

Андрей Прахов

Blender

3D-моделирование и анимация

Руководство для начинающих

Санкт-Петербург

«БХВ-Петербург»

2009

УДК 681.3.06
ББК 32.973.26-018.2
П70

Прахов А. А.

П70 Blender: 3D-моделирование и анимация. Руководство для начинающих. — СПб.: БХВ-Петербург, 2009. — 272 с.: ил. + CD-ROM — (Библиотека ГНУ/Линуксцентра)

ISBN 978-5-9775-0393-8

Руководство для начинающих по трехмерному моделированию и анимации в свободно-распространяемой программе Blender. На простых примерах показаны типичные приемы работы в описываемой области. Рассмотрены варианты создания трехмерных моделей различными способами, в том числе новейшими инструментами скульптурного моделирования, работа с NURBS, поверхностями, текстурами и материалами, использование шейдерных эффектов, возможности имитации физических явлений, а также все виды анимации, включая скелетную.

Прилагаемый CD содержит файлы рассмотренных в книге примеров, а также свободно-распространяемое ПО: установочные файлы Blender для Windows и Linux, короткометражный фильм Big Buck Bunny и набор текстов.

Для широкого круга пользователей

УДК 681.3.06
ББК 32.973.26-018.2

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Евгений Рыбаков</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Владимир Красовский</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Елены Беляевой</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 25.03.09.
Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 21,93.
Тираж 2000 экз. Заказ №
"БХВ-Петербург", 190005, Санкт-Петербург, Измайловский пр., 29.

Санитарно-эпидемиологическое заключение на продукцию
№ 77.99.60.953.Д.003650.04.08 от 14.04.2008 г. выдано Федеральной службой
по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов
в ГУП "Типография "Наука"
199034, Санкт-Петербург, 9 линия, 12

ISBN 978-5-9775-0393-8

© Прахов А. А., 2009
© Оформление, издательство "БХВ-Петербург", 2009

Оглавление

Введение	1
Глава 1. Интерфейс программы	5
1.1. Система окон и меню.....	5
1.2. Манипуляции в 3D-пространстве	12
1.3. Объекты, группы и связи.....	15
1.4. Концепция сцен и слоев	20
Глава 2. Работа с <i>Mesh</i>-объектами	23
2.1. Прimitives и их структура.....	23
2.2. Основы редактирования	25
2.3. Симметричное моделирование	35
2.4. Режим скульптуры	46
2.5. Свободное рисование с <i>Retopo</i>	51
2.6. Дополнительный инструментарий.....	54
Глава 3. Кривые, поверхности NURBS	69
3.1. Основные понятия.....	69
3.2. Работаем со сплайнами.....	70
3.3. Поверхности NURBS	79
3.4. Особенности работы с текстом	83
3.5. Эффекты для текста	89
3.6. Дополнительный инструментарий.....	91
3.7. Свойства отображения объектов	94
Глава 4. Материалы и текстуры	97
4.1. Что такое материал	97
4.2. Базовый цвет.....	98

4.3. Встроенные шейдеры	103
4.4. Отражение и преломление	108
4.5. Мультиматериалы	114
4.6. Эффекты <i>Halo</i>	116
4.7. Рамповые шейдеры	119
4.8. Использование текстур.....	122
4.9. Рельефные карты.....	129
4.10. Карты смещения.....	136
4.11. Работаем с UV/Image Editor	137

Глава 5. Анимация..... 147

5.1. Основы анимации.....	147
5.2. Простое управление с <i>Timeline</i>	150
5.3. Использование кривых IPO.....	154
5.4. Как заставить объект следовать по траектории.....	162
5.5. Деформация объекта.....	167
5.6. Создание и редактирование скелета.....	172
5.7. Обычная и инверсная кинематика	175
5.8. Нарращиваем "мясо"	186
5.9. Редактор действий для арматуры	190
5.10. Нелинейная анимация.....	196

Глава 6. Физика в Blender..... 199

6.1. Создание и использование частиц.....	199
6.2. Работа с тканями	207
6.3. Физика мягких тел	211
6.4. Силовые установки	216
6.5. Создание жидкостей	220

Глава 7. Свет, камера, окружение225

7.1. Источники света и тени	225
7.2. Работа с камерой.....	234
7.3. Настройка окружения.....	237

Глава 8. Система рендеринга Blender.....245

8.1. Основы обработки.....	245
8.2. Дополнительные возможности	247

ПРИЛОЖЕНИЯ	251
Приложение 1. Список основных горячих клавиш.....	253
Файловые операции	253
Выбор режима работы окна <i>Buttons Window</i>	253
Управление просмотром сцены	254
Создание и редактирование объектов	255
Приложение 2. Инсталляция	257
2.1. Установка программы в Windows	257
2.2. Установка программы в Linux	260
Приложение 3. Описание компакт-диска.....	261
Предметный указатель	263

Введение

В последние годы мир трехмерной графики претерпевает бурное развитие, внедряясь во многие сферы нашей деятельности. Телевидение, Интернет, печатная продукция и, конечно же, игры активно используют возможности 3D. На данный момент имеется немало программных пакетов для работы с трехмерной графикой, но несомненными лидерами являются 3ds Max и Maya. Вот только цена подобных программ исчисляется тысячами долларов. Другой путь получения качественного контента — использование свободных и открытых приложений.

Движение Open Sources (Открытые исходники) как альтернатива коммерческим продуктам зародилось довольно давно. Сейчас это выглядит не просто одним из средств самовыражения, а, скорее, своеобразной философией, затягивающей в свои объятия тысячи и тысячи людей. Linux, GIMP, OpenOffice.org, Mozilla Firefox — этот список можно продолжать до бесконечности. Многие из вас не только слышали о них, но и, возможно, использовали в своей повседневной деятельности. Эта книга посвящена одной программе из обширного лагеря Open Sources — программе для трехмерного моделирования Blender.

Многие, слыша сочетание слов "бесплатная программа", представляют себе некачественный сырой продукт. Однако реалии жизни показывают, что не всегда коммерческая программа является самой удобной и надежной, часто бывает наоборот. Это же, в принципе, касается и открытых приложений, т. к. ничто в мире не идеально. В случае с Blender вы получаете не только мощный и удобный инструмент, но и дружелюбное сообщество его пользователей, готовых всегда прийти на помощь.

Итак, Blender — это программа для моделирования, анимации, обработки видео, визуализации, которая даже может выступать в качестве игрового конструктора. Правда, здесь мы не будем рассматривать последнюю возможность в силу того, что эта тема слишком обширна и может потребовать написания

отдельной книги. По своим профессиональным качествам программа вполне может потягаться с такими монстрами, как 3ds Max и Maya, и даже более того — открою небольшой секрет — она гораздо эффективнее вышеперечисленных. Сначала непосвященному программа Blender может показаться излишне запутанной и необычной, но ведь книга для этого и написана, чтобы легче было понять и, возможно, влюбиться навсегда в эту уникальную программу. На момент написания книги последней версией программы была 2.48. Именно ее вы сможете найти на диске, прилагаемом к книге.

История развития Blender по-своему интересна и необычна. Сначала это был внутренний проект для собственных нужд, разрабатываемый компанией NeoGeo, лидером по созданию 3D-анимаций в Европе. После "отпчкования" дочерней компании NaN (Not a Number) проект стал доступен для широкой публики. Причем основная программа была бесплатной и лишь некоторые модули являлись коммерческими. По тем временам это была своего рода революция, т. к. проекты такого уровня стоили многие тысячи долларов. Мощь и бесплатность сыграли свою роль. Вскоре программа Blender могла похвастаться целой армией поклонников. В 2000 году после признания программы на выставке SIGGRAPH компания получила инвестиции в виде суммы 4,5 миллиона евро, и это ей позволило развернуться по-настоящему. Уже буквально через полгода вышла новая версия, имеющая в своем составе встроенный игровой движок. К сожалению, неверная политика в организации бизнеса постепенно привела к банкротству и, соответственно, к остановке работы по улучшению кода. Однако и после развала компании разработчики не забросили свое детище. Была создана некоммерческая организация Blender Foundation, и борьба за программу продолжилась. Так как приложение на тот момент принадлежало инвесторам компании NaN, то было принято совместное решение о выкупе кода. К всеобщему удивлению, требуемая инвесторами сумма в 100 000 евро была собрана всего за два месяца. После этого программа стала полноценным членом движения Open Sources.

Итак, давайте рассмотрим некоторые ключевые особенности Blender:

- ◆ программа абсолютно бесплатная и распространяется по лицензии GNU GPL. Вкратце, пользователь Blender имеет право использовать, распространять или копировать ее. Весь контент, созданный в Blender, является собственностью создателя и может распространяться уже по другой лицензии, в том числе коммерческой (полный текст лицензии GPL вы можете найти на прилагаемом диске в директории с инсталлятором программы);
- ◆ если вы программист, то ничто не мешает скачать исходники программы и изменить их по своему желанию, а затем распространять полученную модификацию (см. лицензию);

- ◆ программа кроссплатформенная. Имеется поддержка Windows, Linux, Mac OS;
- ◆ отзывчивость и бесплатность каналов поддержки;
- ◆ новые стабильные версии появляются с похвальной быстротой. Так как программа имеет чрезвычайно маленький исполняемый файл для этой категории приложений (10—15 Мбайт), то скачивание ее из Интернета не представляет особых сложностей;
- ◆ наличие встроенного игрового движка позволяет создавать интерактивные приложения реального времени или игры;
- ◆ использование языка программирования Python (Питон) для написания скриптов как для обычного режима программы, так и для игрового.

Эта книга в первую очередь ориентирована на начинающих пользователей, незнакомых или малознакомых с программой. Поэтому некоторые инструменты и возможности Blender пропущены в угоду ясности и простоте изложения.

По мере чтения вы будете знакомиться с особенностями работы в программе Blender и ее интерфейсом. Большинство операций в программе проще выполнять, используя горячие клавиши. *Приложение 1* содержит список наиболее употребляемых комбинаций клавиш. Подробное руководство по установке программы для операционных систем Linux и Windows вы можете найти в *приложении 2*. Большинство глав сопровождается примерами на диске. В определенных случаях для закрепления материала вам может понадобиться рабочий файл проекта. Поэтому ознакомьтесь заранее с описанием компакт-диска, приведенным в *приложении 3*. И под конец введения, а точнее начала книги, несколько полезных ссылок:

- ◆ <http://www.blender.org>. Официальный сайт некоммерческой организации Blender Foundation — разработчика программы Blender;
- ◆ <http://blenderartists.org/forum>. Крупнейший форум пользователей Blender;
- ◆ <http://b3d.mezon.ru>. Документация по программе на русском языке.

А теперь открывайте следующую страницу и окунайтесь в безбрежное море под названием Blender. Счастливого плавания!

Автор (andrej.prakhov@gmail.com)

ГЛАВА 1



Интерфейс программы

Часто Blender упрекают в излишней перегруженности интерфейса и несоответствии его другим популярным пакетам 3D-моделирования. С одной стороны, это верное суждение, особенно для пользователей, которые имеют опыт работы с подобными программами. Впервые запуская Blender, новички теряются в обилии вкладок, кнопок и окон. К сожалению, встроенной справочной системой программа не обладает, и пользователю остается обращаться за советом к "старшим товарищам" или в Интернет. Если с англоязычной документацией проблем, как правило, не бывает, то с русскоязычной литературой весьма и весьма туго. Однако стоит заметить, что подобной "болезнью" страдают почти все приложения Open Source (Открытое программное обеспечение).

Но не так страшен черт, как его малюют! На самом деле программа обладает чрезвычайно продуманной системой интерфейса, и по достижении некоторого уровня общения с ней начинаешь замечать неудобства работы с другими пакетами. Тут и полностью настраиваемый внешний вид, поддержка разных вариантов оформления, свободно перемещаемые и масштабируемые окна и панели.

По прочтении данной главы вы сможете свободно разбираться в терминологии программы и ее базовых функциях. Также рассматриваются основы навигации в трехмерном пространстве, понятия сцен, объектов, групп и связей.

1.1. Система окон и меню

При первом запуске программы Blender вы увидите два окна — основное, для моделирования, и второе, с кнопочными панелями. В действительности, их целых три, просто верхнее скрыто от глаз пользователя и содержит системные настройки. Если вас заинтересовало его содержимое — потяните курсор

ром горизонтальную линию, разделяющую основное меню и рабочую область. Итак, вы встретили первую особенность интерфейса Blender. Дело в том, что каждое окно программы имеет свой заголовок, который по желанию пользователя может находиться в любом месте окна. В данном случае у первого окна он расположен снизу, а у двух остальных, как и положено, — сверху. Теперь попробуйте провести указателем мыши по окнам программы. Обратите внимание, что при наведении курсора заголовок окна под ним меняет свой цвет на более светлый. Таким способом программа предупреждает, что текущее окно является активным и к нему будут применяться любые действия, совершенные пользователем.

Основные элементы стандартного окна программы Blender (рис. 1.1):

1. **Заголовок.** Содержит меню, кнопки и поля, характерные для этого типа окна.

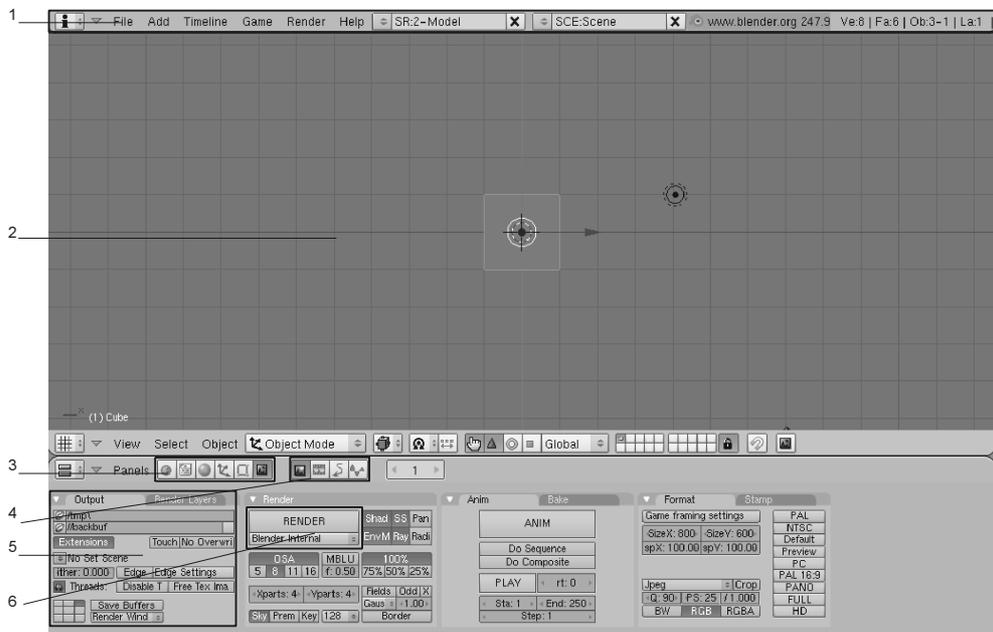


Рис. 1.1. Вид программы при первоначальном запуске

2. **Рабочая область.** Основное содержание окна.
3. **Первичные кнопки.** Глобальные функции, вынесенные для быстрого переключения.
4. **Вторичные кнопки.** Содержание их меняется в зависимости от активной первичной кнопки.

5. **Панели.** Интерфейсный элемент для группировки функций, схожих по назначению.
6. **Кнопки и поля.** Элементы для непосредственного выполнения задачи.

В любой момент вы можете настроить внешний вид программы под свои нужды — добавить новые окна, изменить их размеры и содержание. Все сделанные настройки сохраняются в текущем файле проекта.

Как вы уже поняли, для изменения структуры окон необходимо подвести курсор к границе между зонами и, удерживая нажатой левую кнопку мыши, потянуть разделитель в нужном направлении.

Не сложнее разбить рабочую область программы на нужное количество окон. Для этого существует особое меню, которое вызывается нажатием правой кнопки мыши. Состоит оно из трех пунктов:

- ◆ **Split Area** (Разделить область). При выборе этого пункта появляется разделительная полоса, указывающая место для разрезания окна. Переместите ее в нужное место, и вы получите две новые области;
- ◆ **Join Areas** (Объединить области). Эта функция противоположна только что рассмотренной. Служит для слияния двух смежных окон. При использовании появляется схематичная стрелка, указывающая направление соединения окон. Просто подвигайте мышью для получения желаемого результата;
- ◆ **No Header/Add Header** (Убрать заголовок/Добавить заголовок). Если по каким-то причинам вам необходимо скрыть заголовок, то воспользуйтесь этим пунктом.

Совет

После скрытия заголовка программа всегда восстанавливает его внизу окна. Если это вас не устраивает, заголовок можно вернуть на законное место, воспользовавшись контекстным меню. Вызывается оно щелчком правой кнопки мыши на заголовке и содержит три пункта: **Top** (Верх), **Bottom** (Низ) и **No Header** (Убрать).

Задачи при работе с программой бывают самые разные — моделирование, анимация, построение композиций видео и многое другое. Точно так же разнятся и инструменты для их выполнения. Поэтому очень заманчивой выглядит возможность Blender использовать заранее настроенные виды интерфейса. По умолчанию программа предлагает пять настроек, оптимальных для моделирования, работы с материалами и анимацией, скриптования, обработки конечного результата (рис. 1.2).

Здесь вы также можете сохранить свою собственную настройку, выбрав пункт меню **ADD NEW** (Добавить новый). Для быстрого перемещения между

ними используйте горячие клавиши <Ctrl>+<Left> или <Ctrl>+<Right>. Кроме того, нажатие клавиши <Ctrl> совместно с курсорными клавишами <Up> и <Down> позволяет раскрывать или соответственно сжимать активное окно программы.

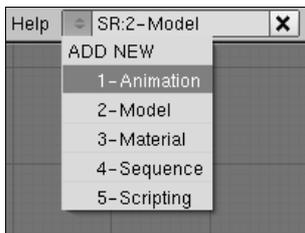


Рис. 1.2. Меню для управления схемами интерфейса

Для удобства разработчики Blender сгруппировали все возможные окна программы в специальное меню. Расположено оно всегда в одном и том же месте: слева на заголовке каждой области. Таким образом, сам пользователь решает, что будет находиться в конкретном окне, будь то кнопочная панель или окно браузера картинок. В этом главное отличие интерфейса программы Blender от других известных пакетов 3D-моделирования, где, как правило, настройки окон сильно ограничены.

Это меню открывает глобальные возможности Blender, поэтому настала пора заинтересоваться его содержимым. Программа предлагает следующие типы окон:

- ◆ **Scripts Window** (Окно скриптов). Расширить функциональность Blender можно, используя дополнительные скрипты на языке программирования Python (Питон). Это окно служит для выбора и запуска всех скриптов, установленных в программе, а также управления ими;
- ◆ **File Browser** (Управление файлами). Просмотр, загрузка, запись файлов рабочего проекта программы;
- ◆ **Image Browser** (Просмотр картинок). Удобный встроенный браузер для просмотра графических файлов, поддерживаемых Blender;
- ◆ **Node Editor** (Редактор узлов). Инструмент для работы с материалами и текстурами объектов в режиме конструктора;
- ◆ **Buttons Window** (Окно управления). Основные функции и возможности программы;
- ◆ **Outliner** (Планировщик). Просмотр имеющихся в сцене объектов, редактирование структуры и связей. Вся информация предоставлена в виде дерева;

- ◆ **User Preferences** (Пользовательские настройки). Окно, содержащее настройки самой программы Blender и главное меню программы;
- ◆ **Text Editor** (Текстовый редактор). Blender позволяет не только использовать уже готовые скрипты на языке Python, но и писать свои собственные. Кроме того, это окно можно использовать для добавления комментариев к своему проекту;
- ◆ **Audio Window** (Управление звуком). Позволяет прослушивать, просматривать треки и подгонять их под анимационное действие;
- ◆ **Timeline** (График времени). Основное окно для управления ключевой анимацией. Запись, редактирование меток, просмотр результата;
- ◆ **Video Sequence Editor** (Редактор видео). Одно из несомненных достоинств Blender — это предоставление пользователю возможности полноценного редактирования видео. Данное окно позволяет управлять отдельными медиафайлами, их длительностью и местоположением, переходами между файлами, видеозффектами;
- ◆ **UV/Image Editor** (Редактор картинок и UV-наложение). Основное окно для управления текстурами, размещением их на объекте и настройками наложения;
- ◆ **NLA Editor** (Редактор нелинейной анимации). Удобное средство для схематичного просмотра анимации и ее редактирования;
- ◆ **Action Editor** (Редактор действий). Еще один способ управления анимацией, а точнее, ее различными последовательностями;
- ◆ **Ipo Curve Editor** (Редактор анимационных кривых). Основной инструмент для точного редактирования анимации. Предоставляет схематичный интерфейс, отображающий анимационные ключи и кривые для управления;
- ◆ **3D View** (3D-окно). Основная рабочая область для просмотра и навигации по сцене, создания и редактирования объектов, текстурирования и анимации.

Как уже было сказано ранее, Blender позволяет свободно распоряжаться видом, масштабом и местоположением своих функциональных панелей. Причем изменять можно как все содержимое окна целиком, так и отдельно взятую панель. Удобнее всего работать с помощью мыши и горячих клавиш (*см. приложение 1*).

Наведите курсор на окно **Buttons Window** и покрутите колесико мыши. При этом рабочая область начнет перемещаться по горизонтали. Если нажать и удерживать дополнительно клавишу <Ctrl>, вы сможете управлять масштабированием содержимого окна. Как видите, все очень просто и эффективно.

Полный список клавиш для управления общим содержимым окна:

- ◆ <WM> (колесико мыши) — горизонтальное перемещение;
- ◆ <Ctrl>+<WM> — масштабирование;
- ◆ <Shift>+<WM> — вертикальное перемещение;
- ◆ <MMB> (средняя кнопка мыши) — свободное перемещение в любых направлениях.

Кроме того, имеется возможность управления размещением панелей. Для этого существует специальное меню **Panel Alingment**, вызываемое щелчком правой кнопки мыши на любом месте панели:

- ◆ **Horizontal** (Горизонтально) — выравнивание всех панелей по горизонтали;
- ◆ **Vertical** (Вертикально) — выравнивание панелей по вертикали;
- ◆ **Free** (Свободно) — отключение любых привязок, панели отпускаются в "свободное плавание".

При выборе последнего пункта меню вы можете размещать панели в любом порядке, как заблагорассудится. Подобный подход к интерфейсу позволяет скомбинировать отображение его элементов наиболее удобным способом. Кроме того, при недостатке места на экране ничто не мешает свернуть любое окно соответствующей кнопкой в его заголовке.

Теперь рассмотрим основы работы с файлами в программе. На первый взгляд управление ими, как и сам интерфейс, выглядит весьма непривычно. Все основные функции находятся, как положено, в меню **File** (Файл) заголовка верхнего окна, но на этом сходство заканчивается. Забудьте о привычных комбинациях клавиш! Так, для создания нового проекта необходимо нажать <Ctrl>+<X>. Чтобы загрузить или сохранить рабочий файл, используйте клавиши <F1> и <F2> соответственно. Для быстрого сохранения пригодится <Ctrl>+<W>. Любые операции, связанные с файлами, требуют подтверждения в виде небольшого меню, которое появляется точно под курсором.

Совет

Во время работы бывает необходимо сохранять разные варианты проектов. Blender поддерживает инкремент имени файла путем добавления к нему очередной порядковой цифры. Для этого используйте клавиши <+> или <->.

Все основные файловые операции, такие как сохранение, загрузка, объединение, импорт и экспорт, совершаются в специальном диалоговом окне **File Browser** (рис. 1.3).

