01" pose pasted on Mia

By default, the Gravity, Translation, and Rotation options are active in the Pose Controls window. This means that the translation and rotation of the pasted pose match the translation and rotation of the selected effector on the current character. The Gravity option ensures that the feet stay at the original level of the pasted pose (normally floor level).

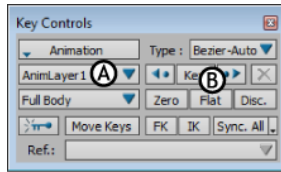
- 3 Select the Hips effector, then go to frame 0 in the Transport Controls.



A. Hips effector selected B. Current frame is 0

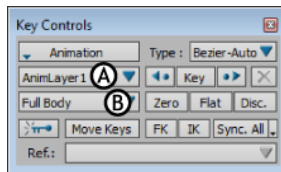
- 4 In the Key Controls window, select AnimLayer1 from the Layer menu (A), then click Flat to set a Flat keyframe (B). You can also press *Ctrl-K* on the keyboard to set a Flat keyframe.

NOTE Bezier-Auto should be selected in the Type menu.



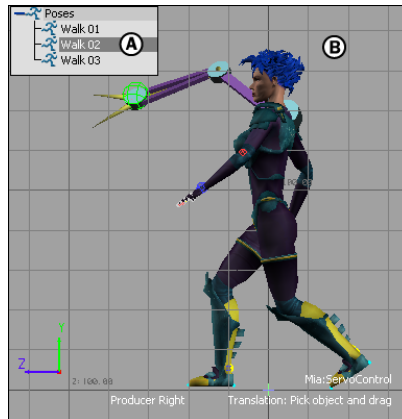
Key Controls A. AnimLayer1 selected B. Flat key

The keyframe is set on AnimLayer1 (A) for the position of the character's full body (B), as indicated in the Key Controls window. The keying mode in the Key Controls (B) reflects the one selected in the Character Controls window.



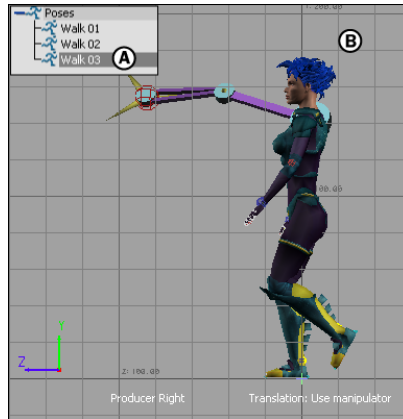
Key Controls A. AnimLayer1 selected B. Full Body keying mode selected

- 5 Go to frame 5 and do the following:
 - Double-click the Walk 02 pose to paste it on Mia.
 - Press *Ctrl-K* to set a (Flat) keyframe.



Frame 5: A. Second pose for the walk cycle B. "Walk 02" pose pasted on Mia

- 6 Go to frame 10 and do the following:
 - Double-click the Walk 03 pose to paste it on Mia.
 - Press *Ctrl-K* to set a (Flat) keyframe.



Frame 10: A. Third pose for the walk cycle B. "Walk 03" pose pasted on Mia

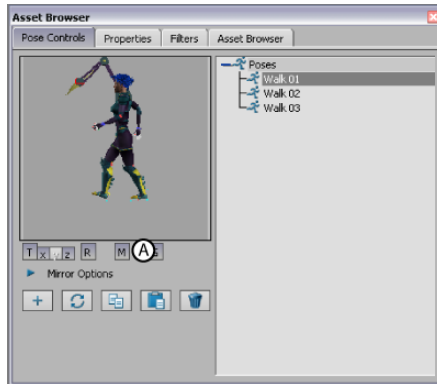
- 7 Drag the Timeline indicator through the animation to view the step you created.

The interpolation between the three keyframes creates the movement for one step.

Mirror poses

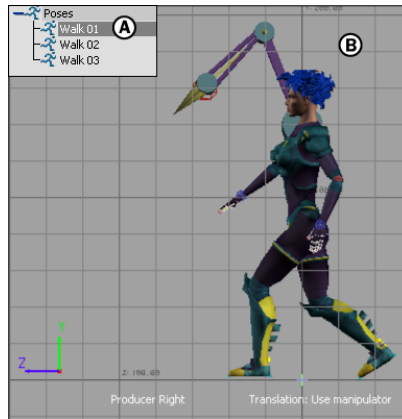
In the following procedure, you mirror the three poses from the Pose browser to create the second half of the walk cycle. By keyframing these mirrored poses after the original poses, you complete the short walking animation.

- 1 In the Pose Controls window, activate the Mirror option (A).



Pose Controls: A. Mirror option activated

- 2 Go to frame 15 and do the following:
 - Double-click the “Walk 01” pose (A).
The “Walk 01” pose is pasted and mirrored onto the character (B).
Because you mirror-pasted the pose, the left leg is now forward, and the right leg is behind to continue the walk cycle on the other side.
 - Press *Ctrl-K* to set a (Flat) keyframe.

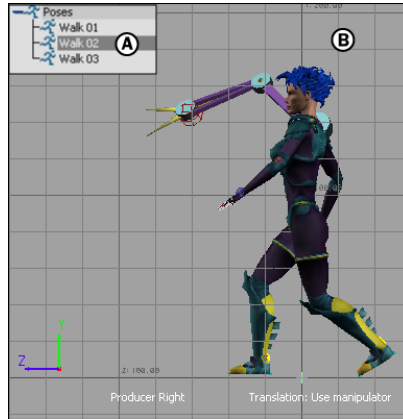


Frame 15: A. First pose for the second half of the walk cycle B. “Walk 01” pose pasted on Mia

- 3 Go to frame 20 and do the following:
 - Double-click the “Walk 02” pose (A).

The “Walk 02” pose is pasted and mirrored onto the character (B).

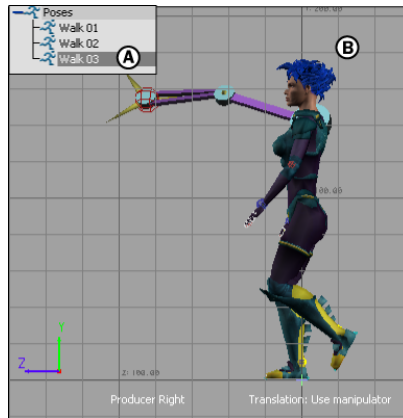
- Press *Ctrl-K* to set a (Flat) keyframe.



**Frame 20: A. Second pose for the second half of the walk cycle
B. “Walk 02” pose pasted on Mia**

- 4 Go to frame 25 and do the following:

- Double-click the “Walk 03” pose (A).
The “Walk 03” pose is pasted and mirrored onto the character (B).
- Press *Ctrl-K* to set a (Flat) keyframe.



**Frame 25 A. Third pose for the second half of the walk cycle
B. “Walk 03” pose pasted on Mia**

Your animation now consists of six keyframes. The first three keyframes were mirrored onto the left side of the character for the last three keyframes, creating a complete walking movement. To complete a full animation cycle, your take should begin and end with the same position.

- 5 On the Action timeline, copy the keyframe at frame 0 to frame 30 by C-dragging the keyframe from frame 0 to frame 30.



Keyframe at frame 0 copied to frame 30

The animation now begins and ends on the same position, creating a complete cycle.

Play the animation

- 1 Click on the Action timeline, then press *Ctrl-Shift-A* to frame the animation on the Action timeline to its full length of 30 frames.



Animation framed A. Loop option

- 2 In the Transport Controls window, click Loop (A), then click Play. As the animation plays, each loop shows a full walk cycle.

In your animation, the movement may be a bit choppy, and the feet may slide on the floor. You can smooth your movement by adjusting the animation's function curves in the FCurves window.

Summary

In this tutorial, you created poses on a character, set keyframes of these poses at different points, and created a walk cycle.

NOTE You can also create a loop if you want to create a walk cycle using the Story window.

In the next tutorial, [Retargeting Character Animation](#) on page 59, you retarget animation and a Character Extension from one character to another.

Retargeting Character Animation

6

This tutorial guides you through the procedures necessary to retarget animation from one characterized character to another. Since the source character includes a Character Extension and the target character does not, you must also retarget the Character Extension.

This tutorial shows you how to:

- [Prepare the scene](#) on page 59
- [Save the character animation](#) on page 61
- [Create a scene](#) on page 63
- [Load character animation](#) on page 65
- [Play the animation](#) on page 68

The following assets are required for this tutorial:

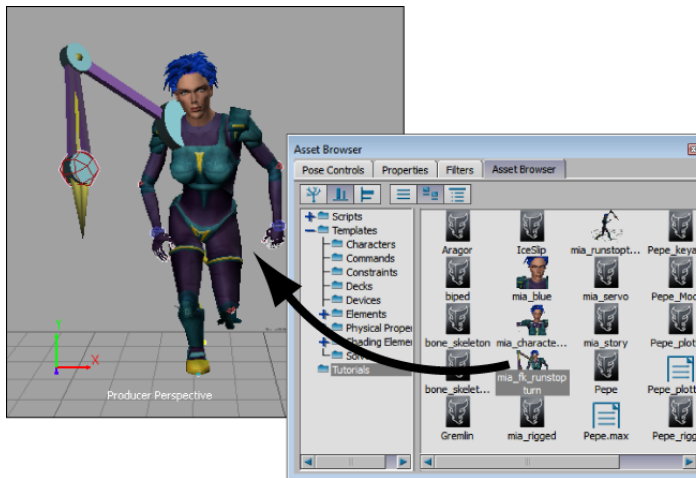
- *mia_fk_runstopturn.fbx*
- *Gremlin.fbx*

NOTE The tutorial assets can be found in the *Tutorials* folder in the Asset Browser and in the *Tutorials* folder in the MotionBuilder directory on your system.

Prepare the scene

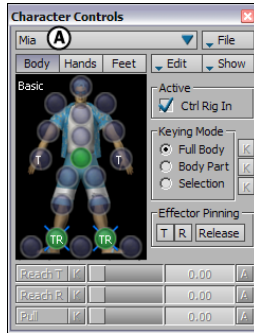
In the following procedure, you prepare the MotionBuilder scene and open the files needed to start this tutorial.

- 1 From the menu bar, select File > New, then select Layout > Editing (or press *Ctrl-Shift-3*).
MotionBuilder displays a new 3D scene using the Editing layout. This layout displays all the windows you need for your work in this tutorial.
- 2 Click the *Tutorials* folder in the Asset browser.
- 3 Drag the *mia_fk_runstopturn* asset (*mia_fk_runstopturn.fbx* file) into the Viewer window as shown in the following figure, then select FBX Open > All takes from the contextual menu that appears.
A model named Mia appears in the Viewer along with her “Servo arm” Character Extension.



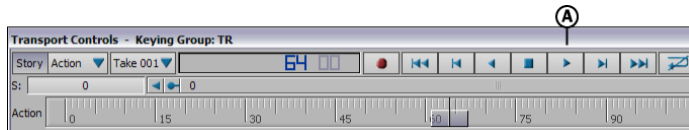
Mia_fk_runstopturn asset dragged into the scene

- 4 In the Character Controls, click Character Controls and make sure Mia is selected in the Current Character menu (A).



Character Controls A. Mia is the Current Character

- 5 In the Transport Controls, click Play (A) to view the animation on the Mia character.

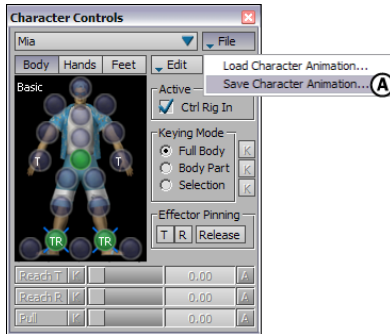


Transport Controls A. Play button

Save the character animation

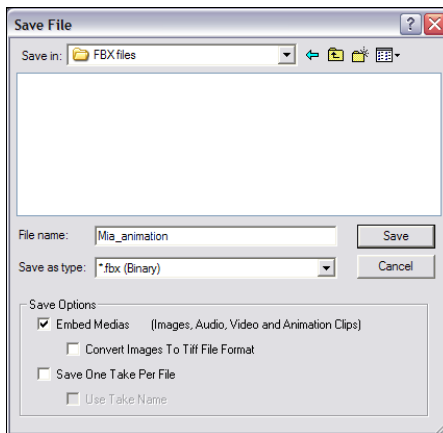
In the following procedure, you save the character animation using the Save Character Animation option. This prepares the animation to be easily loaded using the Load Character Animation option.

- 1 In the Character Controls window, select File> Save Character Animation (A).



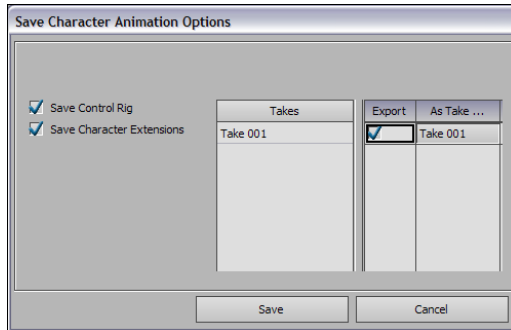
Character Controls A. Save Character Animation option

- 2 Navigate to where you want to save the character animation, enter a file name, and click Save.



Save File dialog box

- 3 In the Save Character Animation Options dialog box that appears, activate the Save Control Rig option and the Save Character Extensions option, then click Save.



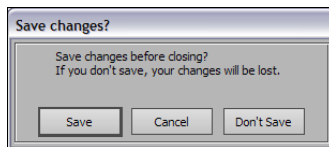
Save Character Animation Options dialog box

The animation and Character Extension are saved as an *.fbx* file.

Create a scene

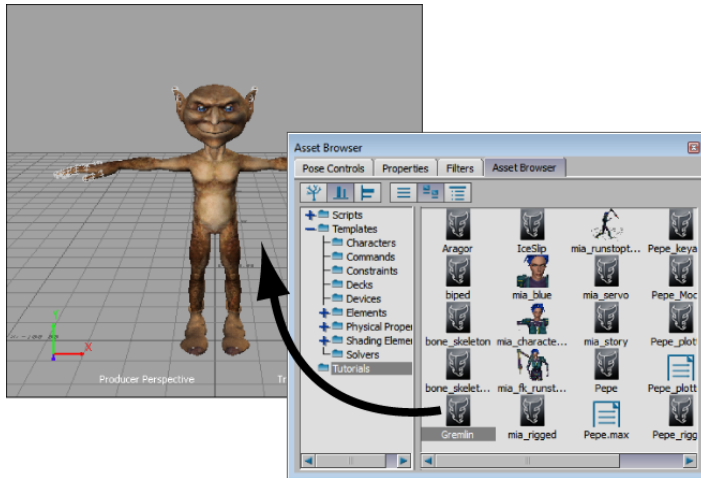
In the following procedure, you create a scene.

- 1 Press *Ctrl-N* to create a scene.
- 2 In the Save changes dialog box appears, click Don't Save.



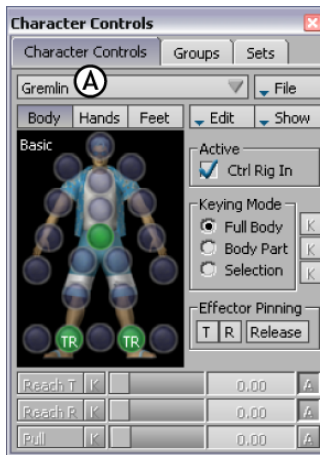
Save changes dialog box

- 3 From the Asset browser, drag the *Gremlin* asset into the scene, and select FBX Open > No animation.



Gremlin asset dragged into the scene

- 4 In the Character Controls, make sure Gremlin is selected in the Current Character menu (A).

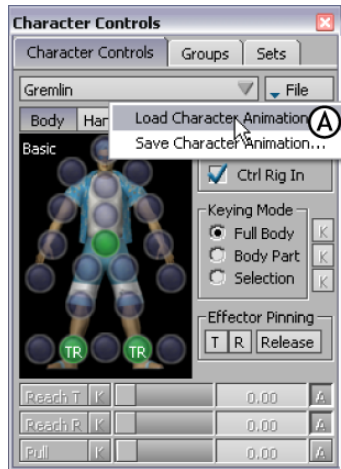


Character Controls: A. Gremlin is the current character

Load character animation

In the following procedure, you load the character animation you saved earlier.

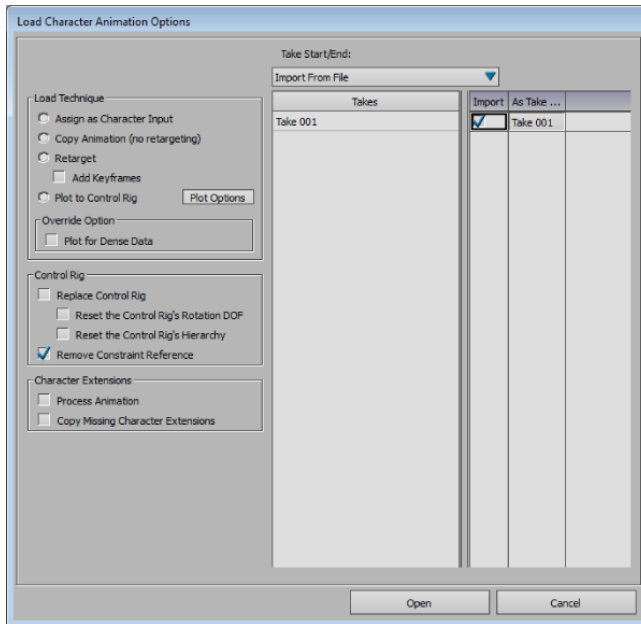
- 1 In the Character Controls window, select File > Load Character Animation (A).



Character Controls: A. Load Character Animation option

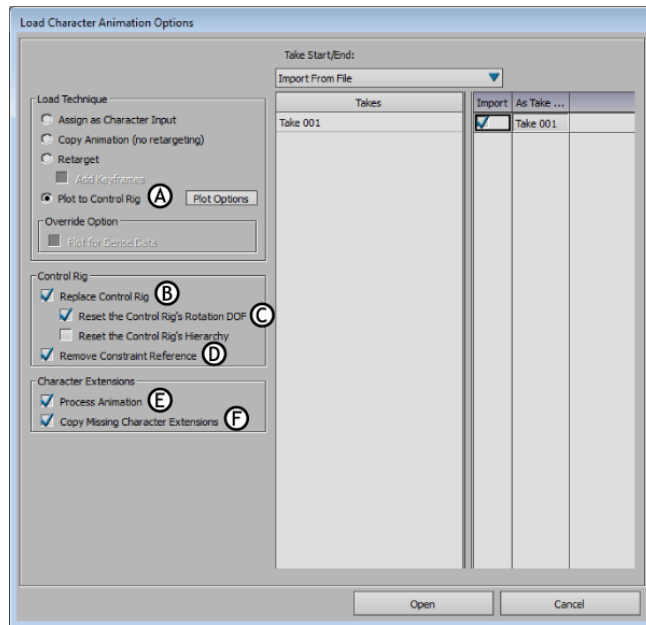
- 2 Navigate to select the *.fbx* file you saved earlier in this tutorial and click Open.

The Load Character Animation Options dialog appears.



Load Character Animation Options dialog

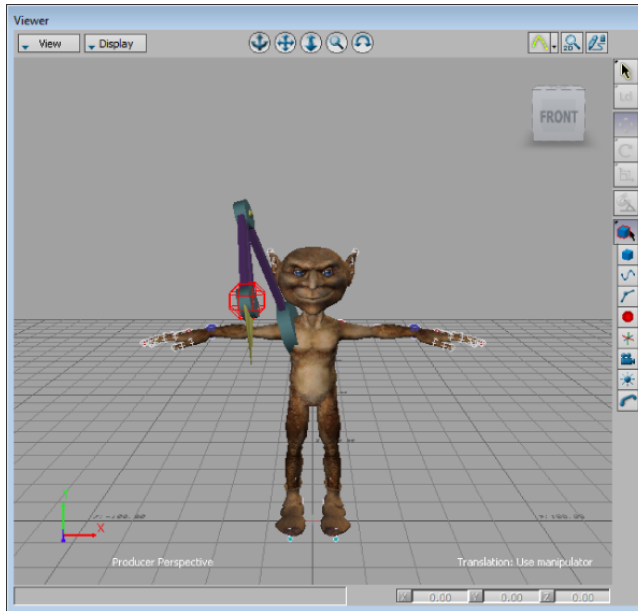
- 3 Choose the following settings:
 - In the Load Technique area (A), select the Plot to Control Rig option.
 - In the Control Rig area, activate the Replace Control Rig option (B).
 - In the Control Rig area, activate the Reset Control Rig's Rotation DOF option (C).
 - In the Control Rig area, make sure the Remove Constraint Reference option is activated (D).
 - In the Character Extensions area, activate the Process Animation option (E).
 - In the Character Extensions area, activate the Copy Missing Character Extensions option (F).



Load Character Animation Options settings

4 Click Open.

Mia's animation, Control rig, and character extension are loaded onto the Gremlin character.



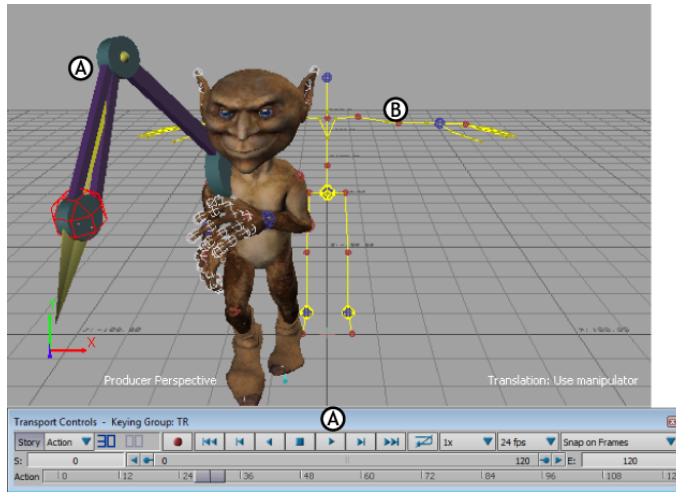
Mia's animation and Control rig loaded onto the Gremlin character

Because Mia's Servo arm is parented to her right shoulder FK effector, the Servo arm is attached in the same way to the Gremlin character.

Play the animation

Play the result animation.

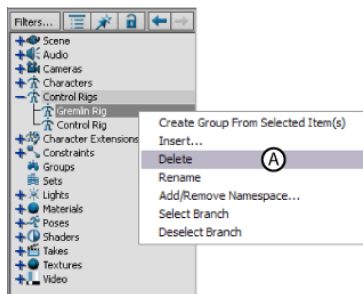
- 1 Click Play in the Transport Controls (A) to play the animation.



A. Gremlin using Mia's animation B. Gremlin's Control rig

Notice how both the Servo arm and the animation are transferred onto the Gremlin character (A). The Gremlin's original Control rig is left in the middle of the scene (B). This happens because you selected Replace Control Rig in the Load Character Animation Options dialog box. Gremlin's Control rig has been replaced by Mia's.

- 2 In the Scene browser, expand Control Rigs and right-click Gremlin Rig (Gremlin's original Control rig) and select Delete to clean up the scene.



Scene browser: A. Contextual menu Delete option

Summary

In this tutorial you retargeted animation from one characterized character to another and you transferred the Character Extension from the source character to the target character.

In the next tutorial, ([Editing Character Animation](#) on page 71), you edit animation on a layer from your original animation, then merge all layers.

Editing Character Animation

7

This tutorial guides you through the procedures necessary to modify animation by creating layers of animation. You modify the animation plotted to the character's Control rig on two separate layers, then combine the original animation and your modified animation.

This tutorial shows you how to:

- [Prepare the scene](#) on page 71
- [Modify the Character Extension animation](#) on page 73
- [Modify the head animation](#) on page 78
- [Plot the animation](#) on page 82
- [Play the resulting take](#) on page 83

The following asset is required for this tutorial:

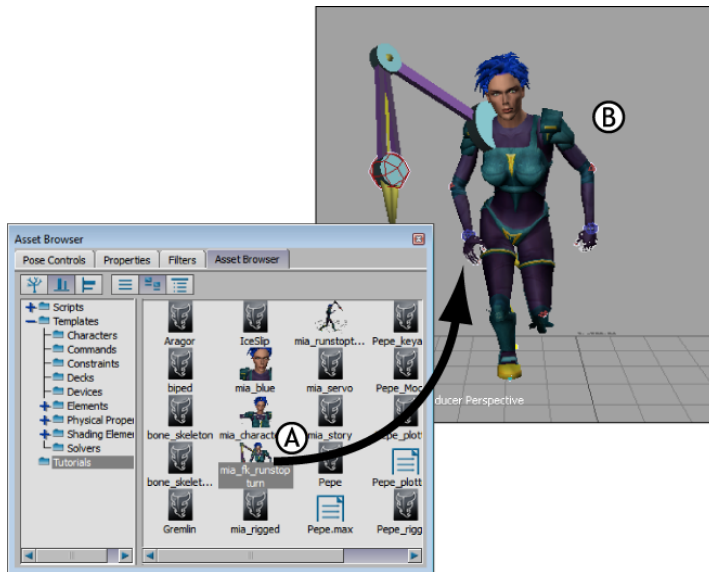
- *mia_runstopturn.fbx*

NOTE The tutorial assets can be found in the *Tutorials* folder in the Asset Browser and in the *Tutorials* folder in the MotionBuilder directory on your system.

