Contents

Returning to the Reference Pose ........................................ 79
Selecting All the Deformers ........................................... 80
Removing Envelopes ...................................................... 80
Modifying Point Assignment Using Bounding Volumes ........... 81
Bounding Groups ............................................................. 81
Creating Bounding Volumes ............................................. 83
Associating and Dissociating Bounding Volumes ................. 83
Combining Bounding Volumes ............................................ 86
Changing Deformer Assignments Manually ......................... 88
Reassigning Points to Specific Deformers ......................... 88
Removing Deformers for Specific Points ............................ 89
Modifying Envelope Weights ............................................ 90
Tools for Modifying Envelope Weights ............................... 90
Painting Envelope Weights ............................................. 93
Editing Weights Numerically .......................................... 97
Smoothing Weights ....................................................... 100
Mirroring Envelope Weights Symmetrically ....................... 100
Freezing Envelope Weights ............................................. 103
Modifying Display Colors .............................................. 104
Using Envelope Presets ................................................... 105
Changing Reference Poses .............................................. 106
Adding and Removing Deformers ..................................... 107
Adding Deformers ............................................................ 107
Removing Deformers ...................................................... 108
Modifying Enveloped Objects .......................................... 109
Deforming and Moving Points on Envelopes ....................... 109
Transforming Envelopes ................................................... 110
Enveloping Tips & Tricks ................................................ 111
X-Ray Mode ................................................................. 111
Muting Envelopes ......................................................... 112
Combining Envelopes with Soft Body Deformations ........... 112
Using Volume Deformations with Envelopes ...................... 112
Importing Envelopes from SOFTIMAGE|3D ....................... 112

Chapter 4

Animating Skeletons ...................................................... 113
Animating Skeletons with Kinematics ............................... 114
Forward and Inverse Kinematics ..................................... 114
Animating with Forward Kinematics ............................... 116
When to Use Forward Kinematics ................................... 116
Animating Bone Rotations ............................................. 118
Aligning the Root's Orientation to Match the First Bone ...... 119
Changing the Bone's Axis Alignment ................................. 120
Animating with Inverse Kinematics ................................. 121
When to Use Inverse Kinematics ................................... 121
Basic Concepts for Inverse Kinematics ............................. 122
Contents

Animating the Effector .................................................. 123
Controlling an IK Chain’s Orientation ............................ 126
Blending Between FK and IK Animation ....................... 127
Blending and Constraints ............................................. 129
Blending IK with Motion Capture Data (FK) .................... 129
Ghosting FK/IK Blending ............................................. 130
Choosing How an IK Chain Reaches Its Goal ............... 132
Choosing a Solver ....................................................... 132
Calculating Joint Rotations with the Solver Angle ........ 133
Forcing the Effector’s Position .................................... 133
Setting the Solver’s Error Threshold ............................ 134
Setting and Blending the Effector’s Orientation ............. 135
Setting the Effector’s Orientation Inheritance .............. 135
Blending and Linking the Effector’s Orientation .......... 136
Changing the Joint’s Preferred Angle ......................... 137
Resetting Bones ......................................................... 137
Constraining the Chain’s Up Vector (Direction) .......... 138
Constraining the Up Vector to an Object ..................... 139
Constraining the Up Vector without an Object ............. 140
Setting the Roll to Correct the Orientation .................. 141
Removing Up Vector Constraints .................................. 141
Rolling an IK Bone ...................................................... 142
Constraining the Preferred Axis of Rotation ............... 144
Setting Restrictions on Movement ............................... 146
Displaying Rotation Limits and Stiffness ..................... 146
Setting Rotation Limits ............................................. 146
Setting Joint Stiffness .............................................. 149
Creating Subchains for Independent Movement .......... 150
Defining a Joint’s Behavior ........................................... 150
Making a Joint into a Pseudo-root ............................... 151
Modifying a Biped’s Motion on Tracks ....................... 154
Getting Set Up .......................................................... 155
Creating the Motion Tracks ......................................... 155
Importing Motion Capture Files ................................. 158
Importing Acclaim Files ........................................... 158
Importing Biovision Files .......................................... 159
Setting Up the Skeleton .............................................. 160
Setting Other Import Options ..................................... 162

Chapter 5

Character Rigging ......................................................... 163
Basics for Setting Up a Control Rig ............................. 164
Tips for Rigging .......................................................... 165
Displaying Chain Elements and Their Shadows .......... 166
Using the Character Development Kit ....................... 167
Getting Started with Predefined Rigs .......................... 168
Loading Biped Rigs ........................................... 168
Loading a Quadruped Rig ....................................... 168
Setting Up the Biped or Biped Dog Leg Guides ............... 169
Loading the Biped Guides ..................................... 169
Duplicating or Deleting Objects in the Guide .................. 171
Creating a Biped or Biped Dog Leg Rig ........................ 173
Features to Help You Use the Rig ............................... 174
Creating the Rig .................................................. 176
Setting Up the Torso Spine ...................................... 178
Creating a Belly Control ...................................... 179
Setting Up the Head, Neck, and Ears ......................... 180
Setting Up the Arms and Fingers ............................... 181
Setting the Roll Division ....................................... 182
Selecting Volume Controls for the Skin (Envelope) .......... 183
Creating Shadow Rigs ......................................... 184
Using the Biped or Biped Dog Leg Controls ................... 185
Tips for Using the Rigs ......................................... 186
Creating Quadruped Rigs ....................................... 190
Setting Up the Quadruped Guide ............................... 190
Creating a Quadruped Rig ..................................... 191
Using the Quadruped Controls ................................ 196
Creating Control Splines for Facial Rigs .................... 197
Creating Hierarchy Setups from a Guide ..................... 199
Controlling the Binding between Models ...................... 201

Chapter 6
Tools for Character Animation .................................. 203
Making Character Animation Easier ............................ 204
All-Purpose Tools for Increasing Productivity ................ 205
Custom Toolbars ............................................... 205
Synoptic Views ................................................. 205
Organizing and Protecting a Character’s Elements ............ 206
Layers .............................................................. 206
Groups ............................................................. 206
Using Consistent Naming Conventions ....................... 206
Locking Scene Elements ...................................... 207
Tools for Easy Animation ....................................... 208
Ghosting Animation ............................................ 208
Setting a Neutral Pose with Zero Values ..................... 208
Rotoscopy for Making Models and Animating ................ 209
Marked Parameter “Memory Sets” ............................ 209
Preferred Transformation Tool (Transform Setup) .......... 210
Creating Custom Sliders ...................................... 210
Creating Your Own Property Editor with Proxy Parameters 210
Contents

Using the Animation Mixer for Character Animation ................. 212
The Importance of Models ........................................ 212
Editing Existing Material .......................................... 213
Blocking Out Rough Animation ................................. 213
Using the Mixer and Fcurves ................................. 213
Relational Views with the Mixer and Animation Editors ...... 214

Index ................................................................. 215
Roadmap
Where to Find Information

The SOFTIMAGE®|XSI® package includes a comprehensive set of learning and reference materials. Use this Roadmap to find the information you need to get up and running quickly and effectively.

Start with the Setup & Licensing guide to install, configure and license all components.

Familiarize yourself with the Limitations & Workarounds for this version, at www.softimage.com/support.

Explore The Principles of XSI—a collection of video clips on DVD-ROM providing overviews of the core features, tools and workflows in XSI. New features for this release are also showcased here.

If you are new to XSI, work through the Tutorials to learn XSI in the context of basic productions. The tutorials provide a full-color set of lessons showing you how to perform typical tasks step-by-step. You can install the scenes from the Media CD.
XSI Online Guides
A complete online library of HTML-based procedural guides on how to use the XSI toolsets (you can install these guides from the Documentation CD):
- Fundamentals
- Animation
- Character Animation
- Nonlinear Animation
- Modeling & Deformations
- Shaders, Lights & Cameras
- Rendering & Compositing
- Simulation
- Customization

Online Help
On-screen reference information on interface elements, commands, and parameters.
To access:
- Click the ? button in any property editor or data view that displays this symbol.
- Choose the Help > Contents and Index menu.

Documentation CD contains:
- Fully updated digital versions of XSI, XSI SDK, BatchServe, and mental ray® documentation.
- Browse documentation on CD, print from PDF, or install XSI user guides on your system and access them in XSI from the Help > Online Guides menu.

www.softimage.com
Visit our web site for tutorials and more. Use the embedded Net View to surf XSI Local for samples and tools, or surf XSI Net for scripts, add-ons, and other resources.

XSI SDK Online Guides
A complete online library of HTML-based Developer and Reference guides on how to use the SOFTIMAGE|XSI SDK toolset. To access: open the script editor and click ?.
Context-sensitive help on all scripting commands, objects, methods and properties is available through the online Scripting Reference. To access: open the script editor and choose Help > Scripting Reference or click F1 while you are working in the editing pane.
Document Conventions

The following are ways that information is displayed in the XSI documentation.

**Typography Conventions**

<table>
<thead>
<tr>
<th>Type style</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bold</td>
<td>Menu commands, dialog-box and property-editor options, and file and directory names.</td>
</tr>
<tr>
<td>Italic</td>
<td>Definitions and emphasis.</td>
</tr>
<tr>
<td>Courier</td>
<td>Text that you must type exactly as it appears. For example, if you are asked to type <code>mkdir style</code>, you would type these characters and the spacing between words exactly as they are appear in this book.</td>
</tr>
<tr>
<td>&gt;</td>
<td>The arrow (&gt;) indicates menu commands (and subcommands) in the order that you choose them: <em>Menu name &gt; Command name</em>. For example, when you see <em>File &gt; Open</em>, it means to open the <em>File</em> menu and then choose the <em>Open</em> command.</td>
</tr>
</tbody>
</table>

**Visual Identifiers**

These icons help identify certain types of information:

- **Notes** are used for information that is an aside to the text. Notes are reminders or important information.
- **Tips** are useful tidbits of information, workarounds, and shortcuts that you might find helpful in a particular situation.
- **The 3D icon** indicates information about differences in workflow or concepts between SOFTIMAGE|3D and XSI.
- **Warnings** are used when you can lose or damage information, such as deleting data or not being able to easily undo an action. Warnings always appear before you are about to do such an action!

**Keyboard and Mouse Conventions**

XSI uses a three-button mouse for most operations. These are referred to as the *left*, *middle*, and *right* mouse buttons. In many cases, you will use the different buttons to perform different operations; always use the left mouse button unless otherwise stated.

- The two-button mouse is not supported in XSI.
- If you are using a left-handed mouse or other pointing device, substitute *primary button for left button*, and so on.
This table shows the terms relating to the mouse and keyboard.

<table>
<thead>
<tr>
<th>When this term is used...</th>
<th>...it means this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click</td>
<td>Quickly press and release the left mouse button. Always use the left mouse button unless otherwise stated.</td>
</tr>
<tr>
<td>Middle-click</td>
<td>Quickly press and release the middle mouse button of a three-button mouse.</td>
</tr>
<tr>
<td>Right-click</td>
<td>Quickly press and release the right mouse button.</td>
</tr>
<tr>
<td>Double-click</td>
<td>Quickly click the left mouse button twice.</td>
</tr>
<tr>
<td>Shift–click, Ctrl–click, Alt+click</td>
<td>Hold down the Shift, Ctrl, or Alt key as you click a mouse button.</td>
</tr>
<tr>
<td>Drag</td>
<td>Hold down the left mouse button as you move the mouse.</td>
</tr>
<tr>
<td>Alt+key, Ctrl+key, Shift+key</td>
<td>Hold down the first key as you press the second key. For example, “Press Alt+Enter” means to hold down the Alt key as you press the Enter key.</td>
</tr>
</tbody>
</table>

**Comments?**

If you have anything you’d like to say or ask about the XSI documentation, please do not hesitate to e-mail us at education@softimage.com.
Softimage Customer Service

Technical support in North America for SOFTIMAGE|XSI is provided directly from Softimage Customer Service. Immediate assistance for any technical issue is available through hotline, and through electronic and web support services.

Softimage resellers working together with Softimage Customer Service provide support worldwide.

Licensing Support

You can request your new license file through the license request form provided at softimage.com/licensing or by contacting your reseller directly.

SDK Support

Support for the XSI SDK is separate from product support for XSI. The SDK support program provides a private web-based discussion forum where you can talk to fellow developers and submit support questions through our web-based technical support form. The web also provides answers to frequently asked questions, source code examples and case studies. Late-breaking SDK information, workarounds, tips and techniques can also be found in the knowledge base.

Training Support

If you’re interested in SOFTIMAGE|XSI training, you’ll find a complete overview of courses, education centers, and training programs at softimage.com/education.

Hotline Support

If you’ve purchased a maintenance contract to receive support directly from your Softimage reseller, you’ll find assistance for contacting your reseller at softimage.com/Corporate/Sales/Buy/ In all other cases, contact Softimage Customer Service at the following numbers and during these hours:

**World Wide**

Tel: 1 (514) 845-2199
Fax: 1 (514) 845-8252

**North America**

Tel: 1 800 387 2559
Fax: 1 (514) 845-8252

**UK and International**

Tel: + 44 1 753 650 670

<table>
<thead>
<tr>
<th>Area</th>
<th>Tel</th>
<th>Fax</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>World Wide</td>
<td>1 (514) 845-2199</td>
<td>1 (514) 845-8252</td>
<td>9 AM to 7 PM (EST)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2 PM to 12 AM (GMT)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1400 to 2400 (UTC)</td>
</tr>
<tr>
<td>North America</td>
<td>1 800 387 2559</td>
<td></td>
<td>9 AM to 7 PM (EST)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2 PM to 12 AM (GMT)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1400 to 2400 (UTC)</td>
</tr>
<tr>
<td>UK and International</td>
<td>+ 44 1 753 650 670</td>
<td></td>
<td>9 AM to 6 PM (GMT)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0900 to 1800 (UTC)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4 AM to 1 PM (EST)</td>
</tr>
</tbody>
</table>
Electronic Support

If you have an active maintenance contract and would like to reach us electronically, you will find a Support Request form at http://www.softimage.com/Support/XSI/TechnicalSupport/supportrequestform.asp.

For general inquiries about Softimage support services, send an email to support@softimage.com.

Web Support

The Support and Download sections of softimage.com provide quick access to a wide range of resources from the XSI teams and user community. Downloads—including Presets, Scripts, and Quick Fix Engineering (QFEs)—provide the latest solutions for XSI. Online user guides, tutorials, and knowledge-base articles ensure you get the most out of working with XSI. It's like having a dedicated Softimage Customer Service engineer sitting at your desk!

Mailing Lists

If you have an e-mail account, you can join the worldwide network of XSI users exchanging ideas. To subscribe to the XSI discussion group, send an e-mail to majordomo@softimage.com with subscribe XSI as the body of your message. Use the message lists for information on related groups. Visit softimage.com/Archives/Help.htm for more information about our discussion groups and mail server.

The discussion groups are provided for technical exchanges between customers. Although customer service is not provided through these discussions, we do contribute.

Student/Teacher Program Support

If you’ve purchased SOFTIMAGE|XSI through the Student/Teacher program, installation support is offered by Softimage (see the support contact information above). For usage support, you are welcome to join the XSI mailing lists.
Softimage Customer Service Addresses

North America
Softimage Customer Service
3510 Saint-Laurent Boulevard
Montreal, Quebec
H2X 2V2 Canada

UK and International
Softimage Customer Service Europe
Pinewood Studios
Pinewood Road
Iver Heath, Buckinghamshire
SL0 0NH United Kingdom
Chapter 1  Character Animation Basics
What Is Character Animation All About?

Character animation in XSI is all about bringing your characters to life. Even though you’re working in a virtual environment, your job is to make characters seem believable in their movements and expression. In XSI, you’ll find everything you need: from envelopes and skeletons to control rigs and weight painting, it’s all there.

For tutorials on building and animating a character from scratch, see the Character Animation lessons in the Tutorials guide.

For information about some general tools that are useful for working with character animation in XSI, see Chapter 6: Tools for Character Animation on page 203.

Creating a Skin (Envelope)

Skeletons are usually covered with an enveloping model or object (generically referred to as an envelope), which can be deformed (stretched, twisted, or compressed), depending on how the skeleton components are moved, rotated, or scaled. You usually create an enveloping model first, then create the skeleton to fit inside it.

Constructing Skeletons

A skeleton provides a framework for a model and can be used to pose or deform it intuitively, just as the human skeleton provides a framework for the body. The structure of your skeleton determines every aspect of how you move, from picking up a pencil to running. Similarly, a XSI skeleton provides an interior framework for a model, and how you build a skeleton will determine how your model will move when animated.

For more information about skeletons, see Chapter 2: Building Skeletons on page 29.
Applying the Envelope to the Skeleton

After you've created the skin and a skeleton to fit inside it, you can apply the skin to the skeleton. Applying the skin (called enveloping) also involves setting how the different parts of the body are weighted to the different bones in the skeleton.

For more information about envelopes, see Chapter 3: Envelopes on page 69.

Making It Move

Skeleton chains are manipulated using inverse kinematics (IK) and forward kinematics (FK). You can use either method on a chain at any time, and easily switch between the two or even blend them on the same chain.

- IK is a goal-oriented way of animating; you move the effector (end) of a chain in place and XSI calculates how the chain reaches that goal.
- FK allows for complete control of the chain's behavior because you rotate each joint's position. Positioning a skeleton's hand to reach an apple means rotating each joint in the arm, from the shoulder to the wrist and fingers.

For more information about animating the skeleton, see Chapter 4: Animating Skeletons on page 113.

Rigging It Up

Part of the process of setting up a character is often to create a control rig. These rigs help you to pose and animate a character more quickly and accurately than without one. While simple characters may not require a rig, a character that is complex or needs to do complicated movements will probably need a rig.

Luckily, XSI has a number of tools to help you create rigs for biped, biped dog leg, and quadruped characters. You can load rig guides that you adjust to match the proportion of your character, then create a rig based on those proportions. There are also tools to help you easily create shadows rigs and manage the constraints between rigs and their shadows rigs.

For more information about creating control rigs, see Chapter 5: Character Rigging on page 163.
Chapter 1  Character Animation Basics

Setting Up Characters within a Model Structure

Models are structures in XSI that make it easy to organize scene elements that need to be kept together. A character’s skeleton hierarchy, controls, envelope geometry, and groups are often kept together within a model.

The main reason for using models with character animation is that they provide the easiest way to import and export characters between scenes, and to copy animation between characters.

You can refine your rigs and character models over the course of a production without fear of lost animation. For example, character animators can start roughing out animation with a simple rig and low resolution proxy model while the other creative work is still being worked out.

As long as you keep the rig controls’ names and their coordinate space consistent, all the animation is kept and can be reapplied as the character and rigging both get more complex.

Another reason to work with models is to easily use the animation mixer. Each model can only have one Mixer “node.” If you have many characters in a scene but they aren’t within models, you have only one mixer node (under the scene root, which is technically a model) for the whole scene which means that you can’t copy animation from one character to another.

When working with characters and the mixer, it’s best to create models at the character level (or higher); that is, don’t create models for each hand, foot, leg, etc. This makes it difficult to animate the character as a whole in the mixer because you won’t have a high-level view of all of your character’s animation. If you want to segregate the different parts within a model, use a compound clip for each different part.

For more information on models in general, see Chapter 22: Models in the Fundamentals guide.

For more information on using models with the animation mixer, see Models and the Mixer in Chapter 1 and Sharing Actions between Models in Chapter 3 in the Nonlinear Animation guide.
The Big Picture

There are many different ways in which to approach character animation, but here’s an outline of some basic steps to get you going.

1. Good animation starts with a story. From it, create character sketches to work out your ideas. Ultimately, these sketches are transformed into the character design model sheets for model and skeleton building in XSI.

2. Develop the story by creating sketches showing the character in different scenes. This storyboard should show the main actions, timing, camera angles, and transitions.
3. Set up the timing (convert time to frames) for the scene and create a 2D animatic.

To do this, scan the storyboard frames and import them into XSI, then apply them as textures onto a grid, and create a flipbook. For information on setting up the frame rate and using the flipbook, see Chapter 3: Playing Animation in the Animation guide.

4. Now you need to change from 2D to 3D to start the initial production of the animation. To start out, create a 3D low-resolution version of the model, or even use primitive objects to create a “stand-in” model.

Using a low-resolution model at this stage lets you work out the animation with the model as a reference, but without being hindered by a slow refresh speed.

It also allows the modelers to work on the details of the model while you're animating. When the final model is ready, you can replace the low-res model with it (see step 10).

See the Modeling & Deformations guide for all your modeling needs.

5. With the model in mind, create the bones for the skeleton and put them in a hierarchy.

It usually simplifies your workflow if the chains and objects that make up your skeleton are organized into a hierarchy before you apply the model as an envelope. For more information, see Creating Chains on page 42 and Setting Up a Basic Skeleton on page 60.

There are also a number of tools that can help you organize the elements in your scene and make work go more smoothly—check out Chapter 6: Tools for Character Animation on page 203 for ideas.

6. Envelope the model to the skeleton you've created and save a reference pose of the envelope before you start animating.

For more information, see Chapter 3: Envelopes on page 69.

7. Develop a basic rig for the skeleton and block out its major actions.

For more information on tools for rigging, see Chapter 5: Character Rigging on page 163.

8. Animate the character in different poses to make the “key” frames.

- For information on keyframes and function curves, see Chapter 4: Animating with Keys and Chapter 5: Editing Function Curves in the Animation guide.
You can also save the skeleton poses as “static actions” which you can then bring into the animation mixer to block out a rough animation. See *Creating Neutral Poses for Skeletons* on page 64 for more information.

9. Now comes all the work of developing the animation between the key poses you have created. In XSI, there is a myriad of tools to help you animate—from keyframes and the animation editor to actions and the animation mixer. Everything about animation is in the Animation guide!

10. Make the final high-resolution model and apply it as an envelope to the skeleton. Then fine-tune the animation and the envelope weighting to solve problem areas.

   - For information on applying the envelope to the skeleton, see *Assigning Envelopes to Deformers* on page 76.
   
   - For information on envelope weighting, see *Modifying Envelope Weights* on page 90.

11. Add the final textures to make your character look beautiful and do some rendering tests.

   - For information on this, as well as lighting, see the *Shaders, Lights, & Cameras* guide.

   - For information on setting up render passes and compositing, see the *Rendering & Compositing* guide.
Chapter 1 • Character Animation Basics

**Using Ready-made Characters**

There are a number of predefined models, skeletons, and animation to get you going with XSI. If nothing else, use these to understand how things are set up in XSI and to play with some of the many animation tools available.

**Predefined Skeletons and Envelopes**

To get you started quickly on creating your own characters, load the predefined skeletons and envelopes. In this way you can quickly create a cast of characters that all share the same topology, making it easy to share animation later.

There are male and female skeletons of different levels of complexities, with controls for various degrees of freedom, such as facial nulls.

**To load the skeletons**

- Choose Get > Primitive > Model and then choose any of the Skeletons you want.

To cover the skeletons, you can load predefined envelopes made of low-resolution polygon mesh models. These models can be used as is for subdivision, enveloping, and other animation, or they can be used as a starting point for modeling your own characters.

**To load the envelope models**

- Choose Get > Primitive > Model > Body Man or Woman.

The skeletons and envelope models are also available in a Netview. The bodies are implemented as models. For more information about working with models in general, see Chapter 22: Models in the Fundamentals guide.
Using Ready-made Characters

Models and Animation

In addition to, and in some cases based on, the predefined skeletons, there are some characters and animation ready for you to use in a Netview.

There is motion capture data and keyed animation that can be applied to these characters. You can drag and drop a character from a Netview into an XSI scene, then drag and drop an animation onto it.

Making Faces

You can get going quickly on facial animation by loading a predefined low-resolution polygon mesh head (male or female) using one of the Get > Primitive > Model > Face commands. This makes it easy to create any number of different faces with the same topology, which makes it easy to copy shape keys between them.

When you choose the command, it opens the Face Maker property editor, which allows you to control the proportions of various parts of the face, such as the eyes, cheeks, mouth, ears, eyes, etc.

For more information, see Getting a Head Start on Facial Animation in Chapter 4 in the Nonlinear Animation guide.
Building Character

You can get going quickly on making characters by loading a predefined male body. This makes it easy to create any number of different characters using the same topology. The body is implemented in XSI as a model. For more information about working with models in general, see Chapter 22: Models in the Fundamentals guide.

The Character Designer lets you get a generic male body, then use parameter sliders to interactively manipulate individual body and head features. You can create many bodies, each with their own distinctive look, yet have all bodies sharing the same underlying topology. The polygon mesh is designed to be used as the basis of a subdivision surface, and is suitable for enveloping.

To get a body

1. Choose Get > Primitive > Model > Character - Man from any toolbar. A body is created and the Character Designer property editor opens.

   The Overview page contains the main sliders that define the overall look of your character. If you want to further define the look, you can use the parameters on the other pages for more fine-tuning.

When you load presets, you need to select an option (such as Ectomorph) on the Overview page, then click Apply for the changes to be applied to your character.

The rest of the pages in the property editor are divided according to each body and face part, such as Skin, Head, Face, Eyes, Nose, Mouth, Body, Arms, and Legs. There’s even a Morph page that contains parameters that morph the character’s head into a dog or elephant’s head!
2. Use the sliders to control the size, position, and other attributes of the body and facial features. You can also transform and deform points and clusters manually, and still adjust the Character Designer properties. All these parameters can be animated, which is useful to doing organic deformation effect. Have fun!

3. If you want to apply hair, there is a predefined cluster of polygons on the head that you can use (see Growing Hair in Chapter 11 in the Simulation guide).

4. When you are finished creating the character, select the Character mesh node beneath the Man_Character model node and freeze it before enveloping to improve performance. Remember that once you freeze, you cannot edit the character’s individual parameters.

To reset the properties
- Click the Reset Character button on the Overview page to reset all option settings in the property editor to the default.
- Click the appropriate Reset button for specific parameter groups or pages to reset only those related options.

To set the polygon mesh resolution
The polygon mesh body is saved with 1 degree of subdivisions.
- Press the - (minus) key to get a lower resolution.
- Press the + (plus) key to get a higher resolution.

To change the Character Designer properties
The Character Designer controls are a custom property found in the Character mesh node beneath the model. To display and modify these properties later:
1. Tree-select the null root of the model.
2. Click Selection on the Select panel.
3. Expand the Man_Character > Character tree and click the Character_Designer custom parameter set icon.
Chapter 2  Building Skeletons
Chapter 2 • Building Skeletons

Anatomy of a Skeleton

Skeletons provide an intuitive way to pose and animate your model. A well-constructed skeleton can be used for a wide variety of poses and actions, in much the same way as the all-purpose skeletons in our bodies can. On the other hand, a skeleton might also be used to provide less flexibility—to ensure that a model cannot move beyond certain restricted conditions. For example, look at your wrist: you can easily bend it straight up or down, but side-to-side motion is very limited.

Analogous to real skeletons, skeletons in XSI are usually made up of bones that are linked by joints that can rotate. However, unlike real skeletons, the combination of bones and joints is referred to generically as a chain in XSI.

This chapter covers what chains are, how to draw them and modify them, how to view their elements in different ways, and how to create a special spine controller. For information on animating skeletons, see Chapter 4: Animating Skeletons on page 113; for information on rigging, see Chapter 5: Character Rigging on page 163.

For tutorials on building a skeleton from scratch, see the Character Animation lessons in the Tutorials guide.

2D and 3D Chains

Chains can be either 2D or 3D. What's the difference?

• 2D chains essentially act as hinge joints, restricting movement so that it’s easier to create typical skeleton actions, such as bending a leg or arm. When using IK, a 2D chain’s joints can only rotate on the Z axis of the root, like hinge joints.

If you create a 2D chain and move the effector, you’ll see that the joints after the first bone all act as hinges. Of course, you can rotate the joints of a 2D chain in any direction with FK, but this is overridden as soon as you invoke IK.

• 3D chains, however, can move any which way they want. Its joints can rotate freely on any axis, like ball joints. This lets you create anything, such as a dog’s wagging tail.

The first joint connecting the root to the first bone on either a 2D or 3D chain is always a ball joint, and it can rotate on any axis.
What Makes Up a Chain?

A chain is a series of bones connected by articulated joints. A chain has four basic components: a root, bones, joints, and an effector. There is no limit to the number of bones a chain can have, but there must be at least one bone in a chain.

You can change the visual display of the chain’s bones, roots, and effectors—see Changing Shapes for Chain Elements and Their Shadows on page 35.

Chain Root

The chain root is a null that acts as the starting point on the chain. Because the first joint is local to the root, the root’s position and rotation determine the position and rotation of the rest of the chain. In a 2D chain, the entire chain extends on the root’s (and first bone’s) XY plane, called the resolution plane (see page 122). A chain root can be the child of another object, such as another chain root or an effector.

You can make any joint in a chain act like a root by creating a subchain—see Creating Subchains for Independent Movement on page 150 for more information.

Bones

The chain bone is a straight line that connects roots, joints, and effectors. A bone always rotates about its joint, which is at the end of the previous bone. The first bone rotates around the root.

The first bone in the chain is a child of the chain root, and all other bones are children of the previous bones in the chain.

The first bone in the chain is position-constrained to the root. As a result, the bone’s position constraint to another object will be superseded by its constraint to the root.

Joints

A joint is the connection between two elements in a chain: there is a joint between the root and the first bone, between bones in the chain, and between the last bone and the effector. The first joint is between the root and the first bone.

A joint can be either 2D or 3D (see page 30 for a description of 2D and 3D chains):

• In a 2D chain, only the first bone can be rotated with three degrees of freedom using IK. Its joint, the root, acts as a ball joint, but all other joints act like hinge joints.

A 2D joint is indicated on a chain by a circle. The angle of the circle indicates the plane along which the chain will unfold, or the resolution plane (see page 122).
A 3D chain’s joints all act like ball joints; they can rotate with three degrees of freedom. A 3D joint is indicated by three circles along the joint’s X, Y, and Z axes.

By default, joints are not displayed: to view them, see Displaying the Joint Circles on page 34.

Effectors

The effector is a special null that is the last part of a chain. Moving the effector invokes IK, which modifies the angles of all the joints in the chain between the root and the effector.

By default, when you create a chain in XSI, the effector is a child of the chain root, unlike in SOFTIMAGE|3D where it is a child of the last bone. This allows you to place the effector anywhere in the chain’s hierarchy or even make it the child of an object outside of the chain. As well, you can animate the effector in local space as opposed to global, as you do in SOFTIMAGE|3D.

For more information about animating the effector, see Animating the Effector on page 123.

You can also draw a chain to keep the effector in the same structure as SOFTIMAGE|3D—see Drawing Chains à la SOFTIMAGE|3D on page 44.

Chains imported from SOFTIMAGE|3D retain their original hierarchical structure: the effector is a child of the last bone instead of being a child of the chain root, as is the default in XSI.