Рассмотрим интерфейс окна **File Browser**:

1. Переход на уровень выше по файловому дереву.
2. Быстрый выбор необходимого диска.
3. Управление размером окна.
4. Сортировка файлов по алфавиту.
5. Сортировка файлов по расширению.
6. Сортировка файлов по времени.
7. Сортировка файлов по размеру.
8. Отображение режима работы окна (запись, сохранение, присоединение и др.).
9. Переключение вида имен файлов.
10. Скрытие системных и невидимых файлов.

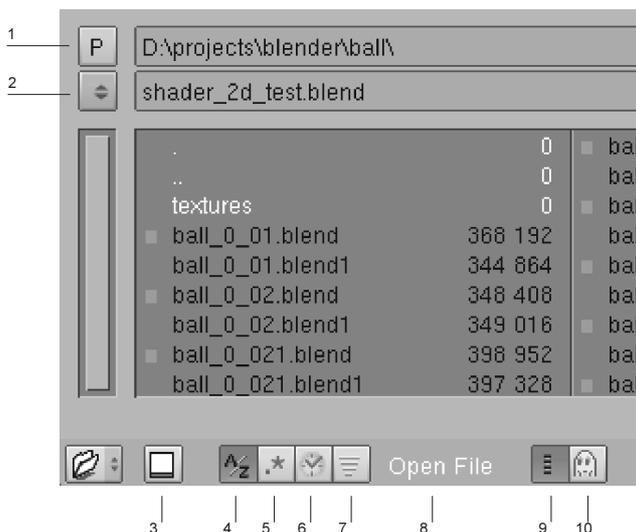


Рис. 1.3. Стандартное окно управления файлами

Внимание!

При закрытии программы подтверждение сохранения изменений в проекте не последует. Не забывайте при необходимости вручную сохранять проект!

1.2. Манипуляции в 3D-пространстве

Изначально концепция программы Blender предполагает, что управление действиями программы удобнее производить с помощью горячих клавиш. Естественно, все функции отражены дополнительно в меню и панелях приложения. Однако добраться до них бывает зачастую сложнее, нежели нажать соответствующую комбинацию на клавиатуре.

Все манипуляции с объектами производятся в окне **3D View**, которое по умолчанию является центральным и занимает большую часть экрана. Следует заметить, что Blender работает с правосторонней координатной системой, когда ось Z расположена вертикально, а оси X и Y смотрят вправо и в глубину соответственно. В зависимости от поставленной задачи вы можете использовать различные режимы отображения сцены, которые отражены в меню **Draw type** (Способ прорисовки), расположенном на заголовке данного окна:

- ◆ **Textured** (С текстурами) — наиболее полный режим прорисовки со всеми материалами, текстурами, шейдерами, тенями. Не рекомендуется использовать при большом количестве объектов сложной структуры;
- ◆ **Shaded** (Затененные) — более простая форма вывода сцены, когда демонстрируются объекты с материалами, освещением и тенями;
- ◆ **Solid** (Сплошной) — по умолчанию является активным. Удобный режим для моделирования, работы с материалами и анимацией. Отличается низким потреблением системных ресурсов в совокупности с хорошим внешним видом;
- ◆ **Wireframe** (Каркас) — все объекты представляются в виде каркасов без материалов и текстур. Этот режим удобно использовать для редактирования модели;
- ◆ **Bounding Box** (Границы объекта) — самый простой режим отображения сцены, когда объекты заменяются параллелепипедами. Это обеспечивает высокую скорость работы программы и низкое потребление ресурсов.

Для эффективной работы с объектами программа Blender предлагает особые режимы демонстрации текущей сцены. В общем-то, они являются стандартными для всех существующих 3D-редакторов. Это отображение сцены с разных сторон, с точки зрения активной камеры и пользовательские установки. Все эти функции можно найти в меню **View** (Просмотр). Однако гораздо удобнее управлять ими с помощью дополнительных цифровых клавиш клавиатуры:

- ◆ <NumPad 0> — камера;
- ◆ <NumPad 1> — передний план;

◆ <NumPad 3> — вид сбоку;

◆ <NumPad 7> — верх.

Оставшиеся цифровые клавиши предназначены для произвольного вращения сцены в соответствующих направлениях. Помимо этого, имеется возможность переключения вида с ортогографической на перспективную проекцию клавишей <NumPad 5>.

Обратите внимание на то, что по отношению к окну **3D View** действуют те же самые правила и способы управления, рассмотренные ранее для панелей. Точно так же можно масштабировать и перемещать рабочую область окна. Единственное исключение в том, что при нажатии средней кнопки мыши вы можете свободно вращать и просматривать сцену в трехмерном виде.

Особый набор функций существует для управления местоположением и масштабом самого объекта в сцене. Причем возможностей и инструментов имеется большое количество.

Вначале следует оговориться, что в Blender существуют две координатные системы — это глобальные и локальные координаты. Обратите внимание на то, что манипуляции можно производить в обеих системах, но по умолчанию активна именно первая.

Примечание

Под *глобальными координатами* объекта подразумевается его местоположение по отношению к самой сцене.

Под *локальными координатами* понимается внутреннее исчисление в самом объекте.

Давайте потренируемся на кубе, который появляется при запуске программы Blender. Если вы до этого уже экспериментировали с программой, то нажмите <Ctrl>+<X> для создания нового проекта. Однако прежде чем выполнять какие-либо действия с объектом, необходимо выделить его среди остальных в сцене. Щелкните правой кнопкой мыши по расположенному в центре окна кубу. Края объекта должны покрыться розовой окантовкой. Нажмите клавишу <G> для включения режима перемещения. Теперь, перемещая манипулятор, вы передвигаете и сам объект. Иногда бывает необходимо произвести перемещение по одной отдельной координате. Для этого нажмите одну из следующих клавиш:

◆ <X> — координата X;

◆ <Y> — координата Y;

◆ <Z> — координата Z.

Обратите внимание на нижнюю область окна, где в момент манипуляций отражается информация о текущих изменениях координат в условных единицах

программы. Для более точного перемещения можно использовать совместно с мышью клавиши <Ctrl> и <Shift>. В первом случае отсчет идет по одной условной единице, другой вариант служит для более филигранной подгонки. Чтобы выйти из режима перемещения без сохранения изменений, нажмите клавишу <Esc> или правую кнопку мыши — для закрепления результата.

Помимо самого перемещения объекта возможна также ротация и изменение его масштаба. Для этого соответственно служат клавиши <R> и <S>. Все сказанное ранее справедливо и для них.

Совет

Быстрое переключение осей можно выполнить, если после активации режимов перемещения, масштабирования или ротации нажать среднюю кнопку мыши.

Однако приведенный способ хорош для задач, когда необходимые действия ограничены одной координатной осью. Для выполнения трансформаций по нескольким осям лучше использовать специальные инструменты-манипуляторы. Необходимая панель расположена на заголовке окна **3D View** (рис. 1.4).



Рис. 1.4. Панель манипуляторов для трансформации объекта

Элементы панели, предназначенные для манипуляции объектом:

1. Включение или отключение панели трансформации.
2. Режим перемещения объекта.
3. Ротация.
4. Масштабирование.
5. Выбор координатной системы для просмотра.

При выборе любого из пунктов объект снабжается стрелками или дугами, охватывающими все возможные направления. Трансформация осуществляется путем захвата и перемещения манипуляторов в нужном направлении. Подобные же действия можно выполнить, если нажать сочетание клавиш <Ctrl>+<Пробел>. В этом случае появляется контекстное меню с соответствующими функциями.

Совет

Ничто не мешает одновременно использовать несколько разных манипуляторов или даже все. Просто выбирайте нужные кнопки на панели, удерживая клавишу <Shift>.

Для более точного манипулирования объектом можно воспользоваться специальной панелью, которая вызывается клавишей <N>. В этом случае имеется возможность введения численных значений в соответствующих окнах.

В завершении этого раздела рассмотрим еще один необычный способ управления объектами, а именно использование "росчерков" мышью. Под этим подразумевается, что при начертании особых фигур в любом месте рабочей области активируются стандартные режимы манипуляции. Попробуйте, например, удерживая нажатой левую кнопку мыши, нарисовать нечто похожее на круг. При отпускании кнопки должен включиться режим ротации. Всего имеется три заготовки фигур, которые может распознать программа. Так, при рисовании прямой включается режим перемещения. Если изобразить нечто похожее на "птичку", можно приступить к масштабированию, а при рисовании окружности — вращению объекта.

Как видите, способов управления объектами в программе предостаточно, и на каком именно останавливаться — решать вам.

1.3. Объекты, группы и связи

Под термином "объект" в программе Blender подразумеваются любые объекты, используемые для построения сцены. Это могут быть примитивы, камеры, свет, вспомогательные элементы и многое другое. По отношению к ним могут применяться трансформации, рассмотренные ранее, составляться группы, выстраиваться иерархии. Наиболее удобный способ добавления нового элемента в сцену — использование меню, вызываемого клавишей <Пробел> и выбором первого пункта **Add** (Добавить). При этом предлагаются следующие группы объектов:

- ◆ **Mesh** (Каркас, меш). Группа примитивов, часто используемая при моделировании. В своем составе содержит: плоскость, куб, круг, сферу, цилиндр, конус, решетку, тор. Другие трехмерные редакторы обычно имеют среди своих примитивов чайник, который является символом 3D-моделирования. В программе Blender его роль играет изображение обезьянки (Monkey);
- ◆ **Curve** (Кривая). Набор примитивов, построенных с помощью кривых. Имеют свои особенности и инструменты для редактирования. Представлены двумя типами: *Bezier* (названа по имени ее создателя Безье) и *NURBS* (Non-Uniform Rational B-spline);
- ◆ **Surface** (Поверхность). Объекты, построенные на основе группы *Curve*, представляют собой более сложные примитивы;
- ◆ **Meta**. Сложносоставные объекты;

- ◆ **Text** (Текст). Особый вид объекта, предназначенный исключительно для создания текста;
- ◆ **Empty** (Пустой). Элемент, являющийся вспомогательным при построении сцены и не проявляющийся при обработке конечного изображения;
- ◆ **Camera** (Камера). Создание дополнительной камеры в текущей сцене;
- ◆ **Lamp** (Лампа). Набор различных по функциональности источников света;
- ◆ **Armature** (Арматура). Специализированные элементы, предназначенные для управления анимацией. При обработке сцены не проявляются;
- ◆ **Lattice** (Решетка). Вспомогательный элемент для редактирования базового объекта.

Прежде чем добавить новый объект в сцену, необходимо указать место для его создания. Для этого существует особый вид курсора, который устанавливается щелчком левой кнопки мыши. В терминологии Blender он называется 3D Cursor. Впоследствии именно он будет являться центром нового объекта (рис. 1.5).

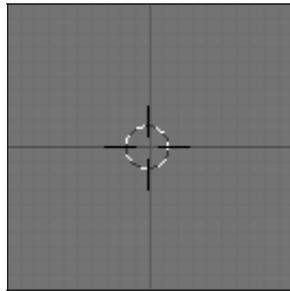


Рис. 1.5. Вспомогательный курсор, предназначенный для позиционирования объекта в сцене

Вот теперь, выбрав подходящее место, вы можете создать нужный объект с помощью меню, описанного ранее.

Иногда в сцене присутствует большое количество объектов, и для упрощения работы с ними используют возможность группирования. Под этим подразумевается создание массива с уникальным именем, с которым впоследствии можно работать как с обычным примитивом. При этом новый элемент добавляется в контекстное меню, о котором писалось ранее, в пункт **Add | Group** (Добавить | Группу). Главной особенностью группы является возможность редактирования ее элементов в первоначальном виде. Причем все сделанные изменения будут отражены в клонах данной группы. Точно так же, как и с обычными объектами, можно производить перемещение, масштабирование или ротацию клона. Начальные базовые элементы изменяться не будут. За-

метьте, что один и тот же объект может принадлежать к различным группам. Кроме того, ничто не мешает разместить базовые элементы в другом слое, а работать с клонами в основном (со слоями мы познакомимся в следующем разделе).

Как обычно, управлять группами можно несколькими способами: из специальной панели, меню или горячими клавишами. Рассмотрим все варианты по порядку.

Каждый объект имеет свои глобальные параметры, просмотреть и отредактировать которые вы можете, нажав клавишу <F7> в окне **Buttons Window** (рис. 1.6).

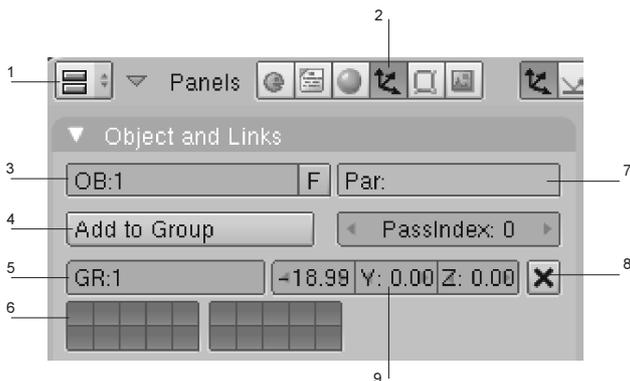


Рис. 1.6. Основная панель объекта

Элементы панели **Object and Links** (Объект и его связи):

1. Значок окна **Buttons Window**.
2. Кнопка, активирующая панель выделенного объекта.
3. Название объекта. Изменить его можно, если щелкнуть левой кнопкой мыши в этой области.
4. При нажатии этой кнопки предлагается выбор: создать новую группу или добавить объект в уже существующую.
5. Название группы.
6. Рабочие слои для этой группы.
7. Название родительского объекта, если он существует (о связях и иерархии — чуть дальше).
8. Кнопка удаления из группы.
9. Глобальные координаты местоположения группы. При изменении их для одного объекта меняются данные для всей группы.

Если вы предпочитаете работать через меню, то необходимые функции расположены на заголовке окна **3D View** — **Object** | **Group** (Объект | Группа):

- ◆ **Add to Existing Group** (Добавить в существующую группу);
- ◆ **Add to New Group** (Добавить в новую группу);
- ◆ **Remove from All Groups** (Удалить из всех групп).

Однако, на мой взгляд, гораздо удобнее использовать контекстное меню, вызываемое клавишами $\langle \text{Ctrl} \rangle + \langle \text{G} \rangle$, в котором предоставлен тот же набор функций.

И последнее, что мы здесь рассмотрим, — это работа с иерархиями объектов. Работая с анимацией, бывает необходимо заставить один объект следовать за другим при любых его манипуляциях. Для этого нужно просто указать, какой объект является главным, а какой ему подчиняется. При этом особенностью такой связи является то, что подчиненными элементами можно манипулировать и это не отразится на родительском объекте. В то же время, любые изменения положения, ротации, масштаба родителя переносятся и на вторичные элементы.

Давайте поэкспериментируем — попробуем создать цепочку из трех объектов, в которой каждый из следующих является потомком для предыдущего. Откройте новый проект Blender и добавьте к кубу два любых объекта (рис. 1.7).

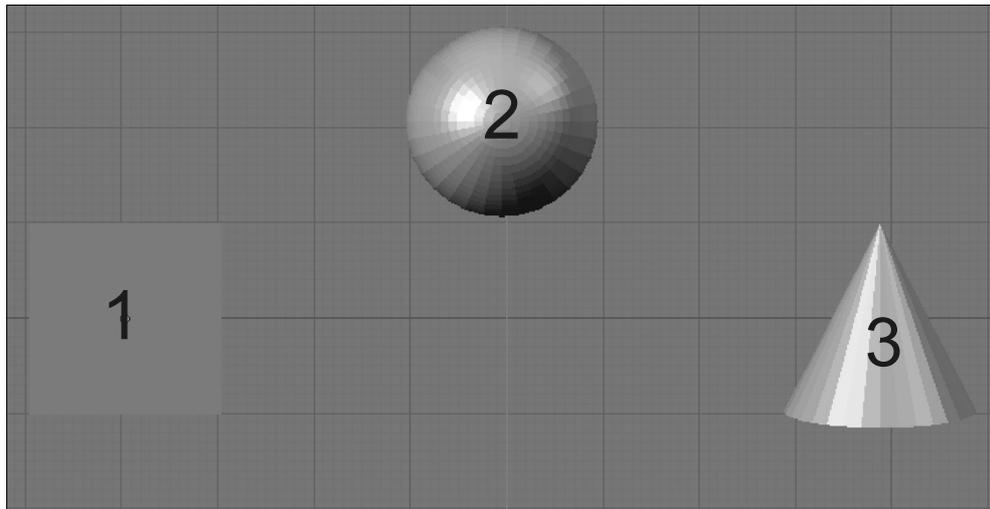


Рис. 1.7. Объекты для примера иерархии

Для создания связи выделите второй объект, который будет являться подчиненным, и, удерживая нажатой клавишу $\langle \text{Shift} \rangle$, добавьте объект номер

один — родителя. Нажмите сочетание клавиш <Ctrl>+<P>. На экране должна появиться пунктирная линия, соединяющая оба объекта. Прделайте то же самое для примитивов 2 и 3 таким образом, чтобы третий элемент стал подчиненным по отношению ко второму. Для контроля правильности полученной иерархии можно воспользоваться окном **Outliner**. Проще всего переключиться на уже имеющуюся схему интерфейса **Animation** в верхнем заголовке главного меню или использовать горячие клавиши <Ctrl>+<Left> (рис. 1.8).

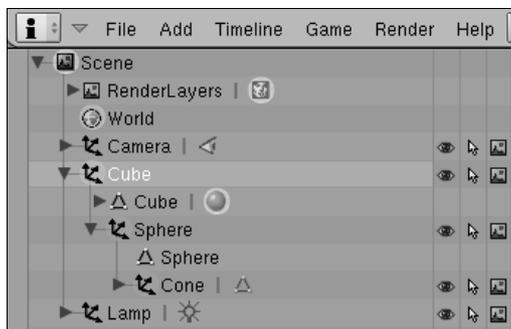


Рис. 1.8. Окно Outliner

Это окно предназначено для вывода информации о иерархии текущей сцены. Здесь вы можете переключаться с объекта на объект, удалять или скрывать их, просматривать все имеющиеся привязки и многое другое. В нашем случае отчетливо видно, что конус принадлежит сфере, а она, в свою очередь, кубу. Чтобы прочувствовать поведение иерархии, попробуйте подвигать кубом. В этом случае все объекты должны следовать за кубом. Теперь отмасштабируйте конус и обратите внимание на то, что изменения не затрагивают другие элементы. В то же время, манипуляции со сферой отразятся на конусе, но не подействуют на куб (*см. пример 1_link.blend на прилагаемом диске*).

Для удаления связи необходимо выделить подчиненный объект и нажать сочетание клавиш <Alt>+<P>. На экране появится следующее меню:

- ◆ **Clear Parent** (Удалить связь). Выделенный объект отсоединяется от родителя и перемещается в корень иерархии сцены. При этом восстанавливаются первичные размер, масштаб и положение объекта до установления связи;
- ◆ **Clear and Keep Transformation** (Очистка связи и сохранение трансформаций). В этом случае удаляется только связь с родителем. Сам объект остается неизменным;
- ◆ **Clear Parent Inverse** (Очистка действия родителя). Позволяет отменить последние манипуляции, связанные с родительским объектом.

1.4. Концепция сцен и слоев

При создании больших проектов, когда количество объектов и анимационных последовательностей исчисляется сотнями, острее становится вопрос о правильном структурировании всей этой массы на сцене. Несложно догадаться, какая головная боль поджидает пользователя, вздумай он проигнорировать нарастающий ком проблем. Все это грозит не только неразберихой в собственном проекте, но и чрезмерным поглощением системных ресурсов и, следовательно, резким замедлением темпа работы. Некоторые задачи такого рода можно решить распределением объектов по тематическим группам. И все же это не выход. К счастью, разработчики Blender предоставили в наше распоряжение простые и мощные инструменты для структурного подхода к организации сцены.

Предположим, вы создали большое количество объектов и решили для удобства объединить их в группу. В целом — верное решение, за одним исключением: теперь у вас в сцене будут находиться по меньшей мере два набора идентичных объектов — оригинальные и их копия в виде группы. Расход системных ресурсов, да и просто рабочего пространства в сцене резко увеличивается. Blender предлагает использовать для этого слои. Представим сцену в виде слоеного пирога, где каждый уровень может содержать свои элементы. У пользователя имеется выбор, какие слои будут визуализироваться в 3D-окне, какие участвовать в обработке конечного результата, а какие просто использоваться для хранения промежуточных объектов. В приведенной ранее задаче было бы разумно хранить оригинальные объекты в одном слое, а саму группу в основном рабочем. Для управления слоями имеется специальная панель, расположенная на заголовке окна **3D View** (рис. 1.9).



Рис. 1.9. Панель для управления слоями

Для переключения слоев имеется 20 кнопок на панели, где активный слой выделяется более темным цветом. Давайте попробуем немного поэкспериментировать.

Откройте новый проект и выделите имеющийся в центре окна куб. Теперь нажмите клавишу <M> для вызова плавающего окна и пометьте в нем любой, на ваш выбор, слой. Объект исчез из окна. Обратите внимание на основную панель управления слоями. В той кнопке, которую вы использовали для перемещения куба, должна появиться точка, сигнализирующая, что данный слой не пустой. Теперь активируйте ее щелчком мыши, и ваш объект снова появится на экране. Если вам необходимо отображение сразу нескольких слоев, просто выделяйте нужные с использованием клавиши <Shift>.

Но и эта возможность программы не полностью удовлетворяет поставленной задаче структуризации больших объемов данных. Давайте рассмотрим последний и самый глобальный инструмент Blender в этой области — управление сценами.

Мы уже не раз в данной главе встречались с понятием сцены. Очевидно, что *сцена* — это совокупность объектов, используемых в текущем проекте. Однако сам проект может содержать в своем составе несколько сцен. И это очень удобно использовать, если ваш проект действительно масштабный. Представьте, что вы создаете очередной шедевр мультипликации. Монтажный план проекта предполагает действие в нескольких декорациях. Вот и получается, что каждую декорацию со всеми актерами и анимациями можно разместить в одном проекте, но в разных сценах. В итоге глобальная иерархия проекта выглядит следующим образом: *проект* → *сцены* → *слои* → *группы*.

Для управления сценами предназначена специальная панель, расположенная в окне **User Preferences** (рис. 1.10).

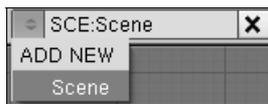


Рис. 1.10. Панель для управления сценами

Здесь вы можете выбрать, переименовать или создать новую сцену. Для последнего действия выберите в меню пункт **ADD NEW**, при этом появится список возможностей:

- ◆ **Empty** (Пустая сцена). Создается новая сцена без каких-либо объектов;
- ◆ **Link Objects** (Ссылка на объекты). Создается ссылка на текущую сцену. Любые изменения в одной сцене отражаются и в другой;
- ◆ **Link ObData** (Ссылка на данные объекта). Равноценно предыдущему пункту за исключением того, что манипуляции с объектами являются индивидуальными для каждой из сцен. Однако любые изменения в геометрии объекта либо его материалах отображаются одинаково для всех копий;
- ◆ **Full Copy** (Полная копия). Создается новая сцена с точным содержимым начальной. Все объекты являются уникальными.

ГЛАВА 2



Работа с *Mesh*-объектами

Погружение в мир трехмерной графики разумнее начинать с изучения основ. Из чего состоит дом? Конечно, из различных строительных блоков. Так и сцена Blender создается из своеобразных кирпичиков. На языке трехмерной графики такие кирпичики называются *примитивами*. Они бывают самые разные — простые, сложносоставные, векторные, важно то, что выбор очень большой и охватывает любые мыслимые задачи. Но начинать лучше все-таки с самого простого, а именно с *Mesh*-объектов.

Работа с *мешами* не отличается большой сложностью и сопровождается солидным количеством инструментов, а также способов моделирования. Здесь простейшее выдавливание соседствует с чувствительной скульптурной лепкой, свободное рисование с низкополигональным редактированием. Думается, каждый найдет для себя подходящий способ моделирования. Итак, приступаем...