Prepare the scene

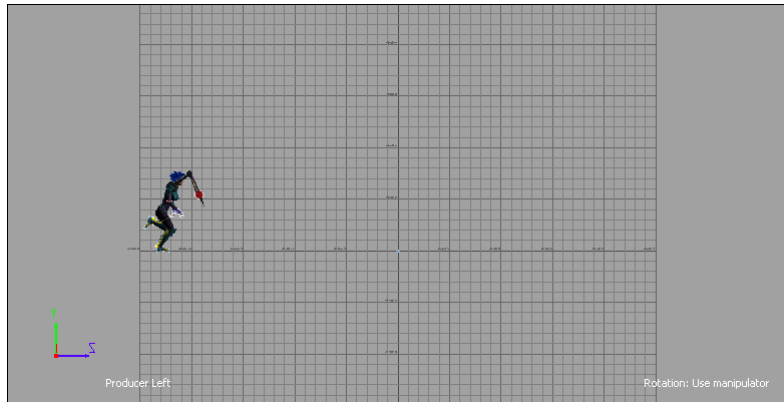
In the following procedure, you prepare the MotionBuilder scene and open the files needed to start this tutorial.

- 1 From the menu bar, select File > New, then select Layout > Editing (or press *Ctrl-Shift-3*).
MotionBuilder displays a new 3D scene using the Editing layout. This layout displays all the windows you need for your work in this tutorial.
- 2 Click the *Tutorials* folder in the Asset browser.
- 3 Drag the *mia_runstopturn* asset (*mia_runstopturn.fbx* file) into the Viewer window (A), then select FBX Open > run_stop_turn180 from the contextual menu that appears.
A model named Mia appears in the Viewer (B).



A. *mia_runstopturn* asset B. Mia character loaded in scene

- 4 Click in the Viewer window, then press *Ctrl-R* twice to switch to Producer Left camera. Zoom out to view the entire grid.



Producer Left camera view

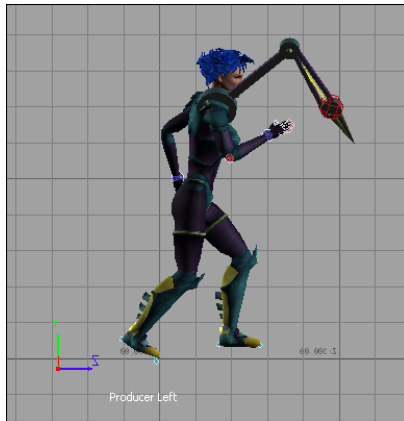
Modify the Character Extension animation

In the following procedure, you create an animation layer and modify the animation of the Servo arm Character Extension.

- 1 Play the entire take (*Ctrl-Spacebar*) to view all the motion, paying special attention to the Servo arm.

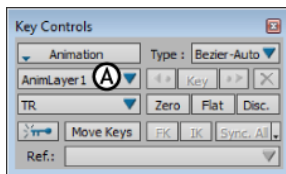
Right now, the Servo arm bounces along in front of Mia, pointing towards the red wire-frame effector. Although this effector is parented to Mia's shoulder, the effector moves enough to cause the Servo arm to jump around while Mia runs.

- 2 Press *Ctrl-Home* to go back to the beginning of the take, then play it again to frame 50.



Mia at frame 50

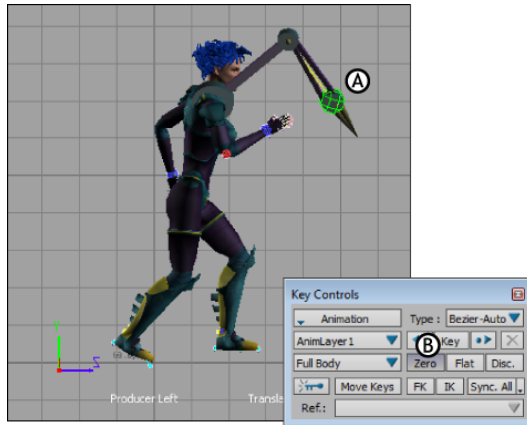
- 3 In the Key Controls, select AnimLayer1 from the Layer menu (A).



Key Controls A. AnimLayer1 selected

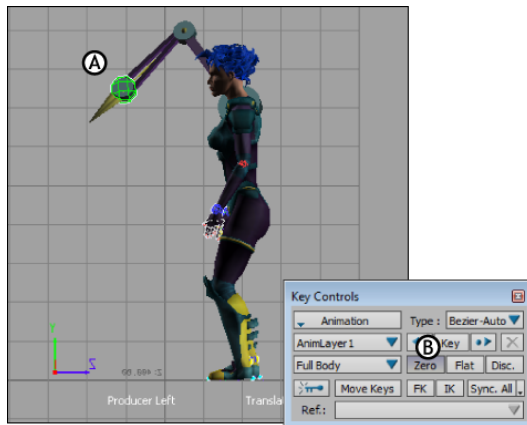
This lets you set keyframes on an animation layer while preserving the original animation on the BaseAnimation layer.

- 4 Select the Mia:ServoControl effector (A) then click Zero in the Key Controls to set a Zero keyframe (B) at frame 50.



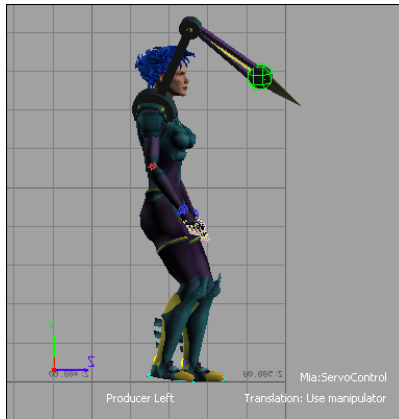
Frame 50 A. Mia:ServoControl effector B. Zero keyframe button

- 5 With the Mia:ServoControl effector still selected, go to frame 150 and set a Zero keyframe.



Frame 150 A. Mia:ServoControl is selected B. Zero keyframe button

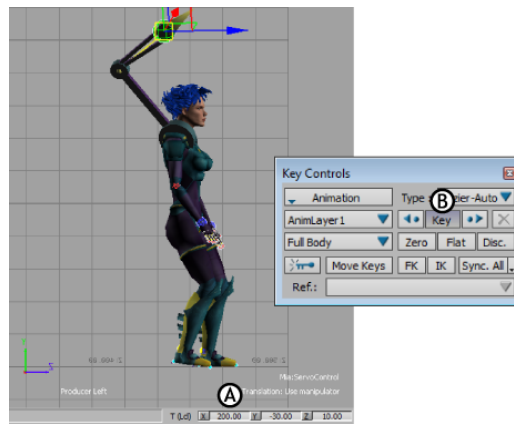
- 6 Go to frame 80.
At this frame, you are going to start modifying the animation so that Mia raises her Servo arm.



Frame 80

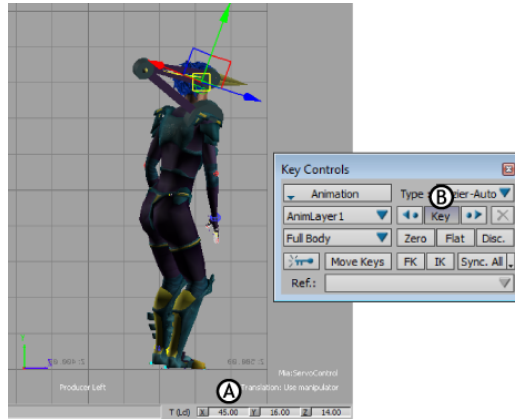
7 Do the following:

- Click in the Viewer window and press *T* to activate Translation mode.
- At the bottom of the Viewer window, set the Translation XYZ values to 200, -30, 10 (A).
- Set a keyframe (B).



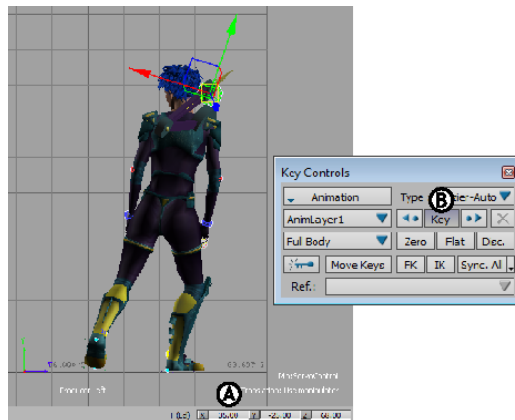
Frame 80 A. Set the Translation values B. Set a keyframe

- 8 Go to frame 120 and do the following:
 - Set the Mia:ServoControl effector Translation XYZ values to 45, 16, 14 (A).
 - Set a keyframe (B).



Frame 120 A. Set the Translation XYZ values B. Set a keyframe

- 9 Go to frame 130 and do the following:
 - Set the Translation XYZ values to 35, -25, 68 (A).
 - Set a keyframe (B).



Frame 130 A. Set the Translation XYZ values B. Set a keyframe

10 Play the animation.

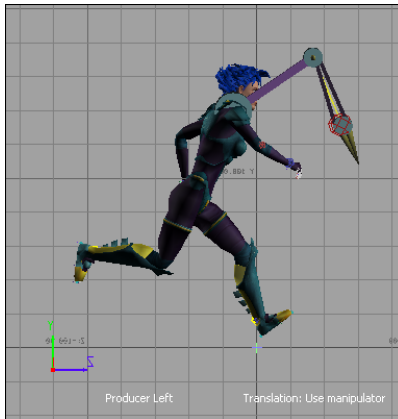
Now, Mia's Servo arm raises up as she slows and turns.

Modify the head animation

In the following procedure, you use another layer to improve the animation by making Mia turn her head as she runs, when she stops, and again just before she turns around.

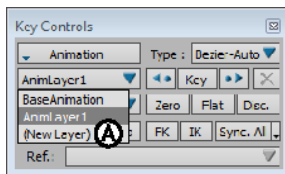
1 Go to frame 30.

At this frame, Mia's head is pointed straight ahead in the direction she is running. You need to modify the motion so that Mia turns her head.



Mia at frame 30

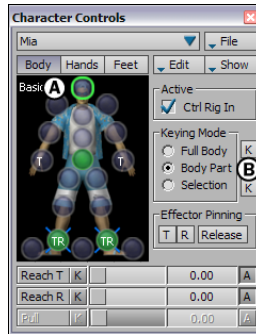
2 In the Key Controls window, select New Layer from the Layer menu (A) to create a layer to modify the head animation.



Key Controls window A. Layer menu > New Layer option

A layer called “AnimLayer2” is added.

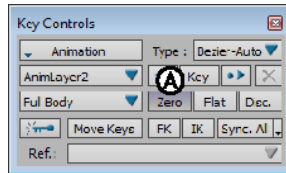
- 3 In the Character Controls window, select the head effector (A), and switch to Body Part keying mode (B).



Character Controls window A. Select the Head effector. B. Switch to Body Part keying mode.

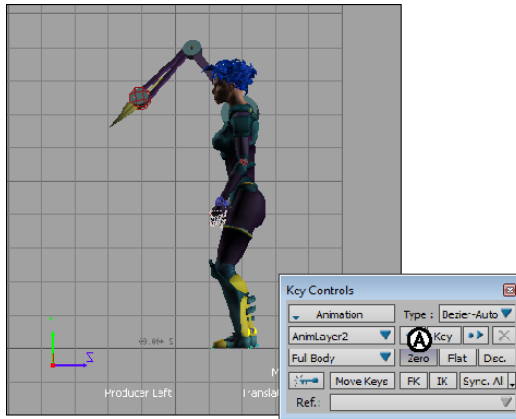
You can use Body Part keying mode as you create animation on this layer, since you only need to set keyframes on the head, not the entire body.

- 4 In the Key Controls, click Zero to set a zero keyframe (A).



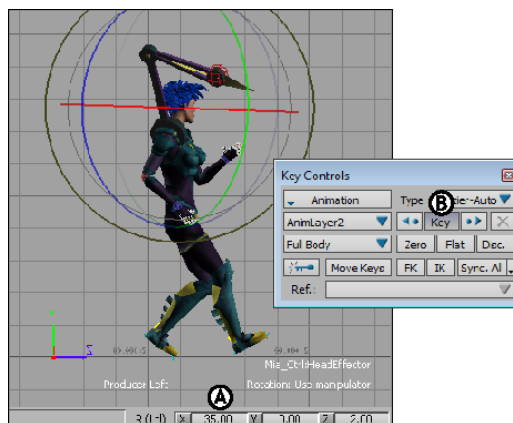
Key Controls A. Zero keyframe button

- 5 Go to frame 150 and set another Zero keyframe.



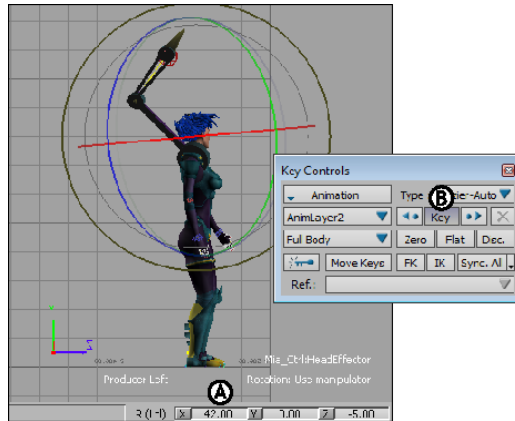
Frame 150 A. Set a zero keyframe.

- 6 Go to frame 60 and do the following:
 - With the Head effector still selected, activate Rotation mode (click in the Viewer window and press R).
 - Change the Rotation XYZ properties at the bottom of the Viewer window to 35, 0, 2 (A).
 - Set a keyframe (B).



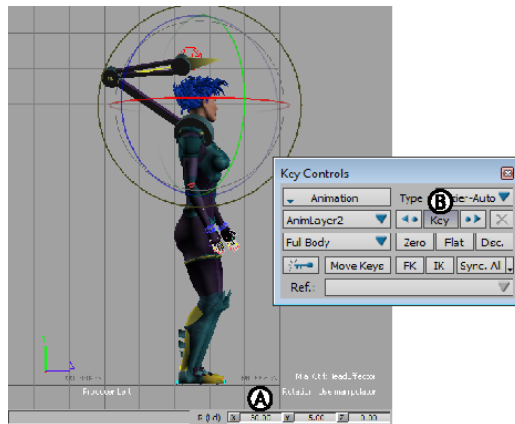
Frame 60 A. Set the XYZ Rotation properties. B. Set a keyframe.

- 7 Go to frame 90 and do the following:
 - Set the Rotation XYZ values to 42, 0, -5 (A).
 - Set a keyframe (B).



Frame 90 A. Adjust the Rotation values. B. Set a keyframe.

- 8 Go to frame 105 and do the following:
 - Set the Rotation XYZ values to 30, -5, 0 (A).
 - Set a keyframe (B).



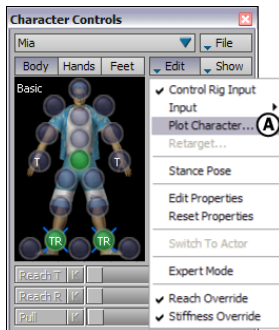
Frame 105 A. Adjust the Rotation values. B. Set a keyframe.

- 9 Deselect the Head effector, and play your animation.
Mia's head turns as she runs.

Plot the animation

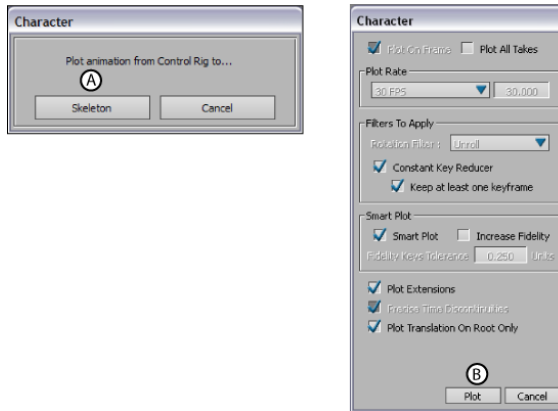
In the following procedure, you plot your animation data to combine the original data with your modifications. Plotting merges all of the animation to the BaseAnimation layer.

- 1 From the Edit menu in the Character Controls window, select Plot Character (A). You can also use the Plot Character button in the Character settings.



Character Controls window A. Select Plot Character from the Edit menu.

- 2 In the first Character dialog box that appears, click Skeleton (A), then make sure the same options are selected as those in the second Character dialog box that appears and click Plot (B).



Character plotting dialog boxes A. Click Skeleton. B. Click Plot.

All the animation data is transferred from the Control rig to the character's skeleton on the BaseAnimation layer of the current take.

- 3 To see the plotted keyframes, select the BaseAnimation layer and Mia's skeleton. Numerous keyframes display on the timeline in the Transport Controls.

You can also see your plotted data in the FCurves window as a series of function curves with numerous keyframes.

Play the resulting take

Play the take and observe your animation.

The animation of the Servo arm rising up and the head turning are merged with the original animation of Mia running and turning around.

Summary

In this tutorial, you modified original animation by setting keyframes on two layers, then merged the animation in one take. In the next tutorial, [Creating a Loop](#) on page 85, you learn how to create a walk cycle using the Story window.

Creating a Loop

8

This tutorial guides you through the procedures necessary to animate a character and create a walk cycle with the Story window.

This tutorial shows you how to:

- [Prepare the scene](#) on page 85
- [Create a Character track](#) on page 86
- [Create poses](#) on page 91
- [Match clips](#) on page 93
- [Process the clips](#) on page 96
- [Test the walk cycle](#) on page 98

The following assets are required for this tutorial:

- *mia_servo.fbx*
- *walkaround.fbx*

NOTE The tutorial assets can be found in the *Tutorials* folder in the Asset Browser and in the *Tutorials* folder in the MotionBuilder directory on your system.

Prepare the scene

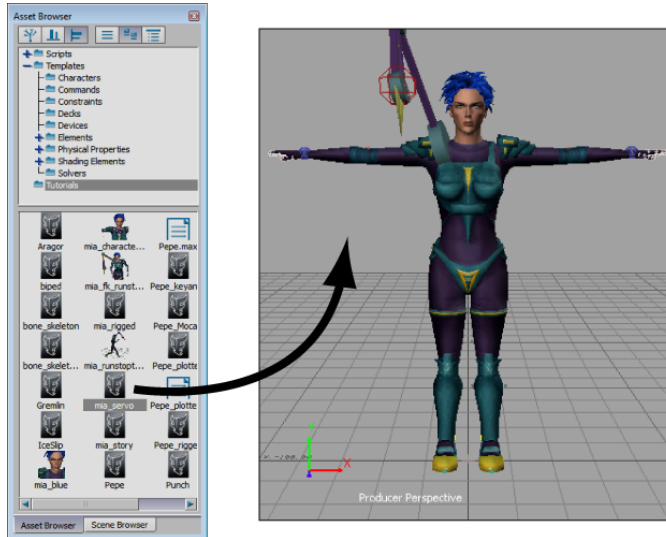
In the following procedure, you prepare the MotionBuilder scene and open the files needed to start this tutorial.

- 1 From the menu bar, select File > New, then select Layout > Story (or press *Ctrl-Shift-5*).

MotionBuilder displays a new 3D scene using the Story layout. This layout displays all the windows you need for your work in this tutorial.

- 2 Click the *Tutorials* folder in the Asset browser.
- 3 Drag the *mia_servo* asset (*mia_servo.fbx* file) into the Viewer window then select **FBX Open > No Animation**.

The *mia_servo* asset appears in the Viewer window, in the T-stance.

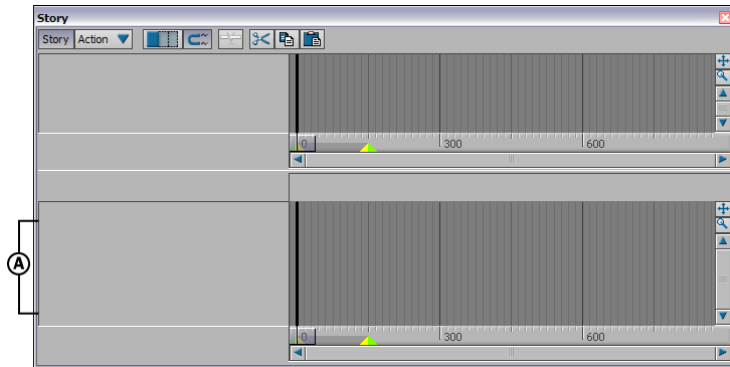


Mia with servo shown in the T-stance

Create a Character track

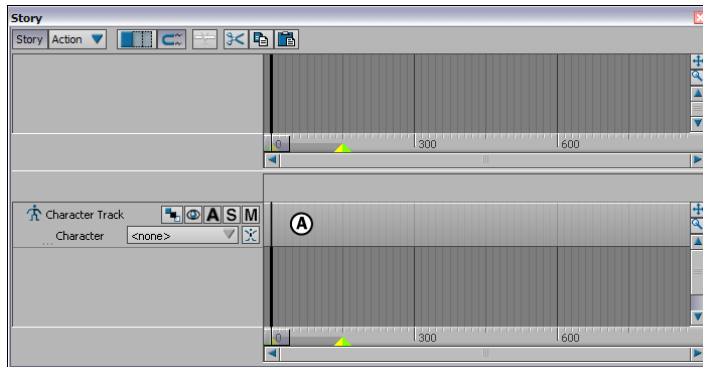
In the following procedure, you create a Character track in the Story window, define the character affected by the track, and add some animation.

- 1 In the Story window, right-click in the Action Track list (A) and select **Insert > Character Animation Track** from the contextual menu.



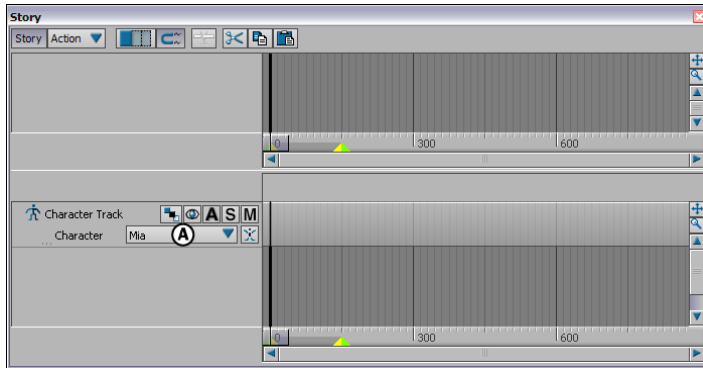
Story window A. Action Track list

A Character Animation track is added (A).



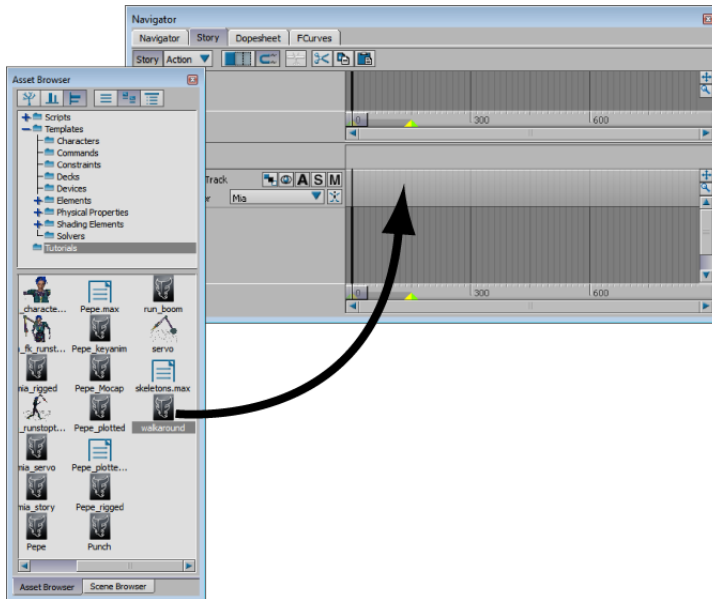
Story window A. Character track

- 2 Select Mia in the track's Character menu (A).