As well, the effector’s orientation and position remain global (and keying it is in global by default), not local.
Finding Skeleton Properties in the Explorer

The following shows where different skeleton elements can be found in the explorer (or using the Selection button on the Select panel to open an explorer specific to the object). By clicking on their icons, you can open their property editors:

- **Root** properties are only its visibility properties and shadow radius for the chain root. To set these, click the select the chain root and press Enter or click the root’s icon in the explorer (not the Chain Root icon—the one for the node above it).

- **Bone** properties are set in the Chain Bone property editor. Select a bone and press Enter or click the Chain Bone icon in the explorer. If you press Enter, you get the chain, joint, and bone property editors all in one.

- **Joint** properties are set in the Kinematic Joint property editor. Select a 2D joint or bone and press Enter or click the Kinematic Joint icon in the explorer.

- **Static KineState** indicates that an envelope is applied to the skeleton element. The StaticKineState is the initial position of the skeleton when the envelope was applied, and is used for envelope calculations.

- **Effector** properties are mostly set in the Kinematic Chain property editor (see below for opening it).

However, to change only effector display and shadow properties in the Chain End Effector property editor, select the effector and press Enter or click the effector’s icon in the explorer (not the Chain End Effector icon—the one for the node above it).

Getting the Chain’s Properties

When you’re working with chains, you often need to open the Kinematic Chain property editor. This property editor contains most of the parameters that affect how an IK chain is solved (especially effector properties).

**To open the Kinematic Chain property editor**

Do any of the following:

- Select any chain element from the same or different chains and choose Create > Skeleton > Inspect Chain Properties or press Ctrl+r.

or

- Select the first bone of the chain and press Enter.

or

- Click the Kinematic Chain icon in the explorer.

The Kinematic Chain property editor for the selected chain is opened. If you had elements selected from several chains, the property editors open in multi-mode.
Chapter 2  •  Building Skeletons

Displaying Chain Elements

There are many ways in which you can control the display of chain elements. You can hide them, like any other object in XSI or choose to display just certain parts of a chain. There is a wide variety of visual styles for displaying bones and effectors, as well as for the bone or effector’s shadow. Many of these display types also make it easy to select a chain element.

Displaying or Hiding Chain Elements

There are a few ways of displaying chains in viewports.

To display or hide chain elements

Do any of the following:

- Click the eye icon in a viewport’s title bar and choose Chains (this is on by default) to see chains in that viewport.

  or

- Choose View > Chains from the main menu bar to toggle chain visibility in all viewports.

  or

- Click the eye icon and choose Visibility Options in a viewport’s menu bar (or press Shift+s) to open the Camera Visibility Property editor. On the Objects page, toggle the Chain Roots, Bones, and Effectors options (these are on by default).

Displaying the Joint Circles

By default, the joints between the bones are not displayed as circles because they can clutter the limited space on a chain. However, if you need to display them, do any of the following.

To display the joint circles

1. Click the eye icon in a viewport’s title bar and choose Visibility Options in a viewport’s menu bar (or press Shift+s) to open the Camera Visibility Property editor.

2. On the Attributes page, select Chain Joint Rotation Limits.

Displaying Critical Zones for IK Chains

Around the root is an area called the critical zone, represented by a double-cone shaped icon. If the effector is translated into the critical zone, the general orientation of the chain will flip.

To display the critical zone

1. Click the eye icon in a viewport’s title bar and choose Visibility Options in a viewport’s menu bar (or press Shift+s) to open the Camera Visibility Property editor.

2. On the Attributes page, select Chain Critical Zone.
The critical zone is blue when the default preferred axis of rotation is used or when a preferred axis constraint is applied to the chain. The critical zone is green when an up-vector constraint is applied to the chain.

When you’re animating a skeleton, it’s common to work with a low-resolution version of the model enveloped to the skeleton. This helps you get a better sense of how the animation works with the final envelope. However, working with enveloped skeletons can make it difficult to select chain elements.

There are two solutions to help with this problem:

• Select a different shape for the bones, roots, and effectors.
• Use a shadow around the bones, roots, and effectors.

There is a wide variety of shapes from which to choose to display bones, roots, and effectors (including having nothing displayed at all). This lets you set up chains that look very different from the standard chain display.

As well, you can have a shadow around each chain element. The shadow is a visual reference that simplifies visualizing and selecting chain elements that are too small or too difficult to see or select. You can set a shadow for any selected root, bone, or effector.

Because the shadows are 3D representations of the chain elements themselves, you don’t need to constrain special control objects or icons for the purpose of rigging, which can eliminate that process (see Chapter 5: Character Rigging on page 163 for more information). As well, you can scale and translate (offset) these shapes without affecting the skeleton’s structure or the envelope’s shape—it’s a display enhancement.
To change the shape of the chain elements and/or their shadows

1. Select any chain element and press Enter. You can also select multiple bones or effectors (you need to select roots separately) to change the shadow radius for all bones or effectors at once.

2. The property editor for the corresponding chain element (bone, root, or effector) appears (such as the Chain Bone property editor for the bone).

3. Select a shape from the **Main Bone/Effector Icon** list to change the shape of the actual chain element (effector and root displays are the same)—see below and the next page for the different display options.

4. Set a shape from the **Shadow Bone/Effector Icon** list to change the shape of the shadow. These are the same shapes as for the Main Bone Icon.
   - You can offset and scale this shadow on each axis in relation to its chain element. For example, the **Scale** is the scale of the shadow in relation to its chain element: the default is 1, meaning a 1:1 ratio (the same size as the element).

If you find that the display size of the bones are too large, set its **Main Bone Icon** display to None, then set the **Shadow Bone Icon** to a bone shape you want (such as Standard) and scale down its size.

   - Select **Lock Shadow Scale** to keep the shadow at a constant size even if the chain element is scaled.

**Shapes for Effectors/Roots and Their Shadows**

Select from any of these shapes for effectors and roots and/or their shadows.

- **Null** - Default XSI style.
- **Rings** - Three concentric rings, each aligned with an axis (X,Y,Z) representing the centre of the root/effector.
- **Arrow Rings** - Same as Rings except with an arrow in each ring to mark the direction of the axis.
- **Box** - Box centred at the root/effector.
Shapes for Bones and Their Shadows

Select from any of these shapes for bones and/or their shadows.

- **Standard** - Default XSI style.
- **SI|3D Style** - Same as Standard except top wedge extends beyond bone's root for a fixed proportion.
- **Pyramid** - Same as Standard except no top wedge.
- **Wedge** - Arched at top, with thickness tapering down to joint. **Wedge II** is the same except top wedge extends beyond bone's root.
- **Box** - Full box from root to joint.
- **Rings** - Elongated spheres from root to joint.
- **Line** - Straight line from root to joint.
- **Cylinder** - 10-sectioned cylinder from root to joint.
- **Cone** - 10-sectioned cone tapering from root down to joint.
X-Ray Shading

X-ray shading is a viewing option that lets you see through objects in Constant, Shaded, and Textured viewing modes. This is very useful when working with envelopes because you can see and select the underlying deformers while still seeing the shaded surface of the envelope.

To use x-ray shading

1. Open the Camera Display property editor using one of the following methods:
   - To affect a single viewport, choose Display Options from the Display Type menu in the top right corner of the viewport.
   - To affect all viewports, choose View > Display Options (All Cameras) from the main menu-bar.

2. On the Display Mode page, select the XRay Shading option and choose an XRay Display Type:
   - Overlay draws the bones on top of the envelope.
   - Screen draws a semi-transparent envelope so that you can see the bones inside.
Selecting Chain Elements

There are various tools in XSI that make it easy to select chain elements in a scene, from quickly branch-selecting a whole chain to using the spreadsheet.

Navigating in a Skeleton Hierarchy

You can use the Alt key plus your keyboard arrow keys to move up and down a hierarchy of chains in all views.

- Press **Alt+up-arrow** to select the next chain element up in the hierarchy (the chain element’s parent).
- Press **Alt+down-arrow** to select the next child down the chain.
- Press **Alt+Home** to select the parent (first object) of the chain, which is the root.
- Press **Alt+End** to select the last child in the chain. If the chain root or a bone is selected, it moves to the next chain effector.

This is particularly useful to quickly flip between a chain root and a chain effector, or to traverse a hierarchy from root to root or effector to effector.

- Press **Alt+left-arrow** to select the previous sibling in the chain.
- Press **Alt+right-arrow** to select the next sibling in the chain.

Remember that a bone is the child of the previous bone, not its sibling. As well, in a default XSI skeleton, the effector is a sibling of the first bone (it’s a child of the chain’s root).

For more information on other shortcut keys for hierarchies, see *Navigating in Hierarchies* in Chapter 13 in the *Fundamentals* guide.
Branch-selecting the Chain’s Root

When you normally branch-select a chain element, the element you middle-click and its children are branch-selected.

However, you can also quickly branch-select a whole chain from its root when you select any part of the chain (such as a bone or effector). For example, if you right-click a bone in a finger, the finger’s root is branch-selected so that the whole finger chain is selected, but not the whole hand (tree).

To quickly branch-select a chain’s root

- Do one of the following:
  - Press the space bar to activate the Object selection tool, and right-click on any chain element in a geometry or schematic view.
    or
  - Select any chain element, press Alt+right-click, and choose Select Chain.
    or
  - Select any chain element and choose Select > Select Chain in the main command area.

If you have reparented the effector so that it is no longer part of the chain hierarchy, it is not included in the selection.

To set a preference for right-clicking a chain element

You can choose how a skeleton is selected when you right-click a chain element.

1. Choose File > Preferences from the main menu bar at the top of the XSI window.
2. In the explorer panel of the Preferences editor, expand the Tools node and click Selection.
3. In the Selection Preferences page, select one of the options in the Upon Selection using RMB group:

- **Chain Select**: Selects the chain's root in branch mode when you right-click any chain element (as described previously). Right-clicking any other type of object selects the entire hierarchy tree.

- **Tree Select**: Selects the whole hierarchy tree when you right-click a chain element or any other type of object (this is how right-clicking worked in versions of XSI previous to version 3.5).

- **Model Select**: Selects the model node when you right-click a chain element.

**Using Selection Filters**

If you have many objects located in close proximity to a chain, it's easy to mistakenly select an object instead of the chain. With any of these Select filter options active, you can make sure that only chain elements can be selected:

**To use the selection filters**

- In the Select panel of the main command area, click the little arrow button and select **Chain_Element** (which includes roots, bones, joints, and effectors), or just **Bone**, **Effector**, or **Joint** from the filter list.

**Using the Spreadsheet**

In the spreadsheet, you can use a special chains query to both select chain elements and get information on them. Anything that is a chain element is listed, giving a host of information: chain type, solver, solver angles, blend FK/IK, force IK, pseudo-root, view and render visibility, etc.

**To use the spreadsheet**

1. Select any chain or skeleton.
2. In the spreadsheet, choose **Query > Chains/Skeletons**.
3. To select an element, right-click on its name and choose **Select Object**.
Creating Chains

While the process of creating a chain is very simple, here are some important points that may have a big impact on the resulting chains.

- **Make sure to draw bones as close to their “natural” angles as possible—which are their preferred angles.** Drawing bones in a straight line can result in unpredictable bending; preferred angles help you predict how a chain will bend. For more information, see *Basic Concepts for Inverse Kinematics* on page 122.

- **Draw the skeleton chains in relation to the default pose of the envelope in which you’re going to use them.** This means you don’t have to spend as much time resizing each bone later.

- **Size does matter, to a certain extent.** While you can usually model regardless of scale, don’t make chains too small (such as a 2- or 3-jointed chain in less than 1 Softimage unit). This is because a small scale limits the amount of precision available for further calculations, such as rotation.

- **You can use snapping when drawing 2D or 3D chains**—see *Chapter 18: Snapping* in the *Fundamentals* guide for more information.

- **You can draw chains in symmetry**—see *Drawing Chains Symmetrically* on page 46.
Creating Chains

- The orientation of a chain root depends on the view in which you're drawing the chain. For example, in the Front view the orientation is 0,0,0, but in the Right view it’s 0,90,0.

- Make sure that Create > Skeleton > Align Root to First Bone When Drawing is on (checked) o change the root’s orientation to match the first bone’s when you draw a chain. See page 119 for more information.

Drawing a Chain

To draw a chain

1. Choose Create > Skeleton > Draw 2D Chain or Draw 3D Chain from either the Model or Animate toolbars.

2. Click in any viewport and the root is created.

3. Move the pointer to a new position and click again. The first bone and joint are created.

- To add bones, continue clicking to create as many as desired.

- To start a new chain, middle-click to end construction of the current chain (without ending chain-drawing mode), then move your pointer to where you want to start the new chain. Repeat steps 2 to 3.

4. To end the chain-drawing mode, right-click.

A 2D chain is created in the local XY plane of the root and its joints rotate about their local Z axes.

A bone’s local X axis always points along the bone—that is, in the direction of the next joint (or if it is the last bone, the effector). To change the axis to a different one, see Changing the Bone’s Axis Alignment on page 120.
By default, when you create a chain, the effector is a child of the chain root. However, you can also create chains with the effector being a child of the last bone, as it is in SOFTIMAGE|3D. As well, the position and orientation inheritance is off, the same as when you import chains into XSI from SOFTIMAGE|3D.

This may be useful if you work back and forth between XSI and SOFTIMAGE|3D and need skeletons to be drawn in exactly the same way in both packages.

To draw SOFTIMAGE|3D-style chains
1. Make sure that Create > Skeleton > SI|3D Skeleton Drawing is on (checked).
2. Choose Create > Skeleton > Draw 2D Chain or Draw 3D Chain and draw a chain as described in the previous section (on page 43).

For information on animating the effector for IK using this setup, see Animating the Effector à la SOFTIMAGE|3D on page 124.

Creating Chains from Curves
In addition to drawing a chain bone by bone, you can create a 3D chain based on a curve. If you're creating a skeleton with a multi-boned chain, such as an octopus' tentacles or a dog's tail, it might be easier to draw and position the curve first and then convert it to a chain.

To create a tail controller for a rig from a curve, see Creating a Tail on page 58.

To create a chain from a curve
1. Select the curve on which you want to base the chain.

   If you're using a closed curve, the effector and root may be in the same location. If you apply an up-vector constraint to the chain, the constraint won’t work if the effector and root are in the same spot.

2. Choose Create > Skeleton > Create Chain from Curve.
3. In the dialog box that opens, specify the number of bones in the 3D chain to be created. The more bones, the more accurately it will reflect the shape of the curve.

   Although the slider only lets you specify up to 20 bones, you can type any value you like in the slider’s text box.

   The chain's root is created at the curve's first point; that is, the first point that was created when the curve was drawn.
4. Select the **Constrain Chain to Curve** option to keep a constraint so that whatever you do to the curve is reflected in the chain, such as animating it. This option makes it easier to animate the curve instead of setting up an IK rig to do this.

**Resetting Bones**

You can reset a bone’s rotation to the value of its preferred angle of rotation. This effectively resets the chain to its “shape” or position when you created it. For more information on preferred rotation angles, see *Changing the Joint’s Preferred Angle* on page 137.

**To reset a bone**

- Select one or more bones and choose **Create > Skeleton > Reset Bone to Preferred Rotation** from the Animate toolbar.
Chapter 2 • Building Skeletons

## Drawing and Duplicating Chains in Symmetry

You can create symmetrical chains when you draw them, or duplicate them in symmetry after you have drawn them.

### Drawing Chains Symmetrically

If you want to draw two chains that need to be mirrored, such as a character’s arms or fingers, you can draw one and have the other one created at the same time in symmetry mode.

**To draw a chain in symmetry mode**

1. Click the **Sym** button in the Transform panel in the main command area to activate symmetry.
2. Right-click on the **Sym** button to select a symmetry plane. A thick line appears down the middle in the viewport indicating the mirroring point.
3. Draw a chain as usual (see *Creating Chains* on page 42) and see its mirror image being drawn at the same time across the plane.

### Duplicating Chains in Symmetry

When you are creating a skeleton, you often need to duplicate chains and then “mirror” them across the correct plane. With the **Duplicate Symmetry** command, you can do both at once.

You can select the plane across which you want the duplicated chain to be mirrored: XY, XZ, YZ. For example, say you’ve created a simple arm chain in the Top view. To duplicate and mirror it for the arm to be on the other side of the body, you would choose the command and then select the YZ plane.
The Duplicate Symmetry command also duplicates constraints (such as position and up-vectors) and chain properties (such as rotation limits), which means that you can duplicate entire rigs for skeletons.

As well, the Duplicate Symmetry command works on any geometrical object or hierarchy, not just chain elements.

You need to apply this command before you animate the chain: it does not work properly on animated chains.

To duplicate and mirror a chain

1. Select the (unanimated) 2D or 3D chain you want to duplicate (middle-click to branch select or right-click for the whole hierarchy).

2. Choose Create > Skeleton > Duplicate Symmetry from the Animate toolbar or Edit > Duplicate/Instantiate > Duplicate Symmetry.

3. In the Duplicate Symmetry dialog box, select Duplicate Constraints to include existing constraints on the chain you’re duplicating. For example, if there’s an up-vector constraint on an arm or leg, you may want to use it on your new chain as well.

   If the chain you want to duplicate is constrained to a null, for example, and you select only the chain and choose this command with Duplicate Constraints selected, the new symmetric chain is also constrained in position to the original null.

   If you select both the chain and the null, the null is duplicated and the new symmetric chain is constrained to the new null.

4. Select Share Parent to automatically make the duplicate a child of the same parent as the original. If you don’t select this option, the new chain will be a child of the scene root.

5. Select Freeze Negative Scaling to do just that on the duplicated chain.

With Freeze Negative Scaling on, this is essentially the same as the behavior with symmetry in SOFTIMAGE|3D.

   If you deselect this option, the scaling for the duplicated chain is negated, switching its handedness. For example, if the original chain is right-handed, the duplicated chain will have a left-handed coordinate system. Note that negative scaling may not be compatible with game engines or motion capture data.

6. Select the Plane of Symmetry to use: XY, YZ, or XZ.
Chapter 2 • Building Skeletons

Adding and Removing Bones

After you have created a chain, you can change it by adding or removing bones. For information on adding or removing bones after an envelope has been set, see Adding and Removing Deformers on page 107.

Adding Bones to a Chain

After a chain is created, you can still add bones to it. The bones are added to the end of the chain, just before the effector.

To add bones to a chain

1. Select the chain and choose Create > Skeleton > Add Bone to Chain from the Animate toolbar.

2. Click at the point where you want to add the bone. The bone is added to the end of the chain. Continue clicking to add more bones.

3. Right-click to end the mode.

Removing Bones from a Chain

While you can’t delete individual bones from a chain, you can branch-select a chain and then delete it. If there are elements that are children in that chain, and you want to keep them, make sure to cut their links before deleting (select the elements and click the Cut button in the Constrain panel), and then reparent them to the chain. For more information on parenting, see Chapter 13: Hierarchies in the Fundamentals guide.

Another way to make it appear as if a bone is removed is to reduce its length to zero, as described in Resizing (Scaling) Chains on page 49.
Resizing (Scaling) Chains

You may need to resize individual bones, chains, or a whole skeleton to achieve the exact structure you are trying to create. For instance, you may find that your character’s forearms are too long for their envelope.

You can scale bones individually or scale a skeleton in branch mode as you would any other object using the Scale commands on the Transform panel. However, there are two tools designed for scaling chains that are easy to use and give you greater control:

- The **Move Joint** tool as described on page 49.
- The bone's **Length** parameter as described on page 51.

**Scaling Bones After Enveloping**

It’s preferable to set the bone’s length *before* you add the envelope to the skeleton so as to not ruin the envelope’s weighting. However, you can also do this after the envelope is on. You can have the new bone length affect the envelope’s deformation or not:

- When you simply change the size of a bone or chain (the deformer), the envelope deforms with it.
- To change the length of the bones *without* deforming the envelope, the process is the same as setting a new reference pose as described in *Changing Reference Poses* on page 106.

  This involves muting the envelope’s operator, scaling the bones and adjusting the envelope’s weights, setting the new reference pose, and then unmute the envelope operator.

**Resizing Bones by Moving Chain Elements**

Using the Move Joint tool, you can interactively resize bones by moving any chain element (root, joint, bone, or effector) in a chain to a new location. The bones that are immediately connected to that chain element are resized and rotated to fit the chain element’s new location.

The Move Joint tools is similar to the Move Joint/Branch command in SOFTIMAGE|3D.

This tool makes it easy to resize bones to do such things as:

- Fit any number of bones inside an envelope.
- Match an existing skeleton’s proportions to a new character by quickly moving its bones into position.
- Resize any number of chains to fit motion capture data.
Chapter 2 • Building Skeletons

- Make the current angle of a joint the new preferred angle of rotation. Because the Move Joint tool acts as an FK manipulator, you can simply click on a joint to update the preferred angle to the current orientation without resizing the bones (see Changing the Joint’s Preferred Angle on page 137 for more information).

- You cannot move a chain element if its position or orientation is affected by constraints or other controlling animation (expressions, scripted operators, or actions in the mixer).

- If you have an up vector or preferred axis constraint on the chain, the Move Joint tool limits the chain’s movement to the resolution plane. You may notice this if you’re trying to snap the chain to points as you resize it. In the chain’s Kinematic Joint property editor, change the Resolution Plane behavior to Default, and then you can move the joint (see page 140).

To resize bones by placing chain elements

1. Choose Create > Skeleton > Move Joint/Branch or press Ctrl+j to activate the Move Joint tool.

2. Move the mouse pointer over any chain element (which is highlighted in white), then click+drag it with the left or middle mouse button.

Only chain elements are “pickable” when the Move Joint tool is active: all other scene elements are ignored.

The type of chain element and the mouse button you click determines which bones in the chain are resized:

- If you left-click+drag (Move Joint) a joint, only the joint is moved and the bones directly connected to it are resized to fit the new position.

- If you left-click+drag (Move Joint) a bone, the bone and its joint are moved, and the bones connected to it are resized to fit the new position. The selected bone is not resized.

- If you middle-click (Move Branch) any chain element, it is moved in branch mode, and only the adjacent bone “above” that element is resized to fit the new position. The selected element and its children are moved as a group but are not resized.
Resizing (Scaling) Chains

- To constrain the movement to the X axis, press Shift while you drag with either mouse button.

- You can use snapping to align chain elements to the grid or another target. See Chapter 18: Snapping in the Fundamentals guide for information.

3. You can continue to select and manipulate any chain element until you deactivate the Move Joint tool by right-clicking or pressing Esc.

Keeping the Move Joint tool active, you can use the Scale, Rotate, and Translate controls in supra mode (that is, keep their associated key pressed down while transforming). It’s an easy way to move elements around to test them, without exiting the Move Joint tool.

For example, activate the Move Joint tool and use it to resize a chain. Make sure the effector is highlighted, then press v to translate the effector (keeping the key pressed) to test how the revised chain will work in IK. When you’re done, release the v key and continue to use the Move Joint tool to tweak the chain.

**Changing a Bone’s Length**

Resizing the bone using its Length parameter preserves the local coordinate system of its children in the chain. This is especially important if you are modifying a skeleton for use with a different envelope.

To change the bone’s length

1. Select one or more bones.

2. Open the Chain Bone property editor (press Enter).
3. Change the length of the bone by dragging the **Length** slider or entering a value in its box.

**Scaling Bones in Hierarchies**

Like other objects in hierarchies, bones try to obey the same rules of transformation inheritance. As a result, when you animate the transformation of a parent object (in this case, scaling), its animation is propagated to its children. This is usually desirable except in the cases where you want a child object to have its own animation.

When you keyframe, it only keyframes the scaling of the manipulated object to avoid destroying any animation on the subsequent nodes (in this case, bones) already there; thus you scale an object and its children follow.

You can key the values for the **Length** parameter to animate a bone changing length. However, if the bone is part of a skeleton and you want this animation to be respected, you probably want to prevent it from inheriting the scaling from its parent.

**To prevent inheriting the parent’s scaling**

1. Select the bone whose length you have animated and open its Local Transform property editor (press Ctrl+k).

2. On the Options page, deactivate the **Scaling** option.

For more information on this, see *Animating Transformations in Hierarchies* in Chapter 2 in the *Animation* guide.
Creating a Spine

Creating a simple and functional spine is often a daunting task in any character setup because of the nature of the spine's movement. There are different ways of solving this problem, but creating a spine with the Create > Skeleton > Create Spine command is the best and easiest way. With it, you can create a quaternion-blended spine for controlling a character the way you like.

This spine is made up of cubes that act as vertebrae, with the top and bottom vertebrae attached to hip and chest control objects that you create. The simplest way of visualizing a spine is a bridge that starts at the hip and extends to meet the chest. Its orientation matches the hip where it starts and the chest where it ends. The vertebrae roll between these two orientations and bank along the path of the spine.

The spine is a mixture of path constraints (to keep it on a spline), distance constraints (to hold the spine together more like a chain than a spline), and up-vector constraints (to keep the orientation to the spine but the roll to the quaternion blend between the chest and hip).

This spine is available as the quaternion spine option for the biped, biped dog leg, and quadruped rigs that you can create in XSI—see Getting Started with Predefined Rigs on page 168.
Chapter 2 • Building Skeletons

To create a spine

1. Create two objects to use as the hip and chest controllers.

2. Orient the positive Y axis of the hip and chest controllers so that they will both be pointing along the axis of the spine. This is very important to do to get a straight spine, and avoid unexpected results when you animate.

3. Choose Create > Skeleton > Create Spine from the Animate toolbar.

Pick the object that is to act as a hip controller (the base of the spine), then pick the object (middle-click) that is to act as a chest controller (the top of the spine).

4. In the Spine Maker dialog box that appears, define the number of vertebrae in the spine. The spine is generated using this information.

   If you can select Implicit Vertebrae, the cubes that represent the vertebrae are created from implicit cubes instead of polygon mesh cubes.

5. Parent the spine into your skeleton or rig setup the way you want.

6. To animate the spine, set keys on the position and rotation of the hip and chest controllers. The spine maintains a fixed length and will reach towards the chest as much as it can.
Scaling the Spine

You can “squash and stretch” the spine to fit it into a number of different character setups, or to adjust to a new animation to which you've retargeted the character (such as with Motion Deform—see Modifying a Biped’s Motion on Tracks on page 154).

The scale of the spine doesn’t change as you move the chest controller, but it can be scaled with the Scale slider that appears when you create the spine, and is located in the Spine Control custom parameter set beneath the spine curve object.

This means that every component can be scaled or animated in proportion from the bones to the spine.

Adjusting the Depth

Both the chest and the hip have nulls that represent depth, called ChestDepth and HipDepth, respectively. These nulls are created as children of the chest and hip control objects. The depth affects how the spine deforms in position and rotation, such as to give the spine a more natural curve.
If you want to modify the depth on the spine, position these nulls in Constraint Compensation mode (click the CnsComp button on the Constrain panel). You can also key in them if you want to animate the depth variance. The depth of the spine is set only through parenting, so the chest and hip objects might move independently from the depth nulls if you have Child Transform Compensation mode on (the ChldComp button).

**Rolling the Spine**

You can roll the spine between ranges of 360° to -360° (a full 720°) which makes it useful for dynamic spine motion, tentacles, dinosaur necks, and many other character setup scenarios.

In a regular quaternion blend, you will never interpolate a rotation blending path greater than 180 degrees. You can visualize this by picturing two points on a sphere. No matter where you position the two points, the shortest path will always represent an angle 180 degrees or less.

Consider Sphere 1 above where you are blending between Object A and B. As B moves away from A the distance you will be blending through increases. In diagram 1, the blend will be between 0° (A position) and 130° (B position).
Creating a Spine

Now if you consider Sphere 2, B has moved beyond 180° to a position 220° away. But at this point the best blend found by a slerp algorithm ends up being from the other side of the sphere, shown in Sphere 3. The angle being blended between will be 140°.

For this reason you will never have a blend angle greater than 180° and less than -180°.

- You can use the spine as a tool for taking the animation from a simple chain and making it seem as if it was done with many more bones. For example, create a simple chain, constrain the spine to it, and then constrain a secondary chain with more bones to the spine.

- The end of the spine will push out to avoid crunching. This also is useful for keying the spine’s translation with fewer keys.

As shown in the illustration on the left, the chest controller in yellow is keyed at moves between positions A and B.

However, this motion will end up crunching the spine at position C. The spine’s end vertebra (small box in green) will push out to avoid the crunch, giving a smooth animation in this case with out any extra keying to correct the movement.

If you plan on crunching a lot, keep a shallower chest depth.
Creating a Tail

In addition to a spine, you can create a tail chain that can be used in any type of rig. For example, while the quadruped rig allows you to create a tail as part of the setup, you could create a separate tail and attach it to a biped dog leg rig to control the tail on a fantastical bipedal monster creature!

A tail is basically a chain where each bone is controlled by a spring so that it reacts to the skeleton’s animation (secondary animation). You can override these springs by keying the controllers in FK to give precise control and posing when required. If you subsequently plot the animation of the tail, make sure to remove its spring operator.

To open a control slider set for modifying the tail’s spring, see Accessing the Springs’ Control Sliders on page 188.

You can create a tail from a single curve or from several 3D objects.

Creating a Tail from a Curve

To create a tail from a curve

1. Draw a curve in the shape that you want for the tail.
2. Select the curve and choose Create > Skeleton > Create Tail from the Animate toolbar.
3. Specify the number of bones in the Tail Maker dialog box that appears. The more bones, the more flexible the tail will be.

The tail is created with controllers at the curve’s control vertices.

4. Parent the tail into your skeleton or rig setup the way you want.
Creating a Tail from Objects

To create a tail from multiple objects

1. Create two or more objects and position them as you want for the shape of the tail.

2. Select the objects in a consecutive order from base to tip. This is important because the bones in the tail are created in the order in which you select them.

   To help select the objects, use the free-form selection tool (press F9) or any selection tool that lets you pick in a definite order. Using a rectangle selection tool selects many objects at once and may not produce the correct shape.

3. Choose Create > Skeleton > Create Tail from the Animate toolbar.

4. Choose the number of bones in the Tail Maker dialog box that appears. The more bones, the more flexible the tail will be.

   The tail is created with the controllers at the object centers.

5. Parent the tail into your skeleton or rig setup the way you want, such as at the base of the spine.

   See Chapter 5: Character Rigging on page 163 for information about rigs.
Setting Up a Basic Skeleton

After you have created the chains for the skeleton, you need to organize them in a hierarchy. This relationship makes it easy to select and animate the different elements in the skeleton.

Although it is possible to build an enveloping model around a skeleton, it is more common (and usually more practical) to build or place a skeleton inside a model. The structure of a skeleton determines how you can manipulate the envelope. As well, it usually simplifies your workflow if the chains and objects that make up your skeleton are organized into a hierarchy before you apply an envelope.

In addition to just chain elements, you can use other types of object, such as nulls or cubes, as envelope deformers and control objects in rigs. See Deformers on page 71 and Basics for Setting Up a Control Rig on page 164.

Creating the Hierarchical Relationships

There are many different ways in which you can set up a hierarchy, depending on the skeleton structure and the kind of movements that it needs to make. A character that only needs to walk needs a simpler setup than a character who is a circus performer.

For an example of creating a basic human skeleton from scratch and setting it up in a hierarchy, see the Character Animation lessons in the Tutorials guide.