2.1. Примитивы и их структура

В предыдущей главе вы познакомились с основным меню для создания объектов в Blender. Интересующие нас **Mesh** (Каркас) находятся самыми первыми в списке. Давайте разберемся со структурой этого типа объектов.

Создайте новый проект и перейдите в окно просмотра камерой (<NumPad 0>). Это необходимо для того, чтобы появилась возможность работы с объектом в трехмерном виде. Выделите находящийся в центре куб и нажмите клавишу <Tab>. Обратите внимание на то, что объект стал полупрозрачным и окрашен нежно-фиолетовым цветом. Это означает, что программа перешла в режим редактирования с полностью выделенной структурой объекта. При ближайшем рассмотрении (увеличьте масштаб колесиком мыши) несложно

заметить, что примитив состоит из вершин, объединяющих ребер и плоскостей (полигонов), расположенных между ними.

Всего программа предлагает восемь примитивов, основанных на Mesh:

- ◆ **Plane (Плоскость)**. Двухмерный объект, который можно использовать для создания зеркал, окон, бумаги. Впрочем, несложно с помощью специальных инструментов программы Blender придать ему третье измерение;
- ◆ **Cube (Куб)**. Базовый объект, используемый при моделировании чаще всего. Состоит из 8 вершин, 12 ребер и 6 плоскостей;
- ◆ **Circle (Круг)**. Двухмерный примитив, обладающий специальным окном, которое появляется при его создании для настройки начальных параметров (рис. 2.1):
 - **Vertices (Вершины)**. От установки этого параметра зависит внешний вид объекта. Так, если вы поставите минимальное значение (а это цифра 3), то получится треугольник. Работать с этим полем можно несколькими способами — использовать стрелки, непосредственно вводить цифры или, нажав левую кнопку мыши, тянуть область в нужном направлении. Такое управление характерно для всех подобных типов панелей;
 - **Radius (Радиус)**. Радиус объекта;
 - **Fill (Заливка)**. Заливка объекта. При нажатом состоянии кнопки **Fill** объект получается сплошным;

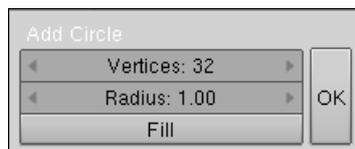


Рис. 2.1. Окно для настройки примитива Circle

- ◆ **UVsphere (Сфера)**. Первая из сфер, предлагаемых программой. Состоит из сегментов, получаемых при пересечении вертикальных и горизонтальных колец. Характерной особенностью этого объекта является изменение размеров отдельных сегментов по мере приближения к полюсам. В своем окне настроек содержит параметры, отвечающие за количество сегментов и колец, а также радиус. При увеличении первых двух параметров можно добиться сглаженности объекта;
- ◆ **IcoSphere (Сфера)**. Характерной особенностью этого типа сферы является то, что она состоит из треугольников одинакового размера. Имеется два параметра: радиус и уровень детализации;

- ◆ **Cylinder** (Цилиндр). Полый или сплошной цилиндр:
 - **Vertices** — количество вершин;
 - **Radius** — радиус объекта;
 - **Depth** — глубина цилиндра, т. е. его высота;
 - **Cap Ends** — наполнение объекта. При нажатом состоянии кнопки **Cap Ends** примитив создается сплошным;
- ◆ **Cone** (Конус) — примитив конической формы, обладающий такими же параметрами, что и цилиндр;
- ◆ **Monkey** (Обезьянка) — специфический объект, являющийся символом программы Blender по аналогии со знаменитым чайником в 3ds Max.

2.2. Основы редактирования

Вот вы и добрались до непосредственной работы с объектами, их формой. Программа Blender обладает богатой палитрой возможностей и инструментов для редактирования примитивов. Используя их, можно добиться практически любого результата, в корне изменив начальный вид объекта. Любые манипуляции со структурой примитива начинаются с выделения необходимого элемента в режиме редактирования. С ним вы уже сталкивались в предыдущем разделе, когда рассматривали Mesh-объекты изнутри. Чтобы переключиться в этот режим, необходимо выделить нужный объект и нажать клавишу <Tab>. Кроме того, имеется специальное меню, позволяющее выбрать этот и многие другие режимы. Расположено оно на заголовке окна **3D View** (рис. 2.2).

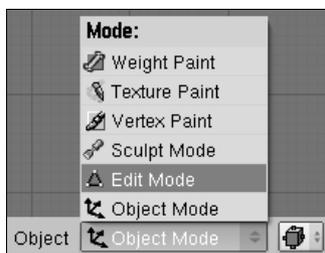


Рис. 2.2. Меню выбора режима работы программы

Меню **Mode** состоит из следующих пунктов:

- ◆ **Weight Paint** (Окраска вершин по весу). Любая вершина объекта может принадлежать к одной из групп, которые используются для облегчения создания анимации. Вес вершины необходим для точного регулирования поведения ее при анимационном действии;

- ◆ **Texture Paint** (Окраска текстуры). При включении этого режима появляются дополнительные инструменты в окне **Buttons Window**. С помощью их вы можете отредактировать текстуру, используя специальные кисти. К примеру, можно на текстуре кирпичной стены изобразить граффити;
- ◆ **Vertex Paint** (Окраска вершин). Любой объект имеет свой базовый цвет, который устанавливается в панели **Material** окна **Buttons Window**. Изначально он однородный, но этот режим позволяет произвольно изменять окраску отдельных вершин. Таким образом, у художника появляется возможность создания разноцветных областей на одном объекте без привлечения дополнительных текстур;
- ◆ **Sculpt Mode** (Режим скульптуры). Мощный способ моделирования высокополигональных объектов, основанный на "лепке" примитива, наподобие работы скульптура. Отсюда и название;
- ◆ **Edit Mode** (Режим редактирования). Особый режим работы программы с отдельным объектом. Именно в нем выполняется изменение примитива в соответствии с задумкой пользователя;
- ◆ **Object Mode** (Объектный режим). Основной режим программы, предназначенный для работы со сценой и ее объектами.

Откройте новый проект и переключитесь в окно просмотра камерой. Выделите куб, затем нажмите <Tab> для начала редактирования. В этом режиме вам подвластно буквально все. Вы можете добавлять или удалять элементы объекта, изменять их масштаб или расположение. Однако до любых манипуляций необходимо выделить нужный элемент в структуре примитива. Для этого существует специальная панель, расположенная на заголовке окна **3D View**, которая позволяет выбрать тип элемента из возможных значений, а также установить особый режим выделения только видимых частей (рис. 2.3).

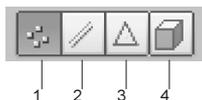


Рис. 2.3. Режимы выделения

Кнопки панели выделения:

1. Вершины.
2. Ребра.
3. Полигоны.
4. Режим прозрачности.

Снимите выделение с объекта, нажав клавишу <A>. Запомните ее, т. к. в дальнейшем она вам не раз пригодится. Эта клавиша может служить и для

обратного выделения всего объекта. При ее использовании в основном режиме программы будут выделены все элементы в сцене. Теперь попробуйте отметить любую из вершин куба, щелкнув по ней правой кнопкой мыши. Если вам удалось попасть по ней, то она окрасится белым цветом, плавно переходящим в серый по соседним ребрам. Кроме того, вокруг нее образуется круг с характерными стрелками-манипуляторами. Теперь вы можете перемещать ее в нужном направлении. Естественно, можно выделять сразу несколько элементов с помощью клавиши <Shift>. В этом случае манипуляторы располагаются в центре, образуемом пересечением прямых из этих вершин. Обратите внимание на то, что с одной активной вершиной вам доступно только ее перемещение, но при выделении нескольких вы можете масштабировать или использовать ротацию.

Теперь поработаем с полигонами объекта. Для этого снимите все сделанные выделения и нажмите на панели выделения кнопку с рисунком треугольника (см. рис. 2.3). Щелкните правой кнопкой мыши ближайшую к вам плоскость, и она примет вид решетки серого цвета. Запомните, что отмеченные полигоны обозначаются программой светло-фиолетовым цветом, а последняя выделенная область показывается в виде серой решетки (рис. 2.4).

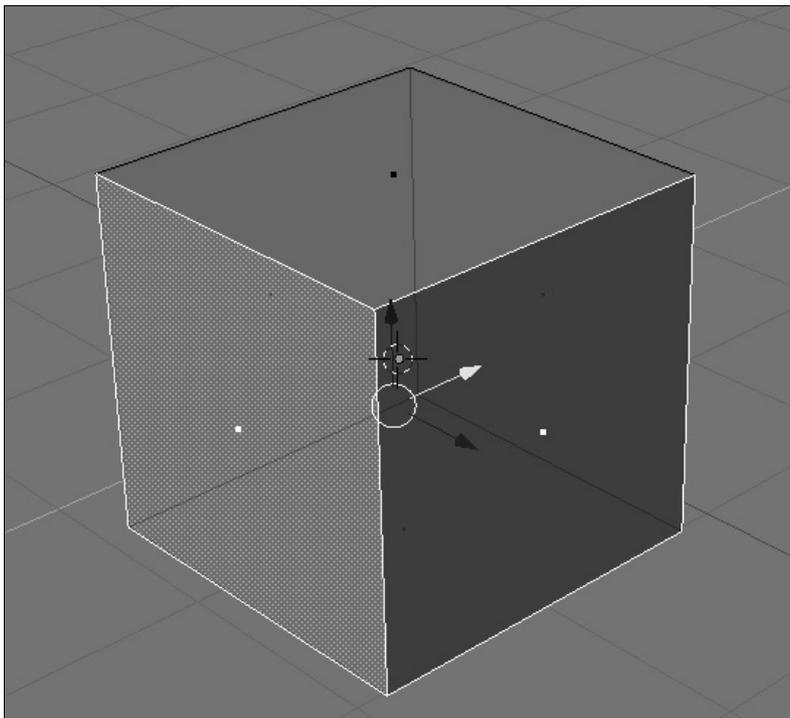


Рис. 2.4. Куб с выделенными плоскостями

Наверное, вы уже столкнулись с тем, что не всегда удастся выделить нужную плоскость. Есть несколько способов решения этой проблемы. Самое простое — развернуть сцену наиболее удобным ракурсом. При этом обратите внимание на точку, являющуюся центром любой из плоскостей. Так вот, щелкнув именно по ней, вы гарантированно выделите нужную область. Дополнительно можно заставить Blender не отображать задние элементы объекта, которые часто попадают в фокус выделения. Для этого на панели выделения имеется особая кнопка с рисунком куба (см. рис. 2.3).

Совет

Иногда бывает необходимо выделить массив элементов, расположенных близко друг от друга. Конечно, можно их щелкать поодиночке, но уж больно это утомительно. Воспользуйтесь горячей клавишей <V> для вызова рамки выделения.

Давайте в качестве разминки создадим простой объект, к примеру куриное яйцо. Удобнее всего в качестве начального примитива использовать UV Sphere. Выберите свободное место в окне **3D View** для эксперимента и создайте нужный объект (<Пробел> | **Add** | **Mesh** | **UVsphere**). Все параметры сферы оставьте по умолчанию.

Яйцо имеет коническую форму с тупым основанием и немного заостренным концом. По логике, для построения его модели необходимо, последовательно выделяя горизонтальные кольца примитива, сужать их по координате Z. Однако если вы все же попытаетесь это сделать, то в лучшем случае получите сильно искореженную модель. Да и времени это займет немало. К счастью, Blender обладает возможностью *пропорционального редактирования* (proportional edit).

Переключитесь в режим просмотра **Front** (Спереди) горячей клавишей <NumPad 1>. Не забудьте, что все манипуляции со структурой объекта производятся в режиме редактирования (<Tab>)! Снимите имеющееся выделение клавишей <A> и, нажав клавишу , отметьте несколько цилиндров в нижней части объекта (рис. 2.5).

Для включения режима пропорционального редактирования нажмите клавишу <O>. Затем немного сместите вниз по координате Z выделенную область (<G> и <Z>). Обратите внимание на появившуюся рамку в виде круга. С помощью колесика мыши вы можете регулировать чувствительность привязки оставшихся полигонов к выделенной области (рис. 2.6). Именно таким образом можно добиться необходимого плавного изменения формы объекта. Для отключения этого режима просто нажмите клавишу <O> еще раз.

Задача, которую мы только что решили, слишком простая и потребовала минимум сил для выполнения. Попробуем создать более сложный объект и за-

одно познакомимся с другими возможностями Blender. Посмотрите на рис. 2.7, где изображен прототип будущей модели. Это выпуклая пятиконечная звезда с двумя центральными точками, к которым сходятся ребра лучей.

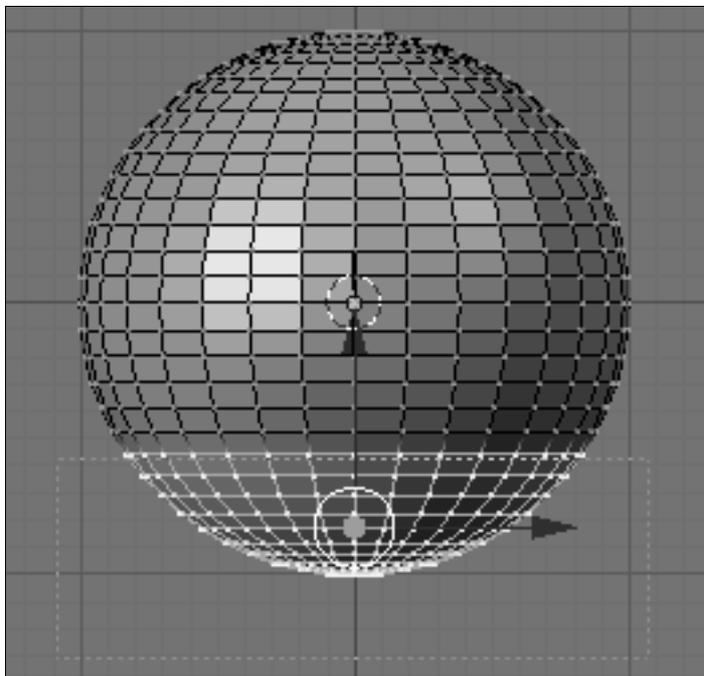


Рис. 2.5. Предварительное выделение полигонов сферы

Как вы думаете, какой примитив лучше всего использовать для создания звезды? Если вы ответили, что куб, то советую запастись большим количеством кофе, т. к. выполнить это будет весьма сложно. Наиболее подходящим начальным примитивом может послужить объект `Circle` (Круг). Его достоинством является возможность задания начального количества вершин. И не смущайтесь, если вспомнили, что `Circle` — объект двухмерный. Проблема решается очень просто, в скором времени вы сами в этом убедитесь.

Создайте новый проект Blender. Так как стандартный куб нам без надобности, его нужно удалить. Для этого в программе служит клавиша `<X>`. Просто выделите объект и нажмите ее. Теперь добавьте в сцену примитив `Circle` с количеством вершин, равным 10, и включенной кнопкой **Fill** (Заливка). Почему десять? Дело в том, что "лишние" пять вершин нам понадобятся для придания необходимой формы объекта. Перейдите в режим редактирования и выделите пять вершин, пропуская каждую вторую. Таким образом, мы отметили точки для создания впадин (рис. 2.8).

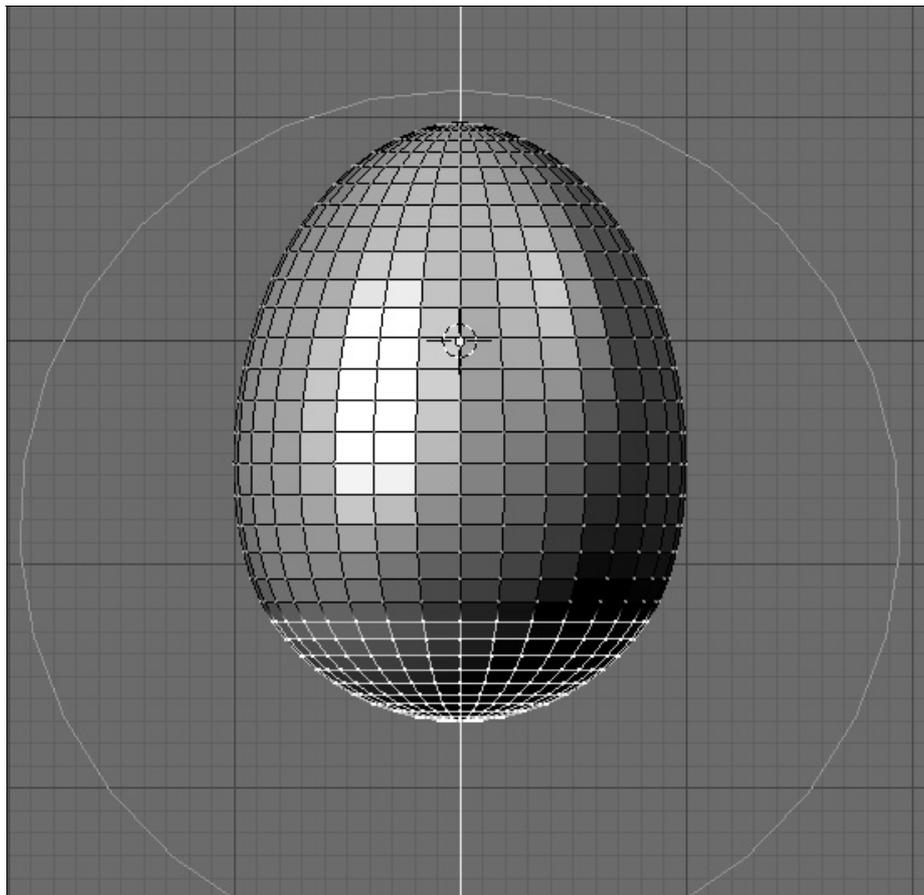


Рис. 2.6. Результат использования пропорционального редактирования

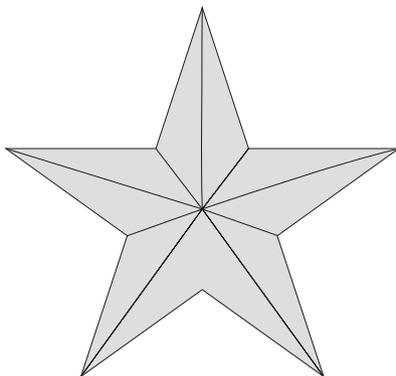


Рис. 2.7. Предварительный эскиз модели звезды

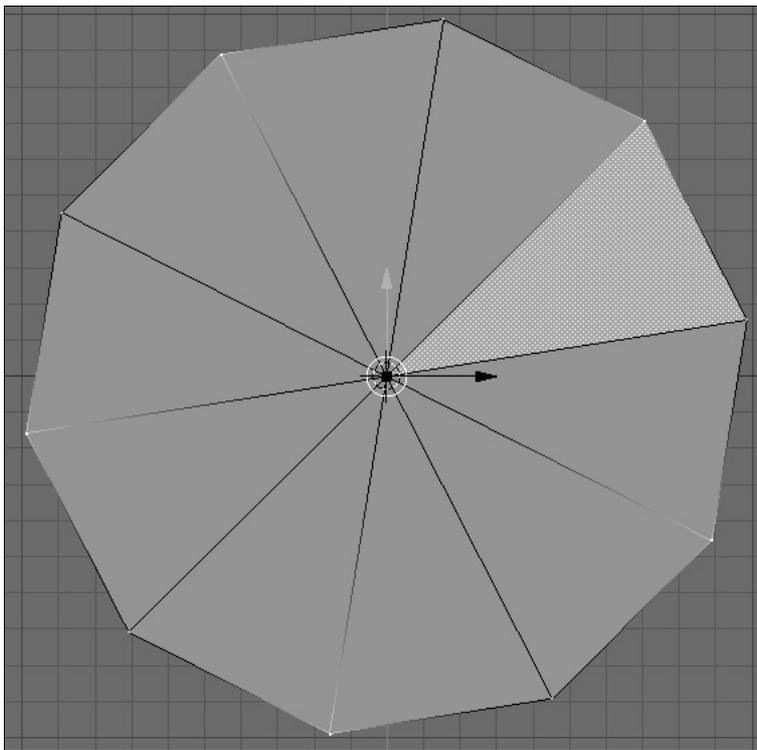


Рис. 2.8. Объект с выделенными вершинами

Теперь наша задача — сместить выделенные вершины ближе к центру для создания вида звезды. Проще всего это сделать, используя масштабирование, а не перемещение. Нажмите клавишу **<S>** и тяните курсор мыши по направлению к центру объекта, пока не образуется необходимая фигура (рис. 2.9).

Для точного создания модели, как в приведенном рисунке, используйте масштабирование с нажатой клавишей **<Ctrl>**. В этом случае смещение будет происходить строго дозированными порциями. Все необходимые данные отображаются в заголовке окна **3D View** в стандартных единицах программы (смещение до значения 0.4000).

В итоге звезда получилась, как на рисунке прототипа. Осталось лишь придать ей объем. Для этого в редакторе предусмотрена функция **Extrude** (Выдавливание), которая доступна как в окне **Buttons Window** (рис. 2.10), так и по горячей клавише **<E>**.

Работать с функцией **Extrude** возможно только в режиме редактирования. Перед этим необходимо выделить нужные элементы структуры объекта. После ее активации появится меню со следующим содержанием:

- ◆ **Region** (Выделенное). Выдавливание происходит для всех выделенных элементов;
- ◆ **Individual Faces** (Конкретные полигоны). Только для активных плоскостей;
- ◆ **Only Edges** (Только ребра). Работа с ребрами;
- ◆ **Only Vertices** (Только вершины). Работа с вершинами.

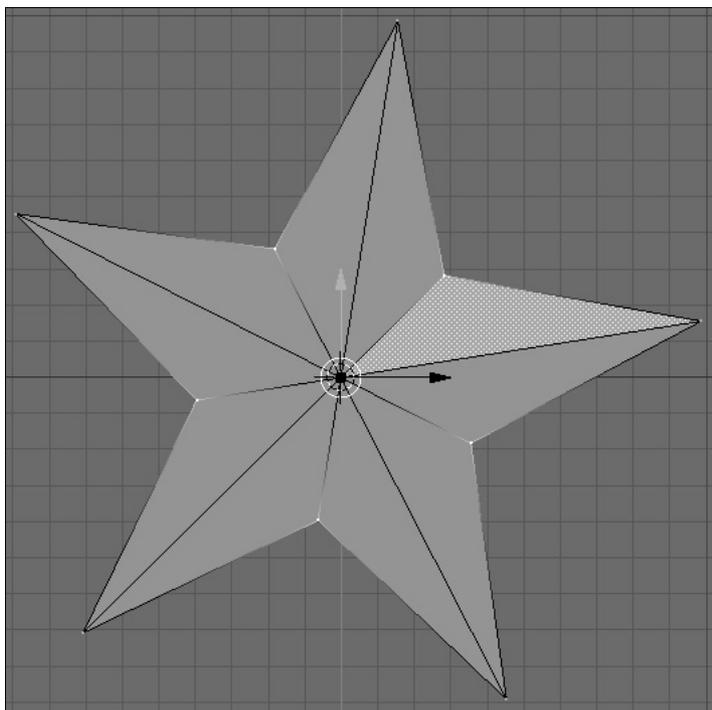


Рис. 2.9. Звезда, но пока двумерная

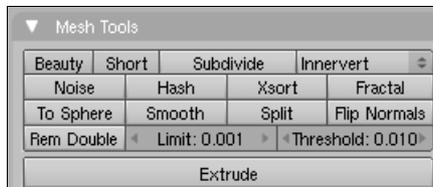


Рис. 2.10. Панель **Mesh Tools** окна **Buttons Window**

В нашем случае необходимо выбрать первый пункт меню — **Region**. Однако перед этим, для удобства просмотра, перейдите в режим просмотра **Front**

(Спереди) клавишей <NumPad 1>. Итак, выделите весь объект клавишей <A>, нажмите клавишу <E> и выберите **Region**. Затем тяните курсор вверх до получения результата, как на рис. 2.11.