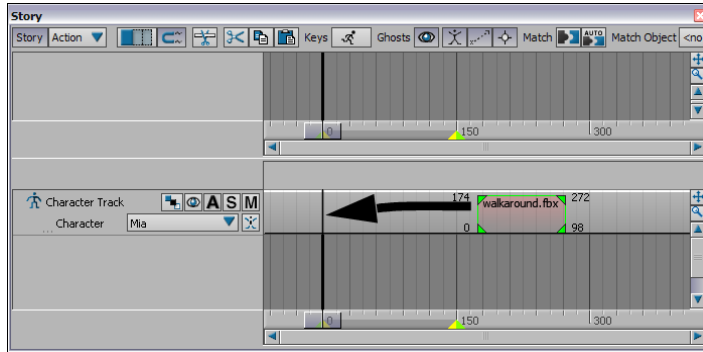
Story window A. Select Mia in the Character menu.

- 3 Drag *walkaround.fbx* from the Asset browser to the Character track.



Drag the *walkaround.fbx* file into the Character track.

- 4 Drag the clip so that it begins at frame 0. The clip should end at frame 98.



Drag the clip to frame 0 on the Character track.

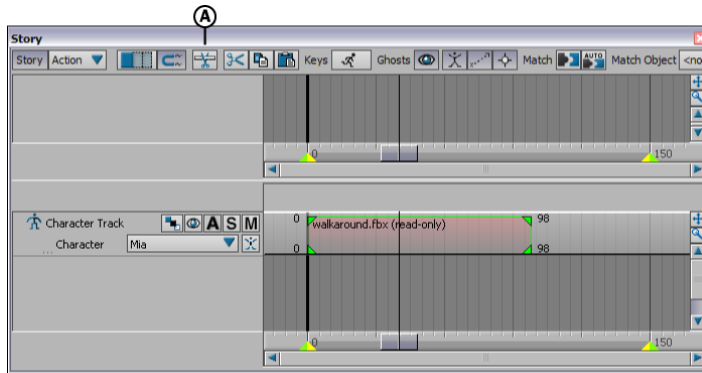
NOTE You can Ctrl-drag to zoom and Shift-drag to pan in the Character track.

- 5 Play the animation (Ctrl-Spacebar).
At frame 0, the character's right foot is in front and the left foot is in back. At frame 98, Mia is turning. If you were to loop the animation at this point, there would be a jump in the walk cycle.
- 6 Go to frame 40. At this frame, Mia's right foot is flat on the ground and her left foot is slightly lifted.
Change your camera view so you can see Mia from the front.



Mia at frame 40.

- 7 With the clip still selected, click the Razor button (A).



Story window A. Razor button

The clip is sliced in two at frame 40.



The original clip is sliced in two.

- 8 Go to frame 75. At this frame, Mia is in almost the same pose as she was at frame 40.



Mia at frame 75.

- 9 Select the second clip if it is not already selected, then click the Razor button.

The second clip is sliced at frame 75, and you now have three clips.



Three clips in the Character track

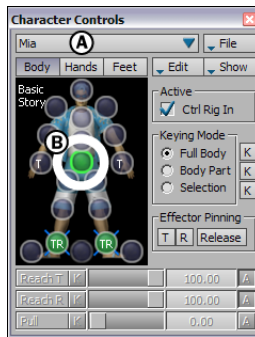
- 10 Ctrl-click the first clip, so that the first and third clips are selected, then press Delete, as you only need the middle clip.
- 11 Drag the remaining clip to start at frame 0.



The clip starts at frame 0 and ends at frame 35.

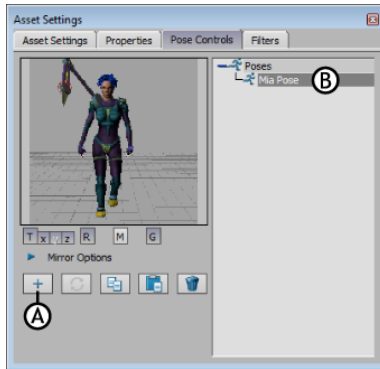
Create poses

- 1 Go to frame 0 (Ctrl-Home).
- 2 In the Character Controls window, make sure Mia is selected in the Current Character menu (A), then select the Hips effector (B).



Character Controls window A. Select Mia from the Current Character menu. B. Select the Hips effector.

- 3 In the Asset Settings window, click the Pose Controls tab.
- 4 In the Pose Controls, click Create (A), then expand the Poses folder to see the pose you created, called “Mia Pose” by default (B).



Pose Controls window A. Create button B. Mia Pose

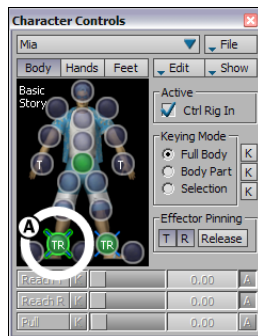
5 Go to frame 35 and do the following:

- In the Story window, activate the Animate option (A) in the Character track. You can only paste poses on a track when the Animate option is active.



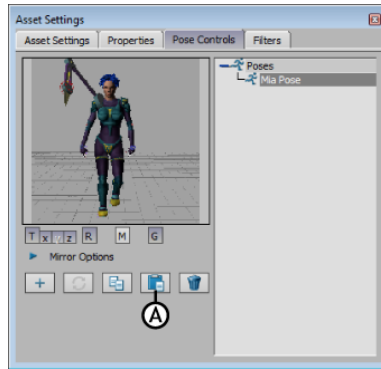
Character track A. Activate the Animate option.

- In the Character Controls window, right-click the Right Ankle effector (A) and select RightAnkleEffector from the menu that appears.



Character Controls window A. Right Ankle effector

- In the Pose Controls window, click Paste (A).



Pose Controls window A. Paste button

A one-frame clip appears on the Character track at frame 35.

This clip contains the data of the pose you pasted. Now Mia starts and stops walking with exactly the same pose.



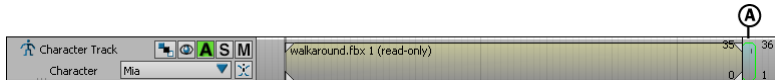
Character track A. New clip at frame 35.

- 6 Jog (J-drag) or use Ctrl Left Arrow and Ctrl Right Arrow to step frame-by-frame through the animation very slowly. Though the animation begins and ends with the same pose, there is a slight jump between the clips. In the following procedure, you remove the jump.

Match clips

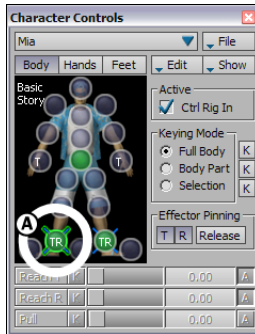
In the following procedure, you match and blend the two clips to remove the jump in the animation.

- 1 Click in an empty space below the track, and press A to zoom in on the clips.
- 2 Select the second clip.



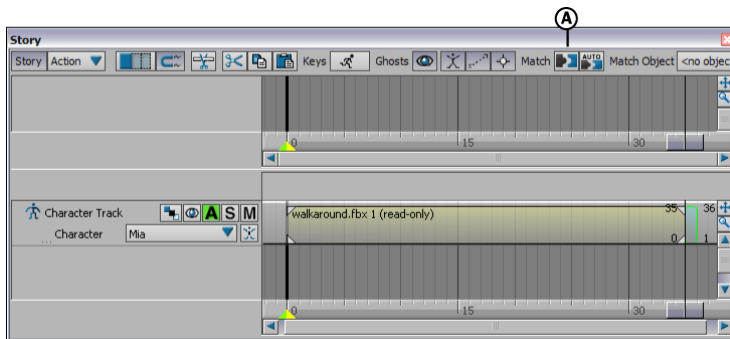
Character track A. Select the second clip.

- 3 Make sure the Right Ankle effector is still selected in the Character Controls window (A).



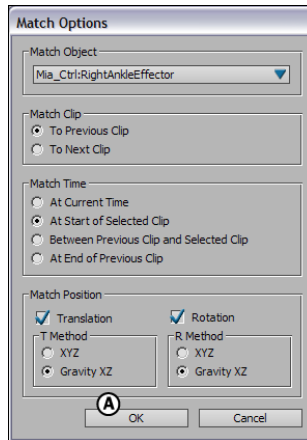
Character Controls window A. Right Ankle effector

- 4 In the Story window, click the Match Options button (A).



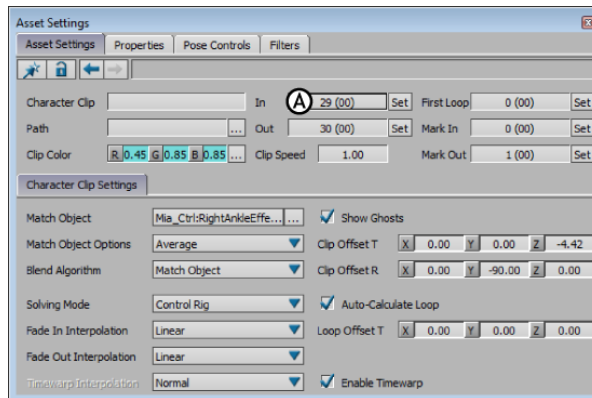
Story window A. Match Options button

- 5 In the Match Options dialog box that appears, click OK (A).



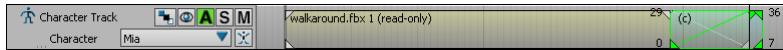
Match Options dialog box A. OK button

- 6 In the Asset Settings window where the Pose Controls display, click the Asset Settings tab, then double-click the second clip in the Story window to display its settings.
- 7 In the Asset Settings, set a value of 29 in the In field (A).



Asset Settings window A. Set the In point to frame 29.

- 8 Enter value of 36 in the Out field in the Asset Settings window.
- 9 The second clip now starts at frame 29, and cross-blends with the first clip to end at frame 36. This blend creates a slightly smoother transition between the clips.



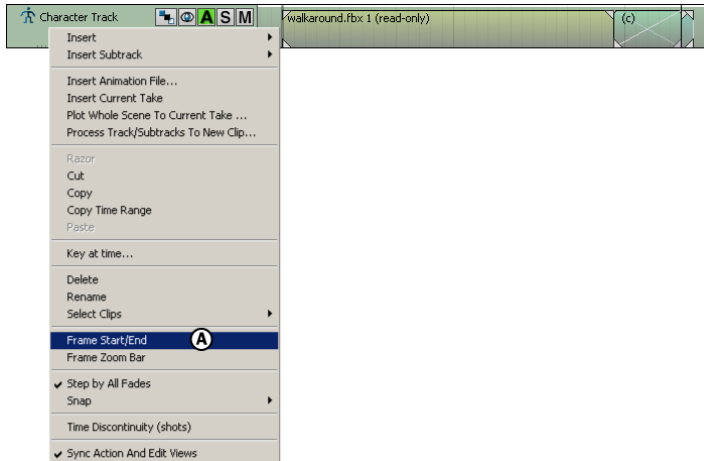
The second clip starts at frame 29 and ends at frame 36.

- 10 Play the animation. Mia walks, starting and ending with the same pose.

Process the clips

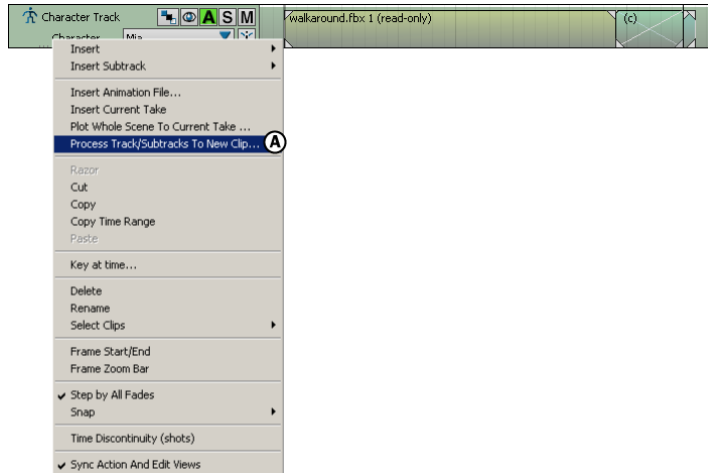
In the following procedure, you process the two clips to save them as a single result clip. Later, you use this new clip to animate a different character.

- 1 Right-click on the Character track near the Character name and select Frame Start/End from the contextual menu (A). The time range is resized to fit the length of the clips.



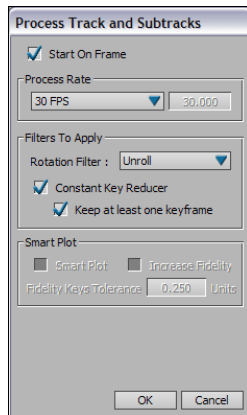
Character track contextual menu A. Select the Frame Start/End option.

- 2 Right-click the Character track again and select Process Track/Subtracks To New Clip from the contextual menu (A).



Character track Contextual menu A. Process Tracks/Subtracks To New Clip option

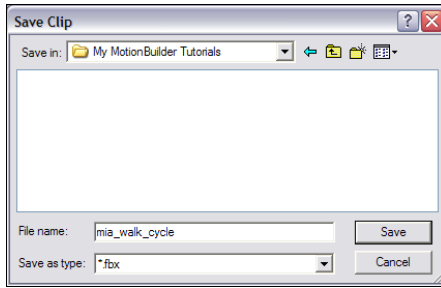
- 3 Click OK in the dialog box that appears.



Process Track and Subtracks dialog box

- 4 In the Save Clip dialog box that appears, save your new clip as *mia_walk_cycle.fbx*.

You need the saved clip to complete this tutorial.



Save Clip dialog box

In the Story window, a second Character track appears containing the new *mia_walk_cycle.fbx* clip (A).

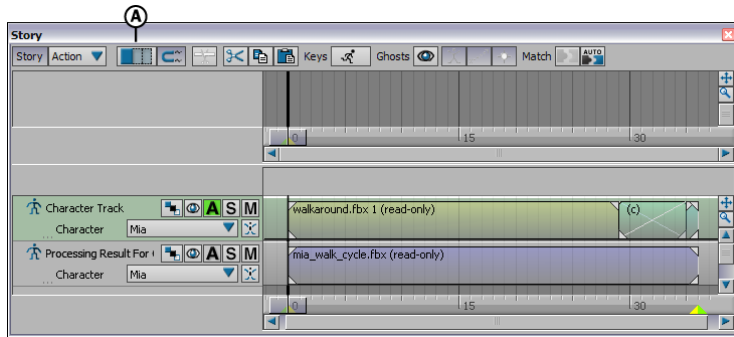


Story window A. New *mia_walk_cycle.fbx* clip

Test the walk cycle

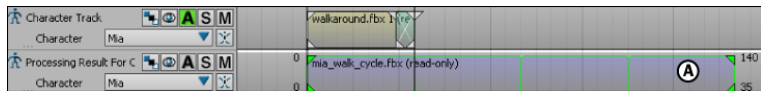
In the following procedure, you test the new walk cycle clip to see if it loops smoothly.

- 1 Make sure that the Loop/Scale option is set to Loop, as shown in (A).



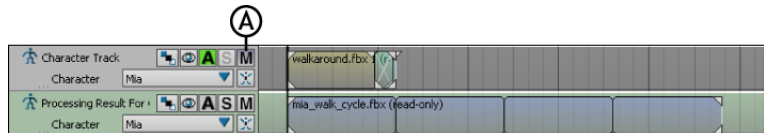
Story window A. Loop/Scale option is set to Loop.

- 2 Zoom out on the Character tracks, then stretch the end of the mia_walk_cycle clip to frame 140 (A). The clip loops four times.



Character tracks A. Stretch the clip to frame 140.

- 3 Right-click any Character track and select Frame Start/End from the contextual menu.a
- 4 Click the first Character track's Mute button (A) to disable the track.



Character tracks A. Click Mute.

- 5 Play the animation. Mia walks smoothly for 140 frames.

Summary

In this tutorial, you took a short clip of animation and turned it into a looping walk cycle. In the next tutorial, [Manipulating Clips](#) on page 101, you learn how to edit character animation by modifying clips.

Manipulating Clips

9

This tutorial guides you through the procedures necessary to modify character animation by manipulating clips.

This tutorial shows you how to:

- [Prepare the scene](#) on page 101
- [Create a turn](#) on page 103
- [Blend two clips](#) on page 107
- [Add a clip](#) on page 108
- [Match clips](#) on page 110

The following assets are required for this tutorial:

- *mia_story.fbx*
- *run_boom.fbx*

NOTE The tutorial assets can be found in the *Tutorials* folder in the Asset Browser and in the *Tutorials* folder in the *MotionBuilder* directory on your system.

Prepare the scene

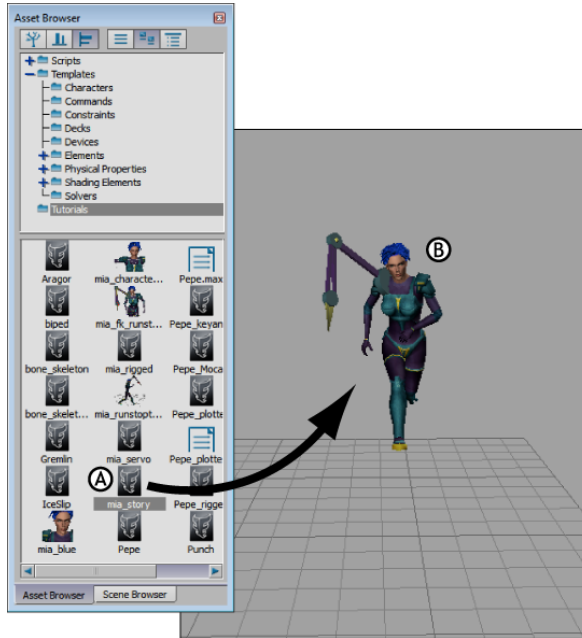
In the following procedure, you prepare the MotionBuilder scene and open the files needed to start this tutorial.

- 1 From the menu bar, select File > New, then select Layout > Story (or press *Ctrl-Shift-5*).

MotionBuilder displays a new scene using the Story layout. This layout displays all the windows you need for your work in this tutorial.

- 2 Click the *Tutorials* folder in the Asset browser.
- 3 Drag the *mia_story* asset (*mia_story.fbx* file) into the Viewer window (A), then select FBX Open > All Takes.

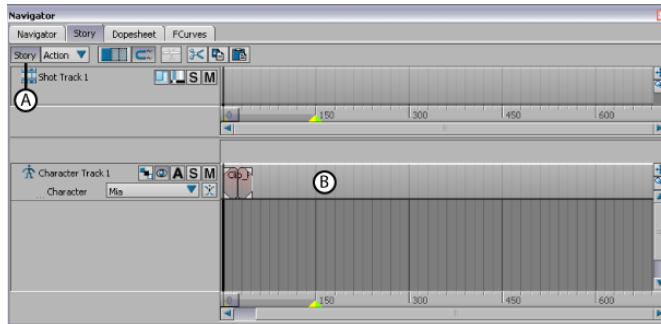
A model named Mia appears in the Viewer window (B).



Loading file A. Drag *mia_story* from Asset browser B. Mia model loads

In the Story window, there is a track with a clip called *Clip_Run_Loop*. Mia is selected in the track's Character menu.

- 4 In the Story window, activate the Story button (A) if it is not already turned on, then click on the Character track (B) and press A to frame the clip.



Story window A. Story button selected B. Character track showing one clip

- 5 Play the animation (*Ctrl-Spacebar*). You may need to zoom out in the Viewer window to see all of the animation.

Create a turn

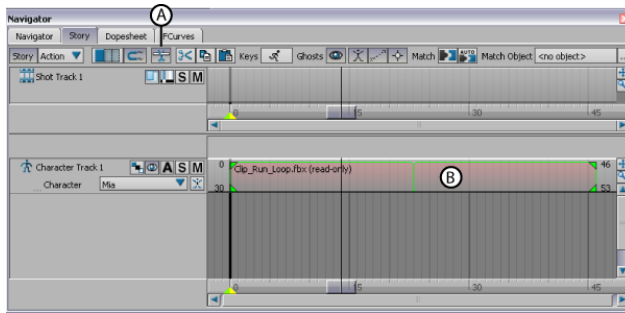
In the following procedure, you slice a clip in two, then rotate a ghost clip vector to make Mia turn as she runs.

- 1 Go to frame 14.
At this frame, Mia's left foot is flat on the ground as shown in the following figure.



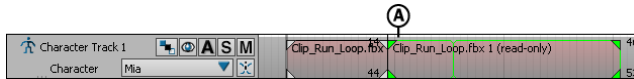
Mia at frame 14.

- 2 Select the clip (B) and click the Razor button (A) as shown in the following figure.



Story window A. Razor button B. Selected clip

The clip is sliced into two clips at frame 14 (A).



Character track A. Clip sliced in two at the current time

- 3 Switch to the X-Ray display mode in the Viewer window, then make sure the Ghost option in the Character track (A) is activated as shown in the following figure.

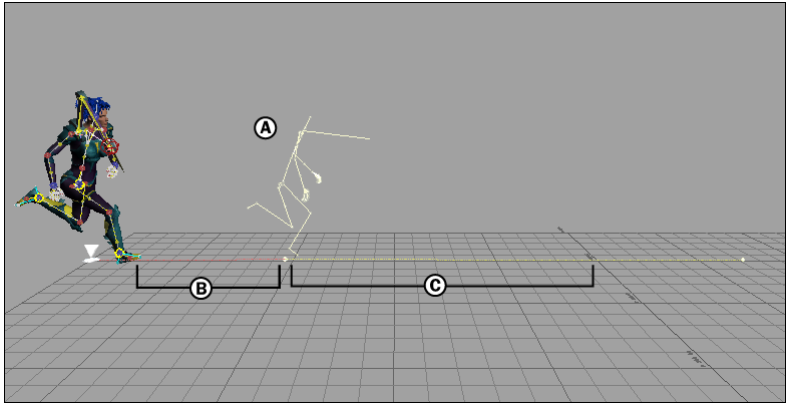


Character track A. Active Ghost option

- 4 Go to frame 0 (*Ctrl-Home*).

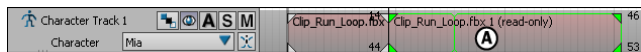
When the Ghost option is active, the ghosts display in the Viewer window as shown in the following figure.

The clip vector ghosts represent the start and end of each clip. For each clip, there is one clip vector ghost that you can select and manipulate (B and C).



Mia model at frame 0 A. Model ghost B. First clip's ghost clip vector C. Second clip's ghost clip vector

- 5 In the Story window, select the second clip if it is not still selected (A).



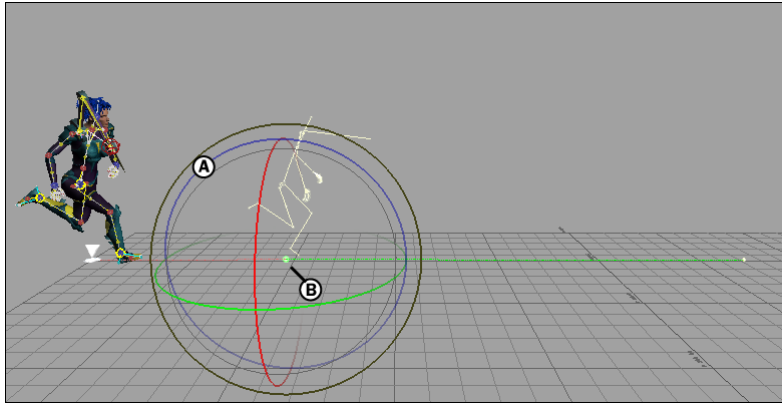
Character track A. The second clip is selected

The ghost clip vector of the selected clip is also selected in the Viewer window.

- 6 Click in the Viewer window and press the keyboard shortcut *R*.

Rotation rings appear at one end of the selected clip vector ghost, as shown in the following figure.

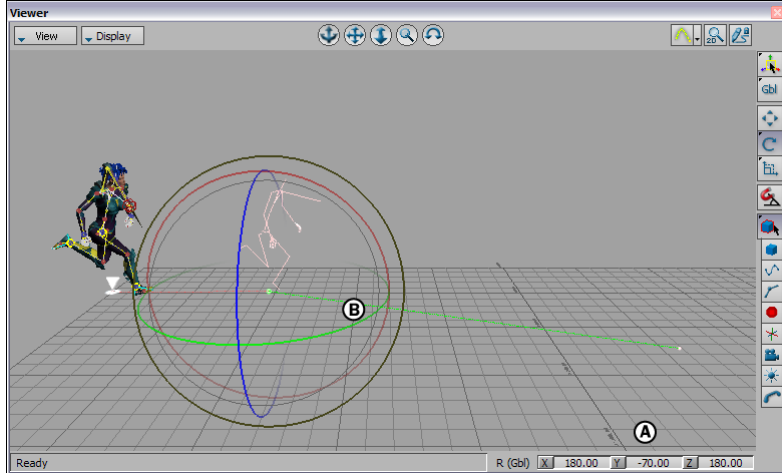
If the rings do not appear at the same point of the clip vector, double-click the clip vector's In point as shown to select it.