The objects and chains of a skeleton are usually connected in a hierarchy. For instance, the hips of your model are connected to the spine, which are in turn connected to the shoulders. The shoulders are then connected to the arms.

To see how a basic skeleton is set up in a hierarchy, load one of the predefined skeletons (see page 24), tree-select the skeleton only, and open a schematic view or the explorer.
Setting Up a Basic Skeleton

To parent objects in a hierarchy

- In the explorer, drag and drop the children nodes onto the node that you want to be their parent.

![Diagram of hierarchy]

Make the right and left leg roots children of the spine effector by dropping their nodes onto the spine effector’s node.

The leg roots are now under the spine effector’s node.

or

- Select the object you want to be the parent, click the Parent button in the Constrain panel, and pick each of the children with the left mouse button. Right-click or press Esc to end parenting mode.

You can also select the children first, click the Parent button, and middle-click on the parent, but it’s usually easier to have the parent object selected first.

For more information on hierarchies in general, see Chapter 13: Hierarchies in the Fundamentals guide.

About the Effector

In XSI, the effector is the child of the chain root, not the last bone in the chain as it is in SOFTIMAGE|3D. However, you can place the effector anywhere in the hierarchy (such as higher up in the tree for easier access) or even make it the child of an object outside of the chain.

For more information on the effector, see Effectors on page 32 and Animating the Effector on page 123.
Chapter 2 • Building Skeletons

Changing the Bones’ Order of Rotation

When you’re setting up a character, you should consider how the bones will be rotating for each body part so that you can choose the proper rotation order for them. While the default order of XYZ works for some body parts, there are certain body parts or movements that can cause gimbal lock with XYZ, but not with another rotation order.

Gimbal lock is a state that Euler angles go through when two rotation axes overlap. The angle values can change drastically when rotations are interpolated through it.

For control over the order in which a bone is rotated about its parent’s X, Y, and Z axes, you can select a Rotation > Order in the bone’s Local Transform > SRT property page (press Ctrl+k).

Figuring Out the Rotation Order

To help you figure out what the rotation order for a bone should be, it’s best to start with the skeleton in a neutral pose with the bone’s rotation at 0 (see Creating Neutral Poses for Skeletons on page 64). Then start posing the skeleton with its most common movements to see where problems can occur.

Mapping the XYZ axes to movement type (such as flexion/extension, medial/lateral, abduction/adduction) can also help to reveal the most useful rotation order. Usually one type of movement will usually be dominant: its axis is the one that you want to be last in the rotation order so that it won’t be affected by the other axes.

For example, let’s compare the XYZ and YZX rotation orders on the humerus bone (biceps) in an arm. With the XYZ order:

- X twists the biceps, which is not that useful in many cases.
- For arms resting at a character’s side, the Z and X approach gimbal lock.
- The back and forth swing of the arm is not easy to animate in FK.

Now by making X the last in the rotation order, the X can be used for animating the back and forth swing an arm has when a character is running or walking. Making it the last also means that the swing axis never changes, which it rarely does because it’s determined by the shoulders.

Meanwhile, Z and Y can still move appropriately:

- Z is natural for medial/lateral movements, like the swing in a punch.
- Y is used for raising and lowering movements. Raising/lowering is usually a much more deliberate movement that comes more from flexing the shoulder muscles. It tends to move along with back and forth or medial/lateral movements, so it makes sense to have it first in the order.

So after analyzing the character’s movements, YZX seems like the best choice for animating the humerus.
To prevent gimbal lock, there are a few different ways in which you can change a bone’s rotation order.

- When you set up character a rig, you can align the chain’s root with the first bone and set the bone’s rotation order to ZXY or YXZ instead of the default XYZ. Then the middle rotation (X) (which causes gimbal lock) is on the least-used rotation (that is, rotating along the length of the bone).

- If an object needs to rotate only along one axis, change the order of axis to match it; that is, if the object is rotating around the Y axis, change the order of axes to YZX.

- If an object is rotating on two axes, change the order of axes to correspond with the first and last axes; that is, if the object is rotating around the X and Y axes, change the order of axes to XZY.

In addition to changing the rotation axes order, you can also convert the rotation angles from Euler to quaternion. Quaternion rotation angles produce smooth interpolation to help prevent gimbal lock.

For more information, see Quaternion Function Curves for Rotation in Chapter 5 in the Animation guide.
Chapter 2 • Building Skeletons

Creating Neutral Poses for Skeletons

When you’re creating a skeleton, it’s a good idea to save it in a default position (pose) before it’s animated or enveloped. This way you have a solid reference point to revert to when enveloping and animating the skeleton. This pose is known as the neutral pose, reference pose, or base pose, and is often the character with outstretched arms and legs, making it easy to weight the envelope and adjust its textures.

A neutral pose that’s useful for envelope weighting and texturing doesn’t always give the best values for animating the skeleton’s position and rotation. To set the neutral pose for optimum animation, see the section below, Creating a Neutral Pose for Animation.

As well, you can store the skeleton’s neutral pose in an action source (Store Skeleton Pose) so that you can easily reapply it to your character any time you want—see Storing a Skeleton Pose in an Action Source on page 68 for information.

Creating a Neutral Pose for Animation

When you’re animating the character, this default pose does not always give you local transformation values that are easy to key. For example, let’s say that you load the default skeleton and want to rotate the finger bones from their outstretched position into a clenched fist. If you look at their local rotation values, you’ll see that they are difficult numbers to use.

To overcome this problem, you can create a neutral pose which uses zero for its local transformation values (0 for rotation and translation, 1 for scaling). When you set a neutral pose, the object’s actual local transformation values are not changed: the neutral pose simply adds a transformation layer that lets
you specify certain conditions for keying. Basically, the neutral pose acts as an offset for the object’s local transformation values, as if there was an intermediate null between the object and its parent in the hierarchy.

When you key the character’s values, they reflect the relative difference from zero, and not a value that’s difficult to use. For example, when you key a hand bone at coordinates (0, 3, 0), you know that it’s 3 units in the Y axis above the neutral pose, so that pose becomes a local reference point for the fcurve values of the bone.

With a neutral pose set, you can select a chain element and enter zero for its local rotation, for example (often referred to as “zeroing out”), and it returns to this neutral pose.

Not only do neutral poses make keying easier, you can also reduce the number of extra nodes in a rig because you don’t need any “spacer” or “parent” nodes that helped you set zero as the value for the bones’ neutral pose.

You can create neutral poses for any type of object, not just chain elements in a skeleton. This means that you can zero out control objects in a rig.

**Setting the Neutral Pose**

You can set neutral poses using either menu commands or an object’s Local Transform property editor.

**To set neutral poses with menu commands**

1. Select or branch-select one or more chain elements (such as a bone) or rig control objects.
2. Position the selected objects into the pose you want to use as the neutral pose.
3. Choose either the **Transform > Set Neutral Pose** command from the main command area or the **Create > Skeleton > Set Neutral Pose** command from the Animate toolbar.

   Both these commands set the neutral values for the object’s total pose: scaling, rotation, and translation values.
You can also set the scaling, rotation, and translation values separately using these commands on the Transform menu:

- **Set Neutral Scaling** stores the current scaling values in the neutral pose and sets the local scaling values to (1, 1, 1).

- **Set Neutral Rotation** stores the current rotation values in the neutral pose and sets the local rotation values to (0, 0, 0).

- **Set Neutral Translation** stores the current translation values in the neutral pose and sets the local translation values to (0, 0, 0).

4. Transform the selected object as you like and key the transformation values. Because you're starting at a base of 0, the values that you enter are offset against it: it’s these values that are displayed in the fcurves.

5. To return to the neutral pose (zero out the transformation), use the Transform > Reset commands or enter 0 for the rotation and translation values and 1 for the scaling values.

**To set neutral poses in the Local Transform property editor**

1. Select or branch-select one or more chain elements (such as a bone) or rig control objects.

2. Position the selected objects into the pose you want to use as the neutral pose.

3. Open the Local Transform property editor for the selected objects (press Ctrl+k) and click the Neutral Pose tab.

4. Click the **Use Current Pose button** to set this pose as the neutral pose.
Creating Neutral Poses for Skeletons

The object’s current transformation values are stored on the Neutral Pose page and the local transformation values (as you can see on the SRT page in this property editor or in the Transform panel) are set to zero (1 for scaling).

5. Transform the selected object as you like and key the transformation values as found on the SRT page in the Local Transform property editor or in the Transform panel.

Because you’re starting at a base of 0, the values that you enter are offset against it: it’s these values that are displayed in the fcurves.

Removing Neutral Poses

When you remove the neutral pose, the object’s transformation values return to what they were before creating the neutral pose. The neutral pose values are added to the local transformation before being reset to the defaults. As a result, the object does not move in global space.

To remove neutral poses

- Select one or more objects or branches that have a neutral pose set and do either of the following:
  - Choose the Transform > Remove Neutral Pose or Create > Skeleton > Remove Neutral Pose commands.
  - Open the Local Transform property editor and click Reset on the Neutral Pose page.

The object returns to the same position it was before you set the neutral pose.
Although you can use the Reset Actor command (see page 79) to reset an enveloped skeleton to its reference pose, you can also use the Store Skeleton Pose and Apply Skeleton Pose commands for a skeleton, enveloped or not.

The Store Skeleton Pose command creates a pose (static) action for the selected skeleton at the current frame. When you store skeleton poses in an action source using this command, the parameters of the bones, joints and chain that define the current pose of the skeleton at the current frame are saved.

To create a skeleton pose
1. Branch-select the entire skeleton hierarchy and store a skeleton pose of your character in its original enveloped position (before it is animated).
2. Choose Create > Skeleton > Store Skeleton Pose from the Animate toolbar.
3. Give the pose an Action Name and set the Default In and Out frames, which also determines the number of frames over which the pose is held. The default is 5 frames.

To restore the default pose you stored
1. Select the pose action source in the explorer in the model's Mixer > Sources > Animation folder, as well as the Project > Sources > Animation > Model folder.
2. Choose Create > Skeleton > Apply Skeleton Pose (or Actions > Apply Action) from the Animate toolbar.

This applies the action you stored and restores the skeleton to the pose you created.

To load the pose as a clip in the animation mixer
Besides applying the pose, you can also load it in the animation mixer as you would any other action to create a clip and choose over how many frames you want to hold the pose. In fact, you can save any number of poses and use them as clips in the mixer to roughly block in pose-to-pose animation.

Do this as you would with any other action source to create a clip and choose over how many frames you want to hold the pose. See Adding Clips to a Track in Chapter 2 in the Nonlinear Animation guide.
Chapter 3  Envelopes
Envelopes

An envelope is an object that deforms automatically, based on the movement of its skeleton or other deformers. You can:

- Assign an object as an envelope on a set of deformers, like bones. Each point in the object is assigned an initial weight to various deformers.
- Use bounding volumes to modify how points are assigned.
- Manually assign specific points to specific deformers.
- Modify how points are weighted, either numerically or by painting.
- Change the reference pose used to calculate the initial weights.
- Add or remove deformers from an envelope.
About Envelopes

An envelope is an object or hierarchy that is assigned as a “skin” to a set of deformers such as IK chains. Envelopes move and deform in response to the movements of their deformers. In this way, for example, a character moves as you animate its skeleton.

Every point in an envelope is assigned to one or more deformers. For each point, weights control the relative influence of its deformers.

Deformers

Deformers are the objects that control the shape of the envelope. You pick the deformers when you first create an envelope deformation. You can still add and remove deformers later if necessary.

Although it’s common to use skeletons as deformers, you can actually use any object. The geometry of a deformer does not matter because the points of the envelope are assigned to its center. (Bones are the exception—points are deformed by the whole length of a bone.) Whether you use a cube, a sphere, or a null, all transformations of the surrounding envelope are relative in size, orientation, and position to the deformer’s center, not its shape. Nulls and implicit objects are good choices because they do not render.

Light bulb as envelope
The light bulb is an object defined as an envelope to the hierarchy of a cube, a cylinder, and a cone.

Hierarchy of objects as deformers
The cube, cylinder, and cone were used to illustrate the example. Nulls would produce equivalent results, but geometric objects are easier to visualize.

Deform the light bulb by altering the skeleton
Translate cube (left); translate cylinder (center); scale and rotate cube (right).
Weights

Each point on an envelope has a total weight of 100, which is divided between the deformers to which it is assigned. For example, if a point is weighted by 75 to deformer A and 25 to deformer B, then A pulls on the point three times more strongly than B.

Multiple Levels of Control

You have several levels of control over how points are assigned and weighted:

- Deformers and weights are initially assigned to points when you apply an envelope deformation. There are several options for controlling how this initial assignment is performed.
- You can use bounding volumes to fine-tune the automatic assignment of deformers.
- You can then select points and reassign them to specific deformers or to no deformers at all.
- You can edit the weighting of individual points by painting or editing numerically.

When you make changes that cause the initial weight assignment to be recalculated—for example, if you add deformers, or reset the reference pose—all modifications you have made to weights are preserved.
Envelope Weight Operator Stack

When you modify the weighting, your changes are added to an operator stack, similar to the geometry operator stack. Once you are satisfied with your envelope, you can freeze the weights to improve performance as described in Freezing Envelope Weights on page 103.

After you have frozen the weights, you can still make modifications, but XSI cannot recalculate the initial assignment.

Differences with SOFTIMAGE|3D Envelopes

There are several important differences between envelopes in XSI and SOFTIMAGE|3D:

- In XSI, envelopes are treated like any other deformation.
- Any object can act as an envelope deformer. There is no need to group objects as a skeleton first.
- There is no longer any distinction between local and global envelopes.
- Rigid envelopes are not supported. You can get the same effect by parenting objects, in combination with either 2-, 3-, and n-point constraints or constraint blending.
- Envelopes and deformers do not need to be in a special hierarchy, but you can arrange them in a hierarchy if you wish.
Before You Envelope

Before you actually set the envelope, there are a few things you should consider.

Modeling the Envelope

While every animator or modeler has their own favorite techniques for modeling envelopes, here are some tips to keep in mind.

Surface Meshes

Surface meshes are a good choice for envelopes because the surface continuity manager eliminates seams at the junctions between surfaces. However, you may experience problems with star junctions.

To avoid these problems, keep star junctions away from areas of high deformation, or hide star junctions in places where they won’t be seen (such as armpits). For more information about surface meshes in general, see Chapter 13: Surface Meshes in the Modeling & Deformations guide.

Subdivision Surfaces

Subdivision surfaces are another good choice for envelopes. If you use the control mesh as your envelope, there are fewer points to weight and manage. This can reduce the amount of work and speed up interaction. For more information about subdivision surfaces in general, see Chapter 9: Subdivision Surfaces in the Modeling & Deformations guide.

Reference Poses

At the moment you create the envelope, the relative positions of the envelope and deformers are very important. This is known as the reference pose (sometimes called a bind pose). The reference pose determines how points are initially assigned and weighted. It’s best to choose a reference pose that makes it easy to see and control how points will be assigned.
In the reference pose, the envelope is not deformed; however as soon as you start to move the deformers, the envelope begins to stretch and move.

After you have set an envelope, you can change the reference pose as described in *Changing Reference Poses* on page 106. You can use *Reset Actor* to return the envelope and its deformers to the reference pose as described in *Returning to the Reference Pose* on page 79.

If the envelope deformers are constrained to other control objects, then *Reset Actor* won’t work. In such cases, you should use the *Actions > Store > Transformations - Current Values* command to store an action for all your deformers and control objects, and *Actions > Apply > Action* to return to the reference pose.

For more information about actions in general, see *Chapter 3: Actions* in the *Nonlinear Animation* guide.

**Freezing Transformations and Geometry**

If you have frozen the transformations of an object, you will likely obtain undesirable results when you use it as an envelope. If this is the case, try freezing its geometry with the *Freeze* button on the Edit panel before using it as an envelope.

Also, if you deformed an object by a curve or surface, you should freeze its operator stack before using it as an envelope.
Assigning Envelopes to Deformers

When you define an object as an envelope on a set of deformers, two things happen:

- Each point on the envelope is assigned to one or more deformers. This determines which deformers influence that point.
- Each point’s weight is distributed between its deformers. This determines how much each of the deformers influence that point.

Setting Envelopes

To assign an envelope to deformers

1. Select the objects, hierarchies, or clusters to become envelopes.
2. Choose Deform > Envelope > Set Envelope from the Animate toolbar.
   - If the current construction mode is not Animation, you are prompted to apply the envelope operator in the animation region of the operator stack anyway. Keeping the envelope operator in the animation region ensures that it is above shape animation as well as any topology operators, giving best results in most cases. It also prevents the envelope operator from being removed if you freeze the modeling or create shape clips.
   - It is recommended to apply the envelope in the animation region unless you will be using the envelope deformation as a modeling tool.
   - For more information about construction modes in general, see Construction Modes and Regions in Chapter 3 of the Modeling & Deformations.
3. Pick the objects that will act as deformers. You are not restricted to skeleton bones; you can pick any object. Left-click to pick individual objects and middle-click to pick branches.
   - You can also pick groups in the explorer—this is equivalent to picking every object in the group individually.
   - If you make a mistake, Ctrl+click to undo the last pick. If you change your mind, press Esc to cancel the entire operation.

You can create a group for your deformers first, then simply pick the group when setting an envelope.

However, if you later add an object to the group, it is not automatically included as a deformer in the envelope assignment. To add deformers after enveloping, see Adding Deformers on page 107.
4. When you have finished picking deformers, right-click to terminate the picking session. The Automatic Envelope Assignment property editor opens.

Each deformer is assigned a color, and points that are weighted 50% or more toward a particular deformer are displayed in the same color.

- To see the deformer colors on points in a viewport, click the eye icon in its menu bar and make sure that Weight Points is on.

- To change the deformer colors, see Modifying Display Colors on page 104.

5. Set the Number of Skeleton Objects to which each point on the envelope is assigned initially.

For example, if you leave this parameter at the default of 2, each point is controlled by the two nearest deformers and its weight is divided between them.
6. Select the assignment method.

- **Distance-based** assigns weights based on the distance between points and deformers.

- **Normal-based** considers the directions of the surface normals in addition to distance.

This option is useful, for example, when enveloping things like legs and fingers. This method never assigns points from adjacent appendages to a deformer. The restriction is that the deformers must be entirely inside the envelope.

7. Move the deformers to see how the envelope deforms.

If necessary, you can now change the deformers to which points are assigned, as well as modify the envelope weights using the methods described in the next few sections.
Assigning Envelopes to Deformers

Modifying Envelope Options

At any time, you can redisplay the Automatic Envelope Assignment property editor and change the options you initially chose (Number of Skeleton Objects, Assignment Method).

To do this, select the envelope and choose Edit > Properties > Animation Properties or Edit > Properties > Modeling Properties and click the Automatic Envelope Assignment tab.

In the explorer, the Automatic Envelope Assignment node is located under the Envelope Weights cluster rather than in the geometry operator stack.

Returning to the Reference Pose

The Reset Actor command returns the envelope and its deformers to their reference poses.

To reset an actor

1. Select an envelope or any of its deformers.
2. Choose Deform > Envelope > Reset Actor from the Animate toolbar.

If the envelope deformers are constrained to other control objects, then Reset Actor won’t work. In such cases, you should use the Actions > Store > Transformations - Current Values command to store an action containing the current transformations of all your deformers and control objects in the reference pose before you move anything, and Actions > Apply > Action to return to the reference pose.

For more information about actions in general, see Chapter 3: Actions in the Nonlinear Animation guide.
Selecting All the Deformers

Sometimes you need to select all the deformers of a particular envelope. You can do this with a single command.

To select all the deformers of an envelope

1. Select the envelope.

2. Choose Deform > Envelope > Select Deformer from Envelope on the Animate toolbar.

Removing Envelopes

To remove an envelope, select or it and choose Deform > Envelope > Remove Envelope. This removes the envelope deformation and any modifications you have made to assignments and weights. Of course, you can always undo the envelope removal.
Modifying Point Assignment Using Bounding Volumes

You can use bounding volumes to modify the assignment of envelope weights. Points can be included or excluded from a deformer’s influence based on whether they are inside or outside a specified volume. Bounding volumes must be implicit objects.

While you don’t need to use bounding volumes, they can provide more control over the assignment of deformers than the default initial assignment. If the initial envelope weights are recalculated, these bounding volumes are preserved.

Bounding volumes are always considered in relation to the reference pose; that is, it doesn't matter whether envelope points are currently inside or outside bounding volumes; what matters is whether they would be inside or outside if the envelope was in its undeformed reference pose.

Bounding volumes affect the initial weighting of envelope points until you freeze the weights as described in Freezing Envelope Weights on page 103.

Bounding Groups

Every deformer has three groups in its Bounding Volumes property. To display these groups, make sure that both Properties and Groups are checked in an explorer’s Filters menu.

You define bounding volumes for a deformer by adding and removing implicit objects from these groups, either by using commands on the Deform > Envelope menu or by modifying the groups directly. These groups are similar to other groups except that they cannot be copied.

For more information about groups in general see Grouping Objects in Chapter 9 of the Fundamentals guide.
Types of Bounding Volumes

There are three types of bounding volumes:

- **Inclusive**—all envelope points within the bounding volume are assigned 100% to the deformer. They are not influenced by any other deformer (unless they are also inside an inclusive bounding volume of another deformer).

- **Exclusive**—all points within the bounding volume are prevented from being assigned to the deformer.
• **Limit**—points outside the bounding volume cannot be assigned to the deformer; points inside the volume may be assigned to the deformer as well as to other deformers as applicable.

**Creating Bounding Volumes**

Bounding volumes can be either implicit spheres or cubes.

*To create a bounding volume*

Do one of the following:

- Choose one of the commands on the **Deform > Envelope > Create Bounding Volume** submenu of the Animate toolbar: **Box** or **Sphere**.

  or

- Choose **Get > Primitive > Implicit > Cube** or **Get > Primitive > Implicit > Sphere** from any toolbar.

The commands on the **Deform > Envelope > Create** menu do exactly the same thing as the ones on the **Get > Primitive > Implicit** menu.

However, the commands on the **Deform > Envelope > Create** menu make it clear which implicit shapes are supported as bounding volumes.

Once you have a bounding volume object, you can scale, rotate, and translate it into place before associating it to a deformer.

**Associating and Dissociating Bounding Volumes**

You can associate or dissociate bounding volumes and envelope deformers by picking interactively using the **Deform > Envelope > Associate Bounding Volume** commands or manually by adding and removing implicit objects from the deformers’ bounding groups.

After you have associated bounding volumes, you can adjust the scaling, rotation, and translation of the bounding volumes to change the way points are assigned to deformers.
Before You Associate Bounding Volumes

Before you associate bounding volumes to deformers, make sure to start with the envelope and deformers in the reference pose—if necessary, use Deform > Envelope > Reset Actor. This allows you to see which points fall inside and outside the volumes in the reference pose.

It doesn’t matter whether envelope points are currently inside or outside bounding volumes when a character is in a particular pose or at a particular frame; what matters is whether the points would be inside or outside a bounding volume if the envelope was in its undeformed reference pose.

It’s not a good idea to parent or pose-constrain the volume objects to the deformers. In such cases, when the deformers move, the bounding volumes move as well, and points become reweighted as they fall inside and outside the new positions of the volumes with respect to the reference pose.

Associating and Dissociating Bounding Volumes Interactively

The commands on the Deform > Envelope > Associate Bounding Volume submenu let you associate and dissociate bounding volumes interactively by picking.

To associate and dissociate bounding volumes interactively

1. Select a bounding volume or a deformer.

2. Choose one of the commands on the Deform > Envelope > Associate Bounding Volume submenu: Inclusive, Exclusive, or Limit. For a description of the different types of bounding volume, see Types of Bounding Volumes on page 82.

3. Do one of the following:
   - If you initially selected a bounding volume, pick one or more deformers. Use the left mouse button to associate, and the middle mouse button to dissociate.
   
   or
   
   - If you initially selected a deformer, pick one or more bounding volumes. Use the left mouse button to associate, and the middle mouse button to dissociate. Note that you can pick any type of implicit, but only cubes and spheres work as bounding volumes.

4. When you have finished picking, right-click or press Esc to terminate the picking session.
Associating and Dissociating Bounding Volumes Manually

You can associate and dissociate bounding volumes and deformers manually by adding and removing the bounding volumes from the deformers’ bounding groups.

To see the bounding groups in the explorer, make sure that both Properties and Groups are checked in on the Filters menu and expand the deformer’s Bounding Volumes node.

To **associate bounding volumes manually**

Do either of the following:

- Drag an implicit cube or sphere onto the appropriate bounding group in the explorer.

  
  or

- Select both the object and the desired bounding group, then choose **Edit > Add to Group**.

To **dissociate bounding volumes manually**

Do any of the following:

- Right-click the bounding volume object within the group, and choose **Remove from Group**.

  or

- Select the bounding volume object, and click **Ungroup** on the Edit panel. If the object is a member of multiple groups, you must multi-select both the object and the group first.

  or

- If you no longer need the bounding volume object in your scene, you can just delete it.
Combining Bounding Volumes

You can combine objects in these bounding groups in various ways. Two examples are given here.

Overlapping Different Bounding Volumes

To achieve a progressively diminishing influence of a deformer, use an inclusive bounding volume around it, which is surrounded by a larger limit bounding volume.

- Points inside the inclusive volume are assigned 100% to the deformer.
- Points in the spillover area (outside the smaller inclusive volume but within the larger limit volume) are probably assigned to the deformer at something less than 100%, as they are farther away.
- All points outside the limit volume are prevented from being assigned to the deformer at all.
Overlapping Inclusive Bounding Volumes

Sometimes you may want to force some points to be assigned to certain deformers, but not so much that they move and deform in an unnatural manner.

To achieve this, create two or more inclusive bounding volumes that overlap near the joint of two bones. The points within the bounding volumes are assigned as follows:

- Points are assigned only to the specified deformers, regardless of the **Number of Skeleton Objects** value in the Initial Envelope Assignment property editor.
- Points in the overlapping region are weighted according to their relative proximity to the two deformers.
- Points that fall within only one volume are assigned with the usual 100% weighting to the associated deformer.
Changing Deformer Assignments Manually

You can change the deformers to which points are assigned. For example, you can specify a list of deformers for specific points or completely exclude points from the deformation.

This is another procedure that is optional, but provides more control over the assignment of deformers than the default initial assignment. Again, if the initial envelope is recalculated, these assignments are preserved.

Reassigning Points to Specific Deformers

You can select points on the envelope and assign them to specific deformers.

1. Select points on the envelope.
2. On the Animate toolbar, choose Deform > Envelope > Reassign Locally.
3. Pick one or more of the original deformers. Left-click to pick objects and middle-click to pick branches. If you make a mistake, Ctrl+click to undo the last pick.
4. If you change your mind, press Esc to cancel the entire operation.

You can only pick an object if it is already a deformer for the envelope. To add deformers to an envelope, see Adding and Removing Deformers on page 107.

4. When you have finished, right-click to end the picking session. The selected points are weighted to the new set of deformers.
Removing Deformers for Specific Points

You can set the weights of individual points on an envelope to zero for all deformers. This effectively excludes these points from the envelope deformation—they remain in the reference pose.

It also removes any manual modifications to the points’ weights that you may have made previously.

1. Select one or more points on the envelope.

2. Choose Deform > Envelope > Assign to No Deformer on the Animate toolbar. The weight assignments of the selected points are set to zero and the points are no longer affected by any deformers.
Modifying Envelope Weights

You can modify the weights of points on an envelope in several ways:

- By painting, as described in *Painting Envelope Weights* on page 93.
- By setting numerical values, as described in *Editing Weights Numerically* on page 97.
- By smoothing, as described in *Smoothing Weights* on page 100.
- By mirroring symmetrically, as described in *Mirroring Envelope Weights Symmetrically* on page 100.

When you have finished weighting, you can freeze the weights as described in *Freezing Envelope Weights* on page 103.

A good first step when modifying envelope weights is to apply a smooth operator on the entire envelope. This can often improve the weighting in many areas at once. After that, you can touch up specific areas with the other methods.

Tools for Modifying Envelope Weights

There are two main tools for modifying envelope weights:

- The weight paint panel is a tool that combines features from the weight editor, the brush properties, and the Animate toolbar into one convenient toolbar. You can use it either for painting weights or setting weight assignments numerically.
- The weight editor is a view that can show the assignments of all the points and deformers at once or just the selected ones. It allows you to set the weight assignments numerically.

In addition, the Brush property editor contains various options for controlling how weights are painted.
The Weight Paint Panel

The weight paint panel combines several features from the weight editor, brush properties, and the Animate toolbar. It is limited to displaying one envelope at a time.

- **Paint options (also available from Brush properties):**
  - Chose a paint mode.
  - Activate Paint tool.
  - Set paint density.
  - Set brush size.
  - Update continuously (on) or only when mouse button is released (off).

- **List of deformers:**
  - Pick a deformer for painting from the 3D views.
  - Display only current deformer's weight map.
  - Click to pick deformer for painting.
  - Right-click for other options.

- **Edit options (also available in weight editor):**
  - Change color of current deformer.
  - Set weight assignment of selected points to current deformer numerically.

- **Enveloping commands (also available on Animate toolbar):**
  - Smooth weights on object or selected points.
  - Reassign points to other deformers.
  - Freeze initial weight assignment and any modifications.

- **Right-click in the list of deformers for a menu with display and other options.**
Chapter 3 • Envelopes

To display the weight paint panel
- Click the weight paint panel icon at the bottom of the toolbar.
  or
- Press Ctrl+3.

The Weight Editor
The weight editor allows you to modify envelope weight assignments numerically. Unlike the weight paint panel, it can display the points and deformers of multiple envelopes at the same time.

To display the weight editor in a viewport
- Choose Weight Editor from the View menu of a viewport.
To display the weight editor in a floating window
Select one or more envelopes and do one of the following:

- Press Ctrl+e.
  or
- Choose Deform > Envelope > Edit Weights from the Animate toolbar.
  or
- Choose Application > Views > Weight Editor from the main menu.
  or
- Click Weight Editor on the weight paint panel.

Brush Properties
The Brush properties control the paint mode, brush size, and other attributes.

To display the Brush property editor
Do one of the following:

- Press Ctrl+w.
  or
- Choose Get > Properties > Brush Properties from any toolbar.

Painting Envelope Weights
You can use the Paint tool to adjust envelope weights. This lets you use a brush to apply and remove weights on points in the 3D views.

When you paint a stroke, a Weight Painter operator is applied to the envelope weights’ stack. Successive paint strokes are combined into a single operator to minimize stored data and increase performance.

If a new paint stroke cannot be added to the previous operator (for example, if the previous operator is a Smooth Envelope Weight operator instead of a Weight Painter), then a new Weight Painter operator is applied on top of the stack.

Overview of Painting Weights
This section provides a quick overview of painting envelopes weights. Each step is described in more detail in the sections that follow.

1. Select an envelope.
2. Activate the Paint tool.
3. Pick a deformer for which you want to paint weights.
4. If desired, set the paint mode.
5. If desired, adjust the brush properties:
   - Use the r key to change the brush radius interactively.
   - Use the e key to change the opacity interactively.
   - Set other options in the Brush Properties editor (Ctrl+w).