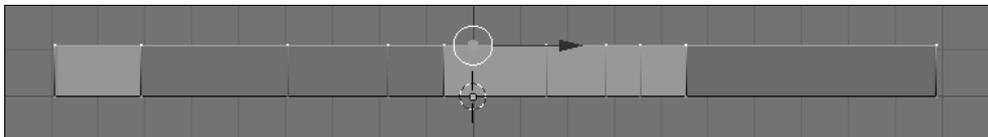


Рис. 2.11. Примитив Circle после применения операции выдавливания

Совет

Как и большинство программ, Blender поддерживает систему отката до первоначального состояния. Если полученный результат по каким-либо причинам вас не устраивает, используйте <Ctrl>+<Z>. Для обратного восстановления служит сочетание клавиш <Shift>+<Ctrl>+<Z>. Кроме того, вы можете просмотреть историю изменений и выбрать необходимый момент с помощью <Alt>+<U>.

Лучи у обычной звезды всегда остроконечные, но не в получившейся модели, и это нужно исправить. Не спешите вручную сдвигать вершины! Есть способ сделать это быстро и элегантно, ведь программа позволяет объединять выделенные элементы в одно целое. Для этого отметьте вершины одного из лучей, развернув сцену на удобный ракурс средней кнопкой мыши (рис. 2.12).

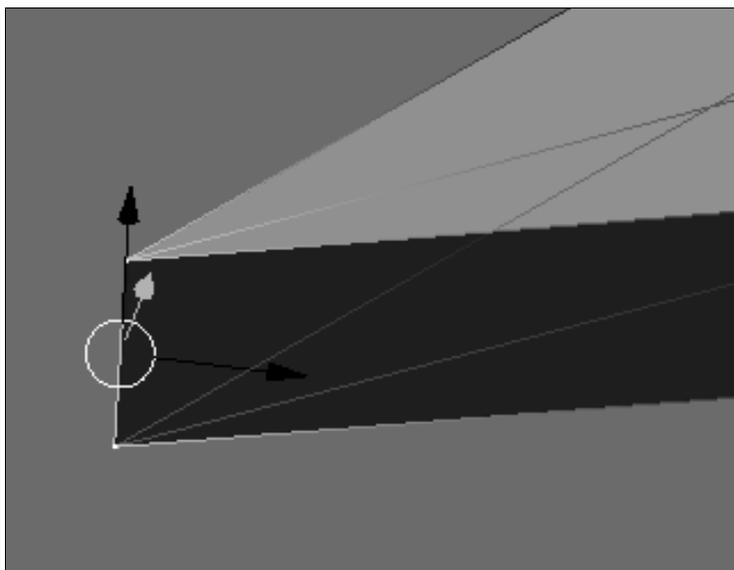


Рис. 2.12. Предварительное выделение вершин для операции объединения

Нажмите клавишу <W> для вызова меню редактирования (те же самые функции вы найдете в панели **Mesh Tools**). Сейчас нас интересует пункт с названием **Merge** (Объединить). После его выбора появляется меню с уточнением места для создания новой вершины:

- ◆ **At First** (По первой). Новая точка будет расположена на месте первой из ранее выделенных;
- ◆ **At Last** (По последней). Расположение на месте последней из вершин;
- ◆ **At Center** (В центре). Высчитывается центр между всеми выделенными элементами, который и будет использован для создания точки;
- ◆ **At Cursor** (По курсору). Координаты вершины выставляются в соответствии с координатами 3D Cursor.

Выберите из списка третий пункт, **At Center**. Программа предупредит об удалении "лишних" точек и выполнит перестроение объекта. Таким же образом обработайте оставшиеся лучи звезды.

Завершающим штрихом для нашей модели станет придание ей выпуклости по центру. Это легко сделать, если выделить центральные точки в режиме просмотра **Front** и, используя масштабирование, немного раздвинуть вширь (рис. 2.13).

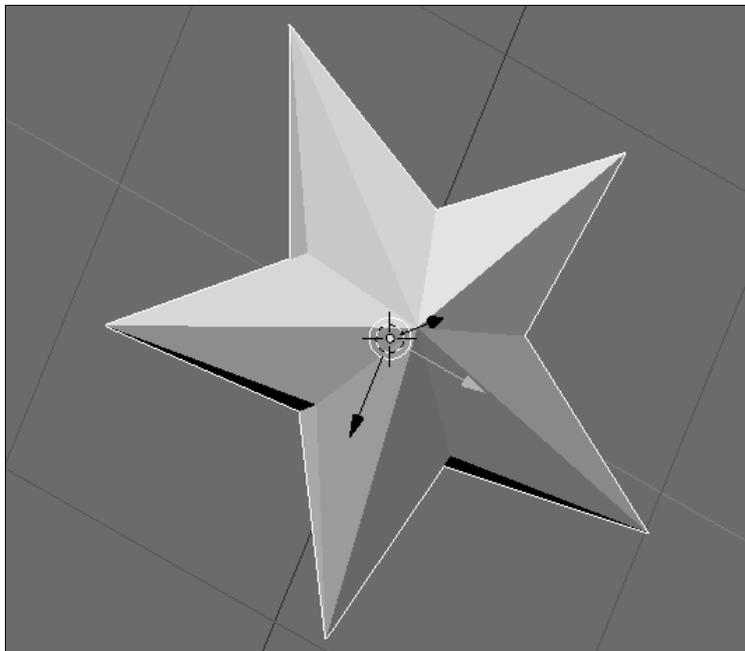


Рис. 2.13. Выпуклая звезда, созданная на основе примитива Circle

Совет

Функция Merge работает только с элементами структуры примитива. Иногда бывает необходимо объединить в один объект несколько примитивов в сцене. Для этого выделите нужные объекты и нажмите сочетание клавиш <Ctrl>+<J>. Естественно, все объекты должны принадлежать к группе Mesh.

2.3. Симметричное моделирование

Если вы внимательно посмотрите на окружающие нас предметы, то заметите, что большинство из них подчиняется законам симметрии. Так, у человека по две руки и ноги, у стола четыре ножки, машина имеет четыре колеса — таких примеров можно найти бесчисленное множество. Почти каждый предмет можно разделить по центру, и при этом выявится определенная симметрия. А раз так, то зачем тратить время на повторение уже сделанного? В этом и заключается принцип *симметричного моделирования*. Сначала создается одна половина требуемой модели, затем производится ее зеркальное отражение и окончательная доработка.

Продолжая практику моделирования, в этом разделе мы создадим модель дивана, следуя принципам симметрии. За основу возьмем примитив Cube.

Обязательным условием для правильного зеркального отражения служит смещение центра объекта в сторону одной его крайней грани. Ведь именно к ней будет примыкать зеркальная копия.

Создайте новый проект (<Ctrl>+<X>) и перейдите в режим редактирования (<Tab>). До начала работы следует переключиться в режим просмотра **Front** (<NumPad 1>). Самым простым способом нужного смещения центра объекта служит разделение примитива на две равные части и удаление одной из половин. В этом случае центр остается на месте, но уже в необходимом нам месте. Для этого выделите весь объект и нажмите клавишу <W>. Появится контекстное меню, в котором нас интересует пункт **Subdivide** (Разбивка). При его выборе структура куба должна получиться, как на рис. 2.14.

Отметьте ненужные вершины рамкой выделения (клавиша) и нажмите клавишу <X> для удаления. В появившемся меню **Erase** (Удаление) выберите пункт **Vertices** (рис. 2.15).

Сейчас мы займемся моделированием основной, весьма приближенной формой дивана с использованием функции *Extrude*. Для удобства редактирования советую отключить режим прозрачности на панели кнопок выделения (см. разд. 2.2). Отметьте оба верхних полигона на объекте и, нажав клавишу <E>, сдвиньте их на три клетки вверх. Для точного перемещения на нужное количество клеток используйте клавишу <Ctrl>. Таким образом, вы получили нужную высоту модели (рис. 2.16).

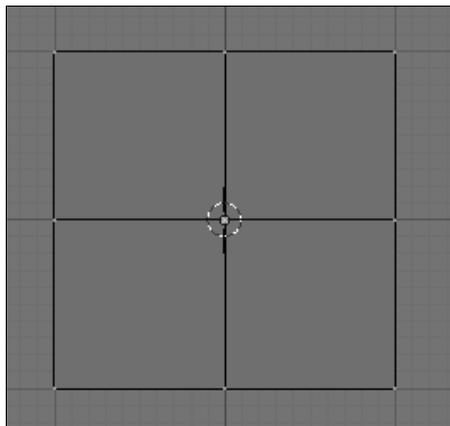


Рис. 2.14. Результат работы функции Subdivide

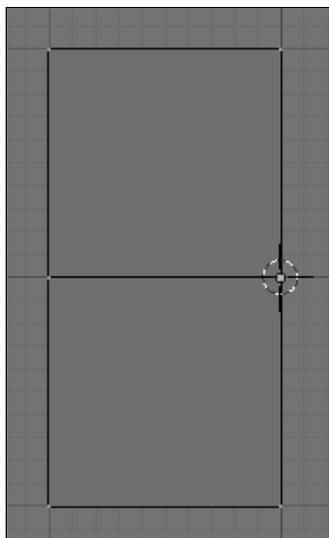


Рис. 2.15. Готовая основа для модели дивана

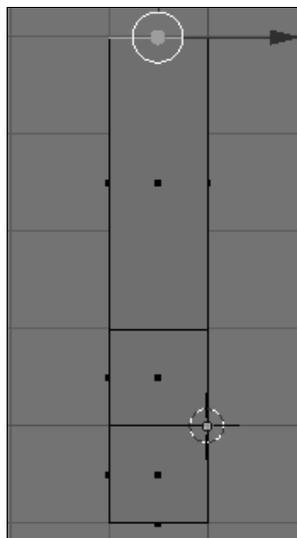


Рис. 2.16. Необходимая высота для спинки дивана

Теперь добавим некоторое количество квадратов по горизонтали. Выделите все полигоны с левой стороны объекта и выдавите их на 3 клетки. У вас должно получиться, как на рис. 2.17.

Для создания сиденья включите режим просмотра **Front** и отметьте центральные нижние полигоны. Перед выполнением самой операции **Extrude** необходимо переключиться в режим просмотра **Side** (Сбоку) клавишей <NumPad 3>. Теперь сместите выделенные полигоны на три клетки по координате Y (рис. 2.18).

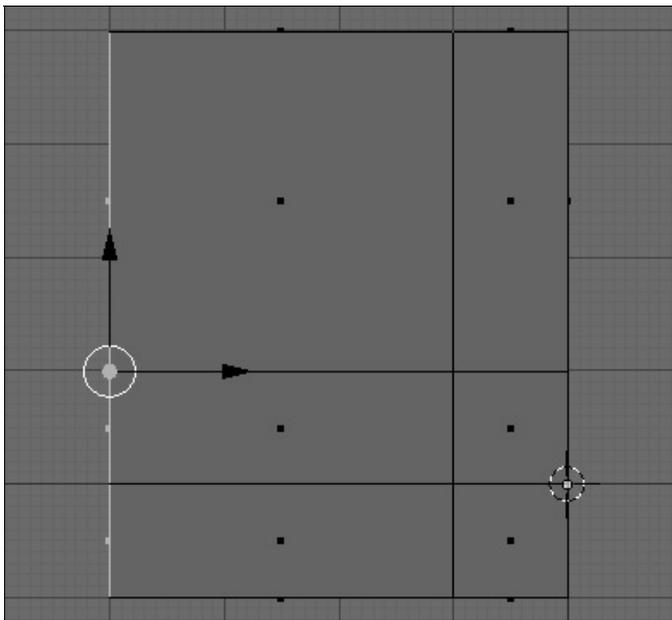


Рис. 2.17. Результат работы функции `Extrude` по горизонтали

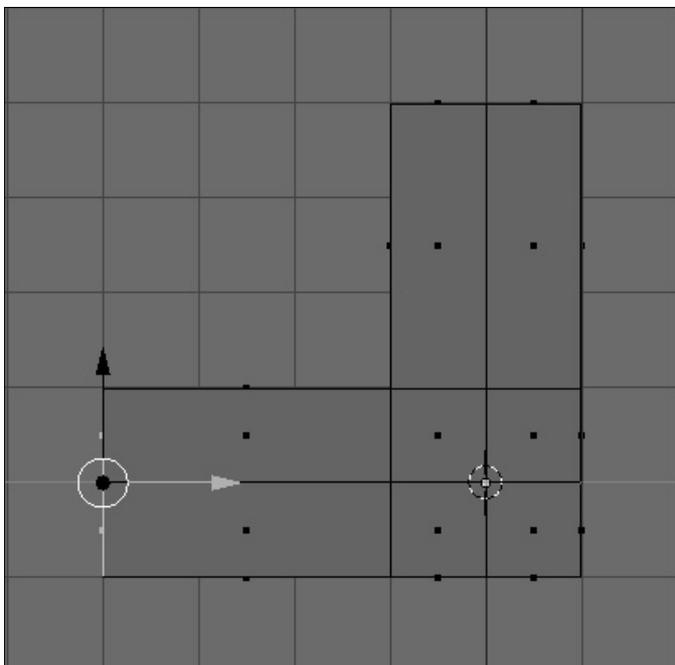


Рис. 2.18. Создание сиденья

Для завершения контура дивана самостоятельно добавьте полигоны подлокотника, как на рис. 2.19. Все его элементы сдвигаются на одну клетку. Используйте технику моделирования, рассмотренную ранее.

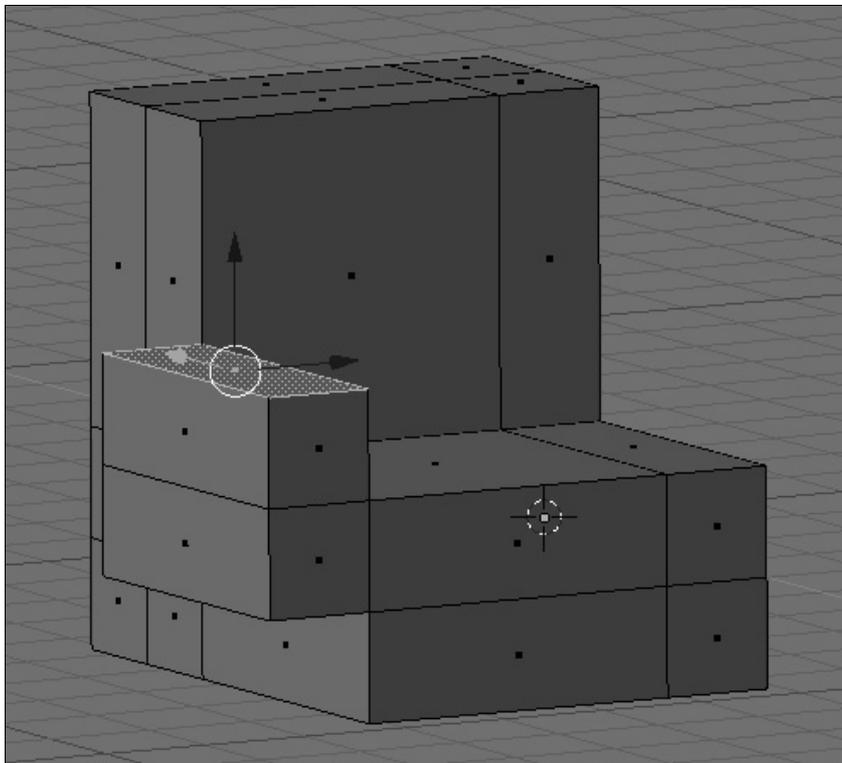


Рис. 2.19. Добавление боковой спинки

Итак, основное мы с вами выполнили. Теперь будем работать над созданием более изящных форм корпуса дивана. Посмотрите на нижнюю часть модели. В реальности такой ровной поверхности у мебели не бывает. Обычно к основанию диван сужается. Попробуем и мы такое сделать. Выделите весь объект (<A>) и произведите двойную разбивку его командой *Subdivide* (<W>). Увеличение количества полигонов позволит точнее выполнить задуманное. Теперь отметьте две нижние линии полигонов (рис. 2.20).

Ваша задача — сместить их по координате Y так, чтобы образовался плавный переход к сиденью модели. Выполнить это можно, используя режим пропорционального редактирования. Включите его клавишей <O>, затем — перемещение клавишей <G> и сдвигайте выделенное по нужной координате (рис. 2.21). Для удобства советуем развернуть сцену средней кнопкой мыши.

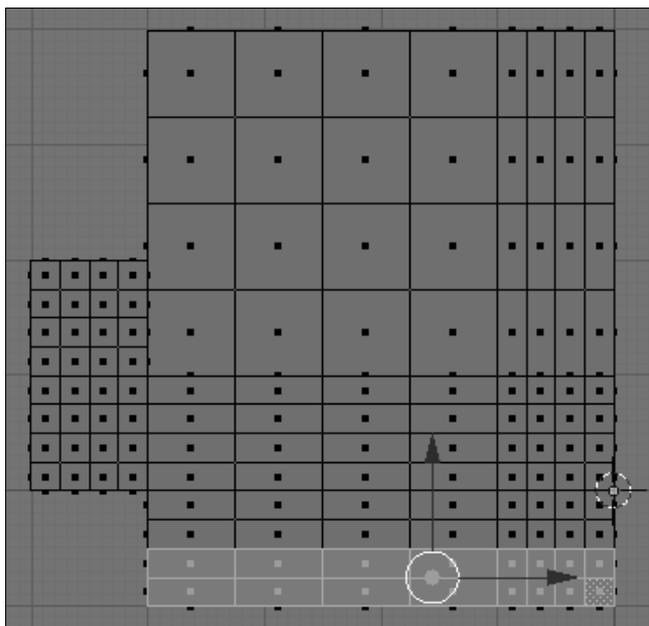


Рис. 2.20. Выделенные полигоны основания дивана

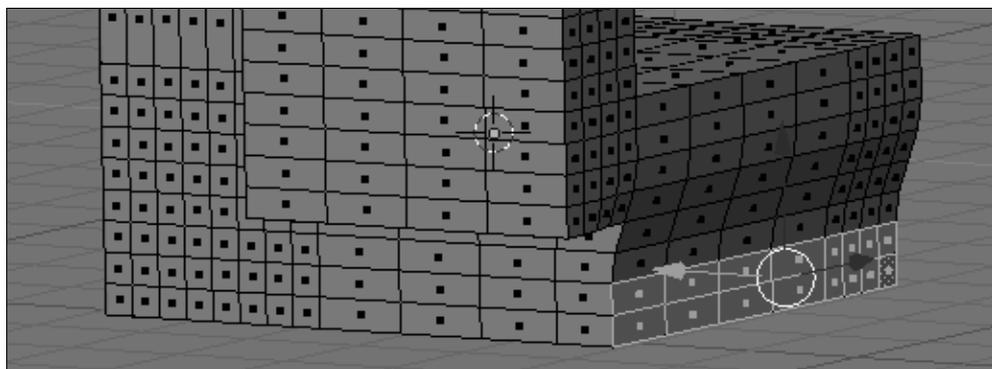


Рис. 2.21. Создание основания дивана

Подобные действия необходимо выполнить для боковой и центральной спинки для придания модели округлости. Принцип все тот же. Выбираете центральные полигоны каждой части дивана и в режиме пропорционального редактирования создаете изгиб. Конечно, у вас может получиться иначе, но в целом картинка должна выглядеть, как на рис. 2.22.

Отдохнем немного от практики и займемся теорией. В программе Blender имеются два способа создания зеркальной копии. Самый простой — восполь-

зваться специальным модификатором `Mirror` (Зеркало). *Модификатор* — это инструмент, позволяющий изменять начальные свойства и структуру текущего объекта. Программа предлагает большое количество модификаторов, расположенных в окне **Buttons Window** на панели **Modifiers**. К любому объекту может быть прикреплено несколько модификаторов. Каждый из них обладает своими параметрами, которые доступны пользователю для редактирования. Так как конечный результат зависит от порядка их расположения, то имеется возможность его изменения. Порядок расположения модификаторов называется *стеком*. Если вас устраивает результат работы модификатора, то его можно удалить, предварительно выполнив операцию применения. В этом случае модификатор исчезает из списка, отразив результат своей работы на структуре объекта.

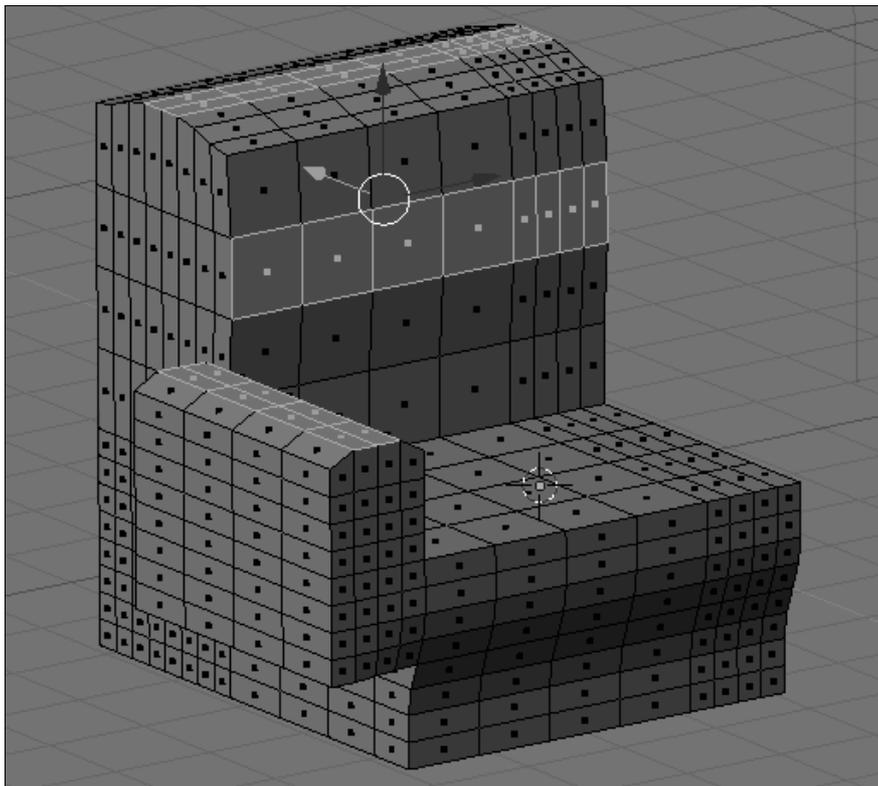


Рис. 2.22. Окончательная версия половины модели

Второй способ создания зеркальной копии — воспользоваться соответствующей функцией из меню **Object**. Однако предварительно необходимо сделать саму копию объекта. Операция копирования в Blender отличается от

других программ. Имеется возможность сделать полную копию объекта или *ссылку* на него. В первом случае вы получите новый объект, являющийся идентичным оригиналу. Для этого необходимо выделить объект, нажать сочетание клавиш <Shift>+<D> и переместить объект в нужное место.

Под "ссылкой" понимается создание копии оригинала с одинаковой внутренней структурой. Редактирование одного затронет и остальные. Однако любые манипуляции, такие как перемещение или масштабирование, для такого объекта являются уникальными. Чтобы создать ссылку, используйте сочетание клавиш <Alt>+<D>.

Какой метод использовать — зависит от требований к модели. В любом случае после создания копии выберите пункт меню **Object | Mirror** и поверните модель по нужной оси. Затем необходимо максимально точно совместить их и произвести слияние командой Join (<Ctrl>+<J>).