A. Rotation rings B. Second ghost clip vector's In point

- 7 In the Viewer window, enter a value of -70 in the Rotation Y-axis field as shown in the following figure (A).
The clip vector turns to Mia's right (B).

TIP You can manually rotate the clip vector by dragging the green rotation ring.



A. Rotation Y-axis field B. Ghost clip vector is rotated

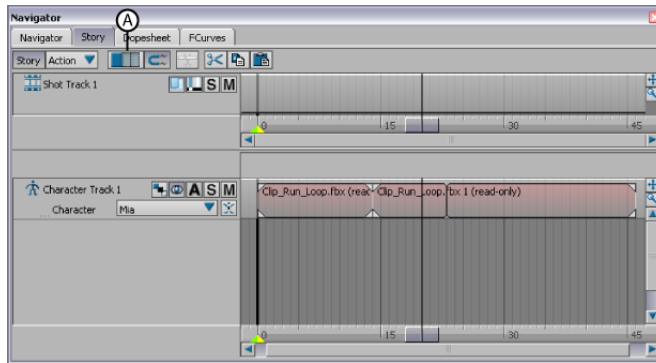
- 8 J-drag in the Viewer window to jog through frames 10 to 20 slowly.

Mia turns as she runs, but her foot jumps slightly at frame 14. You need to blend the clips to remove the jump.

Blend two clips

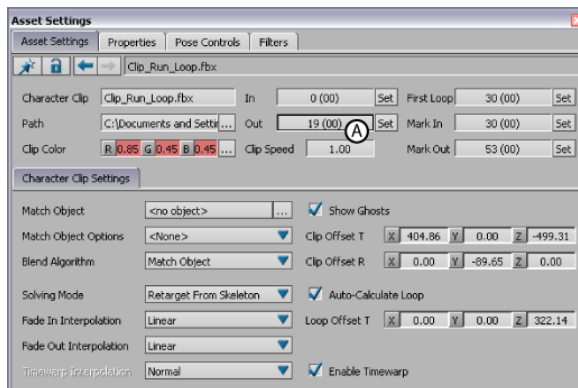
In the following procedure, you blend two clips to remove a jump in animation that occurs when Mia turns.

- 1 Make sure the Loop option (A) is active in the Story window.



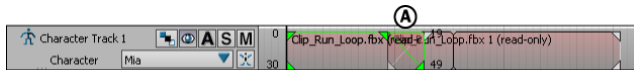
Story window A. Loop/Scale option set to Loop

- 2 Double-click the first clip to display its settings in the Asset Settings window located at the bottom right of the user interface.
- 3 In the Asset Settings, set a value of 19 in the Out field (A).



Asset Settings window A. Out field set to 19

The first clip overlaps the second clip, creating a cross-blend as shown in the following figure (A).



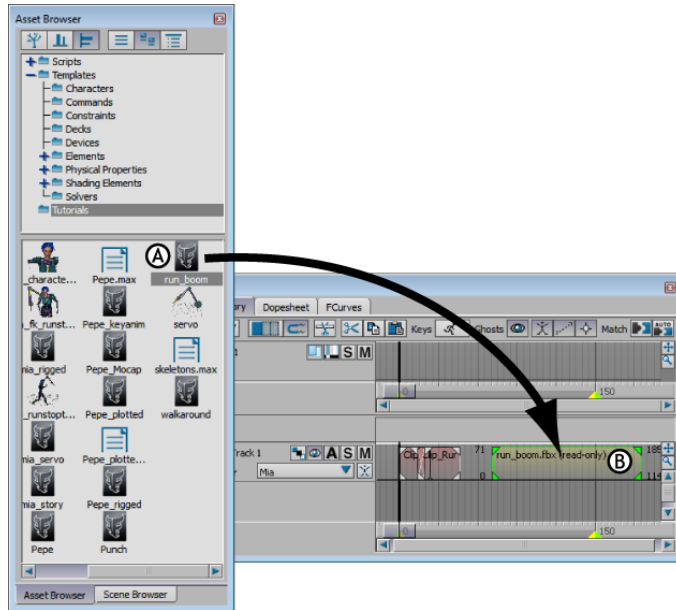
Character track A. Clip cross-blend

- 4 Play the animation.
There is no longer a jump at frame 14.

Add a clip

In the following procedure, you add a clip to the Character track in the Story window.

- 1 *Ctrl*-drag and Shift-drag in an empty space beneath the Character track to zoom out and make room next to the clips.
- 2 From the **Tutorials** folder in the Asset browser, drag the *run_boom.fbx* file onto an empty part of the Character track, to the right of the clips, as shown in the following figure.



run_boom.fbx clip added to Character track

- 3 Drag the clip so that it begins at frame 46.

NOTE It should rest against the end of the second clip, as shown in the following figure.



New clip begins at frame 46

- 4 Deselect the new clip, then right-click the Character track and select Frame Start/End from the contextual menu.
- 5 Play the animation (*Ctrl-Spacebar*).
Mia runs, turns, there is a jump in the animation, then Mia is thrown forward as if propelled by an explosion.

Match clips

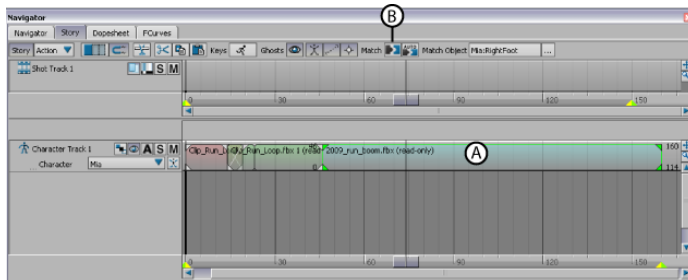
In the following procedure, you match the last clip to the previous clip to remove the jump in the animation.

- 1 Go to frame 0.
- 2 In the Viewer window, switch to the Schematic view and select the Mia:RightFoot node (A).
This node represents Mia's right foot toe use as the matching object.



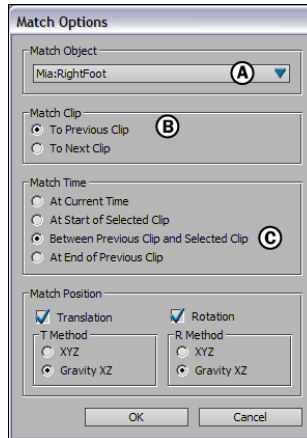
Schematic view A. Mia:RightFoot node selected

- 3 Switch back to the Producer Perspective camera view.
- 4 In the Story window, select the third clip (A), then click the Match Options button (B).



Story window A. Third clip selected B. Match Options button

- 5 In the Match Options dialog box that appears, if not selected, select the Mia:RightFoot in the Match Object menu (A), and select To Previous Clip (B) and Between Previous Clip and Selected Clip (C) as shown in the following figure.



Match Options dialog box A. Match Object B. Match Clip C. Match Time

The Translation and Rotation options are already selected.

- 6 Click OK.
The last clip vector moves to match the previous clip.
- 7 Deselect Mia:RightFoot (Shift-D).
- 8 Play the animation.
Mia runs, turns, then is thrown forward. The jump in the animation is gone.

Summary

In this tutorial, you sliced a clip of running animation in two and rotated one clip vector ghost to make the character turn while running. Then you added another clip with different animation and blended all three clips together in one seamless animation sequence.

In the next tutorial, ([Importing 3ds Max Files into MotionBuilder](#) on page 113), you export 3ds Max skeletons and a 3ds Max character into MotionBuilder, add a Control rig, characterize the skeletons and Character, and animate the

character in MotionBuilder, and then export a Character and animation back to 3ds Max.

Importing 3ds Max Files into MotionBuilder

10

This tutorial guides you through the procedures necessary to import a character created in 3ds Max to MotionBuilder, and then export your work back to 3ds Max as a fully-editable animated character.

NOTE The results of this tutorial are based using the latest version of 3ds Max, MotionBuilder, and 3ds Max FBX plug-ins software products available at the time the tutorial was written.

The tutorial covers the following three kinds of animatable skeletons that originate in 3ds Max and shows you how to bring these skeletons into MotionBuilder for animation using the FBX format.

- Conventional 3ds Max bone system skeletons
- Biped system skeletons
- Skinned characters with skeletons

This tutorial is comprised of three small tutorials that show the same process for different situations. You can single out the interoperability procedure that addresses your needs or complete the whole tutorial to get a well-rounded view of the interaction between MotionBuilder and 3ds Max.

This set of tutorials shows you how to:

- Export two kinds of 3ds Max skeletons and a 3ds Max character into MotionBuilder
- Add a Control rig and characterize the skeletons and character
- Animate the character and prepare it for import to 3ds Max

Following are the major topics in this tutorial:

- [Preparation for this tutorial](#) on page 114
- [3ds Max skeletons](#) on page 115
- [3ds Max Biped](#) on page 127
- [3ds Max Characters](#) on page 137
- [Animating a 3ds Max Character in MotionBuilder](#) on page 144
- [Exporting a Character and its animation back to 3ds Max](#) on page 156

NOTE Although the procedures in this tutorial use the MotionBuilder keyboard shortcuts, you can elect to use the 3ds Max keyboard shortcuts. To do so, from the MotionBuilder menu bar, choose Settings > Keyboard Configuration > 3ds Max. See the 3ds Max keyboard shortcuts topic in the Keyboard Shortcuts chapter of the MotionBuilder Help for a list of all the 3ds Max keyboard shortcuts.

Preparation for this tutorial

To complete this tutorial, you need the current versions of 3ds Max, MotionBuilder, and the 3ds Max FBX plug-ins installed on your system.

If you do not have the current version of 3ds Max, you can download a trial version of the software product from the Autodesk web site at:

<http://www.autodesk.com/3dsmax>.

As for the 3ds Max FBX Plug-ins, you can download the latest version from:

<http://www.autodesk.com/fbx>.

NOTE The default installation directory for the FBX Plug-ins is: *C:\Program Files\Autodesk\FBX\FbxPlugins*.

You can choose to either complete the entire tutorial or parts of the tutorial.

Following are the assets required for this tutorial:

- *skeletons.max*
- *bone_skeleton.FBX*
- *bone_skeleton_characterized.FBX*

- *biped.FBX*
- *Pepe.max*
- *Pepe.FBX*
- *Pepe_rigged.FBX*
- *IceSlip.fbx*
- *Pepe_Mocap.FBX*
- *Pepe_keyanim.FBX*
- *Pepe_plotted.FBX*
- *Pepe_plotted.max*

You can find the tutorial assets in the MotionBuilder Asset browser's *Tutorials* folder as well as in the *Tutorials* folder located in the MotionBuilder directory on your system.

3ds Max skeletons

The following tutorial shows you how to export a conventional 3ds Max bone system skeleton data to FBX format, import to MotionBuilder, and then characterize it so you can animate it.

Following is the asset required for this tutorial:

- *skeletons.max*

Following are the result assets for this tutorial:

- *bone_skeleton.FBX*
- *bone_skeleton_characterized.FBX*

NOTE You can find the tutorial assets in the MotionBuilder Asset browser's *Tutorials* folder as well as in the *Tutorials* folder located in the MotionBuilder directory on your system.

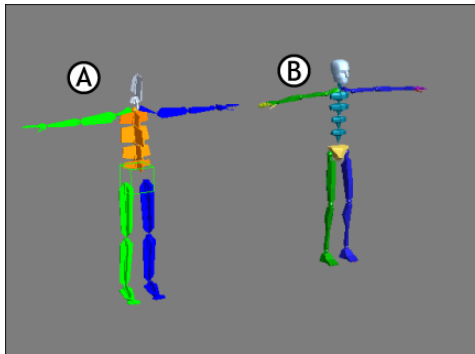
Export a 3ds Max skeleton

In the following procedure, you export conventional 3ds Max bone system skeleton data to FBX format so you can animate it in MotionBuilder.

To export a skeleton in FBX format:

- 1 Launch 3ds Max.
- 2 From the Application menu, select Open > Open.
- 3 In the Open File dialog box, navigate to the MotionBuilder root directory and in the *Tutorials* folder, open the *skeletons.max* scene file. If the File Load: Units Mismatch dialog appears, select Adopt the File's Unit Scale and click Ok.

The *skeletons.max* file opens, displaying two skeletons. The skeleton (A) to the left is created with the conventional 3ds Max bone system, and the skeleton (B) to the right is created with the 3ds Max Biped creation option.



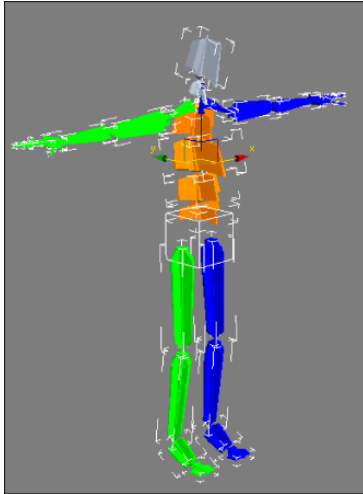
Two skeletons A. Conventional 3ds Max bone system skeleton B. 3ds Max Biped skeleton

The skeletons are positioned in a “T” pose, the stance used by animators for skinning.

NOTE Always place your characters in the “T” stance position before exporting to MotionBuilder. Also, make sure that the skeleton is oriented in the minus Y axis direction. (If you create skeletons with the 3ds Max biped system, they are automatically oriented this way.)

To learn how to export skeletons created with the 3ds Max biped creation option, see [Create and export a 3ds Max biped](#) on page 128.

- 4 Region-select all of skeleton A.



Biped oriented in minus Y direction

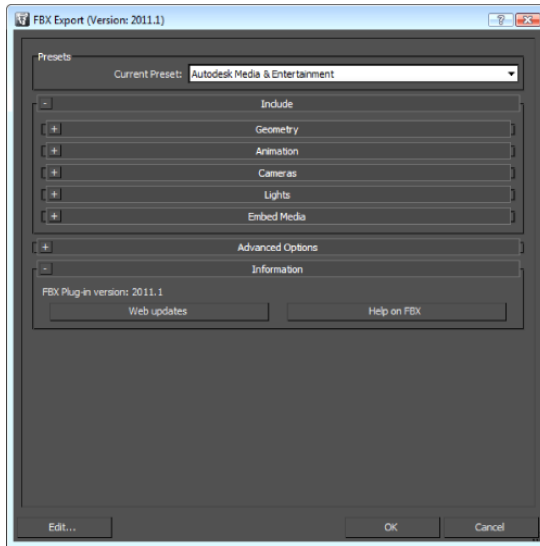
- 5 From the Application menu, select Export > Export Selected.

TIP You need to hold the mouse over Export without clicking for a moment so that Max displays the Export options menu from which you can choose Export Selected.

- 6 In the Select File To Export dialog box, navigate to the default MotionBuilder root directory and in the *Tutorials* folder, name your file `My_bone_skeleton`, save your file as *Autodesk (*FBX)* file type, and click Save.

NOTE If you do not specify a location, the file is automatically saved in the FBX file format to the 3ds Max Export folder.

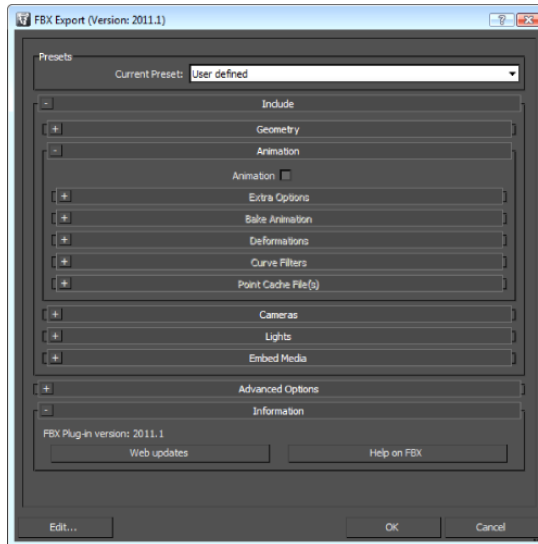
The FBX Exporter opens. This is where you specify how to convert the 3ds Max scene information.



FBX Export dialog box

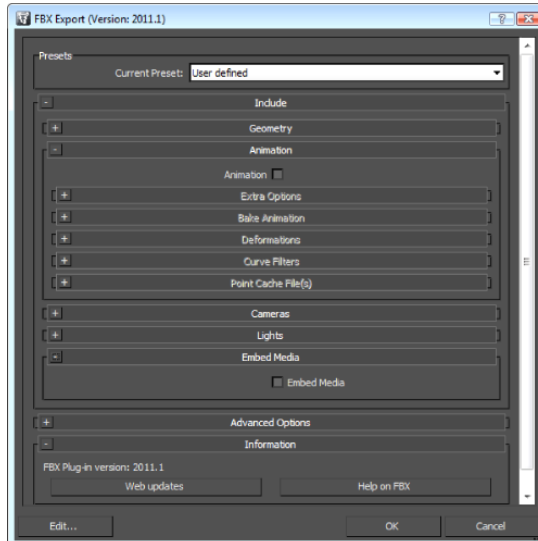
- 7 In the FBX Export dialog box > Include > Animation rollout, disable Animation.

Since you are only exporting the skeleton that has no animation, there is no need to export animation.



Animation option disabled in FBX Export

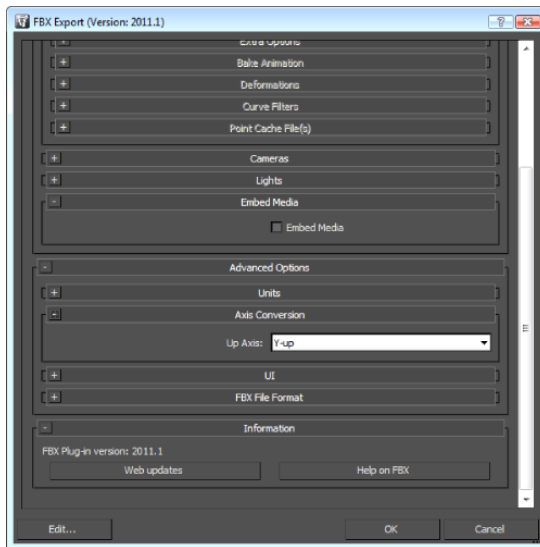
- 8 Expand the Embed Media rollout and make sure Embed Media is also disabled. If Embed Media is not disabled, disable it.



Embed Media option disabled in FBX Export

You only need the Embed Media option if you export a mesh with the character. That way, any texture maps associated with the character are saved with the FBX file. In most cases, you would export a mesh, properly skinned on a skeleton. But in this case, you are only exporting a skeleton, so you do not need this option.

- 9 Expand Advanced Options > Axis Conversion, and make sure Up Axis displays the Y-up option as shown in the following figure.



Up Axis selected in Axis Conversion rollout

This setting assigns the exported character a Y-up axis, the orientation used by objects in MotionBuilder. The Y-up setting is required since objects created in 3ds Max use a Z-up orientation.

- 10 Click OK to export skeleton A as an FBX file to your designated folder. Now that your skeleton is prepared, you can use this FBX file in the next tutorial to import 3ds Max skeletons into MotionBuilder. Once a skeleton is exported to the FBX file format, you can also bring it into other Autodesk software products, such as Autodesk Maya.

Import and characterize a 3ds Max skeleton in MotionBuilder

In the following procedure, you import the skeleton (*My_bone_skeleton.FBX*) you saved in the previous procedure ([Export a 3ds Max skeleton](#) on page 116) and then characterize the skeleton so you can animate it in MotionBuilder. Characterization assigns a Control rig to the skeleton bones and is the name MotionBuilder uses for the process of rigging a skeleton.

NOTE If you did not perform the previous procedure, you can use the *bone_skeleton.FBX* file in the *Tutorials* folder located in the MotionBuilder directory on your system.

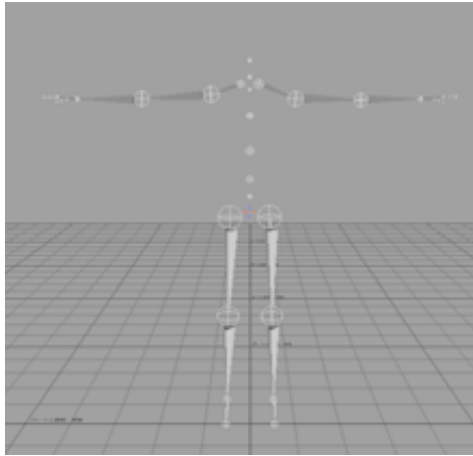
To import a conventional 3ds Max skeleton into MotionBuilder:

- 1 Launch the MotionBuilder software.
- 2 From the MotionBuilder main menu, select File > Open.
- 3 In the Open File window, navigate to the directory where you saved the *My_bone_skeleton.FBX* file and click Open.

NOTE If you did not perform the previous procedure, open the *bone_skeleton.FBX* file located in the MotionBuilder root directory under the *Tutorials* folder.

- 4 In the FBX Plug-in Import Options dialog, leave the default settings unchanged and click Open.

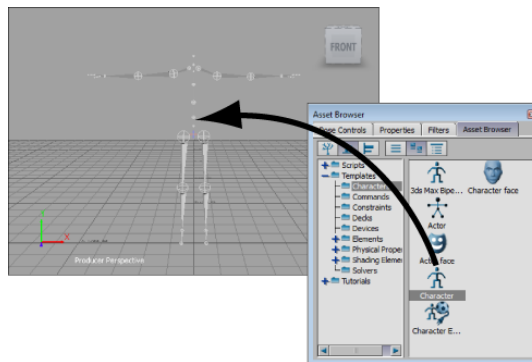
The 3ds Max bone system skeleton loads into MotionBuilder.



3ds Max skeleton displayed in the Viewer window

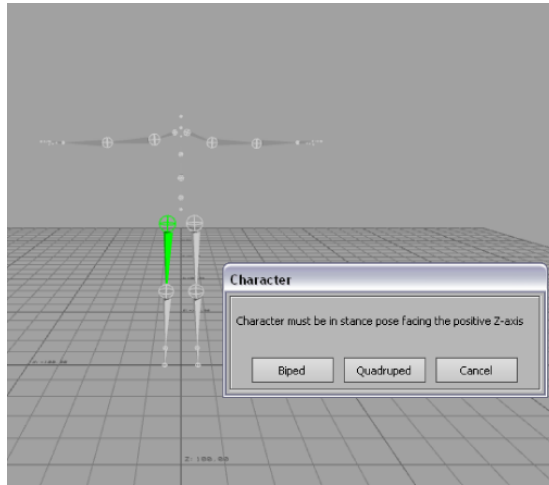
To characterize your skeleton:

- 1 In the MotionBuilder Asset browser, expand Templates > Characters and drag a Character asset onto one of the skeleton bones.



The bone lights up when the asset makes contact with it.

- 2 Click Characterize in the menu that appears.
The following dialog box reminds you that the character must be in a “T” stance pose and face in the positive Z axis (the equivalent of the negative Y axis that you converted when exporting the .max file into the FBX file format).

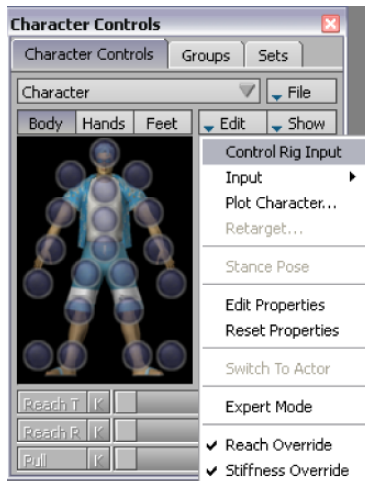


Character must be in T stance for characterization

- 3 Click Biped in the Character dialog box that appears to indicate the type of rigging to apply to the character.

The skeleton is now characterized, which means that it is ready to accept a Control rig you can animate.

- 4 From the Character Controls window, select Edit > Control Rig Input.



Character Controls Edit menu

You must use the Control Rig Input setting if you intend to keyframe your character.

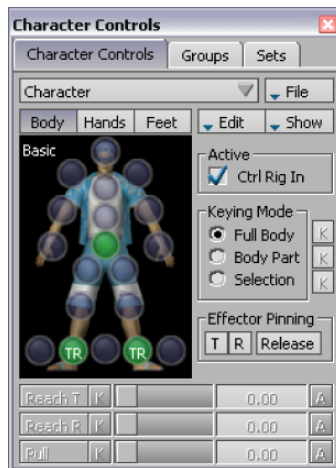
The Create Control Rig dialog box appears.