6. Click and drag to paint on points on the envelope. In normal (additive) paint mode:
   - To add weight, use the left mouse button.
   - To remove weight, either use the right mouse button or press Shift+left mouse button.
   - To smooth weight values between deformers, press Alt+left mouse button.

7. Repeat steps 3 to 6 for other deformers and points until you are satisfied with the weighting.

**Activating the Paint Tool**

*To activate the Paint tool*

Do one of the following:

- Press w.
- or
- Choose Get > Property > Paint Tool from any toolbar.
- or
- Click the Paint tool’s icon on the weight paint panel.

**Picking Deformers**

When you paint weights on an envelope, you are modifying how points are weighted to the current deformer. Picking a different deformer allows you to change how points are weighted to the other deformers of the envelope. The active deformer is highlighted in the 3D views.

*To pick a deformer for painting*

Do one of the following:

- Press d and then pick a deformer in a 3D view.
- or
- Click Pick on the weight paint panel and then pick a deformer in a 3D view.
- or
- Click a deformer’s name in the list on the weight paint panel.
Setting Paint Modes

You can set paint modes from the weight paint panel or the Brush property editor. There are five modes:

- **Add (Normal)** adds weight with the left mouse button.
  - Use the right mouse button or press Shift+left mouse button to remove weight.
  - Press Alt to smooth.
- **Abs (Set Weight)** sets weight to the current Opacity value.
- **Smooth** blends the weights between deformers. As an alternative, you can apply a Smooth operator as described in *Smoothing Weights* on page 100.
- **Reveal** scratches away the previous paint operator to reveal the weight "underneath" it.
- **Erase** removes the influence of the deformer. It is equivalent to Abs mode with Opacity set to 0. This mode is available only on the weight paint panel.

Be careful when erasing weight. Erase mode does not automatically redistribute the weight to other deformers, so points are partially assigned to no deformer until you add weight for a different deformer.

If you want to simply reduce the weight of a deformer on certain points while reassigning the weight to the other deformers that affect those points, it may be better to use the editing tools with the Normalize option instead of painting. See *Editing Weights Numerically* on page 97.

Setting Brush Properties

You can set the brush radius and opacity interactively while you are painting. In addition, you can set these and all other options in the Brush Properties editor. Certain commonly-used options are repeated on the weight paint panel for convenience.

To change the brush radius interactively

- Do one of the following while the Paint tool is active:
  - Middle-click and drag to the right to increase the radius, or to the left to decrease it. Alternatively if the middle mouse button is a wheel, roll it forward or back.
  - Press r and drag to the left or right.

The current radius size in Softimage units is displayed at the bottom of the 3D view.
To change the opacity interactively

- Press e while dragging back or forth with the left mouse button: a number at the bottom of the viewport indicates the current opacity setting.

The opacity controls how much weight is added or removed with each stamp of the brush.

To set other brush options

1. Open the Brush Properties editor by doing one of the following:
   - Press Ctrl+w.
   - Choose Get > Property > Brush Properties from any toolbar.

2. Change options as desired. For a complete description of every parameter, refer to Online Help by clicking the ? in the property editor. In particular:
   - Spacing controls the frequency of the brush stamps in a brush stroke.
   - Surface Coverage controls the degree to which the brush reaches around the sides of a 3D object.
   - The Refresh options determine how the weight map display is refreshed. On Mouse Up refreshes only when you release the mouse button to finish a paint stroke; this gives better performance but makes it more difficult to visualize the stroke. Interactive refreshes continuously as you drag, which may slow down your computer.

Painting Tips

Here are some quick tips to consider when painting weights:

- To paint weights on a subdivision surface created with the geometry approximation method, click the eye icon on a viewport’s menu bar and make sure that Polymesh Hulls is on.

To see the weight maps on the hull, make sure that Subdivision Surfaces is off.

- When painting on NURBS surfaces, you can increase performance by reducing the geometry approximation settings. The Paint tool uses the triangulation of the object to follow its surface. For more information, see Geometry Approximation on NURBS in Chapter 10 of the Modeling & Deformations guide.

- Use the Solo option on the weight paint panel to display the weight map for only the current deformer. This makes it easier to see which points the deformer influences and by how much.
Modifying Envelope Weights

If your envelope has multiple maps, for example, a weight map controlling a Push deformation in addition to an envelope weight map, then you may need to select the envelope weight map explicitly before you can paint on it. XSI remembers the last map selected or applied on each object.

A quick way to select the envelope weight map is to select the enveloped geometry object, then choose Explore > Property Maps from the Select panel and select the map to paint on.

Editing Weights Numerically

You can set envelope weights numerically using either the weight editor or the weight paint panel. The weight editor has a full set of controls, while the weight paint panel has a subset of these.

You can set a weight numerically by selecting a deformer and a point, then adjusting the Weight slider. You can use the Normalize option if desired.

Overview of Setting Weights Numerically

This section gives a quick overview of the process of editing weights numerically. The sections that follow provide more details.

1. Make sure the weight editing options and Normalize are set how you want them.
2. Specify the points you want to modify. The weight paint panel lets you modify the points that you select in the 3D views, while the weight editor gives you more options for specifying points and deformers.
3. Use the slider to adjust the weights.

Setting Weight Editing Options

When you edit weights numerically, the weight options determine how the weights are affected by values you set:

- **Abs** (Absolute) sets the weight to exactly the value you apply.
- **Add** (Additive) adds or subtracts an amount to the current weight.
- **Add %** (Add Percentage) adds or subtracts a percentage of the current weight.

The Normalize Option

The Normalize option automatically adjusts weight values so that they always add up to 100%. For example, suppose a point is weighted 50% to the shin and 50% to the femur. If you set the shin weight to 40%, the femur weight is automatically changed to 60% if Normalize is on.

There are two situations when you would want to turn Normalize off:

- A point is weighted to three or more deformers, and you want to set numerically exact values. After you have set the weight for the first deformer, you do not want that value to change when you set the weight for the second deformer.
You want a “partial” envelope effect. If a point’s weights do not add up to 100%, it behaves as if it is partly assigned to no deformer. Its position is a mix of the reference pose and the positions determined by its existing deformers.

When Normalize is on and you adjust a weight value, the other values are modified proportionally (that is, they are each multiplied by the same factor necessary to keep the total equal to 100). In particular, this means that if a point is already weighted 0 to a deformer, its weight will not change by being normalized.

Even when Normalize is off, a point’s weight assignments cannot add up to more than 100.

Selecting Cells in the Weight Editor

In the weight editor, each cell shows the value by which the corresponding point (represented by the row) is weighted to the corresponding deformer (represented by the column). Selected cells are affected by the weight slider and are highlighted. Non-zero cells are shaded.

To control the weight editor display

Do any or all of the following:

- Right-click in the left-most column to sort the columns according to their values in that row, to hide points, or to expand or collapse envelopes.
- Right-click in the top row to sort the rows according to the values in that column, or to hide deformers.
- Use Show to display only those rows and columns containing the selected cells.
- Use Show All to display all rows and columns.
- Use Lock to prevent the weight editor from automatically updating when you select a different deformer.
- Use Update to show the currently selected envelopes if Lock is on.
- Use Clear to empty the weight editor display.

To view and select cells in the weight editor

The easiest way to select cells for editing is to use the 3D views. Depending on your viewing options, the weight editor updates automatically to show exactly what you are working on.

- Select points or deformers in the 3D views. The corresponding non-zero weights are automatically selected in the weight editor.
- If Filter is off, zero weights are selected as well.
- If Focus is on, the table is automatically updated to show only those rows and columns containing the selected cells.
To select cells manually in the weight editor

You can also select cells manually directly in the table. The corresponding points and deformers are highlighted in the 3D views.

- To select a single cell, click it.
- To add cells to the selection, Ctrl+click them.
- To select a horizontal, vertical, or rectangular range of cells, you can drag across them from one end (or corner) to the other end (or diagonally opposite corner). Alternatively, you can select one end (or corner) and then Shift+click on the other end (or diagonally opposite corner).
- If Highlight is on, the points corresponding to the selected cells are highlighted in the 3D views.
- To select the points that correspond to the cells that are currently selected in the weight editor, click Sync.

Setting Weights Numerically

You can set weights numerically in the weight editor for all highlighted cells or an individual cell. You can also set weights for the selected points and deformer using the weight paint panel.

No matter which method you use to set the weights, the values are adjusted according to the current options (Normalize, Abs, Add, Add%). Also, remember that the deformer weights for an individual point cannot add up to more than 100, so depending on the existing values you may notice that the new weights are not set to the exact value that you typed—instead, they are set to the highest possible value while keeping the total equal to 100.

To set weights for highlighted cells in the weight editor

1. In the envelope weights editor, select one or more cells.
2. Use the Weight slider immediately below the weight editor’s command bar to adjust the values.

To set weights for specific cells in the weight editor

1. Right-click in a cell. The cell contents are selected for editing.
   - If the cell was highlighted, all highlighted cells are affected.
   - If the cell was not highlighted, then only that cell is affected.
2. Enter a new value.

To set weights for selected points and deformer in the weight paint panel

1. Make sure that the desired deformer is active in the Deformers list of the weight paint panel.
2. Select the desired points in the 3D views.
3. Adjust the Deformer Weight value in the weight paint panel.

**Smoothing Weights**

There are two ways to smooth weights:

- By painting in Smooth mode as described in *Painting Envelope Weights* on page 93.
- By applying a Smooth Envelope Weight Operator as described here.

To apply a Smooth Envelope Weight operator to points selected in the 3D views

1. Select the envelope or the points on the envelope that you want to smooth.
2. Do one of the following:
   - Choose Deform > Envelope > Smooth Envelope Weights from the Animate toolbar.
   - Click Apply Smooth on the weight paint panel.

The Smooth Envelope Weight Operator property editor opens.

To apply a Smooth Envelope Weight operator to points selected in the weight editor

1. Select some cells in the weight editor.
2. Click Smooth on the weight editor. The Smooth Envelope Weight Operator is applied to the points corresponding to the selected cells and its property editor opens.

**Mirroring Envelope Weights Symmetrically**

You can mirror the envelope weighting symmetrically. This lets you set up the weighting on one half of your character and then copy the weights to the corresponding points and deformers on the other half.

To mirror weights, your envelope must have:

- A symmetry map to establish the correspondence between points.
- A symmetry mapping template to establish the correspondence between deformers.

**Creating a Symmetry Mapping Template**

A symmetry mapping template establishes the correspondence between deformers. At the same time that you create one, you can allow XSI to automatically add mapping rules between deformers based on position and hierarchy structure, or you can create your own mapping rules later. You also have the option of creating a symmetry map for points at the same time as the mapping template.
Modifying Envelope Weights

To create a symmetry mapping template

1. Select the envelope.

2. Choose Deform > Envelope > Create Symmetry Mapping Template from the Animate toolbar.

3. In the Create Symmetry Map dialog box:
   - Deselect Fill Symmetry Template if you do not want XSI to automatically create mapping rules between deformers. Either way, you can always add, change, and delete rules later.
   - Set Symmetry Axis to the desired plane of symmetry: YZ (X = 0), XZ (Y = 0), XY (Z = 0).
   - Deselect Create Symmetry Map if your envelope already has a symmetry map that you want to use for establishing the correspondence between points.

   For more information about applying symmetry maps manually, see Manipulating Components Symmetrically in Chapter 5 in the Modeling & Deformations guide.

4. Click OK. The Symmetry Mapping Template property editor opens. You can use this property editor to add, change, and delete mapping rules as described in the next section, Editing Symmetry Mapping Templates.

Editing Symmetry Mapping Templates

Use the Symmetry Mapping Template property editor to add, change, and delete the mapping rules that establish correspondence between deformers when mirroring envelope weights. When you have finished editing, you can validate your changes to make sure that you have not entered a deformer’s name incorrectly.

Note that the rules are bidirectional: for example, if you have a rule mapping LShin to RShin, then you do not need a second rule mapping RShin to LShin.

To display the Symmetry Mapping Template property editor

The Symmetry Mapping Template property editor opens automatically when you create a symmetry mapping template. You can also display it manually, for example, if you need to add or modify rules later.

1. Locate the SymmetryMappingTemplate node in an explorer. It is under the model that contains the envelope and its deformers.

2. Click the node’s icon. Its property editor opens.
Chapter 3 • Envelopes

To add a rule

1. Select the row above which you want to place the new rule by clicking its number in the first column.

2. Replace the default text `<from>` in the Map From column with the name of a deformer.

3. Replace the default text `<to>` in the Map To column with the name of the corresponding deformer on the other side of your character.

To modify a rule

- Click in the Map From or Map To column of the corresponding row to change the name of a deformer.

The drop-down list contains all deformers on the envelope; you can select one from this list or enter a name. Use the value `<unmapped>` if you do not want the weight assignments of a particular deformer to be mirrored.

To delete a rule

1. Select the row to delete by clicking its number in the first column.

2. Click Delete Rule.

To validate your changes

- Click Validate.

- Valid deformer names are displayed in black.

- Invalid deformer names are displayed in red. This probably means that you made a typing error when entering the name of a deformer.

- Rules that have not been validated yet are displayed in gray.

Mirroring Envelope Weights

Once your envelope has a symmetry map and a symmetry mapping template, you can mirror weights from selected points on one side of your character to the other side. The effect is not persistent—if you change the envelope weights later, you need to reapply this command.

Symmetry maps are not updated after topological operations. If your envelope already has a symmetry map but you subsequently made modifications that created or removed components on the object, you should delete the old symmetry map and create a new one before mirroring weights.
Modifying Envelope Weights

To mirror envelope weights

1. Select the points whose weight you want to mirror to the other side.
2. Choose Deform > Envelope > Mirror Weights from the Animate toolbar.

   If the envelope does not have a symmetry mapping template, you are prompted to create one now. Click Yes and refer to Creating a Symmetry Mapping Template on page 100 for details.

   The weights you defined for the selected points are transferred to the corresponding points and deformers on the other side of the envelope. A Weight painter operator is applied to the envelope weight map.

Freezing Envelope Weights

When you freeze envelope weights, the weight map’s operator stack is collapsed, removing the original Automatic Envelope Assignment property along with any Weight Painter, Modify Envelope Weight, and Smooth Envelope Weight operators that have been applied. This reduces the amount of stored data and increases performance, but also has a number of other effects:

- The initial envelope weights can no longer be recalculated—it’s as if the envelope was imported as is.
- You cannot add new bounding volumes. In addition, you cannot change weight assignments by moving any previously-existing bounding volumes.
- If you change the reference pose, you can no longer change the initial envelope weights based on the new pose.
- If you add a deformer to an envelope, you can no longer recalculate the weights automatically. The envelope points are all weighted 0 to the new deformer, and you must assign weights manually.

However, you can still add new paint strokes, smooth weights, and edit weights numerically after freezing. In addition, you can still reassign points locally to other deformers.

To freeze an envelope’s weights using the weight paint panel

1. Select the envelope whose weights you want to freeze.
2. Click Freeze Weights on the weight paint panel.

To freeze an envelope’s weights using the weight editor

- Choose Edit > Freeze from the weight editor’s command bar.

   All envelopes that are currently active in the weight editor become frozen. Note that these might not be the currently selected envelopes if Lock is on.
To freeze an envelope’s weight stack from a specific operator down

1. In an explorer, make sure that Clusters is checked on the Filters menu.
2. Locate the envelope weight operator stack. By default, it is under Clusters\EnvelopWeightCls, which is under the envelope’s Polygon Mesh or NURBS Surface Mesh node.
3. Select Envelope_Weights if you want to freeze the entire stack, or an individual operator under it to freeze from that point down.
4. Click the Freeze button on the Edit panel.

Modifying Display Colors

XSI automatically assigns different colors to each deformer. This color is also used to display points that are assigned more than 50% to the deformer, and when painting weights for the deformer. You can change the colors, which may be useful if you find that the colors of two nearby deformers are difficult to distinguish.

To change the display color of a deformer

- Click a color swatch in the weight paint panel or weight editor and use the color editor.

To display or hide weight colors on points

- Do one of the following:
  - To display or hide weight colors in a viewport, click the eye icon in its menu bar and choose Weight Points.
  - To display or hide weights in all viewports, choose View > Weight Points from the main menu.
You can use the commands on the File menu of the weight editor to save and load presets of envelope weights. This can be useful if you want to experiment with modifying weights—you can save the current weights and reload them later if you don’t like the results.

To share presets between different envelopes, the envelopes must meet the following conditions:

- They must have exactly the same topology. This includes both the number of points and their connections.

  If you added points after you created a preset, and then reapply the preset to the modified geometry, the new points are not weighted to any deformer until you assign them manually.

- Their deformers must have the same names.

The easiest way to meet these conditions is to simply duplicate a model containing an envelope and its deformers.
Changing Reference Poses

After an envelope has been assigned, you can change the reference pose of the envelope. The reference pose is the stance that the envelope and its deformers return to when you use the Reset Actor command. It is also the pose that determines the initial weighting of points to deformers based on proximity.

1. Select the envelope.

2. If desired, mute the envelope deformation. This temporarily prevents the deformation of the envelope and gives better visual feedback. To do this, do one of the following:
   - In an explorer, right-click the Envelope Operator and activate Mute.
   - On the Select panel, choose Selection and click the icon of the Envelope Operator to open its property editor. Activate the Mute option. Click the Lock icon to prevent this property editor from being recycled so that you can easily unmute it later.

3. Adjust the positions of the deformers and envelope as desired.

4. Do either or both of the following:
   - If you have moved a deformer, then to reset its reference pose, select it and choose Deform > Envelope > Set Reference Poses from the Animate toolbar. Repeat for other deformers as necessary. Note that you can also branch- or tree-select deformers in hierarchies before choosing this command, as well as select a group containing the deformers.
   - If you have moved the envelope object or points on it, then to reset its reference pose, select it and choose the Deform > Envelope > Set Reference Poses again.

If the envelope weights have not been frozen, the initial weights are recalculated based on the relative positions of the new reference poses of the deformers and envelope. Any manual adjustments you made to the deformer assignments and envelope weights are preserved.

5. Unmute the Envelope Operator.
Adding and Removing Deformers

Adding and Removing Deformers

After you have applied an envelope, you can add and remove deformers.

Adding Deformers

When you add deformers to an envelope, you have the option of recalculating the original initial weights. If the weights are recalculated, any modifications you made to the weights and deformer assignments are preserved.

1. Select the envelope.
2. Choose Deform > Envelope > Set Envelope from the Animate toolbar. If there is already an envelope deformation on the object, the Envelope dialog box opens.
3. Do one of the following:
   - If you do not want to recalculate the initial weight assignments, leave the Automatically Reassign Envelope When Adding Deformers option off. The new deformers are added to the list of deformers, but no points are assigned to them automatically. You can assign points to deformers manually as described in Reassigning Points to Specific Deformers on page 88.
   or
   - If you do want to recalculate weights, turn this option on. The initial weight assignments of all the points on the envelope is recalculated and any modifications you have made are reapplied on top of the new initial weights.

Note that this option has no effect on envelopes that have been frozen or imported from SOFTIMAGE|3D. Envelope weights cannot be automatically recalculated in these cases.

   If you have modified weights manually, you should leave this option off. Otherwise, the result of the new initial weighting combined with the manual modifications will probably not give good results.

   If the envelope weights have been frozen, this option makes no difference. Envelope weights cannot be automatically reassigned if the Automatic Envelope Assignment operator has been removed.
4. Click OK to close the Envelope dialog box.
5. Pick the objects that will act as new deformers. Left-click to pick individual objects, and middle-click to pick branches.
   - If you make a mistake, Ctrl+click to undo the last pick.
   - If you change your mind, press Esc to cancel the entire operation.
6. When you have finished picking deformers, right-click to terminate the picking session.
Removing Deformers

After you remove a deformer, the envelope is no longer affected when the deformer is moved. The weights of points that were assigned to the removed deformer are normalized among the remaining deformers so that the total weights remain 100.

To remove deformers

1. Select the envelope.
2. Choose Deform > Envelope > Remove Deformers from the Animate toolbar.
3. Pick the deformers to remove. Left-click to pick individual objects, and middle-click to pick branches.
   - If you make a mistake, Ctrl+click to undo the last pick.
   - If you change your mind, press Esc to cancel the entire operation.
4. When you have finished picking deformers to remove, right-click to terminate the picking session.
Modifying Enveloped Objects

Sometimes, after carefully assigning weights manually, you discover that you need to make a substantial change to the enveloped object, such as adding points. Luckily, you do not need to redo all your weighting—you can add and move points after enveloping using various techniques.

There are also some issues related to transforming envelopes (rather than the deformers), but it is still possible to scale an envelope if necessary.

Deforming and Moving Points on Envelopes

If you want to apply a deformation or move points on an enveloped object, make sure to first set the construction mode based on what you want to accomplish. For example:

- If you want to modify the base shape of the envelope, set the construction mode to **Modeling**.
- If you want to author shape keys on top of the envelope, for example, to create muscle bulges, set the construction mode to **Secondary Shape Modeling**.

For more information about construction modes in general, see *Construction Modes and Regions* in Chapter 3 of the *Modeling & Deformations* guide.

For more information about construction modes and shapes, see *Creating Shapes in the Shape Modeling Construction Modes* in Chapter 4 of the *Nonlinear Animation* guide.

If you move a point above an Envelope operator and do not bake the resulting MoveComponent operator into a shape, the result may not be correct if you later move the deformers. This is because the MovePoint operator stores data as an offset from the object center, and does not take into account the envelope deformation.

To solve this problem, make sure that **Modify > Component > Relative Mode** is on before moving points on an envelope. For more information, see *Relative Mode* in Chapter 5 of the *Modeling & Deformations* guide.
Transforming Envelopes

Envelopes behave differently from other objects when you transform them:

- If you try to scale, rotate, or translate an envelope, it appears to be unaffected. This is because all of its points are assigned to the stationary deformers.

  If Centers are visible in a 3D view, you can see that the envelope is “really” being transformed.

- If you transform an envelope in Center mode, it appears to react in the opposite way. This is because transforming a center actually applies a compensation to the object’s points.

In general, it is not recommended to transform an envelope because you may get strange results if you later reset the reference pose. Instead, you can put the envelope and deformers in the same hierarchy or model and then transform the whole tree.

Scaling Envelopes

If your envelope is too big or too small, you can fix it by making it a child of the chain root, and then branch-selecting and scaling the chain root.
**Enveloping Tips & Tricks**

Here is some information you may find useful when working with envelopes.

**X-Ray Mode**

X-ray mode is a viewing option that lets you see through objects in Constant, Shaded, and Textured display modes. This is very useful when working with envelopes because you can see and select the underlying deforming elements while still seeing the shaded surface of the envelope.

![X-ray mode with Overlay type](image)

*To turn x-ray mode on or off*

1. Open the Camera Display property editor using one of the following methods:
   - To affect a single viewport, choose **Display Options** from the Display Type menu in the upper-right corner of the viewport.
   - Or
   - To affect all viewports, choose **View > Display Options (All Cameras)** from the main menu bar.

2. On the Display Mode page, select the **XRay Shading** option and choose an **XRay Display Type**:
   - **Overlay** draws the deforming elements on top of the envelope.
   - **Screen** draws a semi-transparent envelope so you can see the deforming elements inside.
Chapter 3  •  Envelopes

**Muting Envelopes**

You can mute envelopes in the same way as other deformations. This gives you faster performance because you can pose a skeleton without continuously updating the envelope as you work.

**To mute an envelope while posing the deformers**

1. Select the envelope.
2. Do one of the following:
   - In an explorer, right-click the Envelope Operator and activate **Mute**.
   - On the Select panel, choose **Selection** and click the icon of the Envelope Operator to open its property editor. Activate the **Mute** option.
   
   Click the Lock icon to prevent this property editor from being recycled so that you can easily unmute it later.
3. Pose the deformers.
4. When the skeleton is in the desired position, unmute the envelope.

**Combining Envelopes with Soft Body Deformations**

If you want to combine a soft body and an envelope deformation on an object, you should apply the soft body operator before the envelope.

Another technique you can use is to apply soft body to an envelope deformer. Note that you cannot apply soft body directly to a skeleton because it has no points; however, you can parent objects to bones, apply soft body to the objects, then use them as envelope deformers.

**Using Volume Deformations with Envelopes**

If you are parenting volume deformers to a skeleton and using them to create bulging muscles on your envelope, make sure that **Envelope Mode** is active in the Proportional Volume Operator property editor. This option makes the volume deformation relative to the parent (in this case, the skeleton deformer), so that the volume deformation gives the expected results as the skeleton moves.

It’s also a good idea to always apply the volume deformation to a cluster, so that the volume deformer does not affect unwanted areas of the envelope as the skeleton bends.

For more information about volume deformations in general, see *Manipulating Points by Volume* in Chapter 5 of the *Modeling & Deformations*.

**Importing Envelopes from SOFTIMAGE|3D**

Because XSI does not support Bézier, Cardinal, or B-spline patches, envelopes on these types of surfaces are lost when they are imported and converted.

However, you can convert the patches to NURBS in SOFTIMAGE|3D and use the **Skin > weightCopy** and **weightPaste** commands before importing the converted NURBS envelope into XSI.
Chapter 4  Animating Skeletons
Chapter 4  Animating Skeletons

Animating Skeletons with Kinematics

Skeletons provide an intuitive way to pose and animate your model. A well-constructed skeleton can be used for a wide variety of poses and actions, in much the same way as the all-purpose skeletons in our bodies can.

How parts of the skeleton move relative to each other is determined by the way your skeleton hierarchy is built, whether and how objects are constrained to each other, and if you have created any expressions to govern their motion.

Before you start animating your character, it is important to understand how animating transformations work in XSI. There are several issues related to local and global animation, as well as animating transformations in hierarchies (such as skeletons). For information on these topics and more, see Animating Transformations in Chapter 2 in the Animation guide.

Forward and Inverse Kinematics

Chains are manipulated using inverse kinematics (IK) and forward kinematics (FK). The method used depends on what part of the chain you manipulate.

* IK is a goal-oriented way of animating: you move the effector (end) of a chain in place and XSI calculates the angles at which the previous joints in the chain must rotate in order for the bone or end of the chain to reach its goal. This process is called solving.

  IK is an intuitive way of animating because it’s how you traditionally think of movement. For example, when you want to grab an apple, you think about moving your hand to the apple (goal-oriented).

  It’s also a quick way of creating poses which you can then animate, keying either the effector’s translation or the bones’ rotation values. For more information, see Animating with Inverse Kinematics on page 121.

* FK allows for complete control of the chain’s behavior because you rotate each joint’s position. Only the angle of the selected joint is affected; all other joint angles are preserved. Positioning a skeleton’s hand to grab an apple means rotating each joint in the arm, from the shoulder to the wrist and fingers.

  However, using FK allows you to create many types of movements that may not be possible to animate with IK alone. For more information, see Animating with Forward Kinematics on page 116.

![For a tutorial on creating a walk cycle for a character, see the Character Animation lessons in the Tutorials guide.](image)

You can blend inverse and forward kinematics on the same chain—see Blending Between FK and IK Animation on page 127.
You cannot key the \textit{global} position or orientation of bones; however, if you wish to animate a bone globally, you can prevent the bone from inheriting its parent’s position or rotation. To do this, open the bone’s Local Transform property page and deselect the appropriate \textbf{Constrain} option (Position or Orientation). See \textit{Animating Transformations in Hierarchies} in Chapter 2 in the \textit{Animation} guide for more information.
Chapter 4 • Animating Skeletons

Animating with Forward Kinematics

When you animate with forward kinematics (FK), you move the skeleton into position, specifying the angle of each joint and then keying its rotational position (orientation). Each movement needs to be carefully planned to create the resulting animation. For example, to bend an arm, you start from the “top” and move down by rotating the upper arm bone, then the forearm bone, and finally the hand bone.

Bones in arm are rotated and keyed in order from top down to move from an outstretched position to raised with a flexed wrist.

Posing a Chain with Inverse Kinematics

While you often need to rotate each bone individually to pose a chain, in some cases you can speed up the process by using inverse kinematics. Move the chain’s effector to place the bones quickly as you like and then tweak the bones’ rotation before keying. When things are in the position you like, use the Key All Bone Rotations command (see page 118) to set rotation keys for all bones in the chain.

To blend FK with IK animation on the same chain, see Blending Between FK and IK Animation on page 127.

When to Use Forward Kinematics

While rotating each joint can sometimes be tedious, it offers you complete control over the orientation of each joint with FK, allowing you to create many types of movements that may not be possible to animate with IK alone.

With FK:

- You can key the exact orientation (in X, Y, and Z) of a joint. This prevents any surprises from occurring when 2D chains flatten on their resolution plane.
- You can control certain joints that are difficult to animate, such as shoulders.
- You can have a movement properly “follow through,” such as making a foot kick a soccer ball properly.
Animating Kraken’s Tentacles with FK

Let’s say you want to animate one of the tentacles on Kraken, the Evil Sea Monster. Currently, the tentacle is coiled, and you want to animate it straightening out.

If you were to use IK by keyframing the effector translation, parts of the tentacle would pass through each other.

Using FK, you would rotate each joint in the chain, starting from the chain root. Although more time-consuming than using IK, the tentacle will not twist into itself.

FK also lets you choose how you want the tentacle to uncoil—from a lazy extension (Kraken is swimming peacefully) to a whip-like snap (Kraken is attacking a submarine).
Chapter 4 • Animating Skeletons

To animate a chain using FK

1. Select a bone, click the **Rotate** button in the Transform panel, and rotate
the joint into position.

   or

   Translate the chain's effector (invoke IK) to move the bones into position,
and then tweak each bone's rotation if necessary.

2. Make sure that the rotation parameters are marked (activate the Rotate
tool if it isn’t already active), and save a rotation keyframe by choosing
**Animation > Set Key** from the Animation panel, clicking the keyframe
icon, or pressing **k**.

3. To animate your object, move the timeline pointer and continue to create
rotation keys for the bone.

   For information on changing the order of the bone’s rotation axes,
see **Changing the Bones’ Order of Rotation** on page 62. This may
sometimes help with gimbal lock problems.

**Keying the Rotation for Multiple Bones**

If you want to key the rotation of several bones at a time, do one of the following:

**To rotate multiple bones**

- Select the bones and rotate them interactively in the **Add** (additive)
  manipulation mode. This rotates the bones in relation to their parent.

  or

- Select the bones and open their Local Transform property editor (one
  property editor opens for the whole selection—just press **Ctrl+k**). On the
  SRT page, enter specific values in the **Rotation** text boxes. This is the same
  as rotating interactively in Add mode.

This time, the joints’ rotations are keyed with some recoil on the first four
joints, creating a whip-like effect. Notice how the second, third, and fourth
joints bend backward compared to their original angles. This creates the whip
effect, which would be impossible with IK.
To key the rotation of all bones in a chain at once

- Select any element of a chain and choose Create > Skeleton > Key All Bone Rotations to set keys for the rotation of all the bones in that chain at the current frame.