Надо заметить, что этот подход требует ювелирной точности на этапе совмещения, да и просто неудобен. Обычно его используют только лишь для точного поворота объекта. Поэтому для доработки нашей модели мы воспользуемся модификатором *Mirror*.

В самом начале раздела уже говорилось, что для зеркального отражения объекта необходимо выполнение некоторых условий. Во-первых, центр объекта должен находиться у крайней грани, которая является стыковочной для копии. Это условие для модели дивана было выполнено. Кроме того, объект должен быть полым, иначе на месте склейки могут появиться неприятные артефакты в виде шва. Во избежание этой ошибки удалите все полигоны с примыкающей стороны рабочего объекта (**Erase | Faces**).

Так как модель у нас уже готова, то пора попробовать модификатор *Mirror* в действии. Выйдите из режима редактирования и нажмите клавишу <F9> для открытия необходимых панелей окна **Buttons Window**. Найдите вкладку **Modifiers**, нажмите кнопку **Add Modifier** и выберите из предложенного списка пункт **Mirror**. Если половина объекта была правильно сделана, то на экране отобразится целая модель дивана. Давайте рассмотрим наш модификатор поближе (рис. 2.23).

Функции панели модификатора, которые являются одинаковыми для всех:

1. Кнопка сворачивания и восстановления окна со свойствами модификатора.
2. Название текущего модификатора. При желании имеется возможность его отредактировать и ввести свое.
3. Включение или отключение модификатора при *рендере* проекта (обработка и вывод сцены в видео- или графический файл).
4. Активация модификатора в объектном режиме программы.

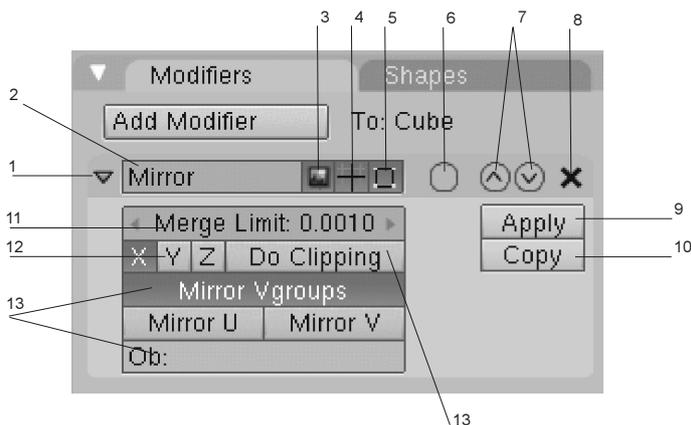


Рис. 2.23. Панель модификатора Mirror

5. Работа модификатора в режиме редактирования.
6. При активации этого поля имеется возможность редактирования области, созданной модификатором так, как будто это настоящий объект. Причем изменения в одной половине отражаются и в другой.
7. Кнопки перемещения модификатора по списку в стеке.
8. Удаление из списка.
9. Применение изменений и закрепление их в структуре объекта. После этой операции модификатор из списка исчезает.
10. Создание идентичной копии активного модификатора.

Функции модификатора Mirror:

11. Расстояние между смежными точками по выбранной оси, при котором произойдет их слияние.
12. Оси для размещения зеркальной копии.
13. Элементы для управления группами вершин.
14. Создание неразрывных швов. Если при слиянии модели появляется разрыв на шве, эта функция принудительно заставит модификатор "закрепить разрыв".

Посмотрите на полученную модель (рис. 2.24). В целом неплохо, но не мешает немного сгладить резкие края (рис. 2.25, а). Для этого существуют различные инструменты сглаживания — от автоматической подборки нужных параметров до специальных модификаторов. Самый простой способ — выделить нужный объект, нажать клавишу <F9> и в открывшейся панели **Link and Materials** (Связи и материалы) нажать кнопку **Set Smooth** (Сгладить).

Картинка при этом станет значительно красивее, однако для нашей модели этого явно недостаточно (рис. 2.25, б). Для отключения сглаживания существует кнопка **Set Solid** (Сплошной), расположенная рядом.

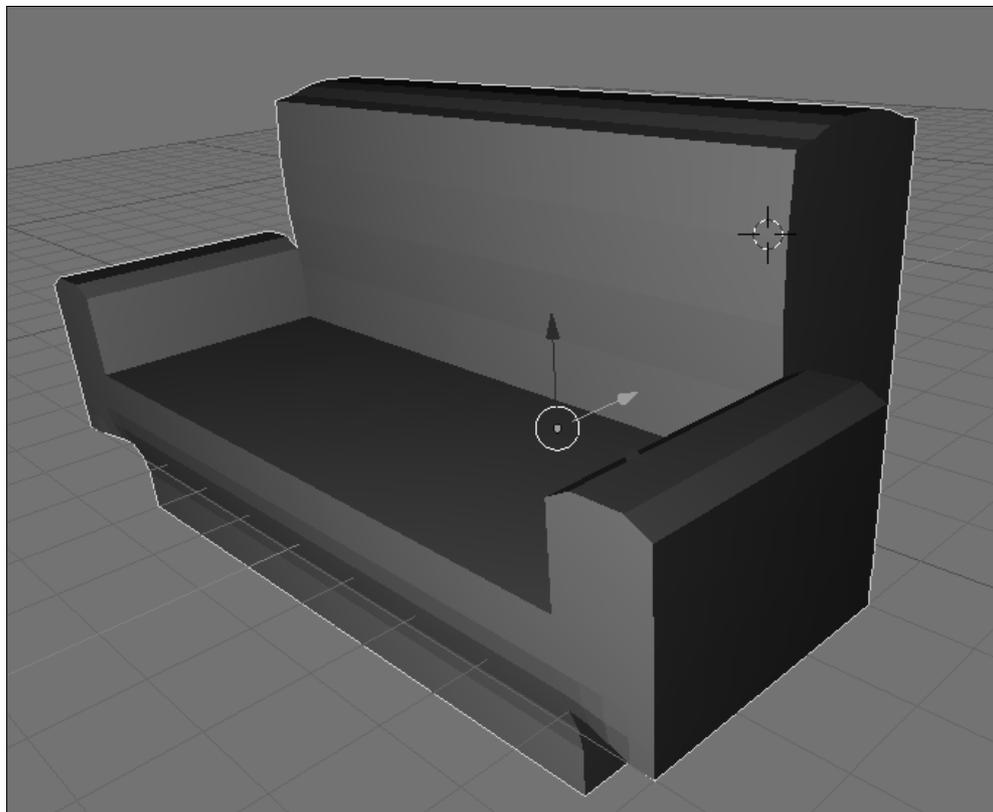


Рис. 2.24. Модель дивана после применения Mirror

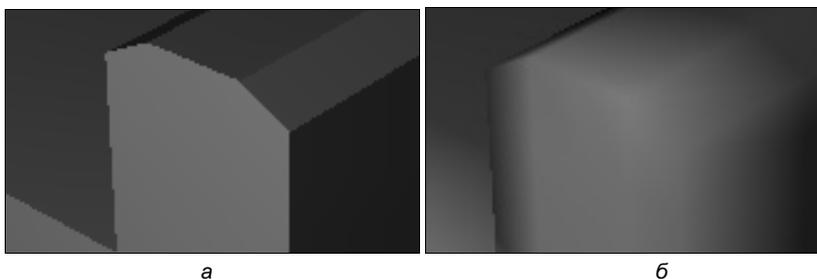


Рис. 2.25. Результат работы сглаживания:
а — до сглаживания; б — после сглаживания

Совет

Функция `Set Smooth` умеет также работать в режиме редактирования. Это можно использовать для придания эффекта сглаживания отдельной группе полигонов.

Если присмотреться к результату работы функции `Set Smooth`, то легко заметить что эффект сглаживания получается за счет игры светотеней и влияния на геометрию объекта не происходит. Нам же необходимо физически придать плавные изгибы каркасу модели. Для этого программа предлагает еще один модификатор меша — `SubSurf` (Сглаживание поверхностей). Его действие основано на разбиении граней объекта "на лету", причем без добавления новых вершин. Это значит, что в режиме редактирования модель предстанет в том же виде, что и до использования `SubSurf`.

Выделите модель в объектном режиме и добавьте в список новый модификатор `SubSurf`. Уровень сглаживания можно указывать в поле **Levels** (Уровни), но не переусердствуйте, иначе это может плохо сказаться на производительности. Для нашей модели оптимальный параметр — 2. Вот теперь можно добавить общее сглаживание (рис. 2.26).

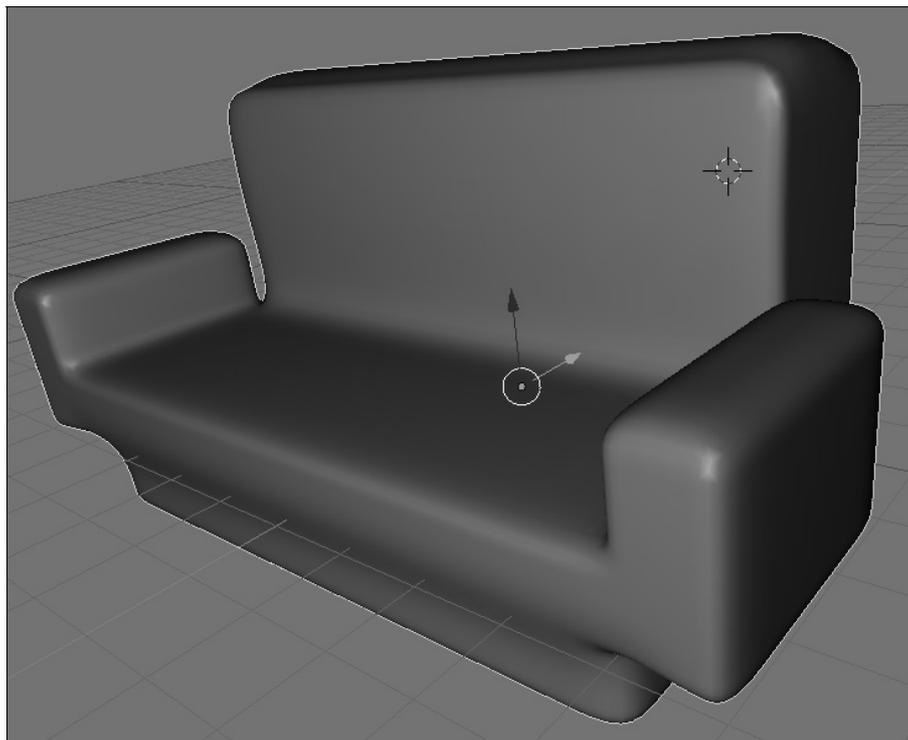


Рис. 2.26. Двойное сглаживание

То, что мы в итоге получили, лишь отчасти напоминает диван. Конечно, если добавить текстуры, карты шероховатости, это прибавит реалистичности, но это темы следующих глав. Поэтому воспользуемся инструментами редактирования для искажения гладкой формы модели.

Присмотритесь к реальному дивану. Наверное, сразу бросятся в глаза натянутости обивки в местах крепления. Попробуем это симитировать. Для этого совсем не нужно изменять геометрию модели. Есть небольшая хитрость, которая влияет на работу *SubSurf*. Обратите внимание на то, что рабочий объект состоит из разного размера квадратных полигонов. Однако Blender умеет работать и с другими полигонами — треугольными. Если в необходимых местах изменить тип полигонов, то при обработке модели модификатор создаст требуемые искажения (рис. 2.27).

Теперь просто нажмите $\langle \text{Ctrl} \rangle + \langle \text{T} \rangle$, и выбранные элементы разобьются диагональными ребрами, тем самым став треугольниками (рис. 2.28).

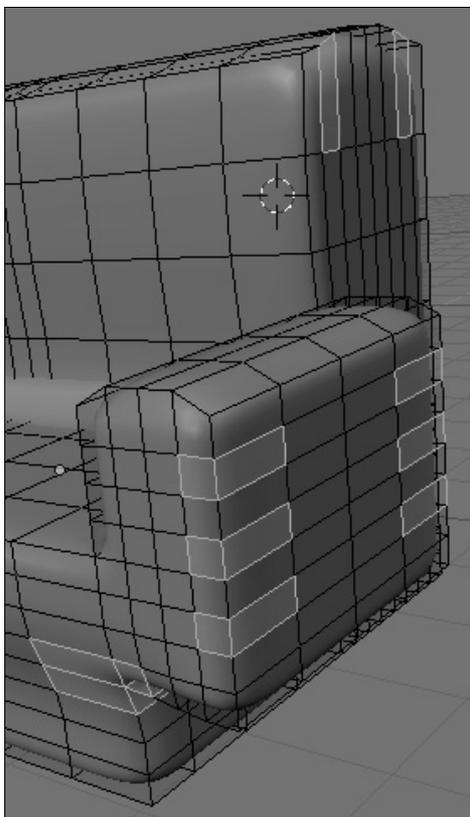


Рис. 2.27. Выделенные элементы для создания искажений геометрии

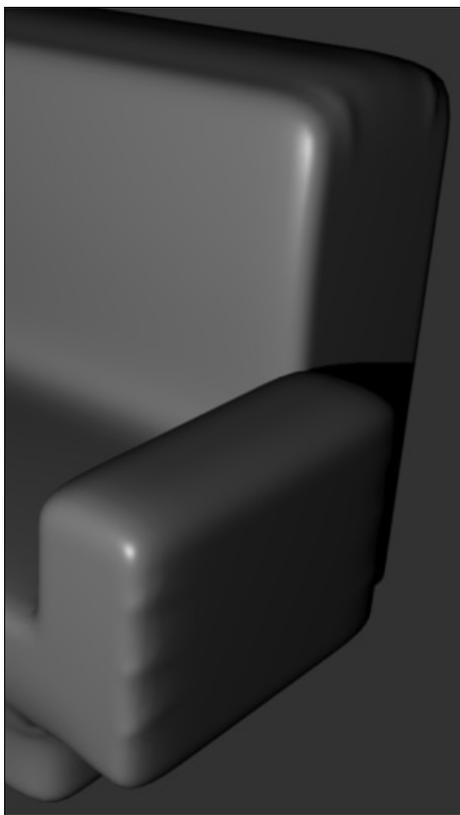


Рис. 2.28. Результат конвертирования выделенных полигонов

Внимание!

Подобный способ редактирования, который приводит к разрушению целостности структуры объекта, на начальной стадии создания модели не рекомендуется. Особенно это касается тех моделей, использование которых предполагается в игровых проектах. Объект должен состоять из однородных элементов: либо квадратных полигонов, как в приведенном примере, либо треугольных. Однако в данном случае, т. к. создание объекта завершено и видимых ненужных искажений не наблюдается, это вполне оправданно.

2.4. Режим скульптуры

Вы когда-нибудь наблюдали за работой скульптора? Мягкими, нежными движениями он касается глины, приминает или разглаживает ее, и под руками мастера возникает очередной шедевр. Нечто подобное предлагает и программа Blender. Естественно, это лишь отчасти напоминает работу скульптора. Может быть, когда-нибудь при появлении более совершенных манипуляторов, чем мышь, этот режим моделирования станет главенствующим, а пока нам придется обходиться стандартными средствами управления. Впрочем, программа Blender умеет работать с графическими планшетами, и если он у вас имеется, то работа упрощается в разы.

Рассматриваемый способ редактирования идеально подходит для создания изящных и пластичных *высокополигональных* моделей. Под этим подразумевается объект с большим количеством полигонов. Однако тут важно не переиграть с объемом элементов, ведь впоследствии это отразится на времени рендеринга конечного результата.

Чтобы лучше разобраться с возможностями скульптурного режима, мы создадим горный ландшафт, основанный на примитиве `Plane`. Забегая вперед, замечу, что есть более удобный путь для реализации задуманного, например воспользоваться картами высот. Но давайте оставим это для следующих глав, а пока создайте новый проект с примитивом `Plane`.

Как уже было сказано, исходный объект уже изначально должен являться высокополигональным. Поэтому воспользуйтесь функцией `Subdivide` для разделения структуры `Plane` на более мелкие элементы. Вполне хватит пятикратного разбиения. Кроме того, включите режим сглаживания **Set Smooth** на вкладке **Link and Materials**.

Для работы с этим видом редактирования существует специальный режим программы — **Sculpt Mode** (рис. 2.29).

Обратите внимание на появившейся курсор в виде окружности (рис. 2.30). Это главный визуальный инструмент, с помощью которого и происходит редактирование объекта.

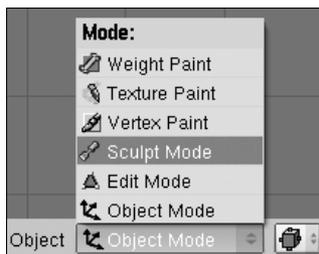


Рис. 2.29. Выбор режима моделирования **Sculpt Mode**

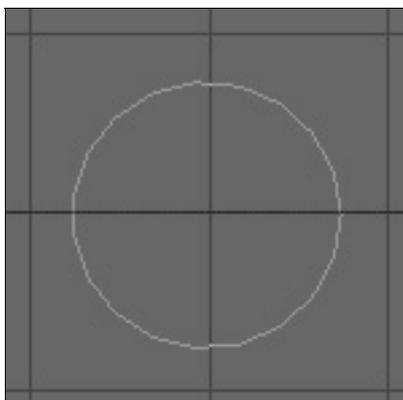


Рис. 2.30. Кисть — основной инструмент **Sculpt Mode**

Скульптурный режим предлагает большое количество опций для регулирования размеров и свойств кистей. Все управляющие элементы сгруппированы на панелях окна **Buttons Window** (<F9>) — это вкладки **Sculpt** (Скульптура), **Brush** (Кисть), **Texture** (Текстура).

Вкладка **Sculpt** содержит основные режимы работы кисти и ее параметры (рис. 2.31).

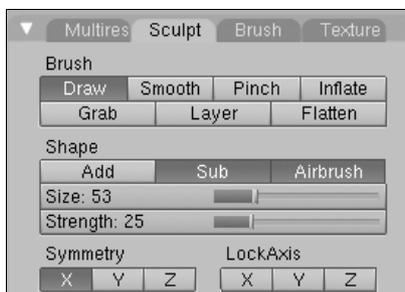


Рис. 2.31. Панель настройки режима работы кисти

Здесь вы можете изменить размер (Size) и настроить силу реагирования кисти (Strength) в соответствующих полях. Пока оставьте все по умолчанию и попробуйте провести курсором по поверхности объекта с нажатой левой кнопкой мыши. Вы увидите, как область под курсором при движении начинает приподниматься. Такой режим работы является основным и носит характерное название **Draw** (Рисование). Для его включения на панели **Sculpt** имеется кнопка с соответствующим именем. Иногда бывает необходимо провести обратную операцию, а именно выдавливание. За это отвечает кнопка **Sub** (Уменьшение), расположенная в области группы **Shape** (Форма). Соответственно кнопка **Add** (Увеличение) служит для переключения в стандартный режим (см. рис. 2.31).

Любые изменения на поверхности объекта с помощью кисти происходят лишь при однократном нажатии левой кнопки мыши и не зависят от длительности ее удержания. В некоторых случаях это просто невыгодно. Помочь с этой проблемой может функция **Air Brush** (Воздушная кисть). Для ее активации включите кнопку с одноименным названием. Теперь при удерживании кнопки мыши будет постоянно производиться работа выбранной кисти (рис. 2.32).

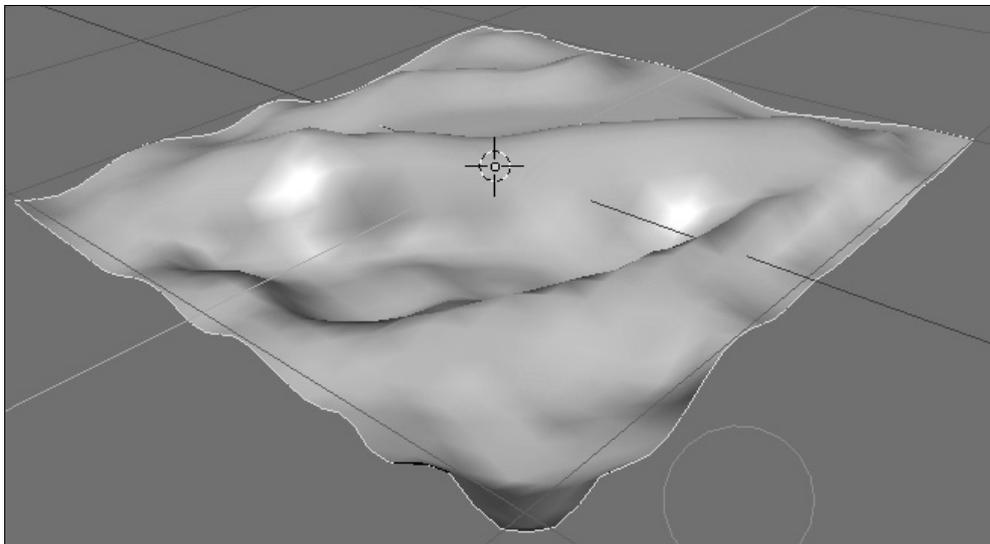


Рис. 2.32. Результат работы функции **Draw**

Предположим, что в определенном месте поверхности вам необходимо сделать остроконечный пик. Конечно, можно перейти в режим обычного редактирования и, используя уже рассмотренный инструментарий, произвести тре-

буемое смещение, а можно воспользоваться специальной функцией режима скульптуры. Для этого служит кнопка **Pinch** (Сжатие). Принцип ее действия — постепенное смещение соседних вершин к центру курсора. Для обратной операции имеется функция *Inflate* (Растяжение). В случае ее применения происходит перемещение вершин от центра по их нормальям.

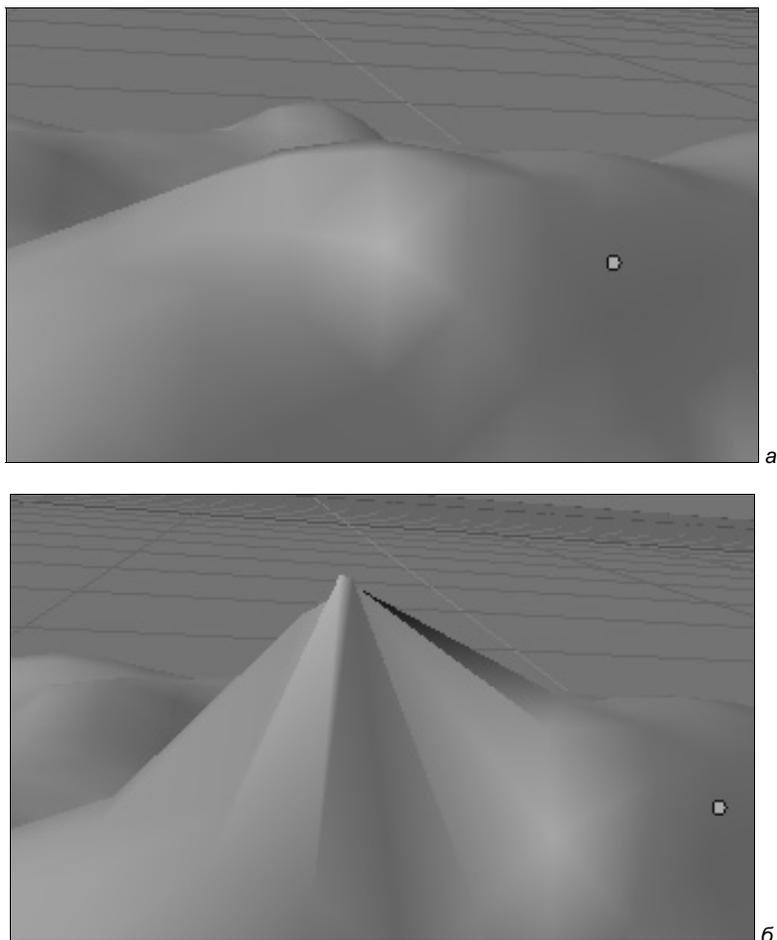


Рис. 2.33. Объект до применения функции Pinch (а) и после (б)

Если вы внимательно посмотрите на рис. 2.33, б, то заметите нежелательные резкие грани по краям получившегося пика. И это притом, что уже включен режим сглаживания! Борьба с подобными артефактами поможет следующая функция — *Smooth* (Сглаживание). Работает она путем постепенного ус-

реднения текущих координат полигонов, попавших в область ее действия. При длительном ее использовании можно добиться абсолютно ровной поверхности (рис. 2.34).