- 5 Click FK/IK.

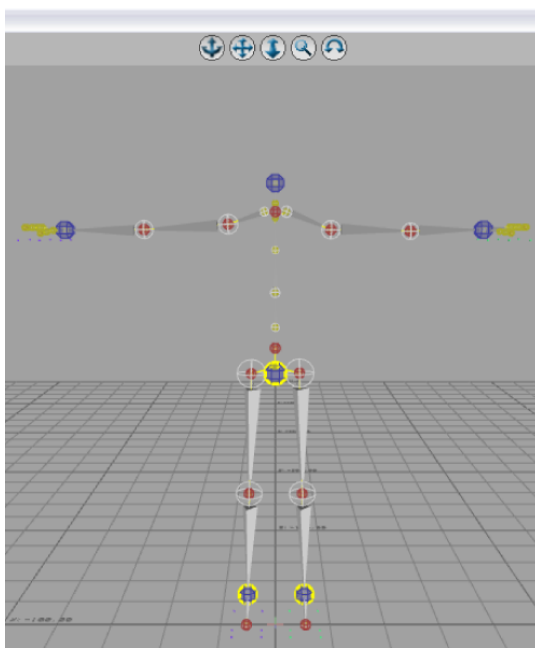
FK/IK is the method commonly used to animate characters.

- 6 In the Character Controls window Active area, activate Ctrl Rig In option.



Ctrl Rig In option activated

This setting activates the Control rig and the Character Controls Character representation to the left of the option. The Character representation is an image of a human form meant to represent the biped skeleton. It contains all the effectors you need to animate the Control rig, as shown in the following figure.

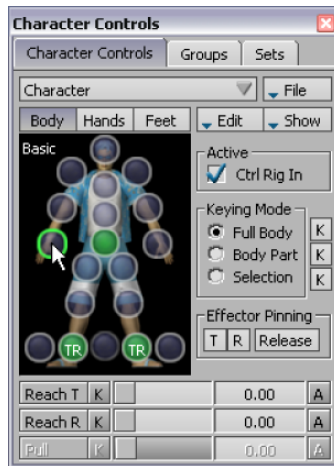


Skeleton with IK/FK Control rig

Your character is now rigged and ready to receive animation.

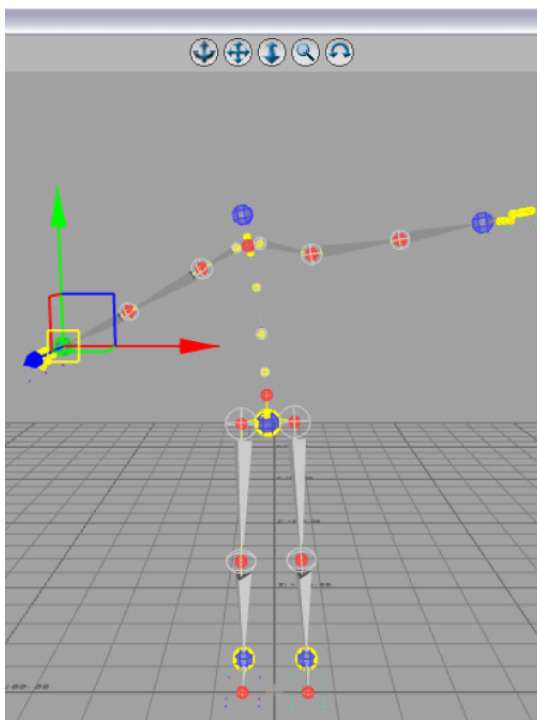
NOTE Rigging a character in 3ds Max using regular FK/IK constraints would have taken more effort.

- 7 On the Character representation, select the Right Wrist effector.



Right Wrist effector selected

- 8 Click in the Viewer window and press *T*.
The transformation handles appear.
- 9 Translate (or move) the hand down as shown in the following figure.



Notice as you move the hand, the arm extends and the rest of the body follows in a natural movement.

You can now transform the characterized skeleton.

NOTE If you wish to see the result of this procedure, open the *bone_skeleton_characterized.FBX* file.

3ds Max Bipeds

This tutorial shows you how to create and export biped skeletons created with the 3ds Max Biped system to MotionBuilder for animation.

Following is the result asset for this tutorial:

■ *biped.FBX*

NOTE You can find the tutorial assets in the MotionBuilder Asset browser's *Tutorials* folder as well as in the *Tutorials* folder located in the MotionBuilder directory on your system.

Create and export a 3ds Max biped

In the following procedure you create a biped in 3ds Max for export to MotionBuilder.

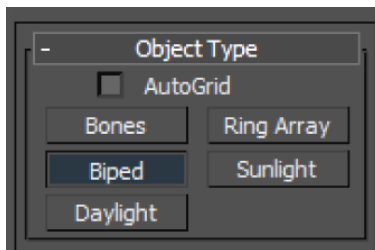
To create a biped in 3ds Max for export to MotionBuilder:

- 1 Launch the 3ds Max software.
- 2 In the Application menu, choose Reset to clear the scene/settings.
- 3 On the Create panel, select Systems.



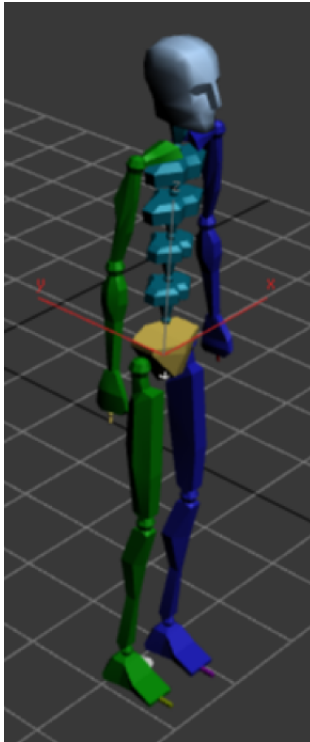
Systems selected in Create panel

- 4** On the Object Type rollout, click Biped.



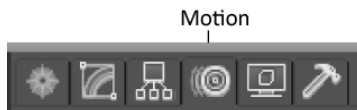
- 5 In the perspective viewport, click and drag to create a biped object.

NOTE The size of the biped is not important.



3ds Max biped object

- 6 On the Create panel, select Motion.



- 7 Select any bone in the biped and then go to the Motion panel > Biped rollout and click Figure Mode.

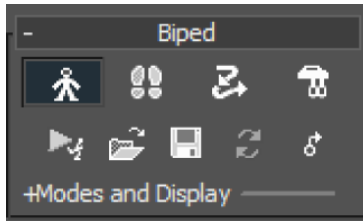
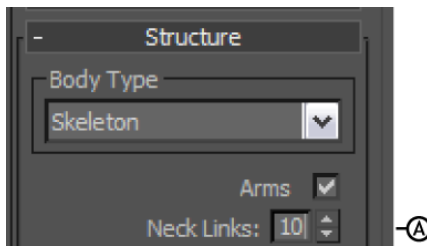


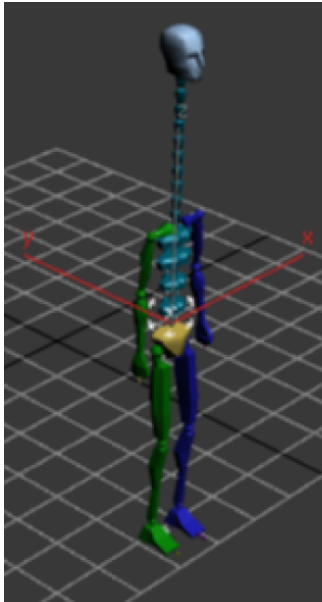
Figure Mode selected

- 8** In the Structure rollout > Body Type group > Neck Links spinner box, enter 10.



Biped with ten neck links

The 3ds Max biped object now has ten neck links as shown in the following figure.



3ds Max biped object with ten neck links

TIP Because MotionBuilder has ten neck link channels, it is good practice to make them available when you create your biped.

- 9 In the Spine links spinner box, enter 10, in the Fingers spinner box, enter 5, in the Finger Links spinner box, enter 3, in the Toes spinner box, enter 5 and in the Toe Links spinner box, enter 3.
- 10 Go to the Motion panel > Biped rollout and click Figure Mode again to exit the input mode.
- 11 From the Application menu, select Export > Export.
- 12 In the Select File to Export dialog box, navigate to the MotionBuilder root directory under the *Tutorials* folder, name your file `My_biped`, save your file as *Autodesk (*.FBX)* file type, and click Save.
- 13 In the FBX Export dialog box, click OK.

Now that your biped is saved as an FBX file, you can use it in the next tutorial ([Import and characterize a 3ds Max biped in MotionBuilder](#) on page 132) to import and characterize in MotionBuilder.

Once you export a biped to the FBX file format, you can bring it into other Autodesk products, such as Autodesk Maya.

NOTE If you wish to see the result of this procedure, open the *biped.FBX* file.

Import and characterize a 3ds Max biped in MotionBuilder

In the following procedure, you import in MotionBuilder the 3ds Max biped (*My_biped.FBX*) you saved in the previous procedure ([Create and export a 3ds Max biped](#) on page 128) so you can animate it. Characterization assigns a Control rig to the biped bones and is the name MotionBuilder uses for the process of rigging a biped skeleton.

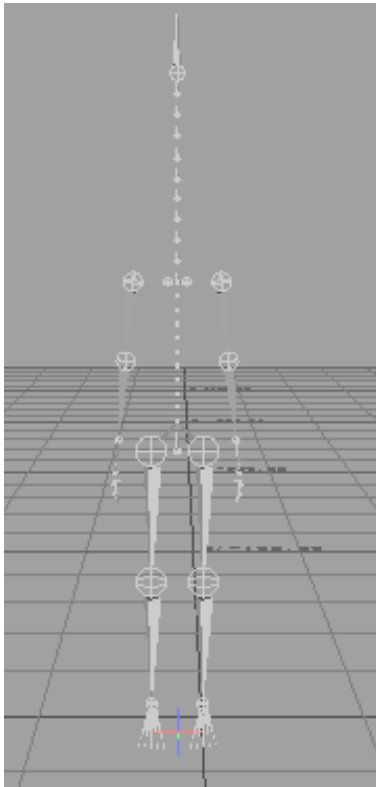
NOTE If you did not perform the previous procedure, you can use the *biped.FBX* file in the *Tutorials* folder located in the MotionBuilder directory on your system.

To import a 3ds Max biped into MotionBuilder:

- 1 Launch MotionBuilder.
- 2 From the MotionBuilder main menu, select File > Open.
Because you exported your 3ds Max to the FBX file format, you can open it in MotionBuilder.
- 3 In the Open File dialog box, select the *My_biped.FBX* file you created in the previous procedure ([Create and export a 3ds Max biped](#) on page 128) and click Open.

NOTE If you did not perform the previous procedure, open the *biped.FBX* file located in the MotionBuilder root directory under the *Tutorials* folder.

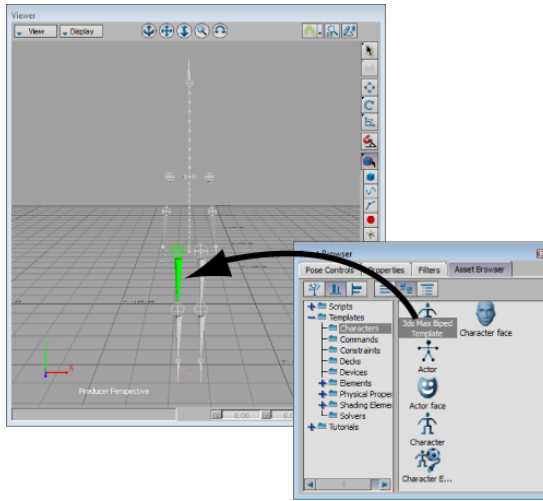
- 4 In the FBX Plug-in Import Options dialog, click Open.
- 5 Position your cursor anywhere in the Viewer window and press *A* to frame all of the biped skeleton.



3ds Max biped skeleton imported in MotionBuilder

To characterize your 3ds Max biped:

- 1 In the Asset browser, expand Templates > Characters and drag the 3ds Max Biped Template on top of the skeleton.



Applying the 3ds Max Biped Template asset to the biped skeleton

The 3ds Max Biped template is specially designed for bipeds created in 3ds Max as they have an unconventional naming structure that MotionBuilder does not recognize.

- 2 Click Characterize in the menu that appears.

The biped skeleton is now characterized. The next step is to add a Control rig so you can animate it.

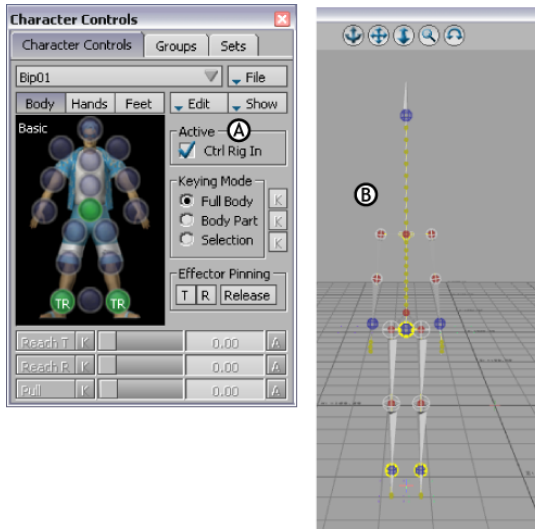
- 3 From the Character Controls window, select Edit > Control Rig Input.



Character Controls Edit menu

You must use the Control Rig Input setting if you intend to animate your biped.

- 4 In the Create Control Rig dialog box, click FK/IK.
FK/IK is the method commonly used to animate characters.
- 5 In the Character Controls window Active area, activate Ctrl Rig In (A).
The Ctrl Rig In setting activates the Character Controls Character representation and displays the Control rig effectors on the biped in the Viewer window (B).



A. Ctrl Rig In activated B. Control rig effectors appear on the biped

The Character representation is an image of a human form meant to represent the biped skeleton. It contains all the effectors you need to animate its Control rig. The biped is now rigged and ready to receive animation.

NOTE Rigging a character in 3ds Max using regular FK/IK constraints would have taken more effort.

- 6 On the Character representation, select the Right Wrist effector.

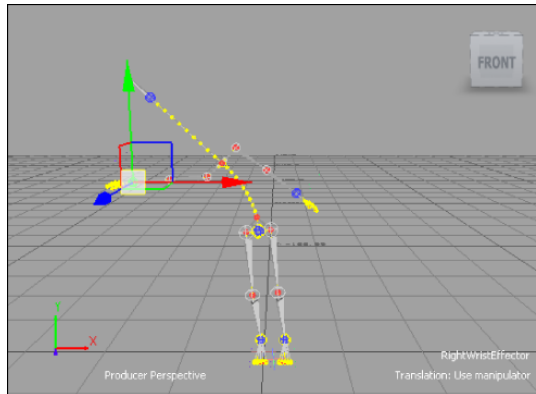


Right Wrist effector selected

- 7 Click in the Viewer window and press *T*.

The transformation handles appear.

- 8 Translate (or move) the hand down as shown in the following figure.



Notice as you move the hand, the arm extends and the rest of the body follows in a natural movement.

You can now transform and animate the characterized 3ds Max biped in MotionBuilder. See [Animating a 3ds Max Character in MotionBuilder](#) on page 144.

3ds Max Characters

The following tutorial shows you how to export a character with a skeleton created and skinned in 3ds Max to the FBX file format, import to MotionBuilder, and then characterize it so you can animate it.

Since a character is a skeleton with skin and textures, the procedure is similar to the procedure shown in the tutorials for exporting 3ds Max skeletons and biped skeletons.

Following is the asset required for this tutorial:

- *Pepe.max*

Following are the result assets for this tutorial:

- *Pepe.FBX*
- *Pepe_rigged.FBX*

NOTE You can find the tutorial assets in the MotionBuilder Asset browser's *Tutorials* folder as well as in the *Tutorials* folder located in the MotionBuilder directory on your system.

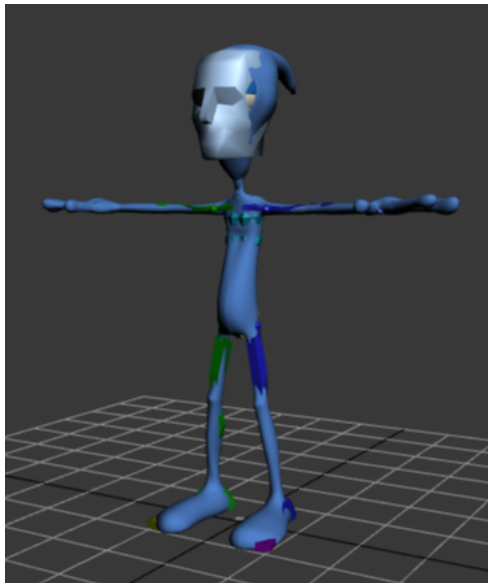
Export a 3ds Max character

The following procedure shows you how to export a 3ds Max character to MotionBuilder.

To export the 3ds Max Pepe character:

- 1 Launch the 3ds Max software.
- 2 In the Application menu, choose Reset to clear the scene/settings.
- 3 Select Open and open the *Pepe.max* scene file located in the MotionBuilder root directory under the *Tutorials* folder.

The *Pepe.max* scene file opens, displaying a biped skeleton inside a mesh.



3ds Max biped skeleton inside a mesh

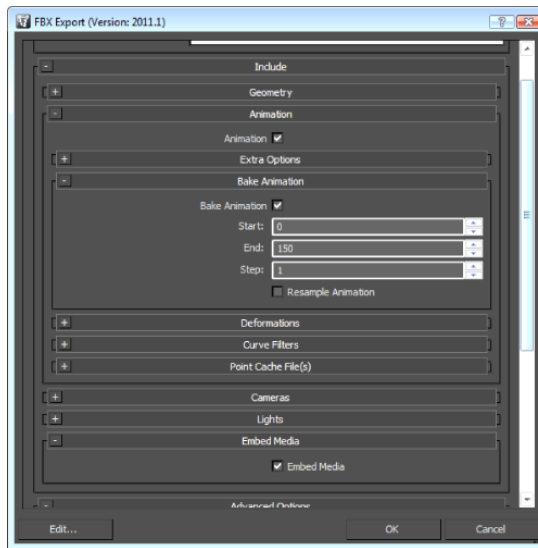
- 4 From the Application menu, select Export > Export.

- 5 In the Select File to Export dialog box, navigate to the MotionBuilder root directory in the *Tutorials* folder, name your file `My_Pepe`, save your file as *Autodesk (*FBX)* file type, and click Save.

NOTE If you do not specify a location, the file is automatically saved in the FBX file format to the 3ds Max Export folder.

The FBX Export dialog box opens. This is where you specify how to convert the 3ds Max character to an FBX file that MotionBuilder can recognize.

- 6 In the FBX Export Include rollout, activate the Animation, Bake Animations, and Embed Media options, then click OK.



Animation, Bake animations, and Embed Media activated

Activating these options enables you to export a mesh with the Pepe character, and export textures and materials assigned to the Pepe character.

Now that your character is in the FBX file format, you can use this FBX file in the next tutorial to import characters into MotionBuilder or other programs that support the FBX file format.

Import and characterize a 3ds Max Character in MotionBuilder

In the following procedure, you import into MotionBuilder the *My_Pepe.FBX* file you saved in the previous procedure ([Export a 3ds Max character](#) on page 138). The Pepe character is a 3ds Max biped, with only a mesh and materials, so some of this procedure is similar to the [Import and characterize a 3ds Max biped in MotionBuilder](#) on page 132. After importing your Character, you characterize it.

NOTE If you did not perform the previous procedure, you can use the *Pepe.FBX* file in the *Tutorials* folder located in the MotionBuilder directory on your system.

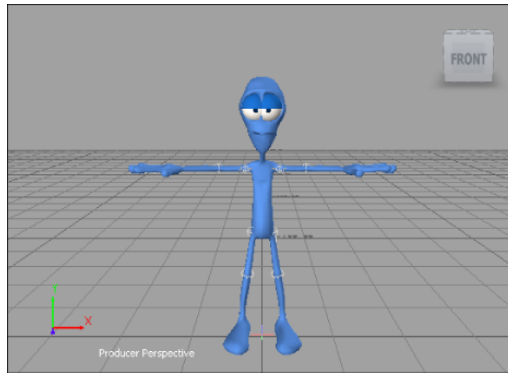
To import a 3ds Max character into MotionBuilder:

- 1 Launch the MotionBuilder software.
- 2 From the MotionBuilder main menu, select File > Open.
- 3 From the Open File dialog box, choose the *My_Pepe.FBX* file you exported in [Export a 3ds Max character](#) on page 138, click Open, then click Open in the FBX Plug-in Import Options dialog.

NOTE You can also import the *Pepe.FBX* file located in the MotionBuilder root directory under the *Tutorials* folder.

The Pepe character appears in the MotionBuilder Viewer window.

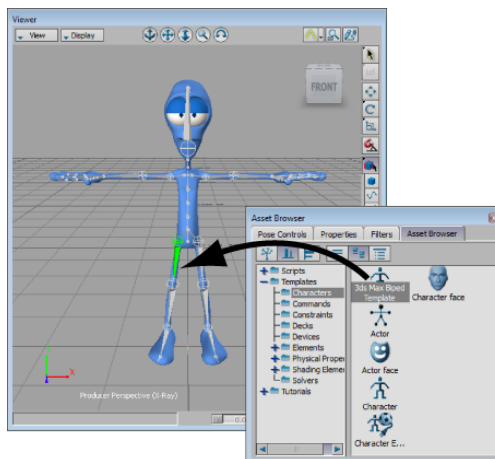
- 4 Position your cursor anywhere in the Viewer window and press *A* to frame all of the character.



3ds Max Pepe character imported into MotionBuilder

To characterize your Character:

- 1 In the Viewer window, press *Ctrl-A* until you are in X-Ray mode. X-Ray mode lets you see through the character skin to the skeleton underneath.
- 2 In the Asset browser, expand Templates > Characters and drag the 3ds Max Biped Template asset on top of the Pepe Character skeleton.



3ds Max Biped Template asset applied to the Pepe Character skeleton

The 3ds Max Biped template is specially designed for bipeds created in 3ds Max as they have a different naming structure that the MotionBuilder Character does not recognize automatically.

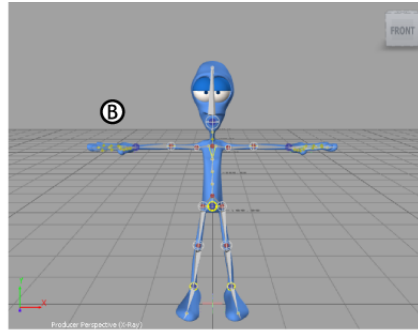
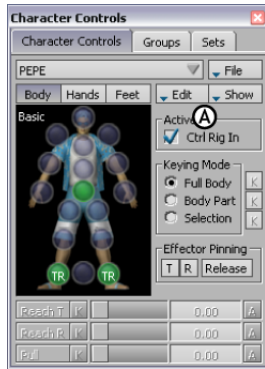
- 3 Click Characterize in the menu that appears.
- 4 From the Character Controls window, select Edit > Control Rig Input.



Character Controls Edit menu

You must use the Control Rig Input setting if you intend use keyframe animation on your biped.

- 5 In the Create Control Rig dialog box, click FK/IK.
FK/IK is the method commonly used to animate characters.
- 6 In the Character Controls window Active area, activate Ctrl Rig In. Effectors appear on the biped as shown in the following figure.

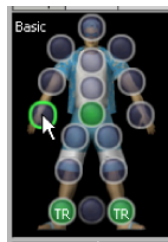


A. Ctrl Rig In activated B. Effectors appear on the character

This setting activates the Control rig and the Character Controls Character representation. The Character representation is an image that shows an image of a human form meant to represent the character skeleton. It contains all the effectors you need to animate its Control rig. The biped is now rigged and ready to receive animation.

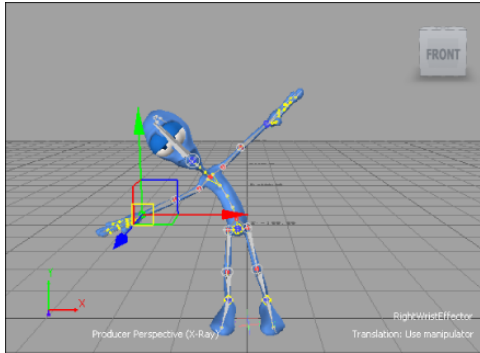
NOTE Rigging a character in 3ds Max using regular FK/IK constraints would have taken more effort.

- 7 On the Character representation, select the Right Wrist effector.