This means that you can select a bone, rotate it, select another bone, rotate, and so forth. Then choose the Key All Bone Rotations command without needing to select the specific bones first.

This may also be useful to do when posing a chain using IK but keying in FK. Select the chain’s effector and move it to place the bones quickly as you like. Then tweak the bones’ rotation before keying. When things are in the position you like, choose the Key All Bone Rotations command to set rotation keys for all the bones in that chain.

Aligning the Root’s Orientation to Match the First Bone

To make it easier to work with skeletons driven by FK, you can change the orientation of the root so that it matches the orientation of the first bone. The orientation of the root is changed so that the first bone’s rotation is set to 0,0,0 in its current position.

This means that when you set up the default pose for your character, you can get the first bone to that position by resetting the rotation. For example, if you want to lift the leg up 30 degrees, you set the rotation to 30.

You can also create a marking set that contains the rotation parameters for all the bones whose rotation you want to key. Then you can key from this. See Marking Parameters for Animation in Chapter 2 in the Animation guide for more information.
Chapter 4  Animating Skeletons

To align the root's orientation to the first bone's

1. Set the chain's first bone's rotation to how you want it.
2. Select any element on the chain and choose Create > Skeleton > Align Root to First Bone. The chain root's orientation changes to match that of the first bone's.

You can also choose Create > Skeleton > Align Root to First Bone When Drawing to automatically orient the root when you draw a chain—see Creating Chains on page 42.

Changing the Bone's Axis Alignment

When you create a chain in XSI, its X axis is aligned along the bone by default—that is, in the direction of the next joint (or if it is the last bone, the effector). However, you can change this so that the bone's Y or Z axis is used instead. You can also offset any axis by any value for a more arbitrary alignment.

For example, when you're importing motion capture data on a skeleton, the bone's Y axis is often the one that is aligned instead of X. To fix this, after importing the skeleton into XSI, you change its bone axis alignment to be Y instead of X.

To change the bone's axis alignment

1. Select one or more bones and open the Chain Bone property editor.
2. In the Bone Axis Offset group, do either of the following:
   - Click the appropriate Set Axis button to quickly set the X, Y, or Z axis as the axis for alignment.
   
   or
   
   - Set any value for the X/Y/Z sliders to offset the bone's axis alignment by an arbitrary value.
Animating with Inverse Kinematics

Inverse kinematics (IK) is an animation method that’s more straightforward than FK—you define the goal of the action by positioning the effector of a chain and XSI calculates how to position the rest of the chain to reach that goal. For example, to bend a leg, you drag the leg’s end effector and the rest of the chain follows.

You save a key for the translation of the effector at different times, and XSI solves the joint rotations, from the root to the effector, based on the effector’s position. You can key the translation of an effector as you would any other object.

To blend IK with FK animation on the same chain, see Blending Between FK and IK Animation on page 127.

When to Use Inverse Kinematics

IK makes it simple and fast to move characters, but you often need to tweak the movements later to compensate for any unwanted movements. These problems occur because there is often more than one way for the chain to reach the goal set for it by the effector, or because the effector goes from position to position, rather than allowing the bones to rotate in a natural arc as you get with FK. Luckily, there are many tools in XSI to help you fix these problems, as described in this chapter.

You can use IK to do the following:

- You can easily try out different movements. Dragging an effector to reach a goal is intuitive for certain types of movements. Consider how you reach for something: you think about moving your hand to grab the object, not moving your shoulder first and eventually your hand.

- Quickly animate simple movements, including 2D chains that have a limited range of movement.

- Set up poses for a chain by positioning the effector, then keying the bone rotations (FK).
Basic Concepts for Inverse Kinematics

There are a number of issues about the way a chain behaves when you invoke IK. Two fundamental concepts you should understand before you start working in IK are a chain’s preferred angles and its resolution plane.

Preferred Angles

When you draw a chain, you usually draw it with a bend (as shown on the left) to be able to predict its behavior when using IK. This bend is called the chain’s preferred angle. When you move the effector, XSI tries to preserve the preferred angle as much as possible. For example, you can prevent a character’s elbow from bending unnaturally because bones are prevented from rotating backward against their joints’ preferred angles when using IK.

As well, having a home state ensures a solution that you can reproduce: whenever the effector comes back to the same place, it always goes to the same chain state.

The preferred angles of a joint are local rotation angles. These angles influence the behavior of the entire chain, depending on the selected IK behavior. For more information on IK behavior, see Choosing How an IK Chain Reaches Its Goal on page 132.

- When using 3D chains, the preferred X, Y, and Z angles of the joints are used, since 3D joints work like ball joints.
- When using 2D chains, only the preferred Z angle of a joint is used. Each joint is constrained to remain on a single plane and can rotate only around its local Z axis when the effector is moved, except the first joint.

To change the preferred angle, see Changing the Joint’s Preferred Angle on page 137.

The Chain’s Resolution Plane

With 2D chains, the preferred axis of a chain (see page 144) is perpendicular to the plane in which XSI tries to keep the chain when moving the effector in space. This plane is referred to as the general orientation or resolution plane of a chain because it is in the space of this plane that the IK system resolves the joints’ rotations to get the effector where you want it.

To get an idea of a resolution plane, extend your arm in front of you and do a biceps curl: as your arm bends, it forms an angle at the elbow. That bend occurs in the plane perpendicular to your body. When you do the biceps curl, that plane (your arm’s resolution plane) remains static unless you change it by rotating your shoulder joint.
The resolution plane of a 2D chain is the plane in which the entire chain lies. This plane corresponds to the local XY plane of the first joint (and all subsequent joints) of a 2D chain.

The general orientation of a 3D chain is also influenced by its preferred rotation axis. However, the joints of a 3D chain do not necessarily lie on the same plane, as do the joints of a 2D chain.

You can rotate a joint in a 2D chain outside of its resolution plane using FK. However, if you invoke IK the chain will snap back to the resolution plane.

**Animating the Effector**

The basic way of animating IK is to set keys for the translation of the chain's effector. Translations on effectors of chains created in XSI are local to the effector's parent (by default, the chain root), as opposed to global in SOFTIMAGE|3D.

In other words, you can drag the entire hierarchy without the effector wanting to “stick” to the scene’s global coordinates as it does in SOFTIMAGE|3D. With the default XSI setup, you don't need to constrain a null to the effector and then animate the null. Of course, you can remove the effector from the hierarchy or parent it to the last bone as you are used to in SOFTIMAGE|3D.

By not having the effector tied to the last bone, you are free to create local animation on the effector that can be translated with its parent. However, many animators prefer to constrain effectors and bones to a separate hierarchy of control objects (control rigs) so that they never animate the skeleton itself directly (see Chapter 5: Character Rigging on page 163).
Chapter 4 • Animating Skeletons

If the effector is the child of an object outside of its own chain, you will not be able to control the effector by manipulating its parent: the parent acts as an “anchor” for any animation on the effector but it can’t create IK behavior (that is, it can’t solve the chain).

The only way to invoke IK is by directly manipulating or keying the effector, or by using the **Force IK** option (see *Forcing the Effector’s Position* on page 133).

For more information on how the effector works with IK, see *Choosing How an IK Chain Reaches Its Goal* on page 132.

**Animating the Effector’s Translation**

Animating the translation of the effectors is the most basic way of animating chains with IK.

For more information on animating transformations in general, see *Animating Transformations* in Chapter 2 in the *Animation* guide.

**To animate the effector’s translation**

1. Select the chain’s effector and click the **Translate** button on the Transform panel.
2. Save a translation keyframe by choosing **Animation > Set Key** from the Animation panel, clicking the keyframe icon, or pressing **k**.
3. To animate your chain, move the timeline pointer and invoke IK by moving the effector to another position and setting another key.

In this example, the chain is animated by keying the effector’s translation at frames 1 and 10. The joint angles for each frame are solved.

**Animating the Effector à la SOFTIMAGE|3D**

By default, when you create a chain in XSI, the effector is a child of the chain root, and you animate the effector in local space. However, if you want to animate as you do in SOFTIMAGE|3D, you can either:

- Draw chains using the **Create > Skeleton > SI|3D Drawing** command (see *Drawing Chains à la SOFTIMAGE|3D* on page 44).
or

- If the chains have already been created using the default command in XSI (see Creating Chains on page 42), you can make the effector a child of the last bone as it is in the SOFTIMAGE|3D structure.

When you animate the effector’s position, you do so in global space, as in SOFTIMAGE|3D. To do this, you must either:

- Deactivate the effector’s Local Transform options for inheriting its Position and Orientation from its parent (see Changing the Animation Inheritance in Chapter 2 in the Animation guide).

or

- Explicitly mark and key the effector’s global transformation parameters (local transformations are marked automatically when you use any of the controls on the Transform panel). See Marking Global Transformation Parameters in Chapter 2 in the Animation guide for more information.

### Animating on a Path

You can also translate chains by constraining them to a path or a trajectory:

1. Create a curve for a path and select the chain’s effector to be constrained to the path.

2. Choose Create > Path > Set Path in the Animate toolbar and follow the steps as described in Setting Objects on Paths in Chapter 7 in the Animation guide.

When the sequence is played back, the effector translates along the path just like any other object. However, since the effector is part of a chain, the principles of IK are applied to the rest of the chain.
Controlling an IK Chain’s Orientation

Although the orientation of the chain’s plane is set up when you draw a chain, there are a number of ways in which you can change it to solve certain problems. For example, you can change a chain’s preferred axis from the default Z or use special constraints to manipulate your chain to prevent flipping.

For instance, you may draw a two-joint arm chain vertically for a character’s arms to swing naturally when it walks.

- If this character raises its arm to point at the sky, the chain will remain oriented in a vertical plane.
- If, however, you want the arm to rotate to another plane, as if the elbow were to rotate up to achieve a more horizontal orientation, then you need to use a special constraint to control the joints’ orientation.

You can control the orientation of a chain to help you avoid problems with the chain flipping. Chain flipping occurs when the effector of a 2D or 3D IK chain gets close to the chain root. When this happens, the general orientation of the chain changes very quickly as the effector crosses the preferred axis.

There are three things you can do to help with this problem:

- Show the chain root’s critical zones—see page 34.
- Constrain the chain’s up-vector (direction), as described on page 138.
- Constrain the chain’s preferred axis of rotation, as described on page 144.
Blending Between FK and IK Animation

When you’re animating a skeleton, you may need to use both FK (bone orientation) and IK (effector position) animation methods on the same chain. For example, arms bend at the elbow like hinges when you use IK, but to get a proper swing from the shoulder, you need to use FK.

In XSI, there are two main ways in which you can have both IK and FK:

- Build two skeletons or rigs: one animated with IK, the other with FK.
  Store one action for the FK animation and one for the IK animation, then load them on separate tracks in the animation mixer.

  By weighting one against the other, you can decide how much influence at any time FK or IK has for controlling the character. See Chapter 5: Character Rigging on page 163 for general information on setting up rigs.

  or

- Blend between FK and IK using the Blend FK/IK slider, as described in this section.

  This is the simplest way of blending between IK and FK animation. This slider controls the influence that IK and FK both have on a chain, blending smoothly the results of bone rotation and effector translation to give you the best of both worlds.

You can use FK/IK blending for:

- Animating with rotations to get a good “whip” effect (FK), and then blending in specific grabbing/punching/kicking (goal-oriented IK) movements.

- Mixing goal-oriented movements against motion capture. The motion doesn’t maintain the “jitter” of the underlying animation because that’s what it is intended to do: precisely reach a point in space.

To blend between FK and IK animation on a chain

1. Animate the chain in FK (key the bone’s rotation parameters), as well as in IK (key the effector’s position parameters or use a position constraint, for example).

2. Open the chain’s Kinematic Chain property editor (select any chain element and press Ctrl+r).

3. Drag the Blend FK/IK slider to set the value you want between FK and IK. A value of 0 is 100% FK, while 1 is 100% IK.

   As you move the slider back and forth, you can see the smooth interpolation between the IK position and the FK-driven chain. You can then decide where you want the blend to start and finish by setting keys for this parameter.
Chapter 4 • Animating Skeletons

How FK/IK Blending Works

The FK is calculated and the bone orientations (preferred angles) from that solution (A) are used to calculate the IK to give a solution (B). The Blend FK/IK parameter then interpolates between solutions A and B in quaternion space to come up with a new set of bone orientations.

Keep in mind that if you pass through a critical zone with the FK, then the chain is likely to flip (not necessarily related to the blending).

Up-vector constraints are taken into account for the IK component of the solution that is blended.

Plotting FK/IK Blending

If you want to plot the blended FK/IK animation on a chain, you need to do the following:

1. Mark the correct parameters to get the correct results:
   - To plot FK, mark the bones’ local rotation parameters.
   - To plot IK, mark the effector’s local position parameters.

2. Choose Tools > Plot > Marked Parameters from the Animate toolbar.

See Plotting Animation in Chapter 2 in the Animation guide for more information.

If you have Affected By Last Bone selected for the effector’s orientation (see page 135), you can select Link with IK/FK Blending (see page 136). This option ties the effector’s FK/IK blend with the chain’s overall FK/IK blend you set here.

You can create marking sets on the bones’ rotation and effector’s position parameters to make it faster to plot the animation repeatedly—see Marking Parameters for Animation in Chapter 2 in the Animation guide.
Blending and Constraints

FK/IK blending works in conjunction with constraint blending, allowing you to blend the FK/IK against multiple constraints. These constraints could be blended themselves, so you can create fairly complex behaviors.

- On the effector, you can combine FK/IK blending with constraints that are also blended.
- You cannot use FK/IK blending if you use constraints to control a bone's orientation with other objects: the bone's orientation must be directly controlled with either fcurves or expressions. You can work around this by rigging with expressions from an object.

See Blending Constraints in Chapter 8 in the Animation guide for more information.

Blending IK with Motion Capture Data (FK)

If you are using motion capture data, it is almost always in FK because FK records the full motion of the bones. Instead of combining the worlds of IK and FK with a special blending rig (one rig for IK, one for FK, and a blend set of controls to a third target rig), you can use the Blend FK/IK slider, saving you a lot of time and effort.

You load the motion capture data either directly driving the bone rotations or as an action source in the animation mixer where you can manipulate it. Then you can move and key the effectors (or whatever drives the chain in IK). Usually, you want to key the blend to make a smooth transition from 100% FK to the IK correction, and back again.

Example

This example shows how you can author new IK on top of FK, which is a fairly common case, but you can also do the reverse: take existing IK animation and create new FK moves to blend against.

However, with this method remember that you need to set FK poses at the start and end of where you want to blend the FK in and out from the IK to avoid any pops.

1. Take one of the standard characters and drop a walk or run cycle motion available from the Net view on the character (see Using Ready-made Characters on page 24 for these). Because these predefined animations are motion capture, they are in FK.

2. Change the position of the hand effector at some point: position-constrain the effector to a target, and then alter the Blend FK/IK value (on the chain's Kinematic Chain page) to see it blend between IK (constrained) to the FK (full motion from the preset clip).
3. Key the Blend FK/IK value at 0 (full FK) where you want to start the transition to IK, then key it at 1 (full IK) where you want to end the transition.

In this way, you are using IK to alter the effect of an existing FK motion, and blending to avoid any snaps. Similarly, you can key the transition back from IK to FK.

**Ghosting FK/IK Blending**

To help you see how the chain is blending between its IK and FK states, you can use ghosting. Ghosts are shown for the full FK and full IK positions of the chains.
For more information on animation ghosting in general, see *Ghosting Animated Objects* in Chapter 2 in the *Animation* guide.

**To ghost the chain's FK and IK states**

1. Select the chain that has FK/IK blending on it and open its Kinematic Chain property editor (press Ctrl+r).

2. Select a ghost display type from the **FK/IK Ghosting** list:
   - **Chain** ghosts the chain's root, bones, and effector.
   - **Branch** ghosts all children of the chain root.

3. Do one of the following:
   - Choose the **Skeleton FK/IK Ghosting** command in the display types menu in any 3D view.
   - Choose **Display Options** in the display types menu to open the Camera Display property editor. Then click the **Ghosting** tab and select **Enable Skeleton FK/IK Ghosting**.

   If you activate ghosting in a 3D view but still don't see the ghosts, check that ghosting is active for the layer to which the objects belong—see *Activating Ghosting for Layers and Groups* in Chapter 2 in the *Animation* guide.

4. On the Ghosting page in the Camera Display property editor, set the ghost colors using the **Before Color / FK Color** and **After Color / IK Color** controls in the Keyframe Options group.
Chapter 4 • Animating Skeletons

Choosing How an IK Chain Reaches Its Goal

When you translate a chain’s effector to invoke inverse kinematics, XSI uses a solver—an algorithm that “solves” how the effector will reach its goal by calculating how the joint angles should change. You can select both the type of solver XSI uses, as well as the solver angle methods for calculating the joint rotations.

To change a chain’s behavior

1. Select any chain bone and press Ctrl+r to open the Kinematic Chain property editor.

2. Use the Solver list to change the chain’s behavior. You can use one of two solvers for a chain: the SOFTIMAGE|3D or XSI Solver (see next section for information on choosing between the two).

3. Choose one of three Solver Angle methods for calculating joint rotations when using IK (see Calculating Joint Rotations with the Solver Angle on page 133 for information on choosing among the three methods).

Choosing a Solver

You can choose to use either the SOFTIMAGE|3D or XSI Solver. Whichever solver you choose, it’s easy to switch between them. If you find that different solvers work best for different situations on the same chain, you can even set keys on the Solver options (click its animation icon) to animate the switching from one solver to the other at specific frames.

- The SOFTIMAGE|3D Solver uses the same algorithm as you are familiar with in SOFTIMAGE|3D to provide compatibility with that program. If you go back and forth between SOFTIMAGE|3D and XSI, you should use this solver to be sure that the angles are calculated the same way for your chains in both programs.

- The XSI Solver uses a different algorithm from the SOFTIMAGE|3D solver, usually providing quicker results when translating the effector. Because the XSI solver is based on the XSI architecture, it works better with elements that are different from (rotation limits and up-vector constraints) or didn’t exist in SOFTIMAGE|3D (such as joint stiffness, pseudo-roots, and bone rolls). It knows how to stabilize and solve the chain more effectively than the SOFTIMAGE|3D solver.

If you’re working exclusively in XSI, you may find this solver more efficient.
Calculating Joint Rotations with the Solver Angle

When you draw a chain, the angles at each joint are seen as the “home state”, known as the preferred angles, of the chain. When you use IK, these preferred angles are used as an initial solution for the IK solver. Reading those angles, the solver computes a solution that depends on these angles and the goal position (as well as any up-vector or preferred axis constraints). For more information, see Preferred Angles on page 122.

Choose one of three Solver Angle methods for calculating joint rotations when using IK:

- **Use Joint Rotations (SOFTIMAGE|3D Behavior)** changes the preferred angles if the joint angles are changed by any means other than IK (such as rotation), which will affect further IK joint calculations. This is the same behavior as in SOFTIMAGE|3D, and chains imported from SOFTIMAGE|3D automatically assume this behavior. IK and FK have the same priority with this method.

  If a subchain is to be solved in IK, expressions on this subchain are ignored. Otherwise, expressions take precedence over changes you make by directly manipulating the chain (transforming it), and direct manipulation changes takes precedence over fcurves (or other local orientation control).

- **Use Preferred Rotations from Joint Properties** calculates rotation values based on the joints’ preferred angles. This means that the joint rotations are always predictable; a given effector position always results in the same rotation values. IK has priority over FK with this method: FK only modifies the preferred angles, and this affects the final IK solution.

  The only way to stay in this joint rotation mode and change the preferred angle is to do so from the Kinematic Joint property editor (see page 137—you can see the result of doing so if Force IK is on).

- **Reuse Joint Rotation at Every Resolution Step** ignores the preferred angles and calculates rotation values based on the joints’ current angles. If you manipulate the chain in FK, and then IK, the chain won’t snap back to its preferred angle. When the chain is extended to its full length, translating the effector toward the root causes the chain to bend in one direction.

  Because unpredictable behavior can occur when you use IK, this method is really only practical for FK and is the option for doing “pose-to-pose” animation. Of course, you can use IK to place the chain approximately where you want, then rotate the bones (FK) and key their rotations.

Forcing the Effector’s Position

Select the Force IK option on the Kinematic Chain property page to trigger IK every time you modify the preferred angles from the Kinematic Joint property page (see Changing the Joint’s Preferred Angle on page 137). This way, you can immediately see the effect on the chain.

Using Force IK lets you test out the effect that manipulating bones has on the IK solution without actually adding animation to or manipulating the effector.
Chapter 4 • Animating Skeletons

If you select **Force IK**, you can temporarily force the effector to stay in place if it is not already animated. Thus, with any **Solver Angles** method, you can edit the preferred angles in the Kinematic Joint property editor or rotate the bones interactively.

When you’re animating long chains (especially 3D chains), the error tolerance on the solver can sometimes get too large and the chain’s effector may “pop” to another location. This may also be a problem when you have stiffness or rotation limits set on 3D chains.

To solve these problems, you can lower the solver’s error threshold, meaning that less distortion is allowed to occur as it solves the chain’s angles.

The error threshold affects only:

- 3D chains with more than one bone using either the SOFTIMAGE|3D or XSI solver.
- 2D chains with more than two bones using the SOFTIMAGE|3D solver.

**To change the solver’s error tolerance**

1. Open the chain’s Kinematic Chain property page.
2. Select **User Defined** as the Error Threshold **Type**.

3. Lower the **Threshold** value until the problem stops occurring. This value is in Softimage units, with 1 being one grid square.

You need to set the Threshold value relative to the chain’s length. This means that generally, the shorter the chain, the lower you need to set the value to solve any problem.

The amount that the camera is zoomed in or out of the view is also a consideration. For example, if the camera is zoomed in closely to the chain, you may need to use a very low value.

The speed at which the solver calculates the chain is affected by a combination of the chain’s length and the Threshold value. As the Threshold value gets lower relative to the chain’s length, the longer it takes to solve. For example, a long chain with a high value could take the same amount of time to calculate as a short chain with a low value.

If the value is too low, the solver may not reach a solution and the effector won’t reach the IK goal.
Setting and Blending the Effector’s Orientation

The rotation of the effector is important to the workings of an IK chain because the effector can be used for parenting other objects to a chain. By default, the effector’s orientation is the same as the last joint’s. In some cases, however, you may want this orientation to be different.

The **Effector Rotation** parameters in the Kinematic Chain property editor provide two methods for changing the effector’s orientation: you can rotate the effector according to global coordinates or to local coordinates to have the same orientation as the last bone.

### Setting the Effector’s Orientation Inheritance

To **change the effector’s orientation**

1. Select the chain and press Ctrl+r to open its Kinematic Chain property editor.
2. Determine the **Effector Rotation** behavior you want:
   - **Select** Affected by Last Bone if you want to use the **Offsets** settings to rotate the effector with respect to the last bone.
   - **Deselect** Affected by Last Bone if you want to rotate the effector freely, according to global coordinates.

For example, say you were using a two-bone chain for a character’s leg, with the effector acting as the foot:

- If you want to keep the foot parallel with the ground during a walk cycle, orient the effector globally by deselecting Affected by Last Bone.
- If you want to animate the angle of the foot with respect to the last bone, select Affected by Last Bone and animate the Offsets settings.

If you import a skeleton from SOFTIMAGE|3D and the effector’s rotation was constrained to the last bone, the **Affected by Last Bone** option is automatically selected. Otherwise, the skeleton uses global rotation coordinates (the option is off).
Blending and Linking the Effector’s Orientation

If you select Affected by Last Bone, you can blend the effector’s orientation between the orientation inherited from the last bone and the orientation coming from constraints or animation. For example, let’s say a character grabs a railing and then jumps over it. You would need a way of blending the hand effector’s orientation inherited from the last bone with the orientation constraint used on the effector to keep the hand on the railing.

You can also link the effector’s FK/IK blending with the chain’s overall FK/IK blending (see Blending Between FK and IK Animation on page 127).

To blend or link the effector’s orientation

1. Make sure that the Affected by Last Bone option is selected for the chain, as described in the previous section.

2. Select either of the Blend Effector Rotation options: Inheritance Blend or Link with FK/IK Blending.

• To blend the effector’s orientation against the last bone:
  - Set the Inheritance Blend value between 1 and 0. This value is blend of the effector’s orientation as taken from the last bone (plus any offset) against any other orientation affecting the effector, such as an orientation constraint on the effector.

  The default value is 1, which fully constrained to the last bone’s orientation. A value of 0 (zero) is fully constrained to anything else affecting the effector’s orientation, which has the same effect as if Affected by Last Bone was off.

• To link the effector’s orientation with the chain’s FK/IK blending:
  - Select Link with IK/FK Blending.

  The orientation of the effector is blended by using the Blend FK/IK value (see page 127) to dictate how much of the “scene level” effector orientation is to be used relative to the rotation given by the last bone.

  For example, let’s say you have animated a chain in FK, and you have also placed a pose constraint on the effector. You can animate the constraint position in and out with the IK/FK blend. If you then select Link with FK/IK Blending, it also animates in and out the rotation of the effector in sync with what you would expect.
Changing the Joint’s Preferred Angle

When you draw a chain, you usually draw it with a bend to be able to predict its behavior when using IK. This bend is called the chain’s preferred angle, as described on page 122. XSI tries to stay as close as possible to the preferred angle when IK is invoked.

You may need to adjust a joint’s preferred angle to get the correct skeleton structure for the animation that you are trying to create. When you change this angle, the IK is solved in a new way, affecting the movement of the whole chain.

To change the preferred angles

1. Select a chain bone and open its Kinematic Joint property editor.
2. Do either of the following:
   - Enter values for the bone’s Preferred Rotation Angles X/Y/Z axes.
   - Keep the Kinematic Joint property editor open so that you can see the Preferred Rotation Angle values, then interactively rotate a bone in a 3D view using the Rotate tool in the Transform panel. This way, you can see how the changes you make affect the way the chain reaches its goal and the preferred angle rotation values that are set.

   You can still rotate the bone and see where the effector will be placed, such as for keying the position of the effector.

3. You can animate the preferred angles of a joint by setting keys for the Preferred Rotation Angles settings.

   When the animation is played back, the keyed values replace the preferred angles of the joint before the chain’s IK is resolved. This means you can change the behavior of a chain as the animation progresses.

   You can edit the preferred angles interactively only when the IK chain uses the SOFTIMAGE|3D solver. For information on IK solvers, see Choosing How an IK Chain Reaches Its Goal on page 132.

   You can select the Force IK option in the Kinematic Chain property editor to trigger IK every time you modify the preferred angles (see page 133). This way, you can immediately see the effect on the chain.

Resetting Bones

You can reset a bone’s rotation to the value of its preferred rotation. This effectively resets the chain to its pose when you created it.

To reset bones to their original rotation

- Select one or more bones and choose Create > Skeleton > Reset Bone to Preferred Rotation.
Constraining the Chain’s Up Vector (Direction)

Using a special up-vector constraint designed only for IK chains, you can constrain the orientation of a chain to prevent it from flipping. This constraint provides full and precise control of the general orientation of your chain’s resolution plane and helps you control where flipping problems will occur. You add up-vector constraints to the first bone of a chain because that is the bone that determines the chain’s resolution plane. If you set an up-vector constraint for a bone that is not the first bone in the chain, a subchain is created starting at that bone’s joint.

Using the chain’s up-vector constraint forces the Y axis of a chain or subchain’s first joint to point to a constraining object. The location of the chain’s effector determines the X axis of the resolution plane and the Z axis is perpendicular to the plane defined by the X and Y axes.

The up-vector constraint specifies the third point of the resolution plane of the chain, while the first two points are the positions of the first joint and the effector.

To view the up-vector constraint in the viewport, select Chain Critical Zone on the Attributes page in the viewport’s Camera Visibility property editor (see Displaying Critical Zones for IK Chains on page 34).

To see how up vector controls work in a rig, create a default rig for a character and then translate the pyramid-shaped controls behind the elbows and knees. See Creating a Biped or Biped Dog Leg Rig on page 173.

The chain’s up-vector constraint is different from a “regular” constraint (as found in the Constrain menu) because it’s not a constraint per se; it is actually a property of the chain’s joint. This means that the up-vector can be duplicated as part of a chain, but removing this constraint cannot be done as for normal constraints (see page 141 for how to remove the up-vector constraint).

For more information on other constraints, see Chapter 8: Animating with Constraints on page 257 in the Animation guide.
You can set the up-vector constraint using either a menu command or from the Kinematic Joint property editor, depending on whether you are using a separate constraining object or not.

**Constraining the Up Vector to an Object**

1. Create a constraining (control) object (such as a null) and position it as close as possible to the chain's root. If you place it too far from the root, you may get undesirable results.

To quickly create a null, you can choose Create > Skeleton > Draw 2D Chain and click once, then right-click to end chain drawing. The advantage is that the null is placed where you click, not at the global origin (0,0,0,) and you can translate it immediately.

2. Select the first bone of the chain and choose Create > Skeleton > Chain Up Vector from the Animate toolbar, or choose Constrain > Chain Up Vector from the Constrain panel.

You always add up-vector constraints to the first bone of a chain because that is the bone that determines the chain's resolution plane.

3. Pick the control object.

You will see the position in the Up Vector XYZ settings on the Resolution Plane page (see next page) but you won't be able to move the object using those settings: they only indicate the position of the control object.

• If you perform any action that controls the bone’s orientation (key the rotation, use an orientation constraint, or use an expression on the rotation), it takes precedence over the up-vector constraint.

• If you have both a preferred axis and up-vector constraints on a chain, the last one applied takes precedence.

• If a bone is not moving with the up-vector constraint, it may be that its rotation is animated (FK) which overrides the constraint.

You can set the up-vector constraint using either a menu command or from the Kinematic Joint property editor, depending on whether you are using a separate constraining object or not.

**To constrain the bone’s up-vector to a control object**

1. Create a constraining (control) object (such as a null) and position it as close as possible to the chain's root. If you place it too far from the root, you may get undesirable results.

To quickly create a null, you can choose Create > Skeleton > Draw 2D Chain and click once, then right-click to end chain drawing. The advantage is that the null is placed where you click, not at the global origin (0,0,0,) and you can translate it immediately.

2. Select the first bone of the chain and choose Create > Skeleton > Chain Up Vector from the Animate toolbar, or choose Constrain > Chain Up Vector from the Constrain panel.

You always add up-vector constraints to the first bone of a chain because that is the bone that determines the chain's resolution plane.

3. Pick the control object.

You will see the position in the Up Vector XYZ settings on the Resolution Plane page (see next page) but you won't be able to move the object using those settings: they only indicate the position of the control object.
While it is common to constrain a bone's up vector to a control object, you can simply make the up vector always point in a certain direction using the controls on the Resolution page in the bone's Kinematic Joint property editor. This is known as a static up vector.

**To create a static up-vector constraint**

1. Select the first bone of the chain and open its Kinematic Joint property editor.

   You always add up-vector constraints to the first bone of a chain because that is the bone that determines the chain's resolution plane.