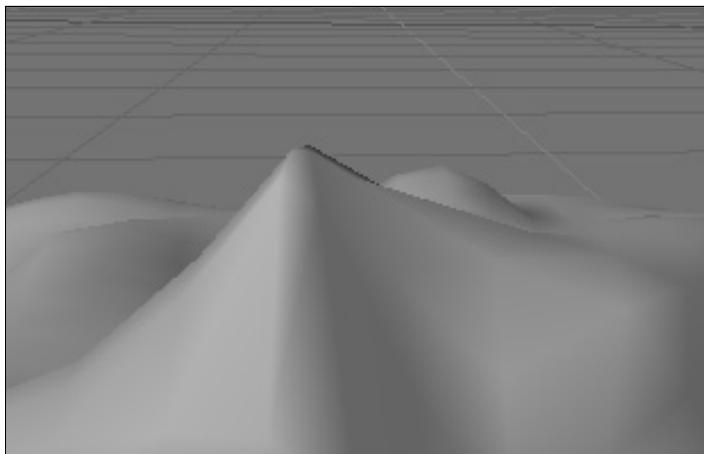


Рис. 2.34. Функция Smooth для режима скульптуры

Режим скульптуры предлагает свой уникальный способ перемещения нужной области на объекте. По сути дела результат следующей функции, *Grab* (Перемещение), идентичен ранее рассмотренному стандартному перемещению с выключенным режимом пропорциональности. Только в этом случае зоной выделения является размер кисти. И этим очень удобно пользоваться — просто передвиньте курсор в нужное место, нажмите кнопку мыши и тяните в нужном направлении.

Забегая вперед, хотелось бы коснуться работы с текстурами в режиме скульптуры. Дело в том, что при изменении геометрии объекта с уже натянутой текстурой можно получить целый ряд артефактов. Как вы думаете, что произойдет с текстурой горы, если ее вытянуть вверх? Правильно. Она сильно растянется, что негативно скажется на конечном результате. Разработчики Blender предусмотрели для этого режима специальные настройки, помогающие избавиться от ненужных искажений. Все они находятся на вкладке **Texture** (Текстура):

- ◆ **Drag** (Тянуть). Перемещение текстуры вслед за движением кисти;
- ◆ **Tile** (Плитка). Размножение текстуры в зависимости от настроек в окне материала объекта;
- ◆ **3D**. Привязка координат текстуры к координатам вершин объекта.

Более подробно о работе с текстурами мы поговорим в *главе 3*.

2.5. Свободное рисование с *Retopo*

Retopo (Remake topology) — это уникальный инструмент, позволяющий из нарисованных геометрических фигур создавать полноценный трехмерный объект. При этом следует учитывать один важный момент при разработке макета: все линии должны пересекаться.

Построение нового объекта начинается с создания пустого Mesh. Это можно сделать двумя способами: либо выделить имеющийся примитив и удалить все его вершины, либо воспользоваться специальным объектом *Empty Mesh* (Пустой каркас).

Создайте новый проект и удалите куб. Теперь добавьте нужный объект из меню **Add | Mesh | Empty mesh**.

Для работы с *Retopo* необходимо перейти в режим редактирования (<Tab>) и на панели **Mesh** нажать кнопку с одноименным названием (рис. 2.35).

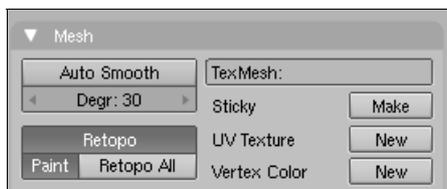


Рис. 2.35. Панель **Mesh** с кнопкой **Retopo**

Нажмите кнопку **Paint** (Рисовать). Это действие откроет специальную панель с инструментами на заголовке окна **3D View** (рис. 2.36).



Рис. 2.36. Панель с инструментами *Retopo*

Инструментальная панель функции *Retopo*:

- ◆ **Pen** (Карандаш). Позволяет свободно рисовать в 3D-окне без каких-либо ограничений. Если нажата вспомогательная кнопка **Hotspot** (Активная точка), то при приближении к краю нарисованного отрезка появится круг. Это дает возможность делать склейки точнее;
- ◆ **Line** (Линия). Рисование прямых линий;
- ◆ **Ellipse** (Эллипс). С помощью этого инструмента можно нарисовать любую закрытую геометрическую фигуру, предварительно настроив желаемое количество вершин в поле **EllDiv**.

Теперь перейдем к практике рисования. Нажмите кнопку **Pen**, если она еще не активна, и попробуйте нарисовать что угодно, удерживая нажатой левую кнопку мыши. Для создания отдельной линии стоит просто отпустить и снова нажать кнопку мыши. Теперь подведите курсор к краю нарисованной линии и дождитесь появления окружности. Следующий отрезок начнется именно отсюда. Как видите, рисовать с помощью **Pen** очень просто.

Попробуем в действии инструмент **Ellipse**. Активируйте его и нарисуйте овал. Теперь поменяйте значение в поле **EllDiv** на 3, а затем на 5. В первом случае появится треугольник, во втором — пятиугольник (рис. 2.37).

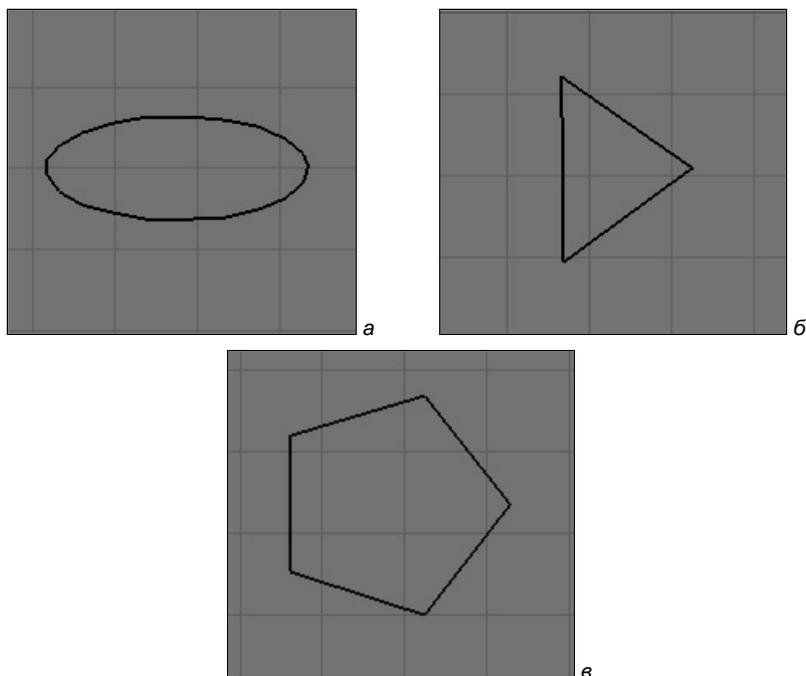


Рис. 2.37. Рисунки, созданные инструментом **Ellipse** со значениями 22 (а), 3 (б), 5 (в)

Обратите внимание на то, что последний созданный рисунок имеет другой цвет в отличие от сделанных ранее. Для удаления такого рисунка нажмите клавишу **<X>** или ****.

На этом обзор инструментов **Retopo** завершен и пора создать нечто осязаемое. Нарисуем, к примеру, букву "Е". Удалите все ранее сделанные заготовки, выберите **Line** и создайте нечто похожее на рис. 2.38.

Для окончательной визуализации объекта осталось только нажать клавишу **<Enter>** (рис. 2.39).

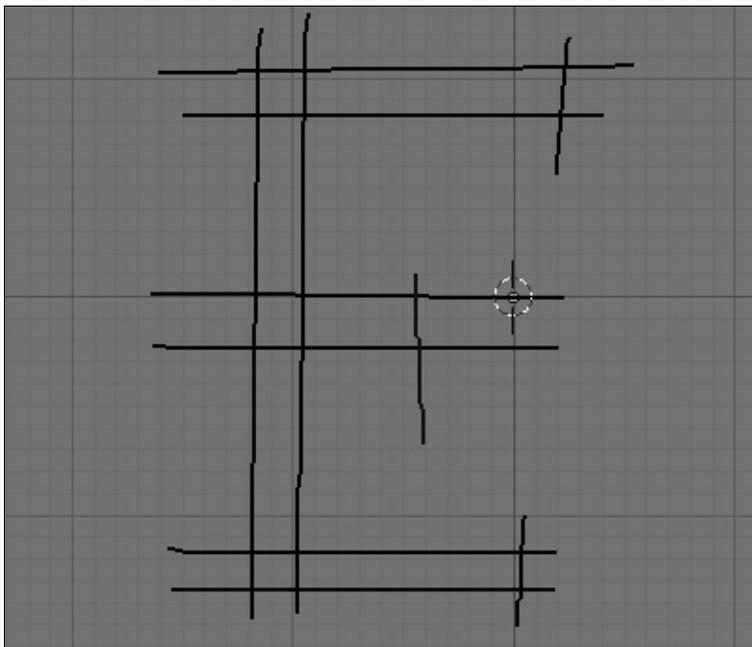


Рис. 2.38. Заготовка для модели буквы "Е"

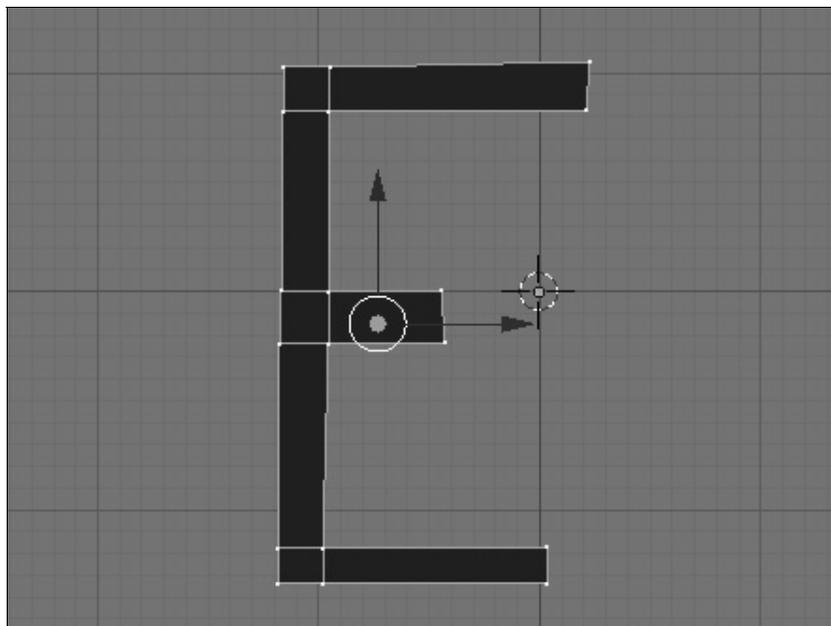


Рис. 2.39. Объект, созданный с использованием Retopo

2.6. Дополнительный инструментарий

Рассмотренные приемы работы с Mesh-объектами являются основными, но не единственными. Разработчики программы создали множество инструментов, позволяющих добиться желаемого результата. Одним из них является объект и модификатор **Lattice** (Решетка). По сути дела, это вспомогательная решетка, имеющая заданное пользователем количество элементов, с помощью которых можно изменять форму модели. Принцип работы заключается в том, что, манипулируя элементами **Lattice**, вы тем самым влияете и на основной объект. Причем сама структура модели не изменяется. Давайте рассмотрим **Lattice** в действии.

Откройте новый проект и пятикратно разбейте инструментом **Subdivide** структуру имеющегося куба. Выйдите из режима редактирования. Теперь создайте новый объект **Lattice**, выбрав в меню **Add** пункт **Lattice**. Кажется, что на экране ничего не изменилось. На самом деле объект **Lattice** слишком маленький, и поэтому увеличьте его масштаб (<S>) так, чтобы он стал немного больше куба. Если теперь войти в режим редактирования (<Tab>), то можно увидеть, что **Lattice** имеет точно такую же структуру, что и обычный **Mesh**. Однако к нему не могут быть применены основные способы редактирования. Единственные настройки объекта расположены на панели **Lattice** окна **Buttons Window** (рис. 2.40).

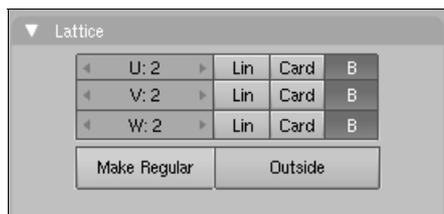


Рис. 2.40. Панель настроек объекта **Lattice**

Выйдите из режима редактирования и увеличьте значения в полях **U**, **V** и **W** до 5. Таким образом вы разбили структуру объекта на большее количество элементов (рис. 2.41).

Запомните следующее правило работы с этим инструментом. Объект **Lattice** должен иметь достаточное количество вершин для выполнения поставленной задачи. Чем больше элементов, тем точнее может выполняться модифицирование. С другой стороны, слишком большое количество элементов затруднит выполнение задачи.

Однако, прежде чем произвести какую-либо манипуляцию, оба объекта нужно связать между собой. Для этого существует специальный модификатор

Lattice, расположенный на панели **Modifiers** окна **Buttons Window** (рис. 2.42). Примените его к кубу.

Теперь в окне **Ob:** нужно указать имя Lattice-объекта. По умолчанию он так и называется. Введите слово "Lattice".

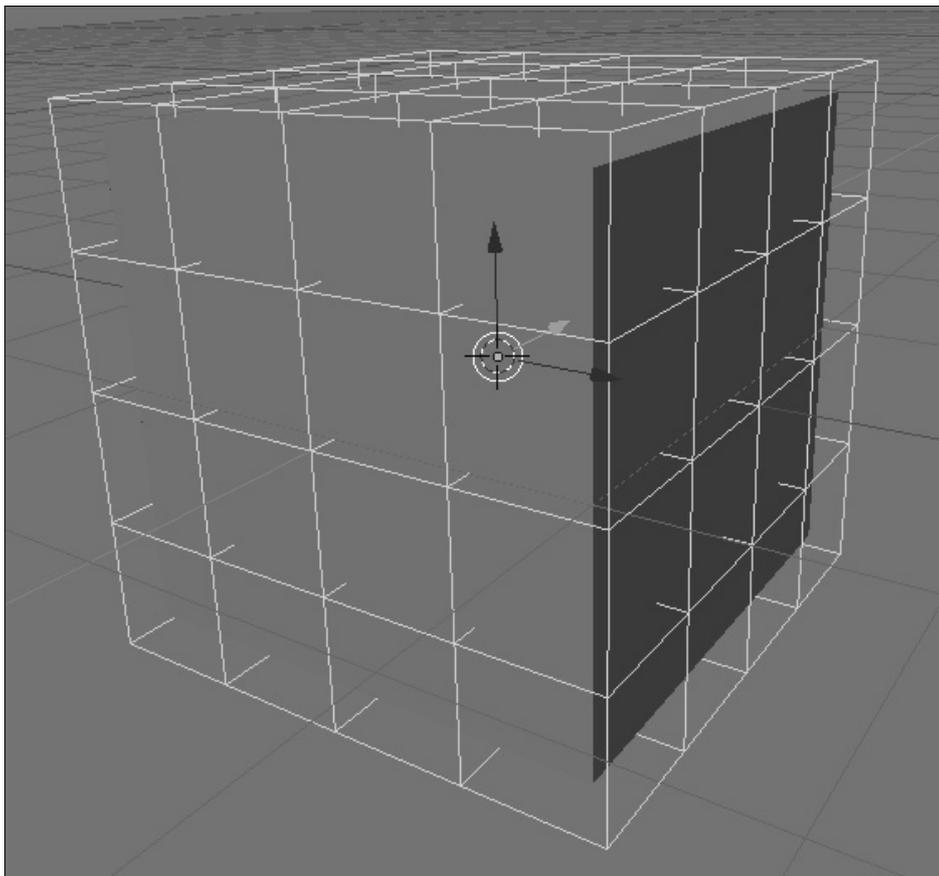


Рис. 2.41. Структура Lattice после операции разбиения

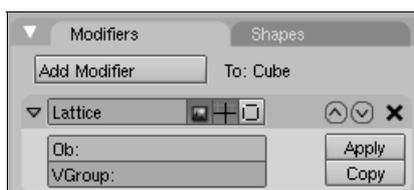


Рис. 2.42. Панель модификатора Lattice

Внимание!

Программа Blender чувствительна к регистру клавиатуры. Поэтому необходимо строго следить за соответствием имен. Если было введено имя несуществующего объекта, то программа после нажатия <Enter> удалит его из поля.

Настало время попробовать инструмент в действии. Перейдите в режим редактирования, отметьте любую точку на *Lattice* и попробуйте ее сдвинуть. Вы заметили, как послушно вслед за ней поползла соответствующая область примитива? Для вершин вспомогательного объекта доступны все возможные манипуляции — перемещение, ротация и масштабирование. Естественно, для нормальной ротации должно быть выделено не менее двух точек. Управление модификацией объекта возможно только с использованием вершин.

Обратите внимание на то, что модифицированный участок примитива не содержит резких граней, все изменения выполнены максимально плавно. Дело в том, что *Lattice* имеет три режима *интерполяции* (способ нахождения промежуточных значений по имеющемуся набору уже известных) вершин объекта, и по умолчанию включен самый мягкий. Выбрать необходимый можно кнопками на панели модификатора. Они трижды продублированы для осей XYZ:

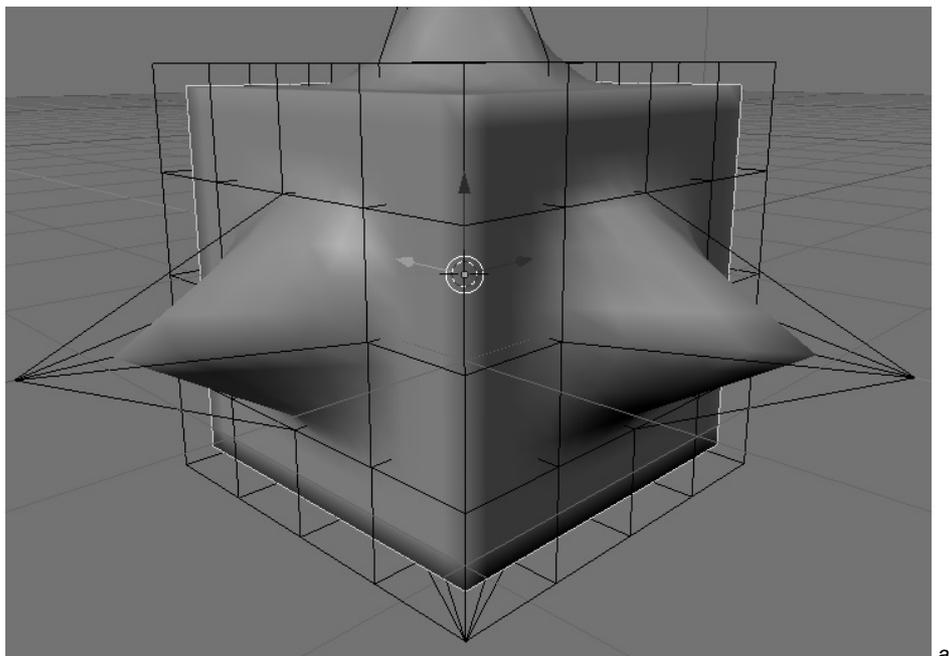
- ◆ **Lin** (Linear). При линейной интерполяции не просчитываются переходы между элементами модифицируемого объекта. Таким образом, полученный результат равноценен такому же изменению при обычном редактировании (рис. 2.43, *а*);
- ◆ **Card** (Cardinal). Нечто среднее между линейной и сплайновой интерполяцией (рис. 2.43, *б*);
- ◆ **B** (B-Spline). Режим сплайна (кривые) обеспечивает наиболее плавные переходы между элементами объекта (рис. 2.43, *в*).

Теперь выделите основной объект и перейдите в режим редактирования. Как вы видите, примитив вновь предстал в своем первоначальном виде. Этой особенностью *Lattice* удобно пользоваться, совмещая обычное редактирование с конечной модификацией модели. Естественно, если применить модификатор к модели и удалить вспомогательную решетку, все изменения зафиксируются в структуре объекта.

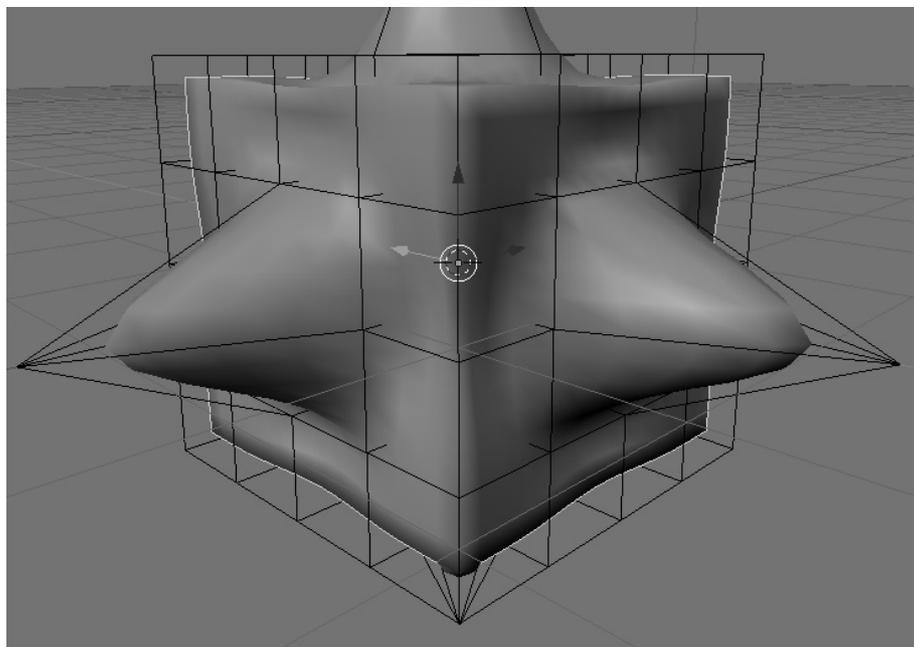
Совет

Работать с *Lattice* удобнее, если поместить вспомогательную решетку в другом слое. В этом случае окончательный просмотр основной модели не отягощен ненужными элементами в 3D-окне.