Right Wrist effector selected

- 8 Click in the Viewer window and press *T*.
The transformation handles display.
- 9 Translate (or move) the hand down as shown in the following figure.



Notice as you move the hand, the arm extends and the rest of the body follows in a natural movement.

You can now animate your character with motion capture and key frame animation.

- 10 Save your file as *My_Pepe_rigged.FBX* in the MotionBuilder root directory under the *Tutorials* folder.

NOTE You can use this file for the next tutorial or you can use the *Pepe_rigged.FBX* file in the *Tutorials* folder located in the MotionBuilder root directory on your system.

See [Animating a 3ds Max Character in MotionBuilder](#) on page 144.

Animating a 3ds Max Character in MotionBuilder

You can animate characters in MotionBuilder two ways: by setting keyframes manually, or using motion capture data. The following tutorials show you how to animate the 3ds Max *Pepe* character you imported in the [3ds Max Characters](#) on page 137 section.

NOTE If you did not perform the previous procedures in section [Import and characterize a 3ds Max Character in MotionBuilder](#) on page 140, you can use the *Pepe_rigged.FBX* file in the *Tutorials* folder located in the MotionBuilder root directory on your system.

In the first of these animation tutorials, [Animate a character using motion capture data](#) on page 145, you animate *Pepe* by using one of the motion capture files that ship with MotionBuilder. In the second tutorial, [Animate a character by adding keyframes](#) on page 151 you refine the animation using keyframes.

NOTE If you already know how to animate characters in MotionBuilder, you can skip this section.

Following are the assets required for this tutorial:

- *Pepe_rigged.FBX*
- *Iceslip.fbx*

Following are the result assets for this tutorial:

- *Pepe_Mocap.FBX*
- *Pepe_keyanim.FBX*

NOTE You can find the tutorial assets in the MotionBuilder Asset browser's *Tutorials* folder as well as in the *Tutorials* folder located in the MotionBuilder root directory on your system.

If you are new to MotionBuilder, take a moment to try a few MotionBuilder navigation techniques using the MotionBuilder keyboard shortcuts:

- Press *Ctrl-Shift* and drag to orbit around the scene.
- *Ctrl*-drag to zoom in and out of the scene.
- *Shift*-drag to pan the scene.

NOTE Although the procedures in this tutorial use the MotionBuilder keyboard shortcuts, you can elect to use the 3ds Max keyboard shortcuts. To do so, from the MotionBuilder menu bar, choose Settings > Keyboard Configuration > 3ds Max. See the 3ds Max keyboard shortcuts topic in the Keyboard Shortcuts chapter of the MotionBuilder Help for a list of all the 3ds Max keyboard shortcuts.

Animate a character using motion capture data

The following procedure shows you how to use keyframe animation techniques in MotionBuilder to refine the motion capture animation.

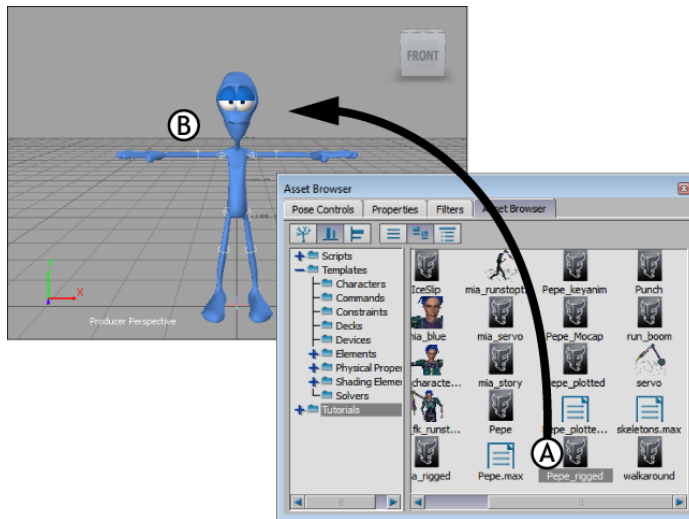
If you did not perform the previous procedures in section [Import and characterize a 3ds Max Character in MotionBuilder](#) on page 140, you can use the *Pepe_rigged.FBX* file in the MotionBuilder root directory in the *Tutorials* folder on your system or the *Pepe_rigged* asset in the MotionBuilder Asset browser's *Tutorials* folder.

To animate the Pepe character using motion capture data:

- 1 Launch the MotionBuilder software.
- 2 From the Asset browser's *Tutorials* folder, drag into the Viewer window the *My_Pepe_rigged* asset (*My_Pepe_rigged.FBX* file) you saved in the previous procedures in section [Import and characterize a 3ds Max Character in MotionBuilder](#) on page 140.

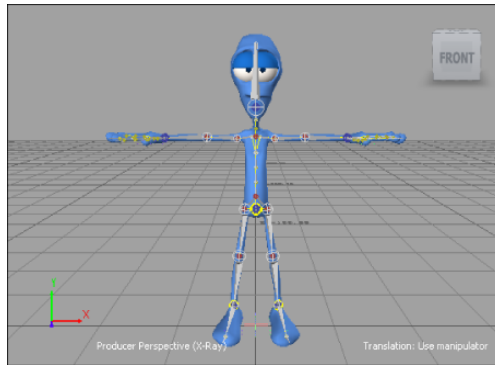
NOTE If you did not perform the previous procedures in section [Import and characterize a 3ds Max Character in MotionBuilder](#) on page 140, drag into the Viewer window the *Pepe_rigged* asset (*Pepe_rigged.fbx* file) into the Viewer window.

- 3 Select **FBX Open > No Animation**.
The Pepe character appears in the Viewer window.



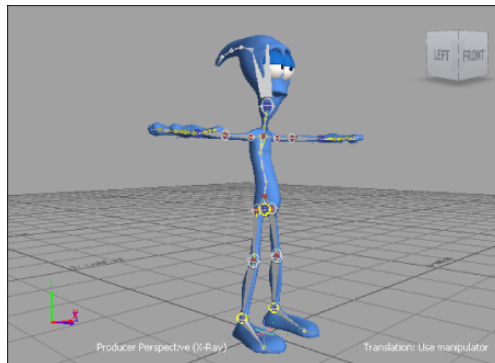
Loading the file A. Pepe_rigged asset in Asset browser B. Pepe loaded in the scene

- 4 Click on an empty area in the Viewer window and press **A** to frame all and zoom in on the *Pepe* character.
- 5 Press **Ctrl-A** until you are in X-ray mode and can see Pepe's skeleton and Control rig.



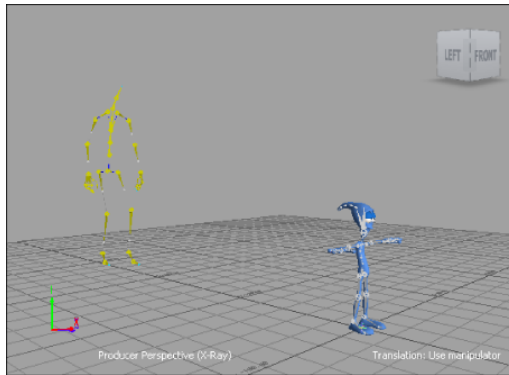
Pepe's Control rig shown in X-Ray mode

- 6 Press *Ctrl-Shift* and drag to orbit until you can see the right side of the *Pepe* character. Use the following image as a guideline.



Orbit around Pepe

- 7 From the Asset Browser's *Tutorials* folder, select the *IceSlip* asset (*IceSlip.fbx* file) and drag it into an empty area of the Viewer window.
- 8 Select FBX Merge > *IceSlip*.
A large yellow skeleton representing the motion capture animation now joins *Pepe* in the scene.
- 9 Zoom out (*Ctrl*-drag down or left) until you can see the yellow skeleton.



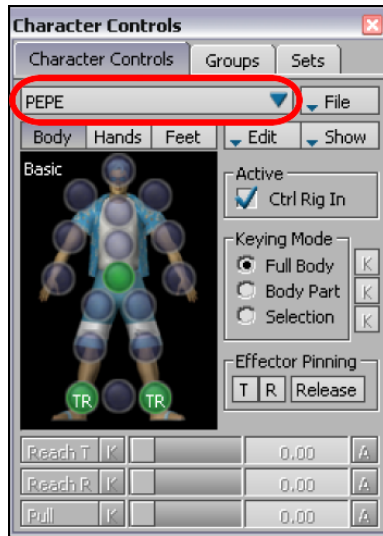
Pepe with skeleton containing motion capture data

- 10 In the Transport Controls, click Play to view the yellow skeleton's *IceSlip* animation.



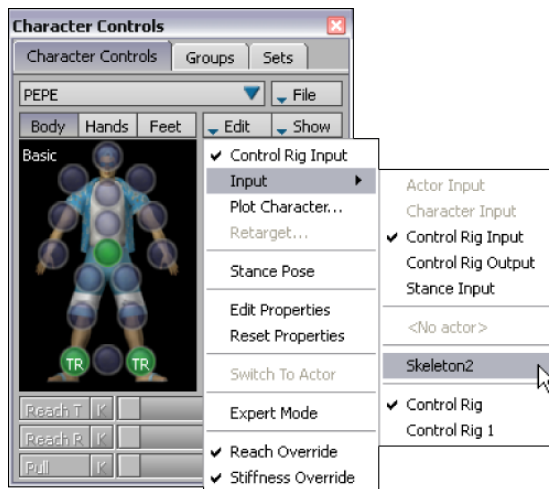
Transport Controls Play button

- 11 In the Character Controls > Current Character menu, make sure PEPE is displayed in the character list.

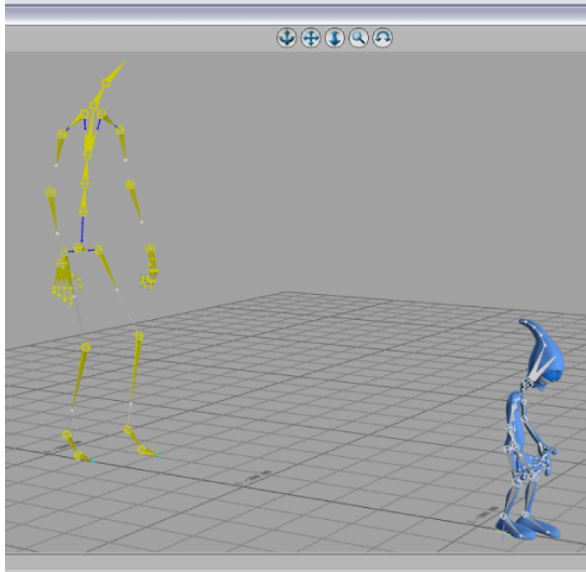


PEPE is selected in the Current Character menu.

- 12** In the Character Controls, select Edit > Input > Skeleton2, which is the name of the yellow skeleton that contains the motion capture animation.



Pepe assumes the skeleton's stance as shown in the following figure.

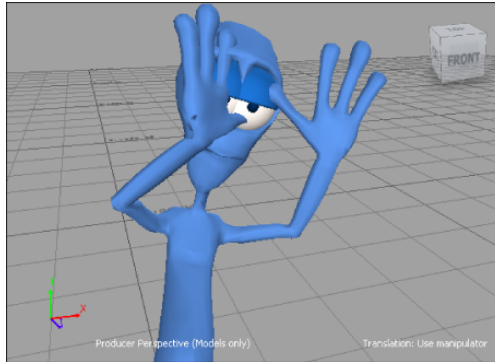


- 13 In the Transport Controls, drag the Timeline indicator (slider bar) to scrub the animation. You can also hold down the */* key and drag left or right in the Viewer window.

The skeleton's animation now drives the *Pepe* character.

- 14 Press *Ctrl-A* until only Pepe is visible, then go to frame 92 and zoom in on the *Pepe* character.
- 15 Press *Ctrl-Shift* and drag to orbit *Pepe*.

If you look carefully, you will notice that one of Pepe's hands passes through his face.



Unwanted hand movement from the motion capture

- 16 Scrub the animation a few times if you cannot see the problem.
The animation that drives Pepe's bone movement is based on a skeleton that has a very different physiology. For example, Pepe's head, hands, and feet are much larger than the skeleton, while Pepe's shoulders are much smaller than the skeleton's shoulder.
- 17 Save the file as *My_Pepe_Mocap.FBX*.
You can use this file for the next part where you add keyframes to correct Pepe's hand movement.

Animate a character by adding keyframes

In MotionBuilder, you can animate characters by setting keyframes manually. This is useful for creating original animation or making changes to motion capture animation.

In the following procedure, you use key frame animation to refine motion capture animation.

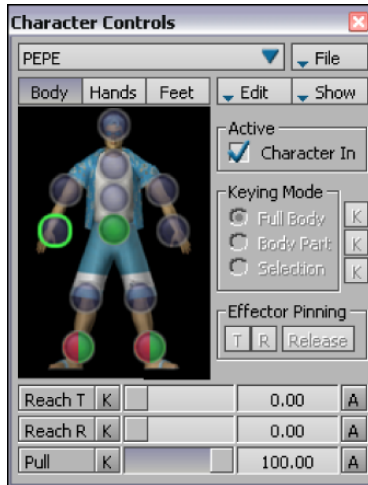
NOTE If you already know how to keyframe characters in MotionBuilder, you can skip the following procedure.

To fine-tune Motion Capture animation with keyframing:

- 1 In MotionBuilder, open your result *My_Pepe_Mocap.FBX* file from the previous procedure ([Animate a character using motion capture data](#) on page 145) to apply the motion capture data to *Pepe*.

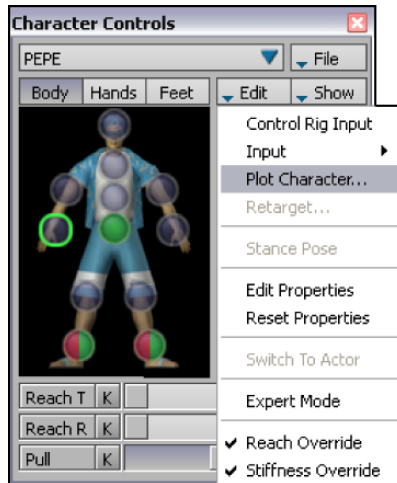
If you did not complete the previous procedure, open the *Pepe_Mocap.FBX* file.

- 2 On the Character Representation of the Character Controls, click the Right Wrist effector.



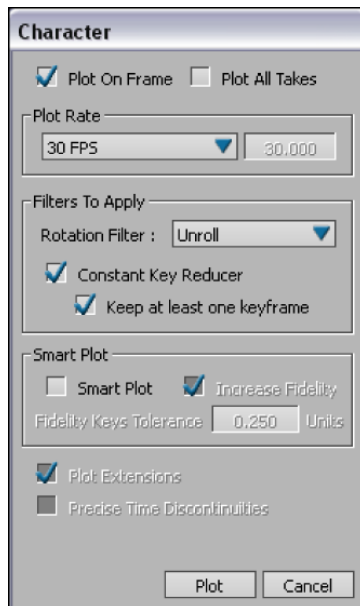
Right Wrist effector selected

- 3 Click in the Viewer window and press *T*.
The transformation handles do not display.
- 4 Try to move Pepe's hand.
Nothing happens because Pepe's animation is controlled by the skeleton, not the Control rig. Before you can keyframe Pepe's motion capture animation, you must plot (or bake) the skeleton animation onto the Pepe character Control rig.
- 5 In the Character Control window, select Edit > Plot Character.



Character Controls' Plot Character Edit menu selected

- 6 On the Character dialog box, click Control Rig.
The Character dialog box appears.



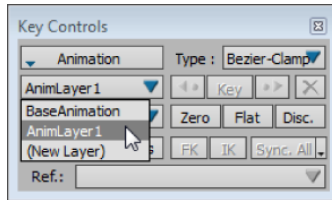
Character dialog box

- 7 Click Plot.

The Plot command creates a key at every frame at the base layer (or *BaseAnimation layer*) of the animation track, making edits difficult. (You can see these keyframes in the Transport controls if you select *Pepe's* wrist effector.)

You can now edit the *Pepe* character using *Pepe's* Control rig.

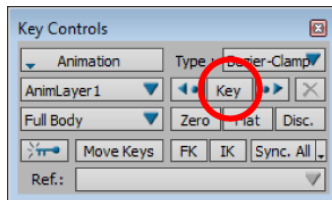
- 8 In the Key Controls, click the Layer menu and select AnimLayer1.



AnimLayer1 selected for adding keyframes

Selecting another layer lets you edit the animation while preserving the original animation on the BaseAnimation layer. When you select AnimaLayer1, the timeline hides the keyframes on the BaseAnimation layer and shows the keyframes set on AnimLayer1 (there are none at the moment).

- 9 Go to frame 80, which is the start of the problematic right hand movement. On the Character Controls window Character Representation, click the Right Wrist effector, then in the Key Controls, click Key.

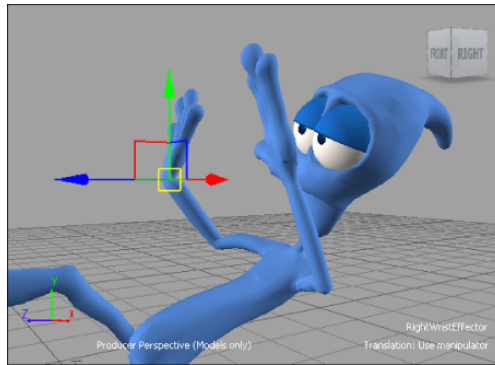


NOTE You can also set a key by pressing *K*.

- 10 Go to frame 105, the end of the problem hand movement, and set another key.

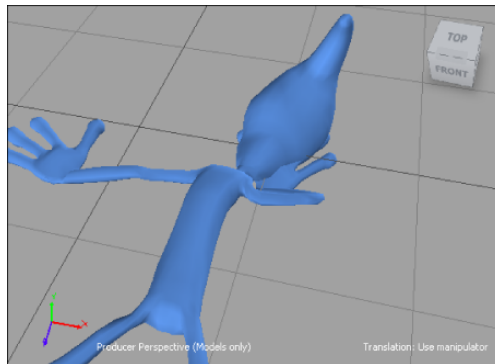
As you take the following steps, all character movement before the first key and after the second key will remain unchanged. Only the character movement between frames 80 to 105 will be modified.

- 11 Go to frame 94, the mid point between the two keyframes you set.
- 12 In the Viewer window, press *T*, move the hand away from Pepe's face on its X and Z axes as shown in the following figure, then set another key.



Hand moved away from Pepe's face

- 13 Press *J*, then drag back and forth to see how the hand reacts to the keys you just created.
- 14 Make any further adjustments to the hand movement as required. Make sure to create a key after each adjustment.
- 15 Advance to the last frame of the animation and adjust your view until you can see the right side of Pepe's body.

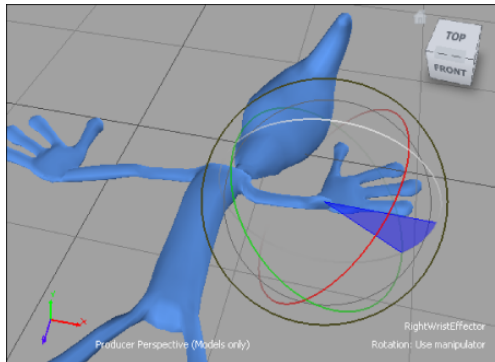


Right hand is too close to the character's head

- 16 Move Pepe's hand away from his body and set a key.

- 17 Press *R* to use the key rotation rings to modify the hand's position until it rests flat on the ground, then set another key.

NOTE You may need to change your view so you can see if Pepe's hand is level with the floor.



Rotation rings used to reposition right hand

- 18 Play back the animation to see the result.
- 19 Make any further adjustments to the character body position and save your file as *My_Pepe_keyanim.FBX*.
If you want to export your FBX file back to 3ds Max, you must bake the animation. See [Exporting a Character and its animation back to 3ds Max](#) on page 156.

Exporting a Character and its animation back to 3ds Max

Because 3ds Max is unable to read the Control rig information that MotionBuilder uses to define character animation in MotionBuilder, you must plot, or “bake”, the animation data onto the character skeleton.

This tutorial shows you how to:

- [Bake animation for export to 3ds Max](#) on page 157
- [Export MotionBuilder scene files to 3ds Max](#) on page 159

The following asset is required for this tutorial:

- *Pepe_keyanim.FBX*

The following assets are the result assets for this tutorial:

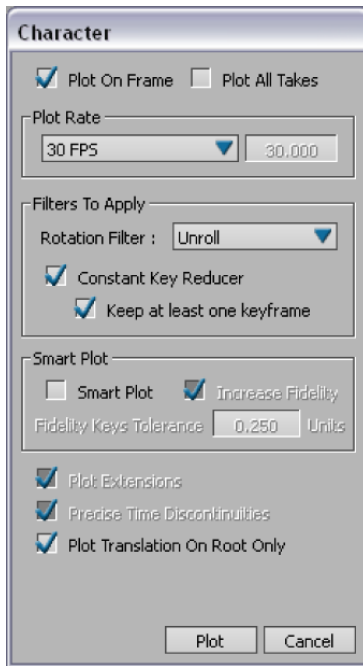
- *Pepe_plotted.FBX*
- *Pepe_plotted.max*

Bake animation for export to 3ds Max

This following procedure shows you how to plot animation on your Character and prepare it for import back to 3ds Max.

To bake animation onto the Pepe character skeleton:

- 1 Launch MotionBuilder.
- 2 Open your result *My_Pepe_keyanim.FBX* file from the previous procedure ([Animate a character by adding keyframes](#) on page 151).
If you did not complete the previous procedure, open the *Pepe_keyanim.FBX* file.
- 3 In the Character Controls, select Edit > Plot Character.
- 4 In the Character dialog box, click Skeleton.
- 5 In the second Character dialog box, leave the default values unchanged as shown in the following figure and click Plot.



Character default settings

The character Control rig is deactivated, but the Pepe character retains all animation information.

To edit the character's movement after its animation has been plotted, go back to the Character Controls, and select Edit > Plot Character > Control Rig again. When you are done, repeat steps 3 to 5 to bake the animation back onto the character skeleton.

NOTE When you save your file, the animated Pepe character in your scene is saved, but so is the yellow reference skeleton. If you want, you can delete the skeleton from the scene, or select the Pepe character and save it to another file for import to 3ds Max, but it is not necessary. You can just as easily strip out the skeleton when you import to 3ds Max.

- 6 From the MotionBuilder main menu, select File > Save As, navigate to the MotionBuilder root directory in the *Tutorials* folder, save your file `My_Pepe_plotted` under the *Tutorials* folder, and click Save.

Export MotionBuilder scene files to 3ds Max

The FBX Plug-in lets you import the entire contents of scenes saved in MotionBuilder, or update only the elements whose names match those in your 3ds Max scene.

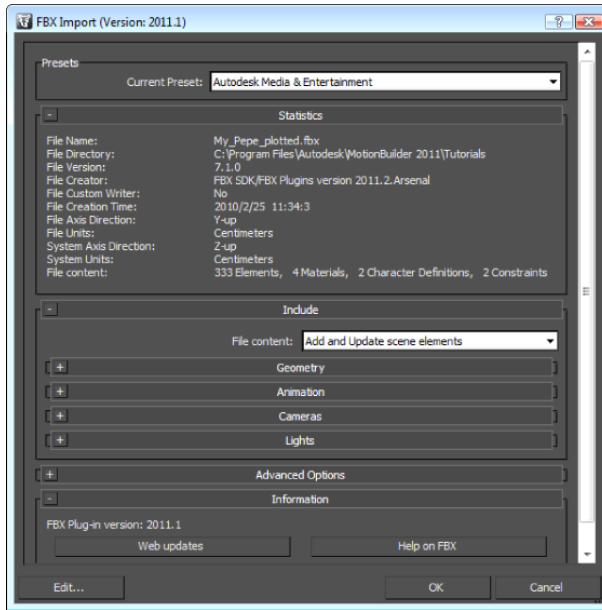
The animation you import from MotionBuilder is now fully editable in 3ds Max.

To import the animated character to 3ds Max:

- 1 Launch 3ds Max.
- 2 In the Application menu, choose Reset to clear the scene/settings.
- 3 From the Application menu, select Import > Import.
- 4 In the Select File To Import dialog box, navigate to the MotionBuilder root directory in the *Tutorials* folder and open your result *My_Pepe_plotted.FBX* file from the previous procedure ([Bake animation for export to 3ds Max](#) on page 157).

If you did not complete the previous procedure, open the *Pepe_plotted.FBX* file.

The FBX Import dialog appears.