2. On the Resolution Plane page, select **Up Vector** from the **Res. Plane** list.

3. Define the up-vector position values in global space using the **Up Vector** X, Y, and Z settings. This is the spot to which the up vector of the bone will point.

   ![Up Vector settings](image)

   You could set up expressions on the Up Vector X, Y, Z settings to control the up vector's position.

4. Select **Use Root Coordinates** to specify a static up vector using the chain root's coordinates instead of global coordinates. There are two main implications of using the chain root's coordinates:

   - The static up vector moves with the chain. Global static up vectors aren’t always useful because the up vector doesn’t move with the chain (such as when a character walks). This is a good alternative to using an up-vector control object when you know the up vector won’t change relative to the root. And without an extra up-vector control object, you can have a simpler character rig setup.

   - You can override the default resolution plane behavior. When the **Res. Plane** is set to **Default**, XSI uses the chain root’s Z axis as the preferred axis. Selecting **Use Root Coordinates** with a static up vector is the same as overriding this default behavior, such as when you want the chain root to have zero rotation.
Constraining the Chain's Up Vector (Direction)

Setting the Roll to Correct the Orientation

Getting the up-vector constraint to work as expected depends on which side of the root you draw the chain: the Y axis of the first bone will be pointing either inward or outward.

If you set an up vector on the chain on the right, it works perfectly. However, if you do this for the left chain (with the left sphere), it flips because the up vector makes the first bone's Y axis point toward the sphere. To correct this, you can set the joint's Roll to 180 (see Rolling an IK Bone on page 142 for more information).

Because the chain's up vector isn't a real constraint, this means that you can't use offsets as you do when compensating typical constraints. To compensate on the position of an up-vector constraining object (such as if it's in front of an arm or leg instead of behind it), set the Roll on the bone where the up vector is applied.

Removing Up Vector Constraints

When you choose the Chain Up Vector command (see page 139), a skeleton up-vector operator is created. To remove the up-vector constraint, you must delete both the operator and reset the resolution plane.

To remove the up-vector constraint

1. In the explorer, delete the SkeletonUpVectorOp operator under the bone's Kinematic Joint property. This disconnects the constraining object from the up vector.

2. On the Kinematic Joint > Resolution Plan property page, set the Res. Plane back to Default.

When you created the up-vector constraint, the Chain/Subchain Behavior on the Kinematic Joint > General page was set to 3D. However, if you change the Resolution Plane back to default, the subchain behavior does not change: you must reset it if you want to return to the previous IK behavior.

For information on subchains, see Creating Subchains for Independent Movement on page 150.
Rolling an IK Bone

With 2D and 3D chains, you can rotate a bone in X. However, you cannot do this with a 2D chain that is solved in inverse kinematics (its effector is translated). The solving of the chain in IK overrides any animation that has been done on a bone’s X or Y rotation. To solve the problem of being able to rotate the IK bone in X, you can set its Roll value.

Roll does not work on 3D chains because the X rotation value is used for solving the chain as well.

When you set the roll of the bone, the rotation of each bone in the chain is relative to its own axis, which rolls the bone. (If a chain has an up-vector constraint, it is considered to be solved in IK.) This rotation is propagated to the following bones in a chain, or added cumulatively to the following bones if they have their own roll defined.

The effect is a “twisting” of each bone on its axis. If you roll the first bone in a chain, it rotates the whole chain around the axis going through the chain root and the effector.
You can interactively rotate the bones in X (locally or globally) in SOFTIMAGE|3D Behavior mode for the Solver Angle (see page 133) if you select the Use Preferred X Rotation as Roll option (see below).

In Preferred Rotations from Joint Properties mode, you can rotate the bones interactively, but it will put the chain out of its resolution plane.

To set a roll for a bone

1. Select a bone on a 2D IK chain. For example, to offset an up vector’s position, set the roll on the first bone of the chain.
2. Press Enter to open its Kinematic Joint property editor.
3. On the Resolution Plane page, set the roll using either of these options:

   - Enter a value for the Roll parameter.
   - Select Use Preferred X Rotation as Roll to use the bone’s preferred X rotation value as the roll value.

Preferred rotations are not continuous, so rolling the bone with this option may make the chain flip when the preferred angles go through a gimbal lock, which is a state that Euler angles go through when two rotation axes overlap. The angle values can change drastically when rotations are interpolated through it. To avoid gimbal lock, you can change the rotation order (see Changing the Bones’ Order of Rotation on page 62).

If you’re using expressions to rotate a bone, put the expression on the bone’s Roll parameter to achieve the effect you want.

- In SOFTIMAGE|3D, the roll was hidden in the preferred X angle of the joint, which is why you could correct the situation by adjusting the preferred X angle. In XSI, the roll is exposed so you can set it directly to do the same thing as in SOFTIMAGE|3D, especially if you select Use Preferred X Rotation as Roll.

- When you import a skeleton from SOFTIMAGE|3D with a chain that has its local X rotation animated, its animation is transferred to its Roll parameter in XSI (and the static values are kept in the preferred angles as well).
Chapter 4 • Animating Skeletons

Constraining the Preferred Axis of Rotation

2D and 3D chains have a preferred axis of rotation around which the chain's joints try to rotate when you translate the effector. The default axis of rotation is the Z axis of the chain's root. In a 2D chain, the second and subsequent joints in the chain then rotate like a hinge to change positions in the plane perpendicular to the Z axis, which is the XY plane.

You can predict the behavior of a chain because of this preferred axis of rotation. For example, draw a 2D chain in the Front view to create a leg. When you translate its effector around its root (as shown in the illustration on the left), the leg moves forward and backward, but not side to side.

When you change the preferred axis of rotation of a chain, XSI tries to rotate the chain around the new axis when you animate the effector.

To constrain the chain's preferred axis

• The process for constraining the preferred axis to a control object is the same as for the up vector (see page 139) except that you choose either Create > Skeleton > Chain Preferred Axis or Constrain > Chain Preferred Axis, and the constraint is offset by 90 degrees.

The values for the Preferred Axis X, Y, Z settings on the Resolution Plane page (see next page) show the position in global space where the preferred axis should point, but you won't be able to move the object using those settings: they only indicate the position of the control object.
If you want to create a preferred-axis constraint without using a control object, the process is the same as for the up vector (see page 140) except that you select Preferred Axis from the Res. Plane list and change the Preferred Axis X, Y, Z settings to define the position of the constraint in global space.

If you have both preferred-axis and up-vector constraints on a chain, the last one applied takes precedence.

If you perform any action that controls the rotation of a joint (key the rotation, use an orientation constraint, or use an expression on rotation), it takes precedence over the preferred-axis constraint.
Chapter 4  Animating Skeletons

**Setting Restrictions on Movement**

When you create a skeleton with many chains, you may find that each chain doesn’t always move the way you would like it to when you invoke IK. Another problem is that, unlike real limbs, chains don’t have that many restrictions in their ability to move. This can cause problems, such as a leg bending in an alarmingly unnatural manner! The way to solve these problems is to set “rules” that determine how much each joint should rotate.

Here are the ways you can constrain joints and effectors to limit their movements:

- Limit the movements of joints by setting **Rotation Limits** and **Stiffness**—see Setting Rotation Limits below and Setting Joint Stiffness on page 149.
- Constrain the chain’s orientation using the up-vector and/or preferred axis constraints—see Controlling an IK Chain’s Orientation on page 126.

Using rotation limits or stiffness on 3D chains may cause the chain to jitter. To help solve this problem, try lowering the error threshold for the 3D chain’s solver—see page 134.

**Displaying Rotation Limits and Stiffness**

To **display rotation limits and stiffness**

1. Click the eye icon in a viewport and choose **Visibility Options** from the viewport menu bar.
2. In the property editor, click the **Attributes** tab and select **Chain Joint Rotation Limits** from the Selected Objects or Unselected Objects area.

**Setting Rotation Limits**

The **Skeleton > Rotation Limits** commands restrict the range of rotation of one or more chain elements. For example, setting rotation limits could prevent a head from rotating 180 degrees (unless, of course, this is the desired effect!).

Local coordinates are always used for the restriction unless the element is not parented, in which case global coordinates are used.
Setting Restrictions on Movement

To apply rotation limits interactively

1. Select the joint to be restricted. It’s usually best to start with the first joint in the chain. Because each bone is parented to the next bone in the chain, you need to consider all three axes and see how changes affect the subsequent joints.

2. Click the Rotate icon in the Transform panel.

If you are working with a 2D chain, remember that the rotation axis for a 2D joint is Z. Rotating only in the Z axis keeps the joint properly in the resolution plane.

3. Rotate the joint to its minimum position, which is the farthest clockwise point (assuming the Z axis points toward you) at which you want it to stop.

4. Choose Create > Skeleton > Set Minimum Rotation Limit from the Animate toolbar.

5. Rotate the joint to its maximum position, which is the farthest counterclockwise point at which you want it to stop.

6. Choose Create > Skeleton > Set Maximum Rotation Limit from the Animate toolbar.

Repeat these steps for the next joint, if necessary.

If you’re using FK chains, you can also set the rotation limits on the chain’s Local Transform > Rotation Limits property page. If you set the rotation limits there as well as with the Skeleton commands here, the most extreme limit values are used. The rotation limit settings on the Local Transform page have no effect on IK chains.

Rotation limits
The bone can only rotate about its Z axis and not beyond the surface of the blocks.

Settings
Set minimum (clockwise direction) and maximum (counterclockwise) allowable values for Z-axis rotation. Restrict rotations in X and Y.

Regardless of the rotation limits settings, IK will continue to prevent joints from bending backward against their preferred angles. However, if both the rotation limits are on the opposite end of the joint’s preferred angle, the preferred angle (and the joint) will flip.

To apply rotation limits interactively

1. Select the joint to be restricted. It’s usually best to start with the first joint in the chain. Because each bone is parented to the next bone in the chain, you need to consider all three axes and see how changes affect the subsequent joints.

2. Click the Rotate icon in the Transform panel.

If you are working with a 2D chain, remember that the rotation axis for a 2D joint is Z. Rotating only in the Z axis keeps the joint properly in the resolution plane.

3. Rotate the joint to its minimum position, which is the farthest clockwise point (assuming the Z axis points toward you) at which you want it to stop.

4. Choose Create > Skeleton > Set Minimum Rotation Limit from the Animate toolbar.

5. Rotate the joint to its maximum position, which is the farthest counterclockwise point at which you want it to stop.

6. Choose Create > Skeleton > Set Maximum Rotation Limit from the Animate toolbar.

Repeat these steps for the next joint, if necessary.

If you’re using FK chains, you can also set the rotation limits on the chain’s Local Transform > Rotation Limits property page. If you set the rotation limits there as well as with the Skeleton commands here, the most extreme limit values are used. The rotation limit settings on the Local Transform page have no effect on IK chains.
To apply rotation limits from the Kinematic Joint property editor

When you add rotation limits to a chain, its joint angles are not recalculated automatically. To update the chain, you must move the effector.

1. Select the joint to be restricted. It’s usually best to start with the first joint in the chain. Because each bone is parented to the next bone in the chain, you need to consider all three axes and see how changes affect the subsequent joints.

2. Open the Kinematic Joint property editor for the selected bone and click the Rotation Limits tab.

3. Click the Rotate icon in the Transform panel.

   If you are working with a 2D chain, remember that the rotation axis for a 2D joint is Z. Rotating only in the Z axis keeps the joint properly in the resolution plane.

4. Rotate the joint to its minimum position, which is the farthest clockwise point (assuming the Z axis points toward you) at which you want it to stop.

5. On the Rotation Limits property page, set the Minimum Angles for the appropriate axes.

6. Rotate the joint to its maximum position, which is the farthest counterclockwise point at which you want it to stop.

7. In the Rotation Limits property page, set the Maximum Angles for the appropriate axes.

8. Check Active to activate the rotation limits for the joint.

The restricted object never rotates beyond the minimum and maximum values set in the Kinematic Joint property editor.

It’s best to activate the rotation limits after setting the values because the preferred angles might be changed by the settings if they were “violating” the limits.
Setting Joint Stiffness

Another way to restrict a joint’s range of motion is to set its stiffness. A joint’s stiffness is a measure of its resistance to bending while using IK. While you may not want to give your character arthritis, a little stiffness in the joints can be a useful thing.

The chain at the top is the original chain. To show how stiffness affects joint behavior, the stiffness of the fourth bone has been set to three different levels, and the effector translated to the same spot.

- On chain #1, the stiffness is 0 and the joints bend evenly, like an accordion.
- On chains #2 and #3, the stiffness is 0.25 and 0.5 respectively and the other joints bend more than the fourth.
- On chain #4, the stiffness is 1 and the fourth joint is at the same angle as in the original chain.

To apply stiffness to a joint

1. Select the bone whose joint is to be restricted and open its Kinematic Joint property editor.
2. In the Joint Behavior area, select Use Stiffness to activate the stiffness for the joint.
3. Set the Stiffness value. This value must be from 0 to 1, inclusively. The higher the value, the stiffer the joint.

When you add stiffness to a chain, its joint angles are not recalculated automatically. To update the chain, move the effector.
Creating Subchains for Independent Movement

You can create independent movement within a chain by creating subchains. Subchains allow you to “break” the chain so that you can move only the subchains. When you move a subchain, it acts like an effector for its parent and like a root for its children. When you move a chain’s effector, IK only works up to the subchain’s root.

You can have several subchains on a chain, each of which can be 2D or 3D, regardless of what the existing chain is.

You can create a subchain by either defining it explicitly in a joint’s behavior or by using a pseudo-root. Pseudo-roots have the added feature of remaining fixed in local space when IK or FK are invoked. Only rotating the root can affect a pseudo-root’s position.

Defining a Joint’s Behavior

You can create a subchain by explicitly defining a joint’s chain behavior. By default, every joint is implicitly defined as 2D or 3D, based on the previous joint’s behavior.

The first joint of each subchain is always a 3D joint, regardless of the subchain’s behavior.

To create a subchain

1. Select the bone that you want to be the first bone in the subchain. Its joint will be used as the root of the subchain.

2. Open the Kinematic Joint property editor for the selected bone.
3. Change the **Chain or Subchain Behavior** setting to 2D or 3D.

The new subchain is defined from the selected bone to the effector or to the next defined subchain, if any. Each subchain has its own plane of resolution.

**To remove a subchain**

1. Select the first bone in the subchain.
2. In the Kinematic Joint property editor for the selected bone, change the **Chain/Subchain Behavior** setting to **Default**.

   The joints in the subchain become part of the previous subchain.

**Making a Joint into a Pseudo-root**

You can also create a subchain by making the joint of the selected bone or effector a pseudo-root (behaving as a root does). The pseudo-root is not a real object: it only appears that the chain is broken at this point. Because of this, it cannot be animated.

Pseudo-roots remain fixed in local space when IK or FK are invoked. You can affect the pseudo-root’s position by either rotating the root or by position-constraining its bone to another object and translating that object. For more information, see *Position Constraints* in Chapter 8 in the *Animation* guide.

---

In this example, the **effector’s joint** has been set as a pseudo-root, so it remains pinned to its location. As the **second bone** in the chain is rotated, the joints after it are automatically recalculated.
To create a pseudo-root using a command

1. Select the bone you want to be as the first bone in the subchain. Its joint is used as the pseudo-root.
2. Choose Create > Skeleton > Break Chain at Bone. This creates a subchain with a pseudo-root at the joint of the selected bone.

To create a pseudo-root using the Kinematic Joint property editor

1. Open the Kinematic Joint property editor and select Joint is Pseudo-Root.

2. If desired, change the Chain or Subchain Behavior setting to 2D or 3D.

To remove a pseudo-root

- Select the affected joint and deselect Joint is Pseudo-Root in the Kinematic Joint property editor.
Creating Subchains for Independent Movement

Moving Joints between Pseudo-roots

You can manipulate a joint between two pseudo-roots by translating it or applying a position constraint to it. A common goal is calculated and applied for the two subchains ending at this “pseudo-effector” or “joint.”

A pseudo-root behaves as a root for the subchain at its right and as a pin (nail) for the subchain at its left. Activating the pseudo-root for a joint places a ball joint at this joint and changes the torsion to a 2D subchain if the chain was 2D, or vice-versa.

Bone between the two pseudo-roots is translated. The pseudo-roots both stay pinned in place while the bones between them move.

The bones to the left of the second pseudo-root are also affected by the bone’s translation. This is because the bone’s rotation has changed and the bones to the left are keeping their local rotation angles.
Modifying a Biped’s Motion on Tracks

The motion deformation tools let you convert the animation from a biped’s feet and center or gravity (usually the hips) into a set of tracks (curves), allowing you to visualize the animation in space. You then deform and edit the tracks just like you would any other curve (move points, use deformers, etc.) and the character adapts to the new shape of the tracks.

Motion deformation lets you modify animation for many different purposes, such as:

- Adapting a character’s animation to a new environment where the terrain or path is different. For example, you could use a lattice or shrinkwrap the curves to a new surface and have the character follow all the bumps and crevices.

- Changing the character’s body proportions by changing the amount of space between the curves.

- Giving different personalities to different characters whose animation comes from the same motion capture data (such as with crowd animation).

For example, you could make one character have a high stepping height while another has a bounce when he walks. With motion capture, this can save you time because you don’t have to reanimate the whole body.

To retarget motion for different kinds of characters, you need a rig that can be dynamically reproportioned, such as the ones you can create in XSI (see Chapter 5: Character Rigging on page 163).
The curves that are created are like train tracks that represent the motion in space and carry the character along them. This allows you to deform the motion and keep the correlation between the feet and the center of gravity. The motion itself keeps a constant length.

**Getting Set Up**

Before you use the motion deform tools, you should make sure that the character whose animation you want to modify is set up correctly.

- The character must be bipedal.
- There must be animation on the two feet and the center of gravity. This can be either IK (effector positions) or FK (bone rotations), but the character must be moving “in space” (that is, no stationary walk cycles).
- Set the timeline to the exact frames of animation you want to process; otherwise the velocity of other animation on the character will be calculated as part of the velocity and ruin the resulting step marks and motion tracks.

**Creating the Motion Tracks**

You must tag three elements that drive the animation for a biped: the two feet and the center of gravity, or hips. The motion deformer takes all three of these elements into consideration because a character’s motion is a result of their combination.

The predefined character rigs (see page 168) are pre-tagged for your convenience.

**To create tracks to modify a character’s motion**

1. One by one, select each of these animated elements in the character, then choose the corresponding command from the Create > Character > Biped - Tags menu on the Animate toolbar: Left Foot, Right Foot, and Center of Gravity.

2. Figure out the Velocity Threshold by doing a test run with the Create > Character > Biped - Define Step Marks command. This creates little icons that represent the stepping positions, allowing you to tune the correct Velocity Threshold.
You need to find the lowest velocity of the animated elements, such as where a foot is on the ground, going from one step to the next. This value needs to be the closest to zero, but not zero.

You can reiteratively test this, starting at a very low threshold and working up until you see that all step marks are present. If a value is too low or high, just delete the step marks and try a new level. You can delete the step marks after you’ve run the Deform Motion command.

3. Once you know the Velocity Threshold, choose **Create > Character > Biped - Deform Motion**. This retargets the character’s motion based on the step marks you set. Three curves are created, representing the motion path for the character.
4. In the Offset Multiplier controls that appear in a property set, you can offset the rotation and position of the tracks, thus warping the animation in space.

5. Deform the tracks like you would any other curve and see how the character responds to the changes.
   
   For example, you could create a chain from the curve, then drag the effector to change the path.
Chapter 4 • Animating Skeletons

Importing Motion Capture Files

You can import motion capture information into XSI using either the Acclaim or Biovision file formats. Both of these file formats contain motion information captured with optical technology.

Importing Acclaim Files

You can import a file saved in the Acclaim Skeleton (.asf) format with an accompanying Acclaim Motion Capture (.amc) file. You can import an ASF file along with one compatible AMC file.

ASF Files

Acclaim Skeleton files (ASF) contain information about the hierarchy and base pose of the skeleton. This information is used to create a skeleton hierarchy (nulls or bones) when imported into XSI. The animation for this skeleton is saved in an accompanying Acclaim Motion Capture (AMC) file.

ASF files are imported into XSI as models placed under the Scene_Root node. The model’s name is the same as the skeleton’s name. The model contains a hierarchy of nulls or bones that reflect the skeleton hierarchy described in the ASF file. The model also contains a base pose action in which the model is initially posed.

AMC Files

Acclaim Motion Capture (AMC) files contain keyframed animation (motion capture data) that can be applied to the skeleton hierarchy in a compatible ASF file.

AMC files are imported into XSI as an action with the AMC file’s name. You can choose to have this action automatically applied as an action clip in the animation mixer (see page 162).
Importing Motion Capture Files

To import Acclaim files into XSI

1. Choose File > Import > Acclaim File from the main menu. The Acclaim Import dialog box appears.

2. In the Skeleton File text box, specify the ASF file to be imported.

3. In the Motion File text box, specify the accompanying AMC file to be imported. This AMC file must be compatible with the ASF file you’re importing.

4. Select how you want to import the skeleton hierarchy—see Setting Up the Skeleton on page 160.

5. Select the other import options—see page 162.

6. Click OK to import the Acclaim files.

Importing Biovision Files

Biovision (BVH) files contain information about the hierarchy of the skeleton. This information is used to create a skeleton hierarchy (you can choose between nulls or bones) when imported into XSI.

BVH files also contain two actions: the base pose action and the motion action. The skeleton uses the base pose by default unless you choose to have the motion action automatically applied as an action clip in the animation mixer (see page 162).

Biovision files are imported into XSI as models and placed under the Scene_Root node. The model’s name is the same as the BVH file name.

Structure of Biovision file elements after importing

Model containing all Biovision hierarchy and animation information.

Base pose of skeleton.

Motion capture animation.

Root of imported skeleton hierarchy.

Group of all animated objects in skeleton hierarchy (see page 162).
To import a Biovision file into XSI

1. Choose File > Import > Biovision File from the main menu. The Biovision Import dialog box appears.
2. In the File Name text box, specify the BVH file to be imported.
3. Select how you want to import the skeleton hierarchy—see Setting Up the Skeleton on page 160.
4. Select the other import options—see page 162.
5. Click OK to import the Biovision file.

Setting Up the Skeleton

You can choose how you want the skeleton hierarchy to be created when you import it into XSI using the following options.

These options are the same for importing either Acclaim or Biovision files.

Select to import the skeleton hierarchy as either bones or nulls.

If you select Bone Chains above, you can then select how to set up the skeleton.

To determine what type of skeleton hierarchy to import

- Select either of the Skeleton Description Mode options:

  - Hierarchy of Nulls creates a skeleton hierarchy of nulls.
    You must then constrain the bones of an existing skeleton or parts of an existing rig to these nulls.
  
  or

  - Bone Chains creates a standard skeleton hierarchy made of bones.
    You can then set the Bones Skeleton Options to define the skeleton setup (see next page).
Importing Motion Capture Files

To define the skeleton setup

If you selected the Bone Chains option as the Skeleton Description Mode, you can decide how you want the skeleton to be set up in XSI.

- Select how you want the skeleton bone connections with the Hierarchy options:
  - Connect with unanimated bones connects all bones, animated or not, as part of the skeleton hierarchy.
  
or
  - Leave unconnected does not connect bones that are not animated.

- Select which structure you want for the resulting skeleton hierarchy with the Chains options:
  - SOFTIMAGE|3D Mode creates SOFTIMAGE|3D-style chains whose effectors are children of the last bone in the chain (see Drawing Chains à la SOFTIMAGE|3D on page 44).
  
or
  - SOFTIMAGE|XSI Mode creates default XSI-style chains whose effectors are children of the chain root (see Effectors on page 32).
Setting Other Import Options

You can select how you want the animation and group information to be set up as you import the skeleton hierarchy into XSI:

<table>
<thead>
<tr>
<th>Import Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crop Motion Values to Limits</td>
</tr>
<tr>
<td>Override Bone Length Multiplier</td>
</tr>
<tr>
<td>Length Multiplier</td>
</tr>
<tr>
<td>Add motion action to Mixer</td>
</tr>
<tr>
<td>Group Animated Objects</td>
</tr>
</tbody>
</table>

- Select **Crop Motion Values to Limits** to apply the DOF (degrees of freedom) limits in the ASF file to the animation in the accompanying AMC file.

  The DOF limits are the minimum and maximum values allowed for the XYZ translation and rotation, as well as movement along the bone. Any values outside the minimum/maximum range are cropped.

  This option applies only to importing Acclaim files.

- Select **Override Bone Length Multiplier** to override the bone length multiplier information in the ASF file to be applied to the animation in the accompanying AMC file.

  If you select this option, you must then set the value with the **Length Multiplier** slider. This is the length (in units) of the bones.

  This option applies only to importing Acclaim files.

- Select **Add motion action to Mixer** to add the accompanying keyframed motion as an action clip in the animation mixer.

- Select **Group Animated Objects** to create a group that includes all animated objects in the skeleton hierarchy.
Basics for Setting Up a Control Rig

Control rigs allow for “puppeteering” a character, helping you easily pose and animate it. Once a control rig is set up properly, you can animate more quickly and accurately than without one.

There are a number of tools in XSI to help you create a rig for your character. You can create rig guides that you adjust to match the proportion of your character, then create a rig based on those proportions. These are available for both biped and quadruped characters. There are also tools to help you easily create shadows rigs and manage the constraints between rigs and their shadows rigs.

Although not described in this chapter, there are a number of standard tools in XSI that can help to streamline your rigging and character animation workflow. See Chapter 6: Tools for Character Animation on page 203 for more information.

For tutorials on setting up character rigs and animating with them, see the Character Animation lessons in the Tutorials guide.
Basics for Setting Up a Control Rig

Character Animation

Tips for Rigging

Neutral pose sets local transformation values to 0,0,0, which makes for an easier base for keying.

- When you set up a control rig, “zero out” the controls. This means that when you set all the controls’ scaling, rotation, and translation to (0, 0, 0), it makes your character snap back to a neutral pose, or “rest” position. An advantage to setting up rigs this way is that fcurves and key values are easier to understand.

For information on how to do this, see Creating Neutral Poses for Skeletons on page 64.

- When you’re creating constraints and rigging, you often need to use nulls. To quickly create a null exactly where you want, choose Create > Skeleton > Draw 2D Chain and click once, then right-click to end chain drawing. The advantage is that the null will be placed where you click, not at the global origin (0, 0, 0,).

- When you create a null, you can resize it by adjusting its Size parameter (select the null and press Enter to open its property editor). This can help with rigging issues where you don’t want to scale the null.

- You can quickly match a null to a control object’s scaling, rotation, and translation: select the null and choose Transform > Match All Transforms, then pick the control object. The null matches the control object’s transformation values.

- You may not need to create a special control rig to animate simpler characters because the effector is a child of the chain root and can be easily put anywhere in a hierarchy or parented to other chains and objects (see Effectors on page 32).

Animating Pivots for Characters

The center pivot is the position about which objects are scaled and rotated when playing back animation. You can change and animate the center pivot, such as for creating a rolling cube.

The center pivot that you can animate is not the same as the pivot you use in the 3D views (by pressing the Alt key) for manipulating objects.

Because you cannot directly animate the transformation of an object’s center, you often create a null, parent the object to it, and then animate the null (see Animating the Transformation of Object Centers in Chapter 2 in the Animation guide). This is a quick and easy way to work for effectors in a skeleton or other objects, but for bones there is a problem. The bones in a skeleton hierarchy cannot be cut from their parent object so that you can insert a null and parent the bone to it. Using the center pivot, you can offset and animate the bone’s pivot regardless of its parenting.

For information on using and animating the center pivot, see Setting the Center Pivot in Chapter 16 in the Fundamentals guide.
These are some other possible uses for pivots in skeletons:

- Create a foot roll rig by constraining an effector to a null and animating the null's center pivot. Using pivots means fewer nulls in the rig, which makes for a less complicated setup that's faster in performance and easier to animate.

- Animate the center pivot of an effector for creating cycle movements with IK, like bicycling. You can also change the pivot's scaling to create more elliptical cycles.

- Change the rotation of a bone's pivot without changing its axis alignment (see Changing the Bone's Axis Alignment on page 120). This way, you can rotate the bone on any axis you like without having to reset each bone's actual axis of rotation.

Displaying Chain Elements and Their Shadows

You can display the bones, roots, and effectors in many different styles to make it easy to view and select them. You can do the same with the elements’ shadow display styles as well, including scaling and offsetting them. See Displaying Chain Elements on page 34 for more information.

Instead of creating a separate object to control chains, it’s often better to use the chain element’s shadow as the control object. This means that you can manipulate the shadows in both IK and FK and then have them control each other without creating cycles.

Jaiqua with shadows displayed as cylinders that have been scaled and offset along the bone.

Notice how easy they are to see and select.
Using the Character Development Kit

The Character Development Kit is a collection of character assembly functions, such as a spine, torso, hip, arms, and heads, which are the set of body parts that are required in typical rigging projects. Using programmed components is an efficient way to produce a large volume of characters. Basically, all you have to do is figure out which components you want, how they map to your character’s envelope, and how they connect to one another.

Many of these commands have the same options available in the XSI interface; for example, you can choose a quaternion spine for the torso, define forearm and biceps roll divisions, or control the location of the leg's up vector (see Creating a Biped or Biped Dog Leg Rig on page 173 and Creating Quadruped Rigs on page 190).

For more information on the Character Development Kit, see its chapter in the SDK Customization Guide by opening a script editor and pressing F1.
Getting Started with Predefined Rigs

Although you can create custom rigs based on your specific character needs, you can also load a number of predefined rigs that are ready for use “as is” or as a base for further modifications. These are default rigs based on the default settings of the proportioning guide.

Loading Biped Rigs

The predefined biped rigs are based on the default settings of the proportioning guide, as described in Setting Up the Biped or Biped Dog Leg Guides on page 169. The rigs are skeletons that include control objects that you can position and orient to animate the various parts of the character’s body. See Creating a Biped or Biped Dog Leg Rig on page 173 for more details about the rig.

To load biped rigs

- Choose Get > Primitive > Model > Biped - Rig from any toolbar to load a basic rig.
- Choose Get > Primitive > Model > Biped - Box, Biped - SI|3D Skeleton, or Biped - Skeleton from any toolbar to load other variations on the basic rig.

These are the same as creating rigs using the Character > Hierarchy from Guide commands (see page 199).

Loading a Quadruped Rig

The predefined quadruped rig is based on the default settings of the proportioning guide, as described on page 190.

The rig is a skeleton that includes control objects that you can position and orient to animate the various parts of the character’s body. See Creating a Quadruped Rig on page 191 for details about the rig.

To load a basic quadruped rig

- Choose Get > Primitive > Model > Quadruped - Rig from any toolbar.
Setting Up the Biped or Biped Dog Leg Guides

To help you make the skeleton’s rig proportional to your setup, you can load a rig-building guide. After you set up the guide as you like it, you can create a biped rig based on this guide.

You can create either a regular human biped guide or a biped dog leg guide, which is the same as the biped except that it has 3-bone legs that are angled like an animal’s—good for fantastical humanoid creatures!