Теперь ответьте на вопрос: как вы поступите, если вам будет необходимо создать модель обычной пустой коробки? Предвижу стандартный ответ:



а



б

Рис. 2.43. Три режима работы Lattice: Linear (а), Cardinal (б), Spline (в)

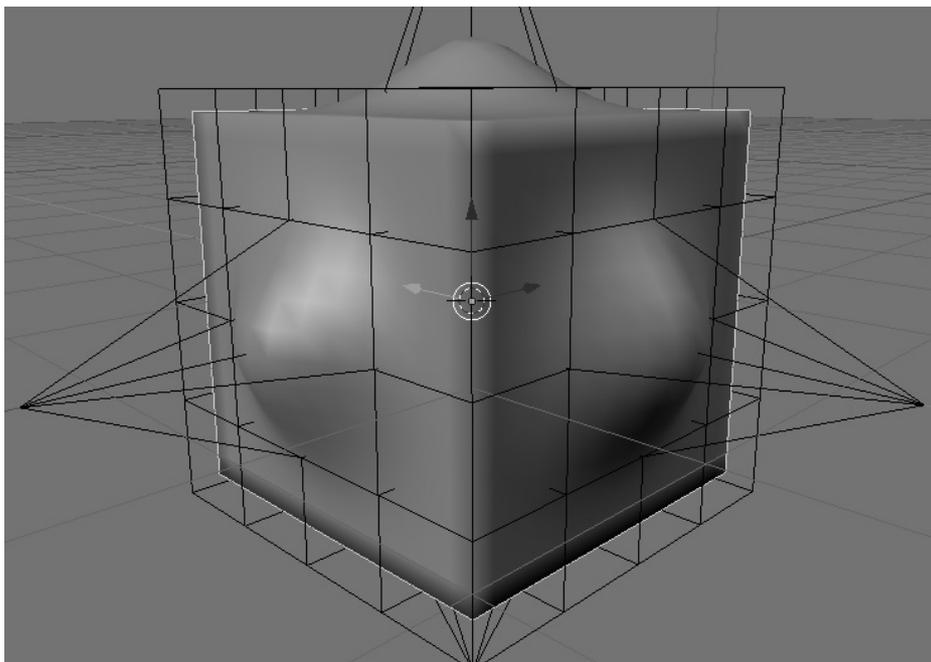


Рис. 2.43. Три режима работы Lattice: Spline (e)

"Использую пять плоскостей, расположенных параллельно друг другу". В принципе, возможно и так. А если было бы нужно вырезать сферу внутри куба?

Рассмотренными ранее способами редактирования это сделать практически нереально. Однако разработчики предусмотрели похожие задачи и создали функции *булевых операций*.

В мире логики и программирования понятие "булевы операции" хорошо знакомо. Под этим понимается вычисление конечного результата на основе двух значений, "Да" и "Нет", и логического условия между ними. В трехмерной графике этот способ применяется для создания уникального объекта на основе двух начальных. Имеется три функции булевых операций:

- ◆ **Intersect (Пересечение)**. Новый объект имеет форму, созданную из пересеченных структур двух примитивов. Все оставшиеся элементы вне пересечения при генерации не используются;
- ◆ **Union (Сложение)**. В результате этой операции новый объект состоит из сложенных структур двух исходных;
- ◆ **Difference (Различие)**. Единственная операция, при которой учитывается порядок расположения начальных объектов. Результат — противополож-

ной операции *Intersect*. Новый объект будет иметь форму тех структур, которые не попали в пересечение.

Для примера создадим модель пустой коробки. Откройте Blender и сделайте копию имеющегося куба (<Shift>+<D>). Теперь сместите второй экземпляр вверх по оси Y. Он у нас является той формой, по которой будет вырезано пространство внутри первого объекта. Уменьшите масштаб формы, как на рис. 2.44.

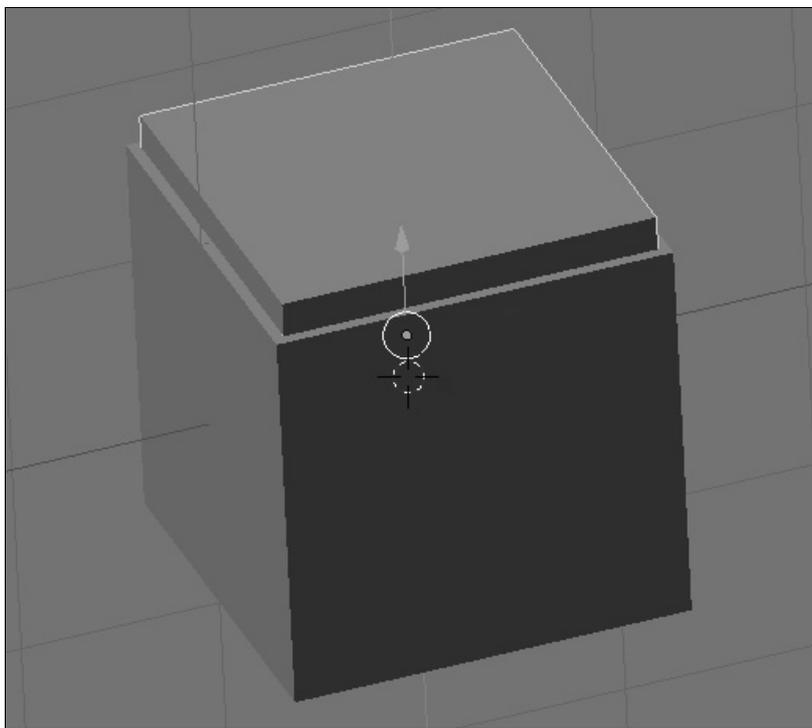


Рис. 2.44. Заготовки для модели коробки

Сначала выделите первый объект, затем добавьте к нему второй примитив (с клавишей <Shift>). Нажмите клавишу <W> для вызова контекстного меню **Boolean Tools** и выберите пункт **Difference**. На первый взгляд ничего не изменилось, но если раздвинуть начальные объекты, то можно увидеть и результат (рис. 2.45).

Существуют определенные требования, при которых булева операция будет правильно произведена. Во-первых, это должны быть только Mesh-объекты. С другими типами объектов она не работает. Во-вторых, необходимо следить за тем, чтобы структуры начальных объектов состояли из однородных эле-

ментов, т. е. полигоны должны быть либо треугольными, либо квадратными. В противном случае результирующий объект будет иметь нарушенную структуру.

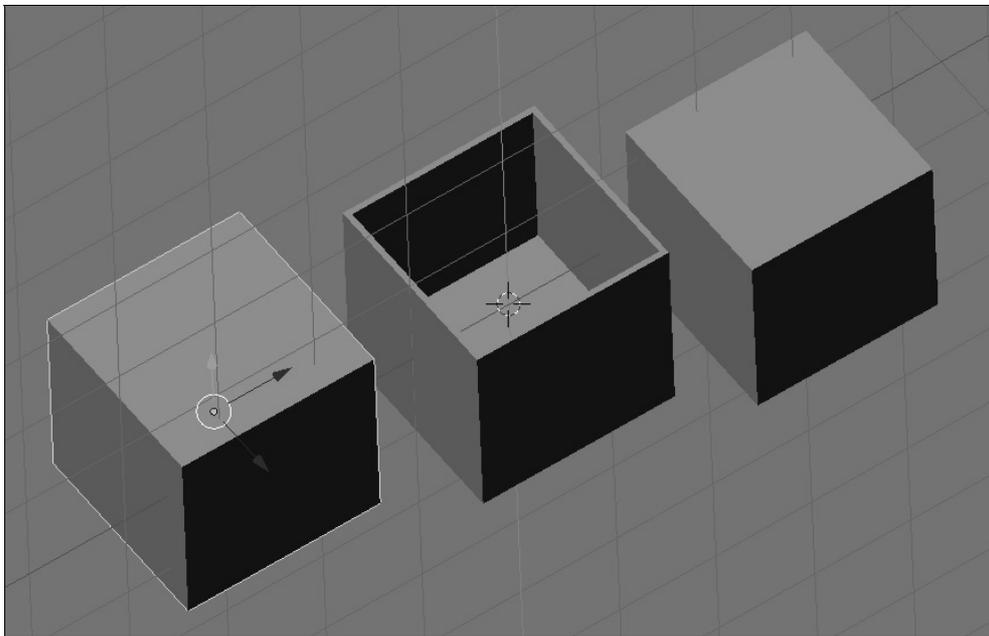


Рис. 2.45. Результат булевой операции *Difference*

При разработке высокополигональной модели иногда бывает нужно создать ее низкополигональную копию. Такая задача ставится для моделей, которые в дальнейшем будут использоваться в игровых проектах. Обычно низкополигональный вариант необходим для непосредственного использования в игровом движке, а высокополигональный — для обработки межуровневых роликов. Задача эта отнюдь не тривиальна и переделать уже готовую модель — большой труд. Однако, если на этапе моделирования это предусмотреть и воспользоваться специальным инструментом *Multires*, процесс переноса происходит безболезненно.

Название *Multires* — это сокращение от двух слов: *multi* (множественный) и *resolution* (разрешение). Суть его заключается в том, что по желанию пользователя можно выбирать разные уровни разрешения модели. Причем при редактировании на одном уровне все изменения автоматически переносятся и в другие. Таким образом, пользователь не только получает возможность работы одновременно в разных разрешениях, но и решать, какой ему уровень необходим для обработки в данный момент. По сути дела, *Multires* — это

модификатор, но расположен он не в обычном списке, а на отдельной панели **Multires** окна **Buttons Window** (рис. 2.46).

Создайте в свободном месте сцены примитив *Monkey*. Это низкополигональная модель головы обезьянки. Если посмотреть на строку статуса вверху окна программы, то можно увидеть, что количество полигонов (Face) составляет всего 500.

Теперь нажмите клавишу <F9> и в открывшейся панели **Multires** нажмите кнопку **Add Multires**. Для добавления уровней разрешения дважды нажмите кнопку **Add Level** (рис. 2.46).

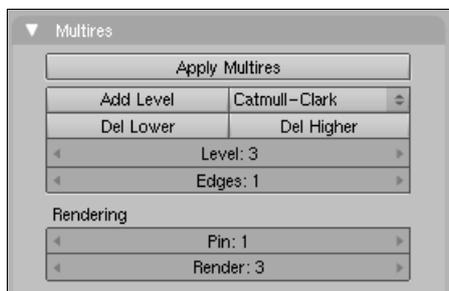
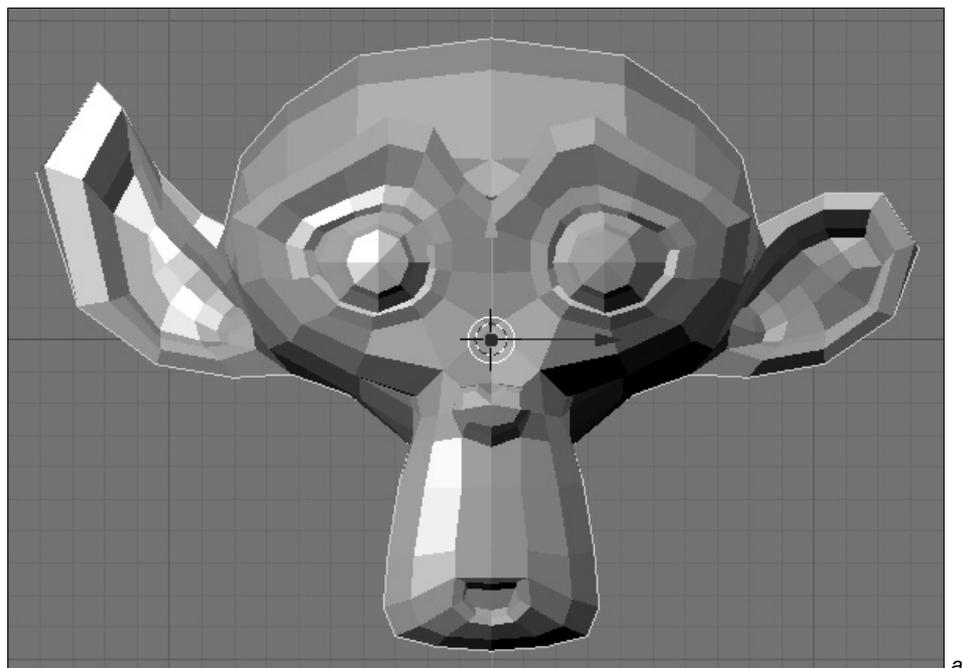


Рис. 2.46. Панель **Multires**

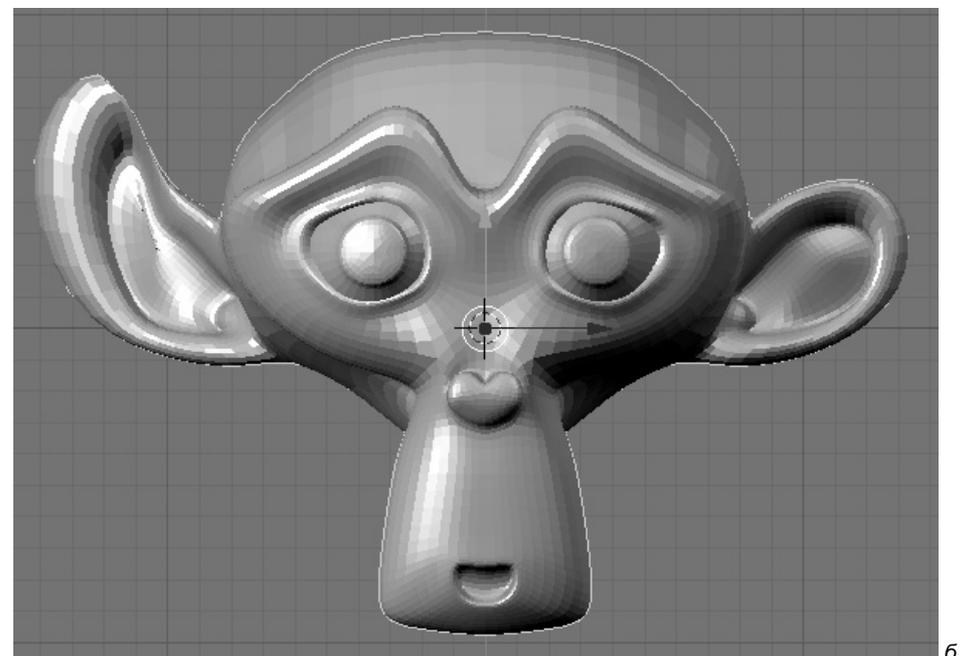
Рассмотрим панель подробнее:

- ◆ **Apply Multires** (Применить). Стандартное применение и последующее удаление модификатора;
- ◆ **Add Level** (Добавить уровень). Добавление очередного уровня разрешения;
- ◆ **Del Lower** (Удалить низкий). Производится удаление самого низкого уровня из списка. Все остальные сдвигаются на один шаг вниз;
- ◆ **Del Higher** (Удалить высокий). Удаление уровня, следующего по разрешению;
- ◆ **Level** (Уровень). Переключение отображения модели на нужный уровень разрешения;
- ◆ **Edges** (Ребра). Управление уровнем отрисовки ребер в окне просмотра. Этим можно воспользоваться, если из-за высокого количества полигонов система начинает тормозить;
- ◆ заголовок **Rendering** (Обработка). Параметры для управления разрешением при окончательной обработке модели.

Переключитесь на самый низкий уровень и попробуйте в режиме редактирования изменить какую-либо часть модели. Теперь, если перейти на более высокое разрешение, эти изменения останутся на месте (рис. 2.47).



а



б

Рис. 2.47. Первый уровень разрешения (а) и последний (б)

При работе над объектом, который имеет часто повторяющиеся элементы, можно значительно сократить время моделирования, если воспользоваться модификатором *Array* (Массив). Простым примером такого объекта могут служить шестеренки или ступени лестницы. Без использования *Array* вам остается разве что воспользоваться операцией клонирования и вручную расставлять все по местам. Все бы ничего, но если таких элементов не два и не три, а добрый десяток, то задача сильно усложняется.

Рассмотрим пример создания модели лестницы с шестью ступенями. В качестве начального объекта имеется первая ступень (рис. 2.48).

Добавляется к объекту модификатор *Array* (рис. 2.49).

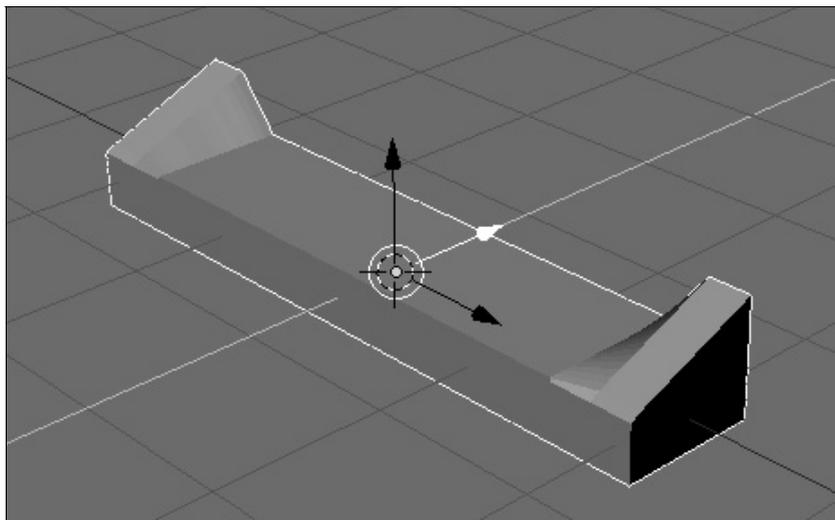


Рис. 2.48. Начальный элемент для *Array*

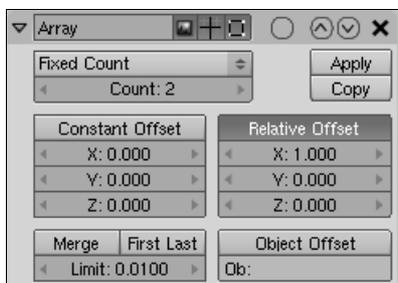


Рис. 2.49. Панель *Array*

Основные режимы работы модификатора *Array* расположены в выпадающем меню:

- ◆ **Fixed Count** (Ограниченное количество). Чаще всего используемый режим (включен по умолчанию). Создает объекты в количестве, указанном в поле **Count** (Счет);
- ◆ **Fixed Length** (Ограниченная длина). Получаемый объект всегда имеет длину, указанную в поле **Length** (Длина). Количество объектов, необходимое для этого, рассчитывается программой;
- ◆ **Fit to Curve Length** (Привязка к размеру кривой). Немного похоже на предыдущий пункт, только в качестве ограничения выступает примитив "сплайн" (см. разд. 3.2).

Для рассматриваемой модели наиболее оптимальным является первый режим, т. к. имеется заранее заданное количество ступеней. В поле **Count** устанавливается необходимое количество элементов.

Обратите внимание на кнопку **Relative Offset** (Относительное смещение), активирующую поля, находящиеся чуть ниже (см. рис. 2.49). Именно здесь устанавливается необходимое смещение генерируемых элементов по осям XYZ.

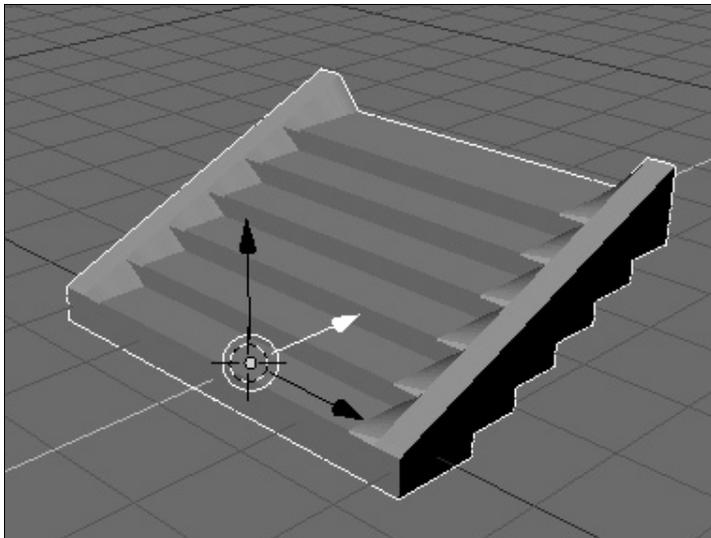


Рис. 2.50. Результат работы модификатора *Array*

Этим возможности модификатора не ограничиваются. Путем специальной опции можно очень гибко управлять результатом работы *Array*. Рассмотрим следующий практический пример.

Все испытания будем проводить на стандартном кубе, который появляется при начальном запуске программы. Создайте дополнительный объект `Empty` и поместите его чуть выше примитива (по координате `Z`). Объект `Empty` является вспомогательным и в обработке результата визуально не участвует. Использовать его мы будем как своего рода рычаг для управления массивом (рис. 2.51).

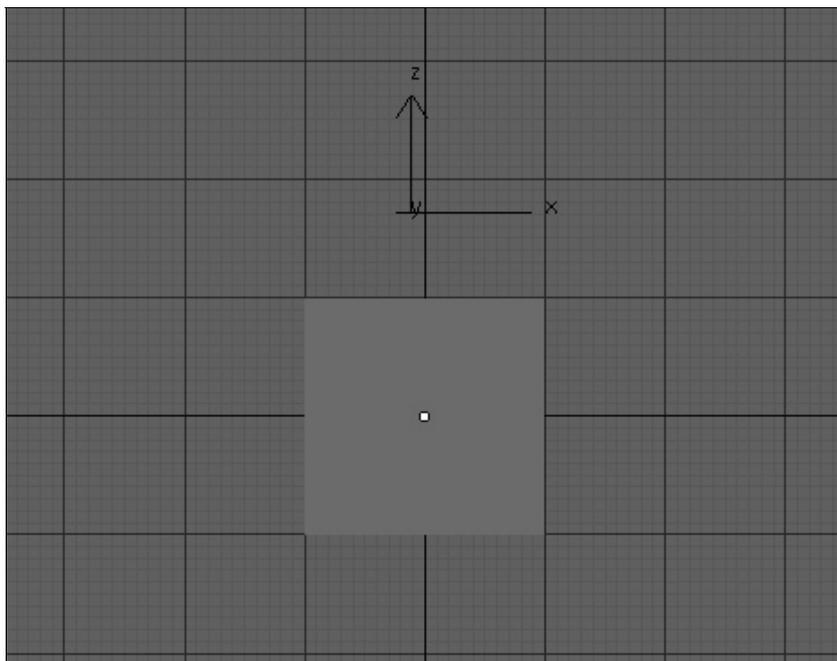


Рис. 2.51. Расположение объектов `Cube` и `Empty`

Добавьте модификатор `Array` к основному объекту. Измените следующие параметры, установленные по умолчанию, на:

- ◆ поле **Count** — 5;
- ◆ кнопка **Relative Offset** отжата;
- ◆ кнопка **Object Offset** (Смещение по объекту) нажата.

Введите в поле **Ob**: название вспомогательного объекта (имя по умолчанию — `Empty`). Для более удобного просмотра результата перейдите в режим просмотра камерой. Теперь, если попробовать перемещать или вращать вспомогательный объект, можно заметить, что все дополнительные элементы послушно следуют за ним (рис. 2.52).