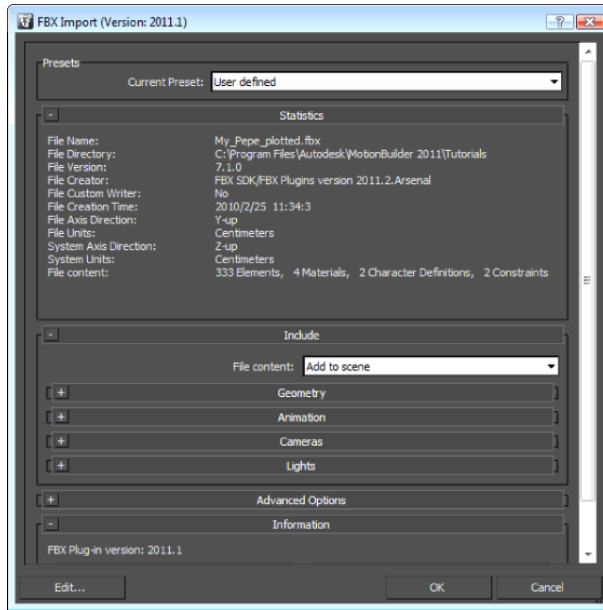


FBX Import dialog

- 5 In the FBX Import window, scroll down to and expand the Include rollout, if it is not already expanded.

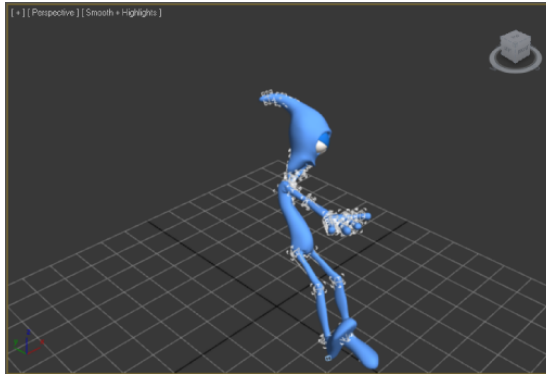
The File Content list displays Add and Update Scene Elements by default. This default setting imports the Pepe character and the yellow reference skeleton. The Update scene elements option updates only the scene elements in 3ds Max that share the same name as those in the imported file.

- 6 Choose the Add to scene option to import animation from MotionBuilder to a new 3ds Max scene.



- 7 Click OK.
- 8 If a Warnings and Errors dialog appears concerning Skin Modifiers Imports, click OK.
- 9 Zoom in on Pepe in the Perspective viewport and scrub the timeline to see how the MotionBuilder animation has been baked into the Pepe character bones.

NOTE To clean up the viewport, region select the yellow skeleton and right-click to obtain the quad menu. From the menu select Hide Selection and the skeleton becomes hidden.



MotionBuilder animation imported in 3ds Max

The animation you created in MotionBuilder now works in 3ds Max.

Summary

In this series of tutorials, you took different skeletons created in 3ds Max and exported them to MotionBuilder as an FBX file. In MotionBuilder, you characterized the bones, and animated the character by plotting it to a skeleton.

Then, you baked the animation back to Pepe's control rig, made a few adjustments to perfect the motion, and baked the refined animation back into Pepe's skeleton for export to 3ds Max.

Glossary

The Glossary describes terms specific to the MotionBuilder software product as well as some of the most common computer graphics and software terms used in the 3D world.

The Glossary also includes acronyms used in the MotionBuilder software product and the most common acronyms used in the 2D/3D world.

Acronyms

Following is a list of acronyms used in the MotionBuilder User's Guide.

- [BCC](#) on page 179
- [BCS](#) on page 180
- [BVH](#) on page 182
- [FK](#) on page 191
- [fps](#) on page 191
- [HSB](#) on page 194
- [HUD](#) on page 194
- [IK](#) on page 194
- [NTSC](#) on page 202
- [NURBS](#) on page 202
- [PAL](#) on page 204
- [Qt](#) on page 206
- [SMPTE](#) on page 211
- [TCB](#) on page 213

- [UCS](#) on page 215
- [UV](#) on page 216
- [VTR](#) on page 217
- [WCS](#) on page 218

Terms

Following is a list and description of terms specific to the MotionBuilder software product and of some of the most common computer graphics and software terms used in the 3D world.

List of terms

The following provides a list of terms specific to the MotionBuilder software product as well as some of the most common computer graphics and software terms used in the 3D world.

0-9

- [3D coordinate space](#) on page 176
- [3D matte](#) on page 176

A

- [Actor](#) on page 176
- [Actor Face](#) on page 176
- [aliasing](#) on page 176
- [alpha channel](#) on page 177
- [alpha-blend](#) on page 177
- [animation](#) on page 177
- [anti-aliasing](#) on page 177
- [artifact](#) on page 177
- [asset](#) on page 177

- [attribute](#) on page 178
- [Auxiliary effector](#) on page 178
- [Auxiliary pivot](#) on page 178

B

- [back plate](#) on page 178
- [Background Color Cancellation \(BCC\)](#) on page 178
- [Background Color Suppression \(BCS\)](#) on page 179
- [Background generator](#) on page 179
- [background plane](#) on page 179
- [BaseAnimation layer](#) on page 179
- [batch](#) on page 179
- [Baud rate](#) on page 179
- [BCC](#) on page 179
- [BCS](#) on page 180
- [bind pose](#) on page 180
- [Biovision Hierarchical Data \(BVH\)](#) on page 180
- [biped](#) on page 180
- [bitplane](#) on page 180
- [Black level](#) on page 180
- [blending object](#) on page 181
- [bone](#) on page 181
- [bound model](#) on page 181
- [bounding box](#) on page 181
- [branch](#) on page 181
- [brightness](#) on page 181
- [buffer](#) on page 181

- [bump map](#) on page 182
- [burst](#) on page 182
- [BVH](#) on page 182

C

- [camera](#) on page 182
- [camera interest](#) on page 182
- [channel](#) on page 182
- [Character](#) on page 183
- [character animation](#) on page 183
- [Character asset](#) on page 183
- [Character Face](#) on page 183
- [Character mapping](#) on page 183
- [character model](#) on page 183
- [child](#) on page 184
- [chroma key](#) on page 184
- [chrominance](#) on page 184
- [clip](#) on page 184
- [cluster](#) on page 184
- [cluster shapes](#) on page 184
- [color burst](#) on page 185
- [color timing](#) on page 185
- [COM port](#) on page 185
- [combiner](#) on page 185
- [command clip](#) on page 185
- [communications port](#) on page 185
- [constrained object](#) on page 186

- [constraint](#) on page 186
- [constraint clip](#) on page 186
- [contrast](#) on page 186
- [Control rig](#) on page 186
- [cross chrominance](#) on page 186
- [cross color](#) on page 186
- [current segment](#) on page 187
- [custom keying group](#) on page 187
- [cut](#) on page 187

D

- [deck](#) on page 187
- [deformation](#) on page 187
- [dense data](#) on page 187
- [device](#) on page 188
- [Done state](#) on page 188
- [dopesheet](#) on page 188
- [Distribution Factor](#) on page 188
- [dummy node](#) on page 188

E

- [effector](#) on page 188
- [Effects send](#) on page 189
- [element](#) on page 189
- [Environment mapping](#) on page 189
- [Expression](#) on page 189
- [expressions constraint](#) on page 189
- [extrapolation](#) on page 189

F

- [.fbx](#) on page 190
- [FCurve](#) on page 190
- [fill](#) on page 190
- [filter](#) on page 190
- [filtering](#) on page 190
- [First Contact balloon](#) on page 190
- [FK](#) on page 191
- [FK effector](#) on page 191
- [FK rig](#) on page 191
- [Forward Kinematics \(FK\)](#) on page 191
- [fps](#) on page 191
- [frame](#) on page 191
- [frame rate](#) on page 192
- [function curve](#) on page 192

G

- [gap](#) on page 192
- [generic channel](#) on page 192
- [ghost](#) on page 192
- [Ghost curve](#) on page 192
- [global coordinates](#) on page 193
- [global keying group](#) on page 193
- [gobo](#) on page 193
- [Guide pose](#) on page 193

H

- [Hardware FC](#) on page 193

- [Head-up Display \(HUD\)](#) on page 194
- [hierarchy](#) on page 194
- [HSB](#) on page 194
- [HUD](#) on page 194
- [hue](#) on page 194

I

- [IK](#) on page 194
- [IK effector](#) on page 195
- [IK rig](#) on page 195
- [interpolation](#) on page 195
- [Inverse Kinematics \(IK\)](#) on page 195
- [IP address](#) on page 195

J

- [jogging](#) on page 196
- [joint](#) on page 196

K

- [key](#) on page 196
- [keyframe](#) on page 196
- [keyframing](#) on page 196
- [keying group](#) on page 196

L

- [latency](#) on page 197
- [layer](#) on page 197
- [Linear key](#) on page 197
- [local blend](#) on page 197

- [local coordinates](#) on page 197
- [local keying group](#) on page 198
- [look at point](#) on page 198
- [loop](#) on page 198
- [luminance](#) on page 198
- [luminance key](#) on page 198

M

- [magnetic mapping](#) on page 198
- [marker](#) on page 199
- [Marker set](#) on page 199
- [Match pose](#) on page 199
- [material](#) on page 199
- [Mipmap](#) on page 199
- [model](#) on page 200
- [moire](#) on page 200
- [morph target](#) on page 200
- [motion capture](#) on page 200
- [motion source](#) on page 200

N

- [Namespace](#) on page 200
- [naming template](#) on page 201
- [National Television System Committee \(NTSC\)](#) on page 201
- [node](#) on page 201
- [noise](#) on page 201
- [Non Uniform Rational B-splines \(NURBS\)](#) on page 201
- [normal](#) on page 201

- [normal map](#) on page 202
- [NTSC](#) on page 202
- [null](#) on page 202
- [NURBS](#) on page 202

O

- [object keying group](#) on page 202
- [occlusion](#) on page 203
- [opacity](#) on page 203
- [OpenGL](#) on page 203
- [Optical editor](#) on page 203
- [optical mapping](#) on page 203
- [Optical root](#) on page 203
- [origin](#) on page 204

P

- [PAL](#) on page 204
- [parameter](#) on page 204
- [parent](#) on page 204
- [parenting](#) on page 204
- [partial occlusion](#) on page 205
- [patch](#) on page 205
- [Phase Alternating Line \(PAL\)](#) on page 205
- [pitch](#) on page 205
- [pivot](#) on page 205
- [pole vector](#) on page 205
- [pose](#) on page 206
- [property](#) on page 206

Q

- [Qt](#) on page 206
- [quadruped](#) on page 206
- [quaternion](#) on page 206

R

- [reference node](#) on page 207
- [Relations](#) on page 207
- [relational constraint](#) on page 207
- [remote port](#) on page 207
- [render](#) on page 207
- [rest pose](#) on page 207
- [retargeting](#) on page 208
- [Rigid body](#) on page 208
- [roll](#) on page 208
- [rotation](#) on page 208

S

- [sample](#) on page 208
- [saturation](#) on page 208
- [scaling](#) on page 209
- [scene](#) on page 209
- [scrubbing](#) on page 209
- [segment](#) on page 209
- [sensor](#) on page 209
- [serial port](#) on page 209
- [shader](#) on page 210
- [Shadow map](#) on page 210

- [shape](#) on page 210
- [shape operators](#) on page 210
- [shuttling](#) on page 210
- [simple constraint](#) on page 210
- [skeleton](#) on page 211
- [skin](#) on page 211
- [SMPTE](#) on page 211
- [solving](#) on page 211
- [source object](#) on page 211
- [Sphere map](#) on page 211
- [Spherical map](#) on page 212
- [spline](#) on page 212
- [stabilizing object](#) on page 212
- [stack](#) on page 212
- [stance pose](#) on page 212
- [SteeringWheels](#) on page 212
- [subcarrier](#) on page 212
- [swapping](#) on page 213

T

- [take](#) on page 213
- [tangent handle](#) on page 213
- [TCB](#) on page 213
- [tessellation](#) on page 213
- [texture](#) on page 213
- [timecode](#) on page 214
- [Timewarp](#) on page 214

- [track](#) on page 214
- [transformation](#) on page 214
- [translation](#) on page 214
- [transparency](#) on page 214
- [trigger](#) on page 215
- [triggering group](#) on page 215
- [T-stance](#) on page 215

U

- [UCS](#) on page 215
- [unlabelled segment](#) on page 215
- [unweighted tangent](#) on page 215
- [up-vector](#) on page 216
- [user channels](#) on page 216
- [User Coordinate System \(UCS\)](#) on page 216
- [UV](#) on page 216

V

- [value](#) on page 216
- [vector](#) on page 216
- [ViewCube](#) on page 217
- [visual keyframe](#) on page 217
- [voice channels](#) on page 217
- [VK ripple](#) on page 217
- [VTR](#) on page 217

W

- [waveform](#) on page 217

- [WCS](#) on page 218
- [weighted tangent](#) on page 218
- [wheel](#) on page 218
- [wheel surface](#) on page 218
- [wheel wedge](#) on page 218
- [wheels](#) on page 218
- [wireframe](#) on page 219
- [World Coordinate System \(WCS\)](#) on page 219
- [world coordinates](#) on page 219

X

- [X-axis](#) on page 219
- [X-coordinate](#) on page 219

Y

- [Y-axis](#) on page 219
- [Y-coordinate](#) on page 220
- [yaw](#) on page 220

Z

- [Z-axis](#) on page 220
- [Z-coordinate](#) on page 220
- [Zero keyframe](#) on page 220
- [zero point](#) on page 220

Term Definitions

The following provides a description of terms specific to the MotionBuilder software product and of some of the most common computer graphics and software terms used in the 3D world.

0-9

3D coordinate space

The Euclidian/Cartesian environment that defines three dimensions by X-, Y-, and Z-axes, and their corresponding coordinate values.

See also [X-axis](#) on page 219, [Y-axis](#) on page 219, and [Z-axis](#) on page 220.

3D matte

A color signal that is used to fill areas of keys and borders. Unlike a regular matte, a 3D matte has depth and is respected by all other 3D objects in the scene, letting you block out parts of a scene and replace it with video footage.

A

Actor

In MotionBuilder, a humanoid model used to link captured optical or magnetic motion data to a character.

Actor Face

In MotionBuilder, a set of magnetic or optical motion data captured from a performer's face, which can be mapped to a Character Face asset.

aliasing

A defect or distortion in a television picture caused by interference between two frequencies, for example the luminance and chrominance frequencies. Aliasing appears as moire or herringbone patterns, straight lines that become wavy, or rainbow colors.

See also [anti-aliasing](#) on page 177.

alpha channel

The portion of each pixel's data reserved for transparency information. 32-bit graphics systems contain four channels: three 8-bit channels for red, green, and blue (RGB) and one 8-bit alpha channel.

alpha-blend

An effect in which you assign pixel values that are solid, invisible, or partially transparent. Alpha-blending is often used in games for special effects such as explosions and weapons discharge. When mapped onto polygons, Alpha-blending simulates semi-transparent objects, such as water and glass.

animation

The process of creating the illusion of moving images by displaying sequential images in rapid succession. In each successive image, two or more values are changed over time, and the items drawn or recorded in the images appear to move.

anti-aliasing

A technique that corrects aliasing by smoothing the edges of diagonal lines on the screen. Without Anti-aliasing, diagonal lines often have a “jaggy” appearance caused by the stair-step effect of the pixels. Anti-aliasing blurs the edges of the lines.

See also [aliasing](#) on page 176.

artifact

An undesirable element or defect in motion capture data. These may occur naturally and can be eliminated in order to achieve a better-quality capture.

asset

In MotionBuilder, any element used to create animation, such as models, textures, and shaders.

attribute

See [property](#) on page 206.

Auxiliary effector

In MotionBuilder, a supplementary effector in a Control rig that corresponds to an existing IK effector. Auxiliary effectors provide additional IK control for a character's reach, and display as a cube on the corresponding IK effector.

See also [effector](#) on page 188.

Auxiliary pivot

In MotionBuilder, a sub-control that lets you translate and rotate an IK Control rig effector from a point other than its current location.

See also [pivot](#) on page 205.

B

back plate

A background image, video clip, or video feed to be displayed on the background plane in a scene.

Back plate can also be spelled as one word in other software products.

Background Color Cancellation (BCC)

A chroma key feature that senses the color of the chroma key backing and replaces it with a complementary color. As a result, the two colors cancel each other. This eliminates the halo or fringing effect surrounding the foreground object in the chroma key.

Background Color Suppression (BCS)

A chroma key feature that senses the color of the chroma key backing and replaces it with an adjustable luminance level. This prevents the backing color from appearing in the chroma key.

Background generator

A video generator that produces a solid-color output which can be adjusted for hue, chroma, and luminance.

background plane

A plane in a scene on which images, video clips, or video feeds are projected.

BaseAnimation layer

The default animation layer to which all other layers are merged when you plot an animation.

See also [layer](#) on page 197.

batch

The process of automating a frequently performed task by storing commands in a script or “batch file”.

For example, batch load refers to the process of loading or processing more than one file with a single command.

Baud rate

The bits per second (bps) rate at which the information carrying capacity of a communication channel is measured.

BCC

See [Background Color Cancellation \(BCC\)](#) on page 178.

BCS

See [Background Color Suppression \(BCS\)](#) on page 179.

bind pose

In MotionBuilder, the position in which a character is weighted, wherein all of the character's limbs should be in neutral positions, neither fully extended nor fully contracted.

Biovision Hierarchical Data (BVH)

A character animation file format that contains skeleton hierarchy information and motion data.

BVH, one of the most popular motion capture data file formats, is mainly used as a standard representation of motion capture of humanoid structures.

biped

In MotionBuilder, a humanoid skeleton that stands on two legs, making contact with the floor using only the feet.

bitplane

The memory in a graphic display device that holds a complete one-bit-per-pixel image.

Black level

The lowest transmittable luminance level that can occur during the active picture portion of a video signal. When viewed on a monitor this signal level is seen as black.

blending object

In MotionBuilder, any selection of nodes or sensors with captured data, a part or complete hierarchy of models with plotted data, or a Control rig that can be used to perform a motion blend.

bone

In MotionBuilder, the connecting lines between the joints that compose an Actor skeleton.

bound model

In MotionBuilder, a 3D model that has a rigid skeleton and is covered by a mesh. The mesh contains a texture and body features that give the model a distinct appearance.

bounding box

Rigid bodies that limit the area in which the eyes, eyebrows, and mouth of a face model can move in an Actor Face.

branch

A part of a hierarchy or tree-based data structure where there is only one route between any pair of nodes. A node on a branch has only one parent.

brightness

Along with contrast, a property that determines the luminance of an object.

buffer

An area of memory used for storing messages.

bump map

Textures that contain two direction vectors, and are used to convey relief in a texture.

See also [texture](#) on page 213.

burst

See [color burst](#) on page 185.

BVH

See [Biovision Hierarchical Data \(BVH\)](#) on page 180.

C

camera

A device for viewing and recording scenes. Each camera sees the scene from a different angle or “vantage point”.

camera interest

Also referred to as a look at point, the focal point of a camera, represented by a null.

channel

A digital effects processing path for video.

A particular signal path.

MotionBuilder uses channels to connect Actor Face assets with Character Face assets to create expressions for 3D models.

Character

See [Character asset](#) on page 183.

character animation

The process of animating objects or models to give the illusion of personality, life, and character. In contrast to other types of animation, objects are meant to appear alive and to appear to act on their own accord rather than to move randomly.

Character asset

In MotionBuilder, also referred to as the Character. The link between a [motion source](#) on page 200 (such as an [Actor](#) on page 176, a [Control rig](#) on page 186, or another character) and a [character model](#) on page 183.

See also [model](#) on page 200.

Character Face

In MotionBuilder, the shapes on a face model which can be driven with live input, recorded motion capture data, devices, and constraints.

See also [shape](#) on page 210.

Character mapping

The process of creating a link between a data source and a 3D model with a skeleton.

See also [Character asset](#) on page 183.

character model

In MotionBuilder, a 3D object composed of a skinned model with a skeleton.

You can animate a character model by linking it to a motion source via a Character asset.

See also [model](#) on page 200, [skeleton](#) on page 211, [motion source](#) on page 200, and [Character asset](#) on page 183.

child

A model or element that is placed below another in a hierarchical structure. For example: Marker2 is parented by Marker1. In the hierarchical structure, Marker2 is the child and Marker1 is the parent.

See also [hierarchy](#) on page 194 and [parent](#) on page 204.

chroma key

An effect that lets you sample out a colored background, and replace it with something else, such as a video layer.

chrominance

A portion of the video signal that contains color information (hue and saturation). Video picture information contains two components: luminance and chrominance.

See also [luminance](#) on page 198.

clip

Each individual instance of animation, audio, commands, constraints, videos, or camera shots in the Story settings.

A portion of data cut off at a defined boundary.

cluster

A collection of vertices that can be linked to objects.

cluster shapes

A shape made of cluster groups by translating, rotating, and scaling the clusters for use in the Shapes Mapping pane.

See also [Character Face](#) on page 183 and [cluster](#) on page 184.

color burst

Also referred to as a burst, a reference for establishing the picture color (hue).

color timing

The synchronization of the color burst phase of two or more video signals. Ensures that no color shifts occur in the picture when the signals are mixed in a switcher or another video device.

COM port

Also referred to as a communications port, a connector for a communications interface.

combiner

A device that controls the way two or more channels work together. It determines the priority of the channels (which picture appears in front and which ones in back) and the types of transitions that can take place between them.

See also [channel](#) on page 182.

command clip

A clip that lets you show and hide models at specific frames in your track. You can also use the Command clip to launch an external application.

communications port

See [COM port](#) on page 185.

constrained object

An object whose movement is determined by the behavior of another object, using a constraint.

See also [constraint](#) on page 186.

constraint

A restriction of the behavior of one object (constrained object) based on the behavior of another object (source object).

See also [constrained object](#) on page 186 and [source object](#) on page 211.

constraint clip

A clip that lets you select, blend, and fade constraints throughout your track in the Story window. To obtain the correct result, the Constraint track must be placed *below* the animation the constraint is meant to affect.

contrast

Along with brightness, a property that determines the luminance of an object.

Control rig

A data source that allows you to create and alter character animation using a combination of an IK rig and an FK rig.

cross chrominance

Also referred to as cross color, moire or rainbow effects in encoded video pictures created when the video encoder misinterprets luminance detail as color information. For example, moire effects on pin-striped clothing.

cross color

See [cross chrominance](#) on page 186.

current segment

The segment of optical data that is currently selected in the Optical editor. When a segment is selected and active (not set to Done in the Label pane), it is colored green.

See also [segment](#) on page 209.

custom keying group

Also referred as custom keying mode. A set of user-defined properties that define a character's effector (or an object) in 3d space which will be captured when a keyframe is created.

A custom keying group can be either: global (can be applied to any character or object in a scene); local (only ever applies to the object(s) that are selected at the time the keying group is created); or object (can be applied to any object in a scene).

cut

A section of a take's animation.

D

deck

A video cassette recorder (VCR).

deformation

A method of modeling object surfaces based on a geometric mesh of control points.

dense data

Animation that displays as many keyframes, such as motion capture data or plotted animation.

device

Any hardware instrument with a specific functionality. In MotionBuilder, you can use input devices such as a mouse, or a MIDI device.

Done state

A possible state of a marker in the optical system. When set to Done, the marker is no longer an active optical marker and cannot be used within the Optical settings. Done optical markers can be filtered and modified in the FCurves window.

dopesheet

A visual representation, similar to a traditional cell-animation timing sheet, that provides you with a way of moving keys, modifying timings, and activating and disabling effects over time.

In MotionBuilder, the Dopesheet window is an exploded view of the Action timeline in the Transport Controls window.

Distribution Factor

A slider that lets you adjust how the gradient is distributed between the Shadow and Highlight colors.

dummy node

A node that contains no geometric data that is used as a parent node.

See also [node](#) on page 201.

E

effector

The markers on a Control rig that represent a character's joints. Effectors are visually represented by the cells on the character representation in the

Character settings and can be selected to transform the character's corresponding body parts. There are two types of effectors: FK effectors and IK effectors.

See [FK](#) on page 191 and [IK](#) on page 194.

See also [Auxiliary effector](#) on page 178.

Effects send

A video switcher feature that lets you select a key source to be sent to a digital picture manipulator. The manipulated key and fill video are then returned to the Switcher's keyer for keying ("flying" a key) over background video.

element

A node. All of the objects that make up your scene.

Environment mapping

A form of reflection mapping best suited for situations when you are filming a model from a single point of view.

Expression

A mathematical formula that you use to animate properties and elements.

expressions constraint

Also referred to as Expressions, constraints created using data entered in the Expressions pane.

See also [constraint](#) on page 186.

extrapolation

The method of using a mathematical algorithm to estimate how a curve logically continues, based on the currently known values.