The guides can also be used as a starting point for different rigging styles, and technical directors can write their own proportioning script to attach their rig to the guide.

To load the biped or biped dog leg proportioning guide

1. Do one of the following:
   - Choose Create > Character > Biped Guide from the Animate toolbar or Get > Primitive > Model > Biped - Guide from any toolbar.
   - or
   - Choose Create > Character > Biped Dog Leg Guide from the Animate toolbar or Get > Primitive > Model > Biped Dog Leg Guide from any toolbar.

2. In the Biped or Biped Dog Leg Guide dialog box that appears, select the Guide Style and the type of Symmetry, as described here.
To set the guide style

- Select **Skeleton Structure and Volume** to create a complete guide that includes volume controls (displayed as yellow splines with small manipulator cubes) which you can use to define the body’s volume.

When you later create the rig from this guide, you use these controls to define the volume as set on the Skin page of the Make Biped or Make Biped Dog Leg dialog box (see page 183).

- or

- Select **Volume Controls Hidden** to hide these yellow deform splines, but still have them available in the guide’s structure. In the explorer, you can find them in a group called DetailedGuide, under the biped guide model.

To set the guide’s symmetry

- Select **Full Symmetry** so that when you manipulate any cube, the corresponding cube on the opposite side of the body is manipulated in the same way. This makes it quick and easy to set up matching proportions for both sides of the body at once.

- or

- **No Symmetry** means that there are no constraints between the cube controls on the opposing sides of the body, allowing you to manipulate each part of the body independently.

- or

- You can also have just the **Upper Body** or **Lower Body** be asymmetrical, meaning that you can manipulate the cube controls on those portions of the body independently.

3. Click OK in the dialog box and the guide is created.

4. Translate the red manipulator cubes to proportion the limbs to match the character’s envelope.
5. If you created volume controls, translate the cubes on the yellow splines to match the volume of the character's envelope. These proportions are used when you create the Skin controls for the rig, as described on page 183.

6. After you’ve set up the proportions of the guide for the character, you can create a rig from it, as described in the following section.

To customize your rig, you can duplicate or delete some of the red placement cubes in the guide and the resulting rig will have only the elements you want. For example, if you have a character that needs a three-bone neck instead of a two-bone version, you can select the tip of the head, duplicate it, and move it upwards. Or if your character doesn’t need all its fingers, you can delete the appropriate finger control cubes in the guide.

These are the control cubes the you can duplicate:

- Head control cube can be duplicated $n$ times for more head length.
- Neck control cube can be duplicated $n$ times to create a jaw or snout.
- SpineBase control cubes can be duplicated 3 or more times to create a tail.
- FingerTip control cubes can duplicated to create more finger joints.

These are the control cubes that you can delete:

- FingerTip can be removed down to two cubes, which draws one-bone fingers.
- Entire fingers can be removed by deleting their control cubes. These fingers will not be created in the rig.
Deleting Bones

To delete bones, you branch-select the root of the unwanted (such as a finger) bone in the guide and delete it. The roots of the bones, however, are hidden so here’s how you do it:

To find and delete the roots

1. Select one of the bones you want to delete.
2. Open an explorer and press f to find the selected element.
3. Branch-select the root node for this bone and then delete it.
Creating a Biped or Biped Dog Leg Rig

After you load and adjust the biped guide to match your character, you can create a rig based on these new proportions. The rig is a skeleton that includes control objects that you can position and orient to animate the various parts of the character body.

The rig provides a foundation of rigging components, allowing you to define as much detail as you need. You can get very specific results by setting roll divisions, volume controls, shadow rigs, and more.

Although there are a great deal of styles available from generating a rig this way, it is simple to do further modifications. The rig is composed of standard XSI components that you can delete, constrain, or connect to new objects.

To load predefined rig setups based on default guide values, see page 168.
Chapter 5  •  Character Rigging

Features to Help You Use the Rig

There are a number of features in the rig that can help you use it.

Synoptic View

There is a synoptic view available to control the rig, which you can open by selecting any part of the rig and pressing F3. This lets you easily select and key the rig, reset it, and have detailed control over the hands.

When you scale the rig, the synoptic’s control systems adapt to the new proportions, so you can still select items.

Marking Sets

Each rig control element has a marking set to define which parts of the character are keyed, plotted, and posed by the synoptic view. These marking sets can help you quickly key or adjust the rig.

For more information on marking sets, see Remembering Marked Parameters with Marking Sets in Chapter 2 in the Animation guide.

Action Sources

The biped rigs contain various action sources to help pose the hands or return to the default pose. For example, if you need to return to the rig’s original position (such as to modify or reapply the envelope), you can apply the action source called DefaultPose to the rig.

To apply an action, select it from the rig’s Mixer > Sources > Actions folder and choose Actions > Apply > Action (see Applying and Pasting Animation from Action Sources in Chapter 3 in the Nonlinear Animation guide for more information).
Tools for Animating the Fingers and Hands

Under each hand’s root node, you can find a custom parameter set that contains sliders for rotating each finger bone in the hand. This set is also available by clicking the Sliders button on the Hands page in the synoptic view (select an element and press F3).

You can also use the buttons on the Hands page to easily select and pose the hands. As well, you can apply action sources that can relax, make a fist, or spread the fingers on each hand.
Creating the Rig

To create a biped or biped dog leg rig

1. Modify the proportioning guide to match your character’s proportions (see page 169).

2. Choose Create > Character > Rig from Biped Guide or Create > Character > Rig from Biped Dog Leg Guide from the Animate toolbar.

3. In the Make Biped or Make Biped Dog Leg dialog box that appears, set up the rig elements as you need. The sections following this procedure provide a summary of the options on each property page:

4. Click OK and a new skeleton rig based on the guide’s proportions and the options you selected is created.

See page 178.

Chest: controls rotation and translation of spine.

Hips (inner square): controls rotation angle for step and heel roll.

Leg up-vectors (pyramids): control resolution plane of legs for IK.

Arm up-vectors (pyramids): control resolution plane of arms for IK.

Upper body (outer square): controls rotation and translation of upper body.

Foot (outer cube): controls rotation and translation of foot.

Foot roll (inner cube): controls rotation angle for step and heel roll.

Head (null): controls rotation and translation of head.

Ears: curves are chains driven by springs. Cubes control ears’ rotation.

Head (null): controls rotation and translation of head.

Arm up-vectors (pyramids): control resolution plane of arms for IK.

Hand (cube): controls rotation and translation of hand.

Chest: controls rotation and translation of spine.

Leg up-vectors (pyramids): control resolution plane of legs for IK.

Hips (inner square): controls rotation and translation of hips.

Upper body (outer square): controls rotation and translation of upper body.

Foot (outer cube): controls rotation and translation of foot.

Foot roll (inner cube): controls rotation angle for step and heel roll.
5. After the rig is created, the Biped or Biped Dog Leg Controls appear. These let you scale the spine and upper body volume, as well as set the foot roll angle (and leg extension for the biped dog leg rig). See Using the Biped or Biped Dog Leg Controls on page 185 for more information.

6. Envelope the body geometry to the rig. To envelope this character, weight it to the envelope_group group found in the rig’s model. The contents change based on the rigging options.

   If you have multiple envelope objects, weight all of them to the full envelope group. This makes it easier to paint weights across the multiple surfaces.

7. Translate and rotate the appropriate rig controls and key them to animate the parts of the skeleton.

   See Tips for Using the Rigs on page 186 for more information on using the rigs.

   When you animate the rig, make sure to turn off both constraint and child transform compensations! You can deactivate the CnsComp and ChldComp buttons on the Constrain panel. Otherwise, you end up repositioning the constraints on the rig.

- The biped dog leg foot controls are the same as for the quadruped rig—see Creating the Rig on page 192.

- The Toe (tip) control allows the foot to point in a particular direction or rotate without the toes going below this control object. Select the Toe control and rotate it or translate it to control the foot placement. Unlike manipulating the Foot control, the third bone won’t go under the Toe control object.

- You can also create a separate tail element that you can then use with any rig that you create, such as to create some fantastical creature using a biped dog leg rig. See Creating a Tail on page 58 for more information.

- You can hide the proportional guide after you’ve created the rig so that it’s easier to see the rig.

- It’s a good idea to save a copy of the character’s proportional guide in case you want to change the rigging options later.
Setting Up the Torso Spine

The options on the Chest page in the Make Biped/Biped Dog Leg dialog box let you select the way you want the spine to be set up in the torso area.

- The Torso spine type can be either quaternion or regular skeleton. Both the spine types are compatible in all combinations of rigs and with the shadow rigs.
  - The Quaternion spine bi-directionally distributes the roll from the chest down and the hip up. This is the same spine that is created when you choose Create > Skeleton > Spine see Creating a Spine on page 53 for details).
  - The Skeleton spine is a simple chain that doesn’t allow for independent hip movement as does the quaternion.

- If you selected Quaternion for the spine, you can select the Spine type:
  - Fixed Length creates a spine that can be extended only using the Scale slider in the Biped or Biped Dog Leg Controls (see page 185).
  - Stretches creates a spine that extends to join with the chest icon.

- The Divisions let you specify the number of vertebrae that make up the spine. More vertebrae allow for a more flexible spine.

- The Control Icon for the chest can be a Square or the control icon for the Upper Body can be a Cube.

Spine is of fixed length.
Chest icon is a square
Three divisions in this Quaternion spine.
Six divisions in this Skeleton spine.
Upper body icon is a cube.
Creating a Biped or Biped Dog Leg Rig

Creating a Belly Control

The options on the Belly page in the Make Biped/Biped Dog Leg dialog box let you create a belly control and set the weighting for it. This control may be useful for animating a character that has a large belly or paunch.

1. Select the Make belly option to create a box belly control that extends in front of the hips.

   The curves that connect the box to the hips have a layer of secondary animation that dynamically reacts to the movement of the rig. You can also key the rotation of the box to directly animate the belly.

   To open a control slider set for modifying the belly’s spring, see Accessing the Springs’ Control Sliders on page 188.

2. If you want the distribute the weighting of the belly around the hips (sliding), select Sides of belly slide to even the belly movement’s falloff.

   This option creates a null on either side of the belly control at the hips. You can use these nulls to help distribute the weight of the belly (envelope) around the hips so that it’s not just like a big lump in the front.

3. Then set the Percentage of the slide falloff to control how much of the weight is distributed to these nulls. The higher the percentage, the more weight is spread to these nulls.
Chapter 5  •  Character Rigging

Setting Up the Head, Neck, and Ears

The options on the Head page in the Make Biped/Biped Dog Leg dialog box let you select the way you want the head and neck to be set up, including ears.

- **The Head** can be either a Skeleton or a Quaternion spine:
  - The Quaternion spine head bi-directionally distributes the roll from the head down and the shoulders up. This is the same spine that is created when you choose Create > Skeleton > Spine (see Creating a Spine on page 53 for details).
  
  or

  - The Skeleton head is a simple chain that doesn’t allow for independent movement as does the quaternion. The chain divisions in the neck are determined by the placement and duplication of the guide’s control cubes.

- If you selected Quaternion spine head for the neck, you can select the Head Spine (neck) type:
  - **Fixed Length** creates a neck that can be extended only using the Scale slider in the Biped Controls (see page 185).
  
  or

  - **Stretches** creates a neck that extends to join with the head icon.

- The Divisions let you specify the number of vertebrae that make up the neck. More vertebrae allow for more flexibility.

- You can also create Ear controls, useful if you are making a humanoid creature.

  The curves of the ear controls have a layer of secondary animation that dynamically reacts to the movement of the rig. You can also key the rotation of the cubes on the curves to directly animate the ears.

  To open a control slider set for modifying the ears’ springs, see Accessing the Springs’ Control Sliders on page 188.
Setting Up the Arms and Fingers

The options on the Limbs page in the Make Biped/Biped Dog Leg dialog box let you select the way the arms and fingers are created and animated.

- The **Arm Rotation** options let you choose between two different rotation orders:
  - **YZX** creates the arms with the rotation order of YZX, which often helps reduce the occurrence of gimbal lock. This means that the Y rotation becomes the primary up-and-down motion of the arm and is evaluated first, then Z, then X (roll). See *Changing the Bones’ Order of Rotation* on page 62 for more information.
  - **XYZ**, which is the default rotation order in XSI.

- The **Symmetrical Manipulation of Arms** options determine if the arm rotation values will mirror exactly:
  - **Symmetrical**: the arms are manipulated symmetrically. For example, if you select both forearms and rotate them 30 degrees forward, both arms will rotate down. This also makes it easier to copy animation from the right arm to the left and have it mirror properly. The right arm is scaled to a value of -1.
  - **Non symmetrical**: both arms have positive scaling, but if you rotate them, the motion is not symmetrical. For example, if you select both forearms and rotate them 30 degrees forward, the left arm will rotate down but the right arm will rotate up. This option is useful when you need to transfer the animation data to a platform that doesn’t support negatively-scaled bones.

- The **Arm Attachment** options let you choose between:
  - Making the arms **children of the shoulders** to animate them in FK (rotating the arm bones). *or*
  - Making the arms **children of the hips** if you want to have shoulder movements that are more independent of the hands and arms. For example, this style is useful for characters who do a lot of up-and-down shrugging movements with their shoulders.

- The **Finger** options let you choose between making fingers with **2D chains** or **3D chains**.
  2D chains are constrained to a single plane and work more predictably in IK for typical character fingers. 3D chains allow for a full range of movement on all axes (not restricted to a plane) and can make good abstract tree branch, monster, or robotic tentacle fingers. See *2D and 3D Chains* on page 30 for more information.

To manipulate the fingers, see *Tools for Animating the Fingers and Hands* on page 175.
Setting the Roll Division

The options on the Roll Division page in the Make Biped/Biped Dog Leg dialog box allow you to choose how you want the skeleton's envelope to be weighted at a more refined level when the bone is rotated (rolled).

The roll division reduces the pinching that occurs when appendages with simple bone structure roll. These quaternion-based options create a set of nulls, based on the number of subdivisions that you specify, to distribute the weighting along the bone.

The Biceps and Thigh Roll Subdivision options break up the biceps and thigh rotation along the length of those bones so that the nulls allow more rotation at the start of the bone than the end. For example, the start of the biceps is joined at the shoulder (and armpit), which require more rotation than the elbow. This allows for more refined envelope weighting for these difficult areas.

The ForeArm Roll Subdivision option divides the rotation in the opposite way as the biceps and thigh because more rotation is required at the wrist instead of the elbow.

In this case, the rotation starts at the end of the forearm bone where it joins the wrist. This setup is compatible with animation on the last bone of the arm, hand effector, hand root, or hand bone. It also works with IK and FK, and IK/FK blending.

Although it’s unlikely that you will rotate hands beyond the -180 to 180 degree range that the roll provides, you can extend this range. For example, certain setups may require that you adjust exactly where the center of the seam line is. To do so:

1. Select a hand bone (LHand or Rhand) and open it in the explorer.
2. Double-click the Roll_Compensation parameter and tweak the roll_offset slider to get the range you need.
Creating a Biped or Biped Dog Leg Rig

Selecting Volume Controls for the Skin (Envelope)

When you're animating a rig, you need to take the final body envelope into consideration. This helps you animate appropriately, especially when the body is very large or unusually shaped.

To help you visualize and estimate the body's volume with the rig, you can select the options on the Skin page in the Make Biped/Biped Dog Leg dialog box. Based on the yellow splines you set up with the proportional guide (see page 169), these options create a series of controls and nulls that approximate the body sliding and compressing over the skeleton.

Using these controls, you can allow a character to reach extreme poses more easily than by weighting alone.

- The **Simple Sliding** options (Hips and Armpits) option use 2-point constraints to approximate sliding.

- The **Sliding and Volume Preservation** options (Thighs) do the same thing but outside a bounding volume, forcing the body to keep its shape as it deforms.

- The **Joint Compression** options simulate the elbow pushing the mass of a biceps muscle or the knee pushing the thigh muscle. The volume is placed as an approximation from the guide’s yellow spline volume controls (see page 169) and needs to be fine-tuned in the pose of the compressed joint.

After you adjust the volume controls’ position, rotation, and scale to get them into the poses that work optimally for your character, you should key these values.

If you need to return to the original position where the control splines were generated (such as to modify or reapply the envelope), you can apply an action source to the rig called **DefaultPose** (see Action Sources on page 174).
Creating Shadow Rigs

Shadow rigs are simpler rigs that are constrained to the more complex main rig that is used for animating the character. When you generate the main rig, you can create a shadow rig at the same time.

Shadow rigs are usually used for exporting animation, such as to a game or crowd engine or other 3D software programs. To transfer the animation from the complex rig to the shadow rig, plot the animation while the shadow rig is still constrained in place to the complex rig. Then you can export the shadow rig or just its animation.

To create a shadow rig

3. On the Shadow Rigs page in the Make Biped/Biped Dog Leg dialog box, select the Type of shadow rig to create:
   - SI|3D Skeleton creates a SOFTIMAGE|3D style skeleton hierarchy with the effectors as children of the last bone.
   - XSI Skeleton creates a skeleton hierarchy with the effectors as children of the chain’s root.
   - Null Hierarchy creates a skeleton made of nulls. A null is created at the root of each chain.
   - Box Hierarchy creates a body of boxes that surround the regular rig.
   - Box with IK Arms and/or Legs creates a body of boxes but with the arms and/or legs as IK bone chains.

4. Select whether the shadow rig Hands have Fingers or not.

For example, select the No Fingers option to create “mitten” hands when finger animation is not needed.
When you animate the rig, the shadow rig follows along. You can then plot the animation on to the shadow rig, remove the constraints between the rigs, and the shadow rig is ready to be exported.

For information on plotting, see Plotting Animation in Chapter 2 in the Animation guide.

The shadow rig is constrained to main rig in various ways. To manage these constraints easily, use the Constraints Outside Model Boundaries commands as described on page 201.

Rig with shadow rig attached is animated. Animation is plotted to the shadow rig, then constraints between the two rigs are removed. Animation from rig is plotted onto the shadow rig. It's now ready for export.

After the rig is created, the Biped or Biped Dog Leg Controls are displayed. You can use these to adjust the scale of the spine, neck, and upper body volume, as well as set the foot roll angle. For the Biped Dog Leg, there are some additional controls that set the leg extension.

If you close these Controls, you can find them (a custom parameter set) in the explorer under the Biped or Biped_DogLeg model’s node, as shown on the left.

- The Scale slider dynamically reproportions the quaternion spine's length.

If you selected the Fixed Length type for the rig's spine (see page 178), this is the only way in which you can scale the spine.

If you created a quaternion head spine (neck), there is a separate Scale slider that lets you control its length.
• The Volume Factor slider scales the quaternion spine vertebrae as the length of the spine changes. This may also result in the scale being inherited by the upper body controls (arms, spine, neck, and head). Use positive values to shrink the upper body and negative values to expand it.

If you don’t want the head and arms to be scaled down the vertebrae by the volume factor, select them, press Ctrl+k to open the Local Transform property editor, and deselect the Constrain > Scaling option on the Options page.

• The Left/Right Foot Roll 1 and 2 sliders set the angle at which the break points of the foot roll propagate rotation. For example, you can set it so that it appears as if the character is wearing high heels or is stepping over something.

Another example is if a character is putting out a cigarette with its foot: set both angles to 0 and the roll will pivot from the front of the foot.

Biped Dog Leg Controls

With the biped dog leg rig, there are extra sliders that control the locking of the leg extension control (the middle box in the three foot controls):

• The Left/Right Lock Percentage sliders determine the length percentage that will lock the two lower bones of the dog leg. For example, if this value is set to 1, the two lower bones cannot rotate forward beyond a straight line as you rotate the leg’s extension control. If you set the lock percentage value to 0.95, the bottom leg bone cannot rotate beyond 95% of the length of a horizontally-locked leg.

For creatures like an ostrich that can rotate the two lower leg bones very straight while running, a value like 0.97 would make sense. For creatures like a cat, this maximum reach is not as straight, so a value like 0.95 would work better.

Tips for Using the Rigs

The following tips provide information about how to set up and use the rig.

Default Position, Rotation, and Scaling Values

The rig’s default position has translation and rotation values of zero so that you’re always working from a known base line. This makes it easier to transfer and keep track of the animation. You can get back this position easily by choosing Transform > Reset All Transforms (press Ctrl+Shift+r), as with any other skeleton.

As well, the default scale of all the components is 1, again making it easy to return to this base. You can scale the rig proportions without adding scale by using the bone length sliders and the spine Scale slider (in the Control set).

You can also create your own neutral pose for the rig with position and rotation values of 0 and scaling values of 1. See Creating Neutral Poses for Skeletons on page 64 for more information.
Creating a Biped or Biped Dog Leg Rig

Animating and Enveloping the Rig

- To animate the rig, you translate and rotate the appropriate rig controls and key them to animate the parts of the skeleton. However, make sure to turn off both constraint and child transform compensations! You can deactivate the CnsComp and ChldComp buttons on the Constrain panel. Otherwise, you end up repositioning the constraints on the rig.
- To envelope this character, weight it to the envelope_group group found in the rig’s model. The contents change based on the rigging options. If you have multiple envelope objects, weight all of them to the full envelope group. This makes it easier to paint weights across the multiple surfaces.

Control Objects

The control objects are all curve-based wireframes and have no geometry. This makes it easy to keep the rig and body geometry (envelope) on the same layer because you can turn off the 3D Geometry option in a viewport’s visibility menu (click the eye icon) to have only the rig displayed.

To scale the control objects

If you need to resize the control icons (such as the foot and hip controls), follow these steps:

1. Tag all the points of the control object that you want resize.
2. Scale the object in point mode.
3. Freeze the object’s geometry.

This way you don’t introduce scaling into your rig, which could create problems transferring animation later in a project.

You can use the same process to reposition and change the shape of the controls without affecting their centers.
Scaling or Changing the Shape of the Hip Control

If you need to change the shape of the hip control object, you can find a parameter set with sliders that control many facets of its size and shape, such as the hip sockets' depth or width.

To open the hip control’s custom parameter set

1. Select the hip control.
2. In an explorer, press f to find the Hip control, then expand its node.
3. Click the $ icon to open the control slider set.

Accessing the Springs’ Control Sliders

Several different controls in the rigs are driven by spring operators. These include the ears and belly control, as well as the tail for the quadruped rig (including the tail control you can create independently and parent to your rig—see Creating a Tail on page 58). If you want to access the sliders for controlling the spring to modify it, here’s how you can find it.

To access the scripted operator for springs

1. In an explorer, select the hidden group under the rig’s model and press h to unhide the controls. Many nulls appear.

   Each type of rig control has a different object and path, but the spring can be found under the rig control’s “hidden” effector.

2. Select the appropriate effector for the ear, belly, or tail.
3. In the explorer, press f to find the effector and expand its **Kinematics > Global Transform > Pos** parameters.

4. Click the S icon beneath the parameters to open the spring's slider set.

**Offsetting the Resolution Plane for the Leg's Up Vectors**

If you want to animate the resolution plane for the biped legs as you animate them in IK, here's a good trick!

1. In an explorer, select the **hidden** group and press h to unhide the controls. Many nulls appear.

2. Select middle green null behind the knee called *lastboneUpV/UpV1*.

3. In the explorer, press f to find the null's node and expand its **Kinematics > Constraints** folder.

4. Click the Two Point Cns icon to open the constraint's property editor.

5. Set the **Distance Percentage** value (50% default) to change the position of the control null between the up vector control (100%) and the null below it (0%). Key this value at animate the plane.

If you're animating the legs in the biped dog leg rig, be careful not to move the leg's IK up vector controls too low or else some flipping may occur.
Creating Quadruped Rigs

To help you make the skeleton’s rig proportional to your setup, you can load a rig-building guide. After you set up the quadruped guide as you like it, you can generate a quadruped rig based on this guide.

The guide can also be used as a starting point for different rigging styles, and technical directors can write their own proportioning script to attach their rig to the guide.

Setting Up the Quadruped Guide

To load the quadruped proportioning guide

1. Do one of the following:
   - Choose Create > Character > Quadruped Guide from the Animate toolbar.
   - Choose Get > Primitive > Model > Quadruped - Guide from any toolbar.

2. Translate the red manipulator cubes on the guide to proportion the limbs so that it fits within the final envelope to be used.

When you manipulate any cube, the corresponding cube on the opposite side of the body is manipulated in the same way. This makes it quick and easy to set up matching proportions for both sides of the body at once.
Creating a Quadruped Rig

The rig is a skeleton that includes control objects that you can position and orient to animate the various parts of the character body (see the illustration on the next page).

The rig provides a foundation of rigging components, allowing you to define as much detail as you need. Although there are a great deal of styles available from generating a rig this way, it is simple to do further modifications. The rig is composed of standard XSI components that you can delete, constrain, or connect to new objects.

Unlike the biped rig, the preference angles are aligned for a proper quadruped with the legs pointing inward. You should always make sure that the rig is aligned correctly with the corresponding anatomical locations of the animal character’s bones. It’s worth the time to do research on the animal’s skeletal structures and motion when starting a project. This is good practice for successful animal creation and animation.

To load a predefined rig based on default guide values, choose Get > Primitive > Model > Quadruped - Rig from any toolbar.
Synoptic View

There is a synoptic view available to control the rig, which you can open by selecting any part of the rig and pressing F3. This lets you easily select and key the rig and reset it. When you scale the rig, the synoptic’s control systems adapt to the new proportions so you can still select items.

Creating the Rig

To create a quadruped rig

1. Modify the proportioning guide to match your character’s proportions (see page 190).

2. Choose Create > Character > Rig from Quadruped Guide from the Animate toolbar.

3. In the Make Quadruped dialog box that appears, set up the elements of the rig as you need.

The sections following this procedure provide a summary of the options on each property page, starting on page 194.
4. Click OK and a new skeleton rig based on the guide’s proportions and the options you selected is created.

5. After the rig is created, the Quadruped Controls appear. These let you scale the spine and tail, as well as set the foot roll angle. See Using the Quadruped Controls on page 196 for more information.

6. Envelope the body geometry to the rig.

   You can hide the proportional guide after you’ve created the rig so that it’s easier to see the rig.

   It’s a good idea to save a copy of the character’s proportional guide in case you want to change the rigging options later.

7. Translate and rotate the appropriate rig controls and key them to animate the parts of the skeleton.
Chapter 5  Character Rigging

Setting Up the Chest, Belly, and Head

The options on the Chest, Belly, and Head pages in the Make Quadruped dialog box let you select the way you want the spine, shoulders, belly, and head (neck) to be set up. These options are the same as available for the biped rigs:

- For options on the Chest page, see Setting Up the Torso Spine on page 178.
- For options on the Belly page, see Creating a Belly Control on page 179.
- For options on the Head page, see Setting Up the Head, Neck, and Ears on page 180.

Creating the Tail

On the Tail page in the Make Quadruped dialog box, you can choose whether or not to create a tail for the rig. A tail is basically a chain where each bone is controlled by a spring that reacts to the rig's animation (secondary animation). You can also animate the cube controls on the curve in FK to give precise control and posing when required.

If you select Make Tail, you can then specify the number of vertebrae Divisions. The more divisions, the more chain elements that will trace the inner curve of the tail. The outer cage driven by the spring stays the same. If you need greater control over the tail where it is in a specific pose, like a curl, you can set the FK/IK Blend on this chain and control it in FK, ignoring the outer cage.

- To open a control slider set for modifying the tail’s spring, see Accessing the Springs’ Control Sliders on page 188.
- You can also create a separate tail element. You can then use this tail with any other rig that you create. See Creating a Tail on page 58 for more information.

Creating Shadow Rigs

The options on the Shadow Rigs page in the Make Quadruped dialog box let you create a secondary rig at the same time you’re creating the main rig. The types of shadow rigs you can create are the same as available for the biped rigs.

For options on the Shadow Rigs page, see Creating Shadow Rigs on page 184.
Setting the Roll Division

The options on the Roll Division page in the Make Quadruped dialog box allow you to choose how you want the skeleton's envelope to be weighted at a more refined level when the bone is rotated (rolled). These options create a set of nulls based on the number of subdivisions that you specify, to distribute the weighting along the bone. These options are quaternion-based.

These options are similar to the ones for the biped rigs: see Setting the Roll Division on page 182 for more information.

- The Humerus (upper front leg bone) Roll Subdivision option break up the rotation along the length of the humerus so that the nulls allow more rotation at the top of the bone than at the end.

  For example, the top of the humerus is joined at the shoulder (and “armpit”), which requires more rotation than the leg joint (equivalent to an elbow). This allows for more refined envelope weighting for these difficult areas.

- The Femur (upper hind leg bone Roll Subdivision option divides the rotation in the opposite way as the humerus because more rotation is required at the “wrist” instead of the elbow. This is done with built-in IK/FK blending.

  In this case, the rotation starts at the end of the second bone where it joins the paw. This helps solve the problem when there's animation on the last bone of the leg, paw effector, paw root, or paw bone.

Although it's unlikely that you will rotate paws beyond the -180 to 180 degree range that the roll provides, you can extend this range. For example, certain setups may require that you adjust exactly where the center of the seam line is. To do so:

1. Select a paw bone and open it in the explorer.
2. Double-click the Roll_Compensation parameter and tweak the roll_offset slider to get the range you need.
Chapter 5 • Character Rigging

Using the Quadruped Controls

After the rig is created, the Quadruped Controls are displayed. You can use these to adjust the rig's spine and foot roll.

If you close these controls, you can open an explorer and click the Controls custom parameter set icon under the Quadruped model.

- The Scale slider dynamically reproportions the quaternion spine's length.
  
  If you selected the Fixed Length type for the rig's spine (see page 178), this is the only way in which you can scale the spine.
  
  If you created a quaternion head spine (neck), there is a separate Scale slider that lets you control it.

- The Volume Factor slider scales the quaternion spine vertebrae as the length of the spine changes. This may also result in the scale being inherited by the upper body controls (front legs, spine, neck, and head). Use positive values to shrink the upper body and negative values to expand it.
  
  If you don't want the head and front legs to be scaled down the vertebrae by the volume factor, select them, press Ctrl+k to open the Local Transform property editor, and deselect the Constrain > Scaling option on the Options page.

- The Front/Back Left/Right Foot Roll 1 and 2 sliders set the angle at which the break points of the foot roll propagate rotation.
  
  For example, an elephant walks with a very shallow foot roll angle, which is basically the width of the foot's bottom. Bears, on the other hand, have a back foot more like humans so there is a more exaggerated stepping angle, closer to 20 degrees.

- The Back Left/Right Lock Percentage sliders determine the length percentage that will lock the two lower bones of the hind leg. For example, if this value is set to 1, the two lower bones cannot rotate forward beyond a straight line as you rotate the leg's extension control. If you set the lock percentage value to 0.95, the bottom leg bone cannot rotate beyond 95% of the length of a horizontally-locked leg.
  
  For creatures like an ostrich that can rotate the two lower leg bones very straight while running, a value like 0.97 would make sense. For creatures like a cat, this maximum reach is not as straight, so a value like 0.95 would work better.
Creating Control Splines for Facial Rigs

XSI includes facial animation tools for creating control splines that you can attach to the rig for your character’s body.