F

.fbx

The generic 3D data packager file format. FBX files can be unpacked, read, and used by all major 3D software packages, regardless of which package the data came from, or how it is converted.

FCurve

See [function curve](#) on page 192.

fill

In video keying, the video signal that is inserted into the “hole” cut in the background video by a key signal.

filter

A tool used to clean, manipulate, or modify captured motion data. You can use filters and filtering options to manipulate captured data according to your own specifications and to correct noisy or distorted motion capture data.

filtering

The process of cleaning, manipulating, modifying or otherwise tweaking captured motion data.

See also [filter](#) on page 190.

First Contact balloon

The interactive graphical tooltip that is displayed when the SteeringWheel is pinned during startup.

FK

See [Forward Kinematics \(FK\)](#) on page 191.

FK effector

See [effector](#) on page 188.

See also [Forward Kinematics \(FK\)](#) on page 191.

FK rig

A Forward Kinematics system that lets you control individual pivot points on a model's skeleton.

Forward Kinematics (FK)

A method of moving a hierarchy (such as a limb) in which the lower elements of the hierarchy follow the motion of parent elements. For example, if you rotate the shoulder using forward kinematics, the upper arm, forearm, hands and fingers follow.

See also [Inverse Kinematics \(IK\)](#) on page 195.

fps

Frames per second.

See [frame rate](#) on page 192.

frame

A single image at a specific point in time within an animation.

The individual picture image on a strip of film or a complete television picture made up of two fields.

A frame can be used as a unit of measurement.

frame rate

The rate at which sequences of images are captured or displayed. The frame rate is usually measured in frames per second (fps).

function curve

Also referred to as an FCurve, a graphic depiction of an animated value. The time and value of the animated value displays on two axes: the vertical axis representing the value, and the horizontal axis representing the time.

G

gap

The space before, after, or between a marker's segments that does not contain sensor data.

generic channel

A channel that is a preset facial expression.

See also [Actor Face](#) on page 176 and [channel](#) on page 182.

ghost

A wireframe representation of an unrendered blending object in the Viewer window. Ghosts only display when the Motion Blend or Story windows are open.

See also [blending object](#) on page 181.

Ghost curve

A visual representation of the original curve that displays in the FCurves window as you edit the curve.

global coordinates

Values that define a location relative to the origin of a scene, in the format (X, Y, Z).

See also [local coordinates](#) on page 197, [X-coordinate](#) on page 219, [Y-coordinate](#) on page 220, and [Z-coordinate](#) on page 220.

global keying group

Also referred to as global keying mode. A custom keying group that can be applied to any character or object in an animated scene. A global keying group describes a set of properties (for example, scale and translation properties) for a character or an object that will be captured when a keyframe is created in an animated scene. Global keying groups can be assigned to any object or character in a scene, but only the keying group's properties that map directly to properties the object or character already has defined for it will be affected.

gobo

A filter placed over a light to make it project patterns.

Guide pose

A representation of a Character pose used in Ragdoll solves when you want the character to attempt to conform to a pose. With a Guide pose, the character does not perform unrealistic contortions to ensure that the pose is assumed. Guide poses appear in the Viewer window as a green stick figure.

A Guide pose is a less-exacting version of the Match pose. See also [Match pose](#) on page 199.

H

Hardware FC

Hardware Full Control. A special data transfer protocol that controls the flow of data between specific hardware devices.

Head-up Display (HUD)

See [HUD](#) on page 194.

hierarchy

An organization structure that visually describes the relationship between elements. A hierarchy looks like an inverted tree structure, with an element at the top (referred to as a parent) and with several elements below its predecessor (referred to as children).

See also [parent](#) on page 204 and [child](#) on page 184.

HSB

Three numerical values, where “H” refers to Hue, “S” refers to Saturation, and “B” refers to Brightness.

HUD

Information that is visually relayed to the user without requiring the user to look away from the usual viewpoint.

HUD (Head-up Display) takes its name from the head-up displays used in modern aircraft. The origin of HUD stems from the users being able to view information with their heads “up” and looking forward, instead of angled down looking at lower instruments.

hue

A specific color. For example, you can use the Hue slider in the Color window to set an object’s hue to “green”.

I

IK

See [Inverse Kinematics \(IK\)](#) on page 195.

IK effector

See [effector](#) on page 188.

See also [Inverse Kinematics \(IK\)](#) on page 195.

IK rig

An Inverse Kinematics system that lets you transform hierarchies of bones using IK effectors.

See [Inverse Kinematics \(IK\)](#) on page 195.

interpolation

The process in which a computer program automatically fills in the action between keyframes with in-between frames, creating the illusion of smooth, continuous motion when the animation is played.

In MotionBuilder, in the FCurves window, the interpolation is shown by the shape of the function curve drawn between keyframes of an animation.

Inverse Kinematics (IK)

A method of transforming a group of connected joints (such as a limb) where the movement of the end joint influences all the preceding joints in the chain.

For example, when you transform the wrist joint of an arm, the elbow and shoulder joints are also transformed.

See also [Forward Kinematics \(FK\)](#) on page 191.

IP address

The 32-bit host address defined by the Internet Protocol (IP).

J

jogging

The action of smoothly moving forward and backward through time in a take by J-clicking and dragging in the Viewer window.

joint

The points on a skeleton connected by bones.

K

key

The process of setting a keyframe.

keyframe

A reference point, or key point, that marks the position of an important action or change in a scene at a specific point in time.

keyframing

The action of creating keyframe animation by transforming an object in a scene at a specific point in time and setting a keyframe.

keying group

Also referred to as keying mode. A set of properties for a character's effector or for an object recorded when you create keyframes.

L

latency

The time during which the read/write heads wait for data to rotate into position after the controller starts looking for a particular data track.

layer

A level of animation in a scene, on top of the original function curve data. You can have multiple layers in a scene and make changes to one layer without affecting the others.

See also [BaseAnimation layer](#) on page 179.

Linear key

A luminance key effect in which the gain of the key is approximately one. This preserves the shaping of the key source edges produced by anti-aliased character generators and digital video effects devices.

See also [luminance key](#) on page 198.

local blend

The process of replacing the motion in one track with motion from another track on only part of a hierarchy.

local coordinates

Values that define a location relative to the origin of a selected object, in the format (X, Y, Z).

See also [global coordinates](#) on page 193, [X-coordinate](#) on page 219, [Y-coordinate](#) on page 220, and [Z-coordinate](#) on page 220.

local keying group

A custom keying group that describes the set of properties for an object in 3d space that will be captured when a keyframe is created for that object in an animated scene. Local keying groups can only be assigned to the object(s) selected at the time the keying group was created.

look at point

See [camera interest](#) on page 182.

loop

The area of a take that is designated to continuously play when you click Play in the Transport Controls.

luminance

The luminous intensity of a video signal. The color picture information contains two components: luminance (brightness and contrast) and chrominance (hue and saturation). Probably should remove reference to luminance within definition.

See also [chrominance](#) on page 184.

luminance key

A key effect in which the portions of a key source that are greater in luminance than the clip level cut a hole, or key, in the background video.

M

magnetic mapping

The process of mapping magnetic motion data to an Actor.

marker

Objects used to identify segments. One or more segments, after being labelled or identified, combine to create a marker of continuous data.

In the Optical tool, marker is another term for sensor.

See also [sensor](#) on page 209.

Marker set

A set of markers that map objects containing motion data (such as magnetic markers or optical sensors) to an Actor. This association is then used to drive the Actor. In the Viewer window, a Marker set displays as a group of white markers attached to an Actor.

Match pose

A representation of a Character pose used in Ragdoll solves as a goal stance that the Character must assume during a specified time. Match poses appear in the Viewer window as a red stick figure.

A Match pose is more definitive than a Guide pose. See also [Guide pose](#) on page 193.

material

Material assets are a set of properties that let you change a model's color and simulate how the model's surface reacts to light. These properties may include color, shininess, transparency, reflectivity, and bump. Use materials with textures and shaders for the best results.

Mipmap

A version of an original texture that has been reduced in size to 1 x 1 pixel. This solves the problem of textures with small objects "flickering" as the viewer gets further away.

See also [texture](#) on page 213.

model

The mathematical description of a three dimensional object that is placed in a scene.

moire

A wavy pattern.

morph target

Operators for use with models that have shape animations, also referred to as shape operators.

motion capture

A method of collecting motion data based on the movement of a performer wearing special sensors or markers.

motion source

An asset such as an Actor, character, or Control rig that is linked to a character model through the Character asset to drive the movement of a character model.

N

Namespace

A namespace is a unique path. Each item in a namespace is identified by its own name along with the namespace to which it belongs.

For example, Moon:Alien is a path, but Galaxy:Moon:Alien is a separate path because it does not reference Moon:Alien in the Galaxy but it does increments Moon. Moon:Alien and Galaxy:Moon:Alien are two separate and cooperative elements that do not require renaming.

naming template

An *.fbx* file containing the customized naming conventions used to define a skeleton.

National Television System Committee (NTSC)

See [NTSC](#) on page 202.

node

The individual objects (such as joints, bones, or nulls) that are linked to a model's skeleton structure. Nodes allow you to map between a source and a model.

In the Schematic view, the variously colored tiles that visually represent each asset of a hierarchy.

See also [reference node](#) on page 207 and [dummy node](#) on page 188.

noise

Irregular jumps in a segment of optical data caused by partial occlusion of a sensor on a performer's body during a capture session.

Noticeable distortion in magnetic capture data caused by metallic objects such as aluminium heating ducts interfering with the capture session area.

Non Uniform Rational B-splines (NURBS)

See [NURBS](#) on page 202.

normal

A perpendicular or vector that defines the orientation of something.

normal map

Textures that contain three direction vectors: an X, Y and Z. Unlike a bump map's two vectors, the normal map's three vectors convey height and lighting detail with greater precision, providing heightened realism.

See also [bump map](#) on page 182 and [texture](#) on page 213.

NTSC

Stands for the National Television System Committee (NTSC) as well as for the standard for color television in the United States and other countries established by this Committee.

NTSC is defined by the frame size, a frame rate of 29.97 fps, as well as by the frame aspect ratio and pixel aspect ratio. Although there are various divisions within the NTSC standard format which determine what frame size is used and what pixel and frame aspect ratios are used, the standard frame aspect ratio used by the NTSC standard format is 4:3 (1.333). This format uses a 640 by 480 resolution.

null

An object that you can parent to other objects for additional transformation flexibility. nulls have no specific properties and are simply used to help you build your scene. In the Viewer window, a null is visually represented as a small axis.

NURBS

Non Uniform Rational B-splines. Surfaces and curves that visually represent complex geometric information, used for modeling.

O

object keying group

A custom keying group for objects only (not characters) that describes a user-defined set of properties for object in 3d space in an animated scene when

a keyframe is set. Object keying groups can be assigned to any object in a scene (as opposed to “local keying groups” which are restricted to the object(s) selected at the time the local keying group is created.).

occlusion

A problem with optical motion capture, wherein a sensor is hidden from all but three cameras. This may occur when a performer passes by an obstructing object, or when the performer’s body comes between the sensor and the camera.

opacity

The extent to which an object is transparent. If an object’s opacity is set to 100%, the object displays opaque. If the opacity is set to 0%, the object displays transparent.

OpenGL

A software interface for graphics hardware that supports rendering and imaging operations.

Optical editor

An editor that lets you correct optical data, fix poor gap interpolation, switch swapped markers, and perform other optical data reconstruction.

See also [noise](#) on page 201, [occlusion](#) on page 203, and [partial occlusion](#) on page 205.

optical mapping

The process of mapping optical motion data to an Actor.

Optical root

The main reference for imported optical data, represented in the Viewer window by a sphere.

origin

The point at the center of a 3D scene relative to which every location is defined. At the origin of a scene, the X, Y, and Z coordinates have a value of zero.

The center or reference point of a selected 3D object, relative to which the surface of the object is defined. At the origin of an object, all three 3D coordinate values have a value of zero, written as (0,0,0).

See also [global coordinates](#) on page 193 and [local coordinates](#) on page 197.

P

PAL

Stands for Phase Alternating Line —a standard for color television used in many European, African, and Asian countries.

PAL is defined by the frame size, a frame rate of 25 fps, as well as by the frame aspect ratio and pixel aspect ratio. Although there are various divisions within the PAL standard format which determine what frame size is used and what pixel and frame aspect ratios are used, the standard PAL video signal format sets the video to playback at 25 frames per second which contain 625 lines of pixels in each frame.

parameter

See [property](#) on page 206.

parent

A model or element that has been made the parent of another. For example, in the hierarchical structure, Marker1 is the child and Marker2 is the parent.

See also [hierarchy](#) on page 194 and [child](#) on page 184.

parenting

The act of making one model or element the parent of another.

partial occlusion

A problem with optical motion data that often occurs if a sensor on a performer's body has been placed too close to another sensor, or the sensor becomes partially hidden from one of the cameras during the capture session. The resulting data may display peaks, shifts, or noise.

patch

A type of tessellation, something to do with a model's surface.

Phase Alternating Line (PAL)

See [PAL](#) on page 204.

pitch

A rotation based on the movement up or down the Y-axis. For example an airplane banking up or down.

pivot

The point from which a selected object is transformed.

See also [Auxiliary pivot](#) on page 178.

pole vector

A part of an IK rotate plane handle that begins at the start joint, and along with the handle vector defines the IK handle's reference plane.

The pole vector changes the orientation of the reference plane, so you can change the orientation of the joint chain directly. This is because the joint chain's degree of orientation, or twist, is defined as the difference in orientation between the reference plane and the joint chain plane.

Also known as “up-vector” in other software packages.

pose

A snapshot in time of a selected character or object's position.

property

Also referred to as an attribute or parameter, a value that quantifies a specific characteristic of an object, and can be animated. For example, the fog intensity of a light is a property.

Q

Qt

A cross-platform application development framework, widely used for the development of GUI programs (in which case it is known as a widget toolkit), and also used for developing non-GUI programs such as console tools and servers.

Qt uses C++ with several non-standard extensions implemented by an additional pre-processor that generates standard C++ code before compilation. Qt runs on all major platforms, and has extensive internationalization support. Non-GUI features include SQL database access, XML parsing, thread management, network support and a unified cross-platform API for file handling.

Qt is most notably used in KDE, Opera, Google Earth, Skype, Qtopia, Photoshop Elements, VirtualBox and OPIE. Qt can also be used in several other programming languages; bindings exist for Ada (QtAda),[3] C# (Qyoto/Kimono),[4] Java (Qt Jambi),[5] Pascal, Perl, PHP (PHP-Qt), Ruby (RubyQt), and Python (PyQt).

quadruped

A four-legged skeleton that makes contact with the floor using all four limbs.

quaternion

A complex number made up of four geometric components.

A quaternion adds a fourth element to the [x, y, z] values that define a three-component-vector. A quaternion represents an axis in 3D space and a rotation around that axis.

R

reference node

A null or joint that acts as the root of an entire model and is the parent of the models' Hips.

Relations

See [relational constraint](#) on page 207.

relational constraint

Also referred to as Relations, constraints that perform custom operations on the data of a source object to determine the behavior of the constrained object.

See also [constraint](#) on page 186.

remote port

An additional port provided by a serial device where you can physically connect input and output devices.

render

To generate an image file, a sequence of image files, or movie file(s) using the mathematical descriptions of the objects that compose the scene.

rest pose

The default position at which a Character Face asset is at rest.

retargeting

The process of taking the animation data mapped to one character, applying that animation to another character to drive its animation without having the need to plot (or bake) the animation.

Rigid body

Two or more markers grouped to correct occlusion.

roll

A rotation around the X-axis. For example, the rolling of a log.

rotation

The process of changing all the points on an object to reflect the degree of rotation around each of the three axes.

S

sample

The position of a sensor recorded by each camera at each frame. All samples from each camera generate a three-dimensional representation of each sensor's position in time when processed.

See also [keyframe](#) on page 196.

saturation

A property that helps determine the chrominance and contrast of the color of an object.

scaling

The process of enlarging an object by moving all the points outward from the object's center, or shrinking it by drawing them all in toward that center.

scene

A representation of a three-dimensional world in which objects are placed and animated using a coordinate system.

See also [3D coordinate space](#) on page 176.

scrubbing

The process of moving through an audio track either forward or backward, while the audio is playing. This process is used to find and hear the audio at a specific frame.

segment

The data captured from an optical motion capture session.

See also [current segment](#) on page 187, and [motion capture](#) on page 200.

sensor

A reflector or light source attached to a performer's body. Sensors are tracked by optical cameras during the capture process. Captured data is combined to create segments.

See also [motion capture](#) on page 200 and [segment](#) on page 209.

serial port

Also referred to as a COM port, a port that uses a special communication protocol to control the flow of data between devices, allowing the transfer to be made at a higher speed.

shader

In 3D modeling, the term “shader” is used to describe an algorithm that specifies how a surface responds to light. In MotionBuilder, “shaders” are assets that you apply to models to create different types of effects. Various types of shaders can define how a surface responds to light, or create special effects like cartoon outlines and bump mapping. You can also use shaders to generate shadow maps and light maps for scenes with a static light source. MotionBuilder also supports custom shaders created using the Open Reality® SDK.

Shadow map

A *.tiff* image of the shadow created by the Shadow Map shader, projected onto planes and objects in a scene.

shape

In the Character Face settings, a Character Face modified to portray a particular expression, such as “angry”.

See also [Character Face](#) on page 183.

shape operators

Also referred to as morph targets, special operators for use with models that have shape animations.

shuttling

The action of fast-forwarding or rewinding through an audio track while the audio is playing.

simple constraint

Constraints that use a pre-defined list of constrained objects and source objects.

skeleton

A set of points representing the joints, and of connecting lines representing the bones.

skin

The mesh of vertices that envelopes a 3D character, creating its shape.

SMPTE

Refers to the Society of Motion Picture and Television Engineers (SMPTE) founded in 1916 to advance theory and development in the motion imaging field.

solving

The process of calculating the position of both the forward kinematics and inverse kinematics rigs, then applying these results to the linked model while observing the settings in the Character Settings pane.

The results of calculating each rig. For example, IK solving refers to the result of calculating the IK rig.

In MotionBuilder, the result of all calculations, rigs, and settings when using the character engine.

source object

An object on which a constraint is based.

See also [constraint](#) on page 186.

Sphere map

A reflection type that causes a 3D object to reflect the contents of its scene from only one point of view.

Spherical map

A reflection type that causes a 3D object to reflect the contents of its scene.

spline

A curve that is defined using control points.

stabilizing object

A sensor, a node from a skeleton, or a model from a hierarchy of models that stabilizes the entire blending object and corrects problems such as foot sliding.

See also [blending object](#) on page 181.

stack

A data structure for storing items which are to be accessed in last-in first-out order.

stance pose

The starting or rest pose of a model.

SteeringWheels

Tool set that provides access to 2D and 3D navigation tools.

subcarrier

Also referred to as the SC, in NTSC or PAL video, a continuous sine wave of extremely accurate frequency which constitutes a portion of the video signal. The subcarrier carries picture hue and color saturation information.

swapping

A problem with optical motion capture wherein two markers either cross or pass close to each other, causing the capture system to misinterpret the markers and label the segments incorrectly.

See also [motion capture](#) on page 200.

T

take

A snapshot in time of an animation instance.

NOTE Time can be measured in hours, minutes, seconds, and frames, or in frames per second.

tangent handle

The visual representation of the tangent of a keyframe on a function curve. Tangent handles let you change the slope of the curve on either side of the keyframe.

TCB

Tension, Continuity, Bias.

tessellation

A step in the rendering process in which the shapes of an object's surface mesh are rearranged into triangles.

texture

Various texture assets let you apply image or video files to materials, layered textures, and models. Textures are most commonly used to add detail to a

model's surface without adding to the model's geometry. Use textures with materials and shaders.

timecode

The value that indicates the current position in time of the current take.

Timewarp

A curve that alters the shape of a function curve and changes the timing of an animation.

See also [function curve](#) on page 192.

track

A course along which something moves, or a sequence of events through time.

In MotionBuilder, there are two types of tracks. In the Motion Blend window, tracks can contain motion data and let you blend takes into a single result track. In the Story window, tracks can contain motion data, keyframe animation, audio and video, and let you blend specific types of clips along the timeline.

transformation

The process of changing the points on an object by translation, rotation, and scaling.

translation

The process of moving an object on one or all axes. Translation moves an object without changing its orientation.

transparency

The level of visibility of a object, determined by the opacity setting. When the opacity is set to 0%, the object is transparent.

trigger

In the Animation Trigger window, a device, such as a joystick or keyboard, that allows you to execute motion clips to test the transitions you created between them.

triggering group

A collection of motion clips and the triggers that initiate their execution.

T-stance

The neutral pose of a biped character similar to a “T” pose in which the arms are at a 90-degree angle from the body, the legs are fully extended, the feet are flat on the floor, and the spine is straight.

U

UCS

See [User Coordinate System \(UCS\)](#) on page 216.

See also [World Coordinate System \(WCS\)](#) on page 219.

unlabelled segment

A segment of data that has not yet been labelled to associate it with a marker. This is done during optical cleaning. Unlabelled segments display as blue asterisks.

unweighted tangent

A tangent on a function curve that does not have weight applied to it.

See also [function curve](#) on page 192 and [weighted tangent](#) on page 218.

up-vector

See [pole vector](#) on page 205.

user channels

Custom channels you can create in the Character Face settings.

See also [channel](#) on page 182 and [Character Face](#) on page 183.

User Coordinate System (UCS)

A user-defined coordinate system that defines the orientation of the X, Y, and Z axes in 3D space. The UCS determines the default placement of geometry in a drawing. See also world coordinate system (WCS).

See also [World Coordinate System \(WCS\)](#) on page 219.

UV

U and V texture coordinates. U and V represent vectors in a 1 x 1 pixel image that connects to places on a 3D mesh. The U and V coordinates let you place the texture on the 3D mesh exactly. This placement attaches the texture to the object's surface, and it is mapped to create a seamless effect.

V

value

A number that defines anything from the position, rotation, or scaling of a model, to a material's emissive, ambient, or diffuse color values.

See also [property](#) on page 206.

vector

A straight line segment.

ViewCube

User interface element that displays the current orientation of a model. You can use it to restore and create a UCS, interactively rotate the view, or restore a preset view.

visual keyframe

Arrowhead-shaped tabs that display on the Action timeline and on the timeline in the Dopesheet window to indicate the location of keyframes that are set.

See also [keyframe](#) on page 196.

voice channels

In the Character Face settings, channels that are preset mouth expressions.

See also [channel](#) on page 182.

VK ripple

Visual keyframes ripple.

See also [visual keyframe](#) on page 217.

VTR

Video Tape Recorder.

W

waveform

A graphical depiction of the continuous fluctuation in the amplitude of a sound over time.

WCS

See [World Coordinate System \(WCS\)](#) on page 219.

See also [User Coordinate System \(UCS\)](#) on page 216.

weighted tangent

A tangent on a function curve that has weight applied, letting you stretch the tangent handle and create special curves that you cannot create with unweighted tangents.

See also [function curve](#) on page 192 and [unweighted tangent](#) on page 215.

wheel

A reference to one of the individual user interface elements that make up SteeringWheels.

See also [SteeringWheels](#) on page 212.

wheel surface

Area of a wheel that is used to organize wedges and other buttons.

wheel wedge

A section on the surface of a wheel that is designated for a specific 2D or 3D navigation tool.

wheels

A reference to more than one of the individual user interface elements that make up SteeringWheels.

See also [SteeringWheels](#) on page 212.

wireframe

A manner of displaying objects, such as ghosts.

World Coordinate System (WCS)

A coordinate system used as the basis for defining all objects and other coordinate systems.

See also [User Coordinate System \(UCS\)](#) on page 216.

world coordinates

Coordinates expressed in relation to the [World Coordinate System \(WCS\)](#) on page 219.

X

X-axis

The dimension on which coordinates define the horizontal space of the scene.

See also [3D coordinate space](#) on page 176.

X-coordinate

The value that defines the horizontal space in the scene relative to the origin.

Y

Y-axis

The dimension on which coordinates define the vertical space of the scene.

See also [3D coordinate space](#) on page 176.

Y-coordinate

The value that defines the vertical space in the scene relative to the origin.

yaw

A rotation based on spinning an object using its center as the axis. For example, a record on a turntable.

Z

Z-axis

The dimension on which coordinates define the depth of the scene.

See also [3D coordinate space](#) on page 176.

Z-coordinate

The value that defines depth in the scene, relative to the origin.

Zero keyframe

A keyframe in which the effect of a layer is set to zero at a given time.

A keyframe set to define the start or end of an animation.

See also [layer](#) on page 197.

zero point

The stance of an Actor where both translation and rotation are set to zero.