Control splines are used for facial setups where you want to simulate skin sliding by using a network of curves to simulate where the flow lines of deformation are on your character’s face. Some areas are typically aligned with muscles, like the cheeks or forehead, while others align with boundaries, like eyelids and mouths.

Control splines help you animate different parts of the character’s face (envelope) to deform it, such as for lip synching. For example, you could create control splines for eyelids, eyebrows, and the mouth. For information on shape animation for lip synching, see Chapter 4: Shape Animation in the Nonlinear Animation guide.

You can use entire curve networks for control splines, so you could also use them to frame out areas of the body for complex character deformations.

To create control splines

1. Create one or more curves (splines) drawn over your character’s geometry.

   You can use surface snapping to keep the lines close to the surface (it doesn’t need to be exact). If multiple curves are snapped to meet at a point of the face (such as the corner of an eye), they will be attached by a single controller.

2. Select one or more curves and choose Create > Skeleton > Create Control Splines from the Animate toolbar.

3. In the Controller Setup dialog box, set these options and then click OK:
   - Set the Divisions, which is the number of controls to be created per control spline.
   - Set the Scale to determine the size of the controls.
   - Select one or more Control Types. Each of these control types can be set with no controller, or with nulls, cubes, squares, pyramids, or diamonds:
     - Point Type control objects move the points of the curve and smoothly interpolate without skin stretching.
     - Path Type controls are path-constrained to the splines at length percentages.
     - Offset Type controls are used for the translation (for pinching of wrinkles and facial expressions) and rotation (twisting along the curve) of the splines for areas like the lips.
- Set the color for each type of control for easy identification.

Control splines with created control objects. In this example:
- Point controls are **diamonds**
- Path controls are **nulls**
- Offset controls are **cubes**

Control splines place the objects to be enveloped (the Offset controls) in a group called envelope_group under the scene root. Typically, you would add the objects from this group to the envelope_group for your entire rig.

4. After applying control splines, select the Path control objects and set the up-vector object for their path constraints. For a facial setup, that is usually the head bone or end vertebra.

5. To animate the control splines, set keys on the position and rotation of the Point and Offset controls.
Creating Hierarchy Setups from a Guide

In addition to rigs, as described in the previous section, you can create a variety of typical character hierarchy setups from any proportional guide—biped, biped dog leg, or quadruped. These skeleton hierarchies are similar to the shadow rigs that you can create when you create a rig, as described in Creating Shadow Rigs on page 184.

The hierarchy setup rigs have a synoptic view, similar to what’s available for the basic rig. Select any part of the rig and press F3 to open its synoptic page.

To create setups from a guide

1. Set up and select any proportioning guide (see Setting Up the Biped or Biped Dog Leg Guides on page 169 or Setting Up the Quadruped Guide on page 190).

2. Choose Create > Character > Hierarchy from Guide from the Animate toolbar.

To load predefined hierarchy setups based on default guide values, choose Get > Primitive > Model > Biped - Box, Biped - SI[3D Skeleton, or Biped - Skeleton (XSI style) from any toolbar.
3. In the dialog box that appears, select the Type of hierarchy you want to create:
   - Made of nulls (a null is created at the root of each chain)
   - Made of boxes (as shown above)
   - A simple SOFTIMAGE|3D-style skeleton (effectors are children of the last bone)
   - A simple XSI-style skeleton (effectors are children of the chain root)

4. Set the number of **Lumbar Divisions**, which is the number of bones/nulls/boxes that make up the spine.

5. Click OK and the skeleton hierarchy is created.
Controlling the Binding between Models

When two characters that are in separate models are constrained together (such as a rig and its shadow rig), you need tools to control the binding between them without interfering with the character setup within each model. When you create a shadow rig with an XSI rig (see Creating Shadow Rigs on page 184), it’s automatically constrained to the rig.

For example, let’s say Model A (an animation rig) is constrained to Model B (a hierarchy with motion capture). There are a large number of position, orientation, and chain up-vector constraints that keep Models A and B together. By branch-selecting Model B and choosing the commands, you can select, activate/deactivate, or remove all of the constraints that go from Model B to Model A. None of the constraints between objects within Model B’s structure are affected.

These commands let you control the use of any constraints on a model outside its structure. The definition of a constraint that is external to a model is when the constraining objects are not part of the same model structure as the constrained object. None of the constraints between elements within a model structure (that is, the rig) are affected by these commands.
To control the constraints between models

1. Make sure that each character or rig that is constrained is within a model structure.

2. Branch-select the model to which another model or object is constrained.

3. Choose one of the Create > Character > Constraints Outside Model Boundaries commands from the Animate toolbar:
   - Select selects all the constraints between the models.
   - Activate/Deactivate turns all constraints between the models on or off.
   - Remove deletes all the constraints between the models.
Chapter 6  

Tools for Character Animation
Chapter 6 • Tools for Character Animation

Making Character Animation Easier

There are many standard tools in XSI that are not particularly designed for character animation, but can help your job of character animation go much more smoothly. Some of these tools are simple aids in making it easier to visualize or select elements, while others can solve major animation problems and change the way you work.

This chapter is a guide to some of those tools and techniques. You will find out why the animation mixer is a powerful tool for character animation, how you can set up custom toolbars and synoptic views to streamline the workflow and share information with other, and how to make animation easier with marking sets and transformation setups.

For information on viewing options for character animation, see Displaying Chain Elements on page 34.

For information on tools for rigging, see Chapter 5: Character Rigging on page 163.

For tutorials on building and animating a character using many of the tools described in this chapter, see the Character Animation lessons in the Tutorials guide.
All-Purpose Tools for Increasing Productivity

Both custom toolbars and the synoptic view allow you to set up whatever you like for character animation. They are typically used to make it easy to select and key character elements.

Custom Toolbars

Although these require a bit of work setting up, custom toolbars can save you a lot of time and effort in the long run. You can create a toolbar for a character, then add all sorts of custom buttons that run simple scripts to select different elements or open property editors.

For example, you can have a button that selects an arm's effector or the whole spine, or one that opens a property editor with proxy parameters of frequently-used IK parameters—it's all up to you!

For more information on custom buttons and toolbars, see Chapter 3: Custom Commands in the Customization guide.

Synoptic Views

Synoptic views allow you to quickly access commands and data related to a specific object or model. They consist of a simple HTML image map stored as a separate file outside of the XSI scene file. The HTML file is linked to a scene element by a Synoptic property. Clicking on a hotspot in the image either opens another synoptic view or runs a script.

You can set up a character synoptic view for other members of your team, allowing them to use your character easily. You can include all sorts of information about the character, set up hotspots for selecting body parts, setting keys on different elements, running a script, etc.

To see some synoptic pages, create a biped or quadruped rig (see page 168), select any of its elements and press F3.

For more information on creating a synoptic, see Chapter 12: Synoptic Views in the Customization guide.
Chapter 6  •  Tools for Character Animation

Organizing and Protecting a Character’s Elements

Layers

Layers let you divide up different scene elements into groupings whose visibility, selectability, and renderability can be controlled. You can use layers to break a character down into sections so that you can quickly change selectability and visibility each layer.

For example, separate the model’s geometry, the skeleton, and the constraining objects for the rig each into different layers. This makes it easy to select and key only the elements that you want to.

Layers, however, live only at the scene level, so if you’re importing and exporting models between scenes, they’re not going to include any layer information. This is where groups can be of help.

For more information on layers, see Chapter 14: Layers on page 331 in the Fundamentals guide.

Groups

Groups also let you put certain scene elements together, but unlike layers, groups can be made children of models and exported or imported with them. You can set up the same sort of groupings as you had with layers, except that your character needs to be within a model structure first.

Groups allows you to select multiple objects at a time and are important for sharing materials and setting up texture supports for many objects at a time.

If you’re using a number of methods of organizing your scene elements, beware of conflicting overrides between layers, groups, and render pass partitions. A single object can be told to be visible by its layer, invisible by a group, and then visible again by a partition all at the same time. This could lead to premature baldness from pulling out your hair at render time!

For more information on groups, see Grouping Objects in Chapter 9 in the Fundamentals guide.

Using Consistent Naming Conventions

Use consistent naming conventions such as including a “g_” as a prefix for all of your geometry nodes. That way you can select entire groups of geometry objects just by typing g_* in the Selection text box. It also lets you write name-based utility scripts.

For example, you can write scripts that seek out all left-side elements (”l_”) and duplicate them and rename them to right-side elements (“r_”), or vice versa. This can be a huge time-saver!

Another reason to use naming conventions is to make selection easier. For example, if you add a “c_” as a prefix for all rig control elements, you know which parts are okay to animate. Then when you want to store poses and actions, you can select all the controls by typing c_* in the Selection text box.
When you create a scene or model, there may be certain parts of it that you don’t want to change or that you don’t want anyone else to change (this is particularly true in a workgroup environment). To help reduce the introduction of error, you can use locks to control the type of modifications allowed to parts of a scene. For example, if you’re a technical director, you can use locks to prevent modifications (accidental or otherwise!) to a rigged character. Or you can lock only the animation of certain parameters so that they’re available but can’t be keyed.

You can lock 3D objects, local models (not referenced models), parameters, animation mixers, groups, layers, clusters, and passes.

You can also apply locks at different levels: topology, animation, value, and all. These levels let you lock scene elements so that only certain types of modifications are restricted. For example, you cannot add or remove points from geometry that has been locked at the topology level.

For more information, see Chapter 23: Locking and Tagging Scene Elements in the Fundamentals guide.
Tools for Easy Animation

There are many tools in XSI made specifically for animation, but these are some general tools that are particularly useful for character animation.

Ghosting Animation

Animation ghosting, also known as onion-skinning, lets you display a series of snapshots of animated objects at frames and/or keyframes behind and/or ahead of the current frame. This lets you easily visualize the motion of an object, which can help you improve its timing and flow.

Drawing ghosts only on keyframes lets you quickly see where a character’s keys are. If you add or remove a key with ghosting on, the ghosts are immediately updated to reflect the change.

You can use different display modes for ghosting, such as the character’s geometry, motion trails, or simply points.

For more information, see Ghosting Animated Objects in Chapter 2 in the Animation guide.

Setting a Neutral Pose with Zero Values

When you’re creating a skeleton, its neutral pose is often the character with outstretched arms and legs, making it easy to weight the envelope and adjust its textures. However, this doesn’t always give the best values for animating the skeleton’s local transformation values.

To overcome this problem, you can create a neutral pose which uses zero for its local transformation values (0 for rotation and translation, 1 for scaling). Then when you key the character’s values, they reflect the relative difference from zero, and not a value that’s difficult to use. And when you can select a chain element and enter zero for its local rotation, for example (often referred to as zeroing out), it returns to this neutral pose.

For information on how to do this, see Creating Neutral Poses for Skeletons on page 64.
Rotoscopy for Making Models and Animating

Use rotoscoped images of models to act as a template from which you can create skeletons or envelopes. For example, import an image of an animal in an appropriate pose into XSI in the Rotoscopy view, create the envelope to match it, and create the skeleton to fit within the model.

As well, you can use rotoscoped images to help you animate a character, such as setting up and matching to a walk cycle sequence.

For more information on rotoscopy, see Chapter 6: Rotoscopy in the Fundamentals guide.

For an example of using rotoscopy for animation, see the Walk Cycle tutorial in the Tutorials guide.

Marked Parameter “Memory Sets”

A useful tool to help make keying quick and easy is the marking set. This lets you set up which parameters of an object should be marked for keying. Once you create the marking property, it’s easy to activate the marking set so that you can key the parameters in it.

Marking is an efficient way of saving only the information that you need to be animated, and you can also have XSI “remember” which parameters you have marked on an object. For example, if you’re always keying the rotation of a certain bone, you can save a marking memory set with those parameters, and then just call up the set when you need it.

To select only objects that have marking sets, select the Obj < Marking Set > filter from the Selection list (click the arrow button in the Select panel to open the list).

For more information on marking parameters and memory sets, see Marking Parameters for Animation in Chapter 2 in the Animation guide.

For an example of creating marking sets for keying, see the Walk Cycle tutorial in the Tutorials guide.
Chapter 6 • Tools for Character Animation

Preferred Transformation Tool (Transform Setup)

When keying an object, there is a good chance that you will always animate an object using the same parameters. This is possible if you create a transformation setup for it. Transform setups let you specify a preferred transformation tool for individual objects.

This is very useful for the control objects of your rig—for example, select the center-of-gravity control and the Translate tool is automatically activated, select the spine control and the Rotate tool is automatically activated, and so on.

You can apply a transform Setup property using Get > Property > Transform Setup in any toolbar—see Transform Setup in Chapter 16 of the Fundamentals guide for more information.

For an example of creating transform setups for keying, see the Walk Cycle tutorial in the Tutorials guide.

Creating Custom Sliders

Using custom parameters, you can create your own animation controls via expressions or linked parameters. You could control different rig elements or body parts via expressions using custom sliders, such as the movement of a hand and each of its fingers.

Creating Your Own Property Editor with Proxy Parameters

Using proxy parameters, you can bring a number of parameters from different locations in XSI under one “roof”, creating your own property editor. Proxy parameters are clones of existing parameters: any changes you make to them has the same result as using the real thing. Having all pertinent parameters grouped together saves time because you only have to look in one place, allowing you to focus only on the parameters that matter.

When you’re done creating custom property editors, create a custom button that opens them up with one mouse click (see page 205). Alternatively, you can set up a synoptic control page where you have access to all the character’s control panels via a picture of your character (see page 205).
To select *only* objects that have custom property sets, select the **Obj w Custom PSet** filter from the Selection list (click the arrow button in the Select panel to open the list).
Using the Animation Mixer for Character Animation

While you may not use the animation mixer for directly animating a character, it can be an invaluable friend during many other parts of the character animation process.

The mixer is a control layer in XSI that takes priority over the standard controls of fcurves. While fcurves provide a graphical way of visualizing and editing changes in motion, and interpolating between poses, the mixer provides a more abstract interface for visualizing and editing the timing and the mixing of poses and animation clips. For example, you can easily save an action, then offset it in time, cycle it, slow down and speed up the animation, etc.

- For general information on using the mixer, see Chapter 2: The Animation Mixer in the Nonlinear Animation guide.
- For general information on storing and using animation in actions, see Chapter 3: Actions in the Nonlinear Animation guide.

The Importance of Models

Understanding how the mixer works with respect to models is critical to avoid problems. For example, forgetting about parameters in some clips while including them in others can lead to undesirable effects at the frames where the mixer has precedence.

For more information on using models for characters and with the mixer, see Setting Up Characters within a Model Structure on page 20.
Using the Animation Mixer for Character Animation

**Editing Existing Material**

The animation mixer is well-suited for editing existing material and bringing together all the pieces of a character animation. In it, you can assemble all the bits and pieces you’ve imported from different scenes and models and help you build them into a final character.

If you’re modifying someone else’s animation, you don’t really have to deconstruct their work—just add a layer with your own animation. You can even modify the existing animation with a clip effect, acting as a separate and removable layer on top of the original animation.

**Blocking Out Rough Animation**

You can use the mixer to rough out animation without even setting keyframes. Store the character’s key poses as static pose clips, plop them in the mixer, and block them out using automatic transitions and mixing.

This results in a rough pose-to-pose animation, even though you haven’t keyed anything yet. This is an ideal place for tweaking the timing because it’s easy to move clips around in the mixer.

You can also replace existing static poses or add more to the mixer. Since there aren’t any fcurves yet, there’s no fear of messing up timings or complex fcurve interpolations.

**Using the Mixer and Fcurves**

Once you’re happy with the rough animation, you can start marking the animated parameters and keying their values. This is like plotting the animation except that you have complete control over which parameters are keyed and at exactly which frame.

Remember that an action clip overrides any animation on the character at the same frame, but you can use this to your advantage: switch between the rough animation in the mixer and the keyed animation on the character by muting the appropriate mixer tracks. With the mixer tracks on, you can set multiple keys before and after the frames of the poses, mute the tracks and tweak the fcurves to add anticipation, and then follow-through to the animation.

What makes XSI powerful is the combination of using standard keyframing techniques with the editing prowess of the mixer. You can pass animation data back and forth between control levels as needed without being forced to plot animation curves or losing the structure of the underlying keys and their tangency controls. The latter is really important if you need to keep the number of keys minimized or like to animate in a pose-to-pose style with fcurves.

For more information on storing animation in actions and loading them in the animation mixer, see *Storing Action Sources* in Chapter 3 and *Adding Action Clips to the Mixer* in Chapter 3 in the *Nonlinear Animation* guide.
Relational Views with the Mixer and Animation Editors

There are several sample relational views that contain the animation mixer in them. Relational views are simply combinations of different views contained within a single window. Having certain views together in one window can make typical mixer animation tasks easier.

In addition to these sample relational views ready for you to use, you can also create your own custom relational views, as described in Chapter 5: Layouts and Relational Views in the Customization guide.

To open the sample relational views

1. Choose Application > Views from the XSI main menu.
2. Select any of the RV views as described on Relational Views with the Mixer in Chapter 1 in the Nonlinear Animation guide.

Sample relational view called Timeline Mix Fcv is a combination of the mixer with an fcurve editor and a shared timeline.

This view is often useful when you’re editing the fcurves inside the selected action clip.
### Index

#### Numerics

| 2D and 3D chains | 30 |

#### A

| Acclaim (ASF and AMC) files | 158 |

### action sources

- skeleton poses | 68 |

### actions sources

- skeleton poses | 64 |

### AMC (Acclaim Motion Capture) files | 158 |

### animation

- character | 21 |
- retargeting on tracks | 154 |
- sample for characters | 25 |

### animation mixer

- for character animation | 212 |

### articulated chains

- See chains |

### ASF (Acclaim Skeleton) files | 158 |

### Automatic Envelope Assignment property | 77, 79 |

#### B

| base pose | See neutral poses |
| belly control in character rigs | 179 |
| bind poses | See reference poses |
| Biovision (BVH) files | 159 |

### bipeds

- control set | 185 |
- creating dog leg rig from guide | 173 |
- creating rig from guide | 173 |
- foot roll | 185 |
- hierarchy setups | 199 |
- predefined rigs | 168 |
- proportional guide for dog leg rigs | 169 |
- proportional guide for rigs | 169 |
- retargeting motion on tracks | 154 |
- roll division | 182 |
- shadow rigs | 184 |
- tags for modifying motion | 155 |

### blending

- between forward and inverse kinematics on a chain | 127 |
- effector and bone orientation | 136 |
- linking effector orientation with chain's FK/IK blend | 136 |
- blocking out a rough animation | 213 |
- body
  - designing with predefined model | 26 |
  - pre-built models | 25 |
- bones
  - about | 31 |
  - adding to chain | 48 |
  - aligning root with first bone's orientation | 119 |
  - blending with effector's orientation | 136 |
  - changing axis alignment | 120 |
  - creating spines | 53 |
  - inheriting the scaling | 52 |
  - keying rotation for all | 119 |
  - resetting angle (preferred rotation) | 45 |
  - resetting preferred rotation | 137 |
  - resizing | 49 |
  - rolling in an IK chain | 142 |
  - rotating in order | 62 |
  - shadows | 35 |
  - system
    - See skeletons
    - See also skeletons, chains |

### bounding volumes in envelopes

- about | 81 |
- associating | 83 |
- combining | 86 |
- creating | 83 |
- dissociating | 83 |
- exclusive | 82 |
- inclusive | 82 |
- limit | 83 |
- types | 82 |
- breaking chains | 150 |
- brush properties | 93, 95 |

### BVH (Biovision) files | 159 |

### C

| chains | 2D or 3D | 30 |
| adding bones to | 48 |
| animating effector like
  - SOFTIMAGE|3D | 124 |
| animating on path | 125 |
| animating with forward kinematics (bone rotation) | 116 |
| animating with inverse kinematics (translating effector) | 121 |
| blending forward and inverse kinematics (FK/IK blending) | 127 |
| bone axis alignment | 120 |
| bones | 31 |
| breaking | 150 |
| changing shape of elements | 35 |
| controlling orientation | 126 |
| creating | 42 |
| creating from curves | 44 |
| critical zone at root | 34 |
| display of | 35 |
| displaying joint circles | 34 |
| displaying only chain elements | 34 |
| drawing in symmetry | 46 |
| drawing SOFTIMAGE|3D style | 44 |
| duplicating in symmetry (mirroring) | 46 |

### effectors

- 32 |
- 32 |
- elements in explorer | 33 |
- forcing IK behavior | 133 |
- in hierarchy (skeleton) | 60 |
- joints | 31 |
- keying rotation for all bones | 119 |
- Kinematic Chain property
  - editor | 33 |
- orientation of effector | 135 |
- parts of | 31 |
- preferred angles of joints | 137 |
- preferred angles when drawing | 122 |
Index

preferred axis of rotation 144
preventing flipping 126
pseudo-roots (pinning joints) 151
resizing bones 49
resizing by moving joints 49
resolution plane 122
rolling a bone for IK chains (in X) 142
root 31
rotation limits for joints 146
shadows for elements 35
solving IK behavior 132
stiffness in joints 149
subchains 150
up-vector constraints (up direction) 138
See also bones, effectors, joints, skeletons
character animation
animating with kinematics 114
biped dog leg rig 173
biped rig 173
blocking out rough 213
creating custom parameter sets 210
custom toolbars 205
models in 20
naming elements 206
overview of process 21
predefined body 26
predefined skeletons and envelopes 24
quadruped rigs 190
retargeting motion 154
rigs for controlling movement 164
rotoscopy 209
sample animation 25
skeletons 30
storyboards 21
synoptic views 205
transform setups 210
using animation mixer 212
using layers and groups 206
using marking sets 209
character designer 26
Character Development Kit for SDK 167
clusters
reallocating to envelope deformer locally 88
colors
envelopes 104
constraints
blending with chain's IK/FK blend 129
controlling between rig and shadow rig 201
curve (path) with chains 125
preferred axis for chains 144
up-vector for chains 138
control rigs for character animation See rigs
control splines
for facial animation rigs 197
in rigs for approximating body volume 183
critical zone at chain root 34
curves
creating chains from 44
D
defomers
adding to envelopes 107
envelopes 71
removing from envelopes 108
distance-based envelope assignment 78
dog leg biped guide 169
dog leg biped rig 173
drawing
chains 42
duplicating
in symmetry 46
E
ear controls in character rigs 180
effectors
about 32
animating (inverse kinematics) 123
animating like
SOFTIMAGE|3D 124
blending orientation with bone 136
changing orientation 135
forcing position 133
inheriting orientation from bone 135
linking orientation with chain's FK/IK blending 136
shadows 35
SOFTIMAGE|3D chains 32
solving on 132
deffectors of chains See effectors
Envelope > Assign to No Deformer 89
Envelope > Create Bounding Volume 83
Envelope > Create Symmetry Mapping Template 101
Envelope > Mirror Weights 103
Envelope > Reassign Locally 88
Envelope > Remove Deformers 108
Envelope > Remove Envelope 80
Envelope > Reset Actor 79
Envelope > Select Deformer from Envelope 80
Envelope > Set Envelope 76, 107
Envelope > Set Reference Poses 106
envelopes
about 71
adding deformers 107
assigning points to different deformers 88
assigning to deformers 76, 88
bounding volumes 81
combining with soft body 112
deformers 71
display colors 104
distance-based assignment 78
editing weights numerically 97
freezing transformations 75
freezing weights 103

geometry types 74

global 73

importing 112

local 73

mirroring weights 100

modifying properties 79

modifying weights 90

muting 112

normal-based assignment 78

normalizing weights 97

on subdivision surfaces 96

painting weights 93

predefined body 26

predefined human 24

presets 105

reference poses 74, 106

removing 80

removing deformers 108

removing deformers locally 89

resetting reference poses 106

returning to reference pose 79

rigid 73

smoothing weights 95, 100

subdivision surfaces 74

surface meshes 74

symmetry 100

transforming 110

weight operator stack 73

weights 72

x-ray shading 38, 111

error threshold for IK solver 134

exclusive bounding volumes 82

gap (generic attribute painting)  See painting

genus types

for envelopes 74

ghosting

forward and inverse kinematics

blend (FK/IK) on chains 130

gimbal lock

changing order of rotations 62

global envelopes

about 73

groups

objects as skeleton 73

files

motion capture 158

FK  See forward kinematics

FK/IK blending

about 127

ghosting FK/IK states 130

linking effector's orientation with 136

motion capture data 129

plotting animation 128

with constraints 129

flipping

chain's orientation 126

foot roll

biped and dog leg rigs 185

quadruped rigs 196

forcing IK behavior 133

forward kinematics

about 114

animating with 116

blending with inverse kinematics 127

changing order of bone rotation 62

keying rotation for bones 119

posing a chain with inverse kinematics 116

freezing

envelope weights 103

transformations on envelopes 75

G

GAP (generic attribute painting)  See painting

genus types

for envelopes 74

ghosting

forward and inverse kinematics

blend (FK/IK) on chains 130

gimbal lock

changing order of rotations 62

global envelopes

about 73

groups

objects as skeleton 73

h

hierarchies

chains in (skeletons) 60

I

IK assembly  See skeletons

IK  See inverse kinematics

IK/FK blending  See FK/IK blending

importing

Acclaim (ASF and AMC) files 158

Biovision (BVH) files 159

envelopes 112

motion capture files 158

inclusive bounding volumes 82

inverse kinematics

about 114

animating effector 123

animating with 121

blending with forward kinematics 127

forcing IK behavior 133

preferred angle of chain 122

resolution plane for chain 122

rolling a bone in X 142

SOFTIMAGE|3D behavior for solving joint rotations 133

solver angles for joint rotations 133

solvers 132

solving joint rotations from preferred angles 133

translating effector 124

using with path 125

J

joints

about 31

changing preferred angle 137

displaying circles around 34

methods for solving rotations 133

pinning with pseudo-roots 151

preferred angle 122

resizing by moving 49

rotation limits 146

shadows 35

stiffness in (arthritis) 149

f

Face Maker  about 25

facial animation

controls splines for rigs 197

file types

.amc 158

.asf 158

.bvh 159
See also bones, chains, inverse kinematics

K
Kinematic Chain property editor 33
kinematics blending between forward and inverse 127
forward 116
forward and inverse for chains 114
inverse 121

L
limit bounding volumes for envelopes 83
local envelopes about 73

M
marking sets selecting objects with 209, 211
mirroring envelope weights 100
models creating from predefined human 26
for character animation 20
pre-built bodies 25
pre-built heads 25
motion capture data Acclaim files 158
Biovision files 159
blending with inverse kinematics 129
changing bone axis alignment 120
importing files 158
motion deformation about 154
setting biped tags 155
step marks 155
Move Joint/Branch command 49
moving joints to resize bones 49
muting envelopes 112

N
neutral poses skeletons 64
normalizing, envelope weights 97
normals-based envelope assignment 78

O
opacity, brush 96
operator stack envelope weights 73
orientation aligning root with first bone in chain 119
blending effector’s with last bone’s 136
chain effector with last bone 135

P
painting brush properties 95
envelope weights 93
on subdivision surfaces 96
paint modes 95
paths animating chains on 125
plotting IK/FK blend on chains 128
points of articulation See joints polygon meshes pre-built body models 25
pre-built face models 25
poses default for skeletons 64
pose-to-pose animation using animation mixer for 213
preferred angles drawing chains 122
for joints 137
preferred axis of rotation for chains 144
preferred rotation for bone resetting 45, 137
presets envelopes 105

R
radius, brush 95
reference poses about 74
resetting 106
returning to 79
reference poses See also neutral poses
resolution plane
for chains 122
rigid envelopes simulating 73
rigs action sources in 174
arms and finger controls 181
belly control 179
biped and biped dog leg controls 185
biped dog leg proportional guide 169
biped hierarchy setups 199
biped proportional guide 169
Character Development Kit for...
SDK 167
control splines for faces 197
controlling constraints between rig and shadow rig 201
creating biped dog leg rig from guide 173
creating biped from guide 173
creating quadruped from guide 191
default transformation values 186
deleting and duplicating elements in guide 171
ear controls 180
for character animation 164
head and neck controls 180
marking sets in 174
nulls in 165
predefined biped 168
predefined quadruped 168
quadruped controls 196
quadruped proportional guide 190
quaternion rotation spine 178
roll division for bipeds 182
roll division for quadrupeds 195
scaling control objects 187
setting volume controls for approximating body envelope 183
shadow 184
shadows of chain elements 35
springs in controls 188
tail controls 194
zeroing out controls 165, 208
roll division
biped rigs 182
quadruped rigs 195
rolling
bone in an IK chain 142
roots
aligning with first bone’s orientation 119
of chains 31
shadows for chain elements 35
rotations
changing order of for bones 62
keying all bones in a chain 119
limiting joint movement 146
limits for chain joints 146
rotoscopy
for character animation 209
S
scaling
bones 49
bones by moving joints 49
SDK, Character Development Kit 167
shadow rigs
about 184
controlling constraints between it and rig 201
shadows
display for chain elements 35
shape animation
predefined heads with Face Maker 25
skeletons
about 18, 30
animating with kinematics 114
changing shape of elements in 35
creating spines 53
in actions (poses) 68
predefined human 24
putting in models 20
reference poses stored in an action 64
rigs 164
sample animation for 25
setting up 60
shadows 35
storing default poses 64
See also chains and envelopes
skin See envelopes
smooth operator 100
smoothing
envelope weights 95, 100
soft body
combining with envelopes 112
SOFTIMAGE|3D
animating effector in chain like 124
drawing chains in style of 44
effectors in chains 32
importing envelopes from 112
Move Joint/Branch command 49
preferred X rotation as roll for bones 143
solving joint rotations behavior 133
solvers for inverse kinematics 132
error threshold 134
spines
creating for skeleton 53
crunching 57
depth for curve 55
quaternion rotation 53
quaternion rotation in rig 178
rolling 56
scaling 55
springs in rig controls 188
step marks for motion deform 155
stiffness
in chain joints 149
storyboards for animation 21
subchains
creating 150
pinning joints with pseudo-roots 151
removing 151
subdivision surfaces
for envelopes 74
painting weights 96
pre-built face models 25
surface meshes
for envelopes 74
symmetry
drawing chains in 46
duplicating chains in 46
mirroring envelope weights 100
symmetry mapping templates 100
symmetry maps
on envelopes 100
synoptic views
Index

for biped rig 174
for character animation 205
for quadruped rigs 192

T
tail controls for character rigs 194
terminators See subchains
transform setup
  character animation 210
transformations
  envelopes 110
translations
  animating chain’s effector (inverse kinematics) 124

U
up vectors
  constraints for chains 138

V
viewing
  critical zone for chain root 34
  filtering chain elements 34
  joint circles 34
  joint stiffness 146
  rotation limits 146
  shadows for chains 35
  x-ray view of chains with envelopes 38
volume controls
  approximating envelope for rig creation 183

W
weight editor 92, 98
weight maps
  envelopes 93
  on subdivision surfaces 96
weight paint panel 91
weights
  envelopes 72, 90
  mirroring 100
  operator stack 73

X
x-ray shading
  on envelopes 111

Z
zeroing out joints (neutral poses) 64