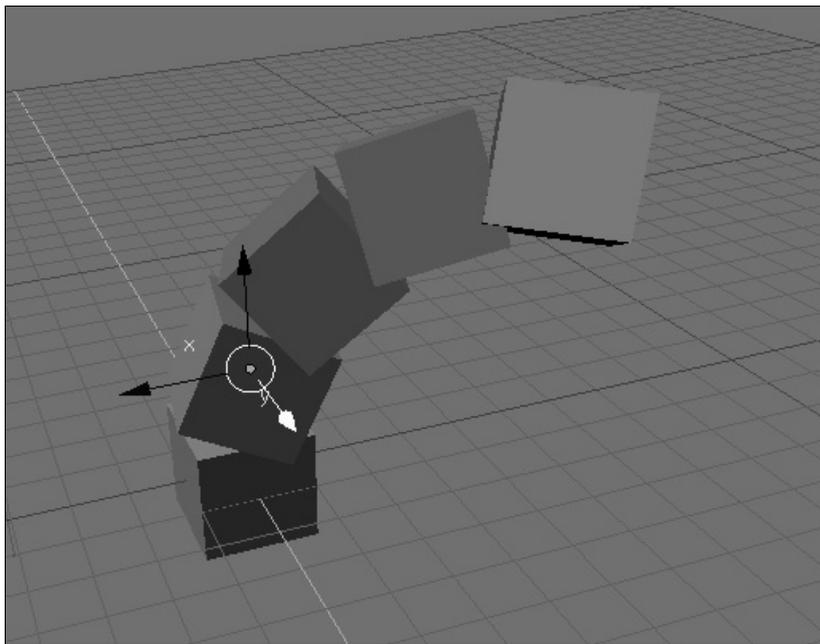


Рис. 2.52. Влияние ротации `Empty` на результат

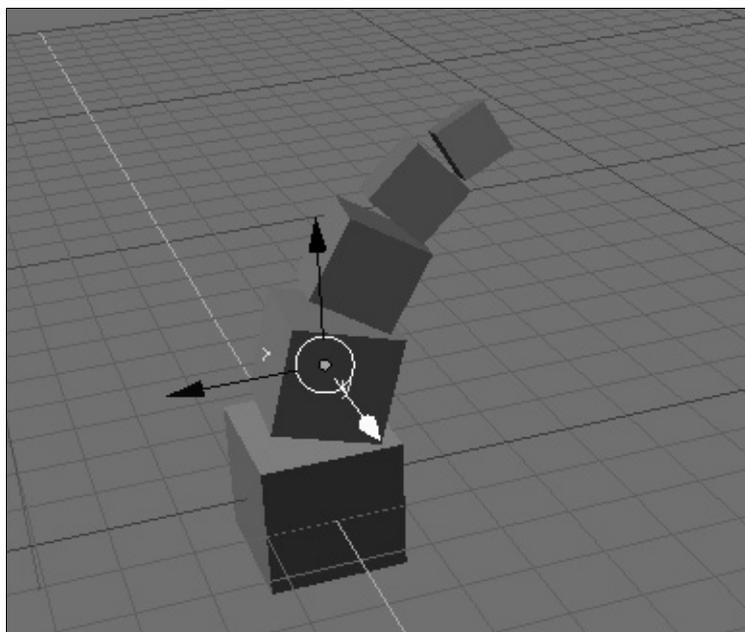


Рис. 2.53. Результат масштабирования `Empty`

Совсем другой вид приобретает результат при использовании масштабирования Empty (рис. 2.53).

Рассмотренные в этой главе способы работы с объектами, разумеется, являются не единственными. Кроме того, нельзя сказать, что какой-то лучше, а какой-то хуже. Чаще всего их используют параллельно. Оптимально, к примеру, создавать основной корпус с использованием симметрии и базовых инструментов, дорабатывать мелкие детали в режиме скульптуры, а необходимые вырезы совершать при помощи булевых функций. Но, конечно же, все зависит от поставленной задачи.

ГЛАВА 3



Кривые, поверхности NURBS

Работая с Mesh-объектами, вы должны были заметить, как тяжело создаются модели с плавными изгибами форм. Конечно, существуют модификаторы и инструменты, позволяющие добиться нужного эффекта, но чревато это избыточным количеством полигонов и увеличением времени разработки. Если вы работали до этого с каким-нибудь векторным редактором, то понятие "кривые" должно быть вам знакомо. По сути дела, *кривая* (иногда их еще называют сплайнами) — это особый объект, форма которого рассчитывается с помощью специальных функций, выполняющихся в реальном времени. Это является главным отличием от Mesh-примитивов, которые представляют собой законченный массив элементов. Именно с кривыми можно добиться создания модели любой геометрической сложности, изобилующей плавными переходами. Ведь ограничений на количество промежуточных точек между главными вершинами кривой просто не существует. Но управлять ими всеми, как вершинами Mesh-объекта, невозможно. С одной стороны, кривая — это чисто двумерный объект. Однако в случае с трехмерным редактором, каковым является Blender, она имеет соответствующие пространственные координаты. Не стоит думать, что при помощи кривых можно создавать только каркасные объекты. Blender имеет немало специальных примитивов, основанных на сплайнах, которые являются сплошными объектами. Вот только подчиняются такие примитивы особым законам и правилам, характерным для семейства сплайнов.

3.1. Основные понятия

В *разд. 2.1* рассматривался пример создания звезды. Задача эта была решена с такой легкостью лишь потому, что в наборе примитивов Mesh находился двумерный объект Circle, обладающий некоторыми характеристиками

сплайнов. Попробуйте, к примеру, создать на основе *Mesh* такой предмет, как ваза. Можно с уверенностью сказать, что это вряд ли получится или займет немало времени.

Существует два типа кривых — сплайны Безье и NURBS (Non Uniform Rational B-Spline — неоднородный рациональный B-сплайн). Необходимо знать их отличия и возможности для эффективного моделирования. Сплайны Безье, или, как их еще называют, простые кривые, оптимально подходят для создания логотипов, надписей, вспомогательных элементов для анимации или выдавливания. NURBS, благодаря своим обширным настройкам, подходят для создания математически точных объектов.

Помимо кривых, в составе примитивов имеются поверхности, состоящие из сплайнов NURBS. В отличие от кривых, поверхности обладают дополнительным измерением: основное называется *U*, а дополнительное — *V*. С этими значениями мы познакомимся чуть позже, а пока замечу, что их можно использовать для создания жидких объектов.

Все кривые можно найти в меню **Add | Curve** (Добавить | Кривая):

- ◆ **Bezier Curve** (Кривая Безье);
- ◆ **Bezier Circle** (Окружность Безье);
- ◆ **NURBS Curve** (Кривая NURBS);
- ◆ **NURBS Circle** (Окружность NURBS);
- ◆ **Path** (Путь). Специальный объект для использования в анимации или выдавливания.

Для поверхностей существует особое меню — **Add | Surface** (Добавить | Поверхность):

- ◆ **NURBS Curve** (Кривая NURBS);
- ◆ **NURBS Circle** (Окружность NURBS);
- ◆ **NURBS Surface** (Поверхность NURBS);
- ◆ **NURBS Tube** (Труба NURBS);
- ◆ **NURBS Sphere** (Сфера NURBS);
- ◆ **NURBS Donut** (Тор NURBS).

3.2. Работаем со сплайнами

Начнем с того, что любая кривая, будь то Безье или NURBS, подчиняется тем же правилам и способам манипулирования, что и примитивы *Mesh*. Это относится к действиям как в пространстве сцены, так и в режиме редактирования. Любая кривая имеет по меньшей мере две контрольные вершины, которые

можно перемещать, масштабировать (при выделении нескольких) или вращать. Для изменения формы кривой между ее точками существуют специальные манипуляторы, или, как их еще называют, рычаги. Причем программа предлагает несколько вариантов, индивидуальных для конкретного типа сплайна. Рассмотрим для начала работу с Безье.

Для Безье имеется четыре типа рычага (рис. 3.1). По умолчанию после создания объекта включен режим *Aligned* (Выровненный). Особенностью его является чуткое реагирование обоих сегментов кривой (до и после активной точки) на изменения положения рычага. В режиме *Aligned* все управляющие рычаги отмечены фиолетовым цветом и лишь активный — светло-розовым (рис. 3.1, а).

Обратите внимание на то, что любой рычаг имеет в качестве центра точку на кривой и две управляющие вершины на нем самом. Принцип работы очень простой. Необходимо выделить нужную вершину рычага и перемещать ее в надлежащем направлении (клавиша <G>). При этом имеется возможность изменять размер активного плеча.

Следующий тип рычага — *Free* (Свободный). Для перехода в этот режим необходимо выделить нужные рычаги и нажать клавишу <H>. (Повторное нажатие вернет режим *Aligned*.) Рычаги в этом случае отмечаются черным цветом. Этим режимом удобно пользоваться, если нужно изменить конкретный сегмент кривой, не затрагивая близлежащие (рис. 3.1, б). При этом можно получить острые углы на стыках.

Особый вид рычага — *Vector* (Векторный). В этом случае плечи рычагов всегда "смотрят" на близлежащие точки (рис. 3.1, в). Если попробовать такой рычаг сдвинуть с места, то он переключается в режим *Free*.

Последний тип рычага — *Auto* (Автоматический). Для получения максимально плавных переходов между точками нужно использовать именно его. В этом случае программа автоматически рассчитывает необходимую длину и направление плеч с учетом уже имеющихся изгибов (рис. 3.1, г). При изменении местоположения плеча такого рычага он также переключается в режим *Free*.

Для демонстрации работы с кривыми Безье сделаем модель логотипа Blender. Условимся, что он должен иметь небольшую толщину, закругленные края и сглаживание. Удобнее всего создавать такого рода объект, имея в качестве обложки в окне **3D View** образец будущей модели (картинку к этому примеру вы можете найти на прилагающемся к книге диске в папке `examples\3\logo.png`). Сделать это можно, выбрав пункт меню **View | Background Image** (Просмотр | Фоновое изображение) окна **3D View**. В появившейся плавающей панели нажмите кнопку **Use Background Image** (Использовать фоновое изображение), а затем **Load** — для загрузки нужного файла картинки (рис. 3.2).

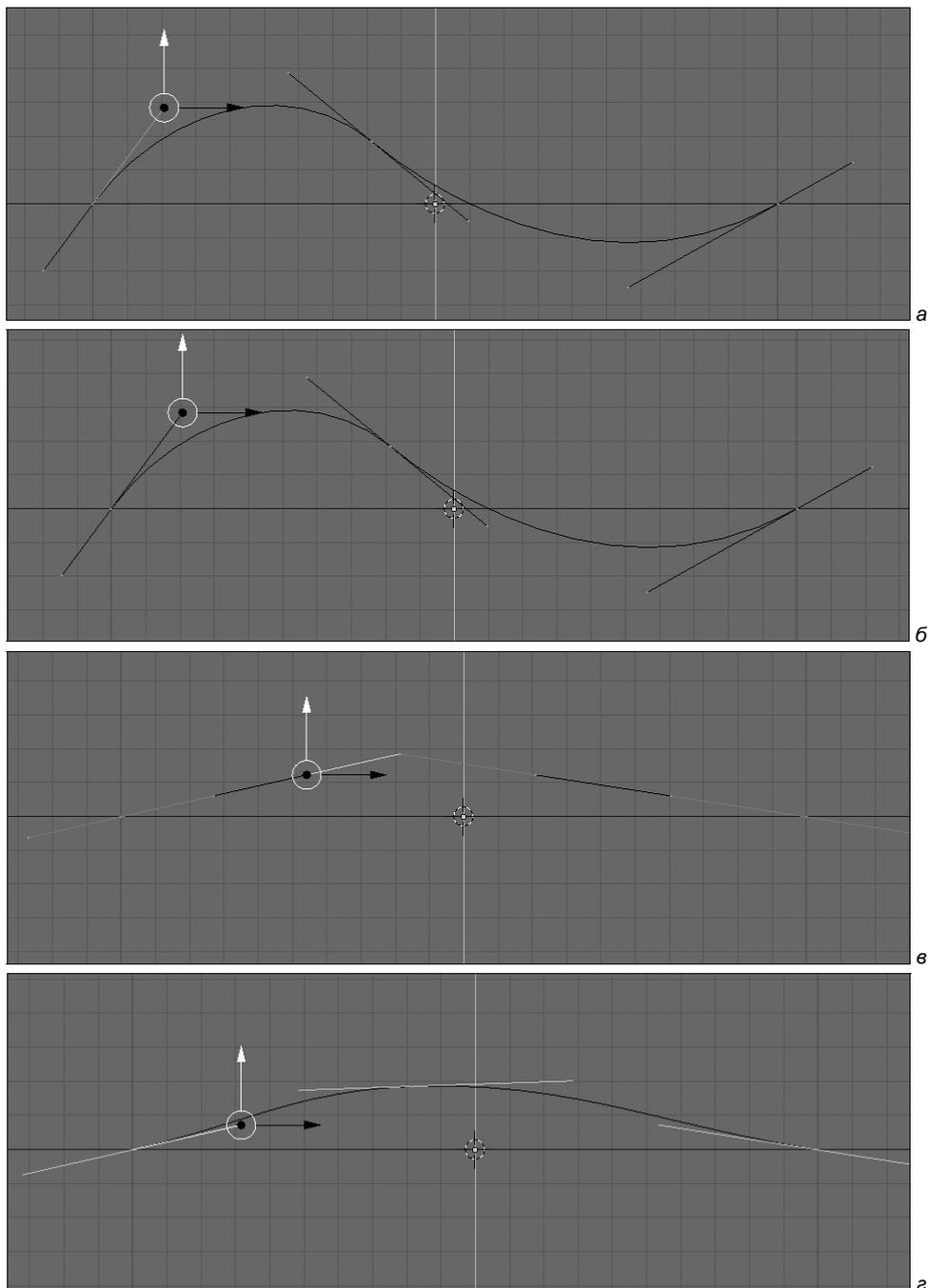


Рис. 3.1. Виды рычагов кривой Безье: Aligned (а), Free (б), Vector (в), Auto (г)

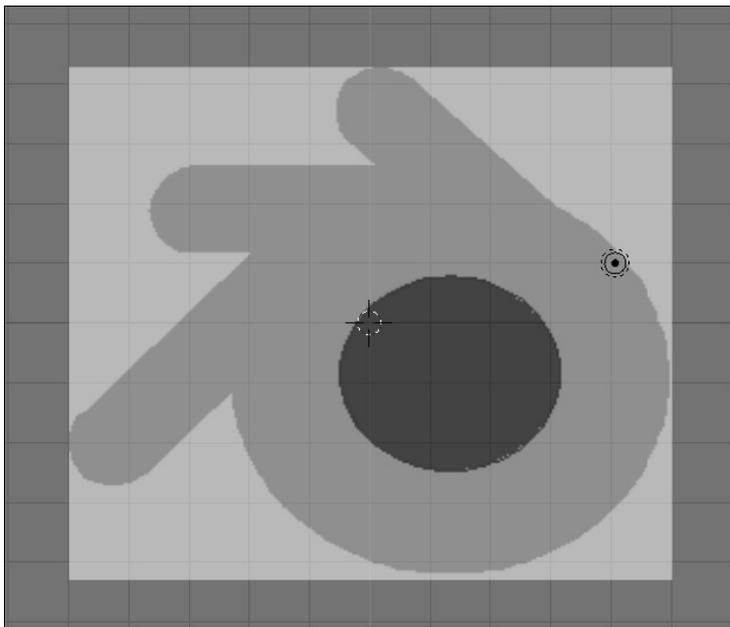


Рис. 3.2. Образец логотипа

Такое фоновое изображение имеет ряд особенностей. Во-первых, оно доступно во всех видах проекции, кроме ручного режима. Имеется возможность регулирования степени прозрачности бегунком **Blend** (Смешивание), управления размерами и местоположением по осям XY. Кроме того, возможно использовать в качестве фона видеофайлы. Запомните, фоновое изображение используется только для упрощения работы и в рендере не участвует.

Отрегулируйте фоновое изображение по своему вкусу и создайте объект **Bezier Curve** (**Add | Curve | Bezier Curve**). Поместите кривую с крайней левой стороны и настройте ее так, чтобы сегменты легли строго по контуру рисунка (рис. 3.3). Для этого совсем не обязательно переключаться в другой режим использования рычагов, используйте те, что активны по умолчанию.

Теперь необходимо продолжить кривую и создать новую точку с краю очередного выступа. Просто щелкните в нужном месте кнопкой мыши с нажатой клавишей <Ctrl>. Обратите внимание на то, что при попытке правильно отстроить кривизну сплайна и создать острый угол изгибается уже готовая сторона модели. Вот тут пригодится другой тип рычага, а именно **Free**. Выделите точку в месте создания угла и нажмите клавишу <H>. Как вы помните, в этом режиме имеется возможность регулирования кривизны сплайна без затрагивания соседних областей (рис. 3.4).

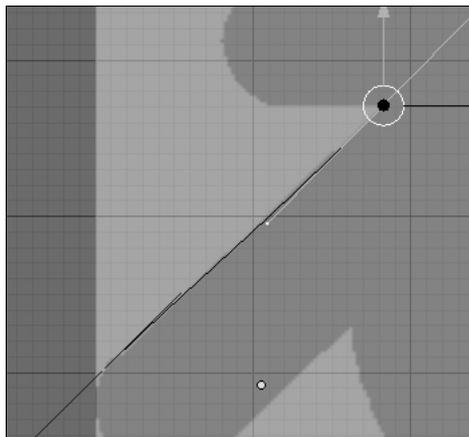


Рис. 3.3. Первый этап создания модели

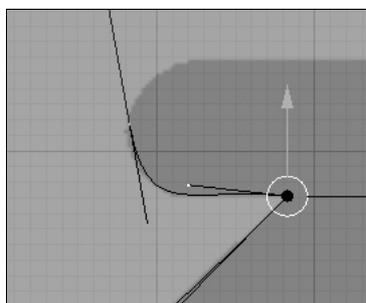


Рис. 3.4. Создание острого угла с рычагом Free

Внимание!

Не допускайте перекручивания кривой при моделировании. Впоследствии это может отразиться на правильности заливки формы и корректности отражения объекта.

Обрисуйте все оставшиеся выступы, используя метод, рассмотренный ранее (рис. 3.5).

Теперь познакомимся со способом разбиения сегмента пополам и создания тем самым новой вершины. Добавьте новую точку у основания первого выступа (рис. 3.6).

Выделите новую и предыдущую вершины. Нажмите клавишу <W> для вызова контекстного меню и выберите пункт **Specials | Subdivide**. После выполнения этой функции на кривой добавится новая вершина. Окончательно доработайте оставшуюся часть модели, как на рис. 3.7.

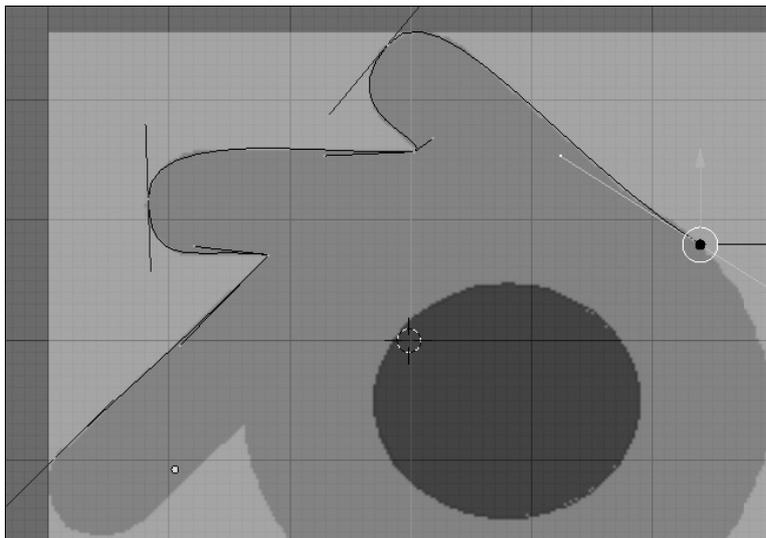


Рис. 3.5. Создание контура выступов

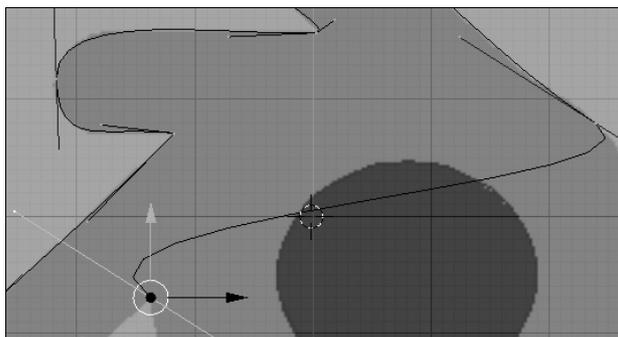


Рис. 3.6. Точка для продолжения контура

Осталось объединить крайние вершины для создания замкнутой кривой и заливки полученного пространства. Просто выделите их и нажмите клавишу <C>. Остальное программа Blender доделает сама. Немного отредактируйте полученный объект в соответствии с формой образца (рис. 3.8).

Основная форма логотипа готова, но осталось проделать отверстие на месте окружности рисунка синего цвета. Решается это очень и очень просто. Не нужно выделять и удалять полигоны, как при работе с Mesh. Необходимо указать область для отверстия в модели, а умная программа сама все рассчитает.

В данный момент созданный логотип перекрыл собою фоновый рисунок и не позволяет правильно отметить нужное место. Существует специальный

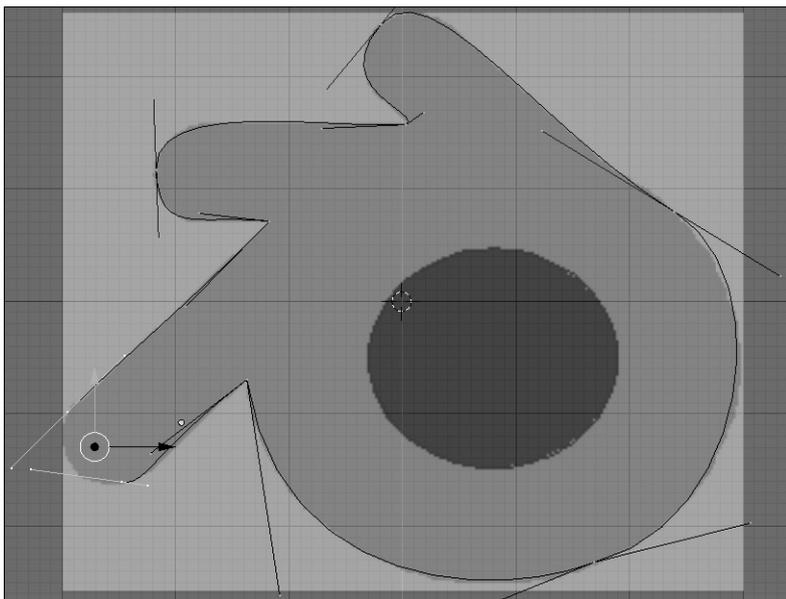


Рис. 3.7. Доработка модели

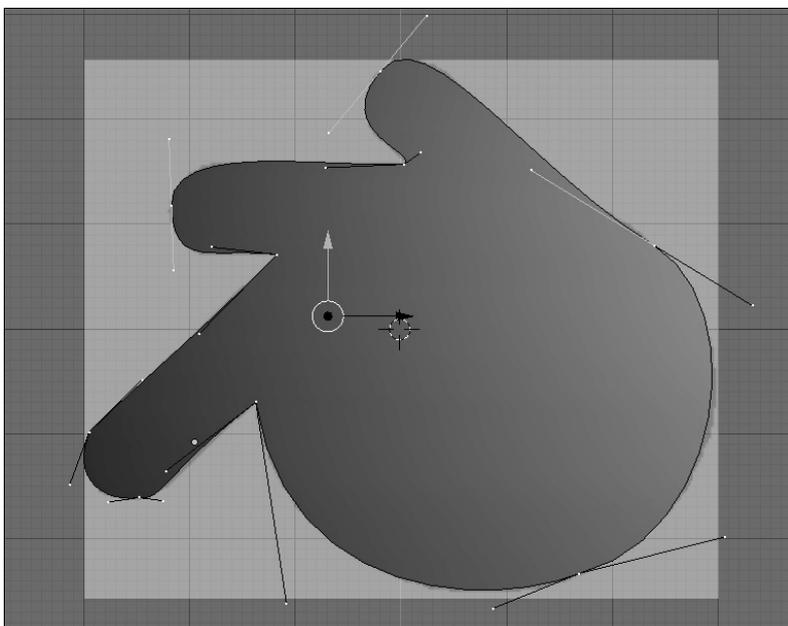


Рис. 3.8. Окончательная доводка контура

режим отображения сцены в 3D-окне, который прорисовывает все объекты в виде каркасов, без заливки (см. разд. 1.2). Для включения его нажмите клавишу <Z> или выберите пункт **Wireframe** меню **Draw type**.

Не выходя из режима редактирования, установите 3D Cursor в центре предполагаемого отверстия и добавьте новый объект — **Add | Bezier Circle**. Отредактируйте круг в соответствии с рисунком, нажмите клавиши <Tab> и <Z>. Если все было сделано правильно, на месте, отмеченном кругом, появится отверстие (рис. 3.9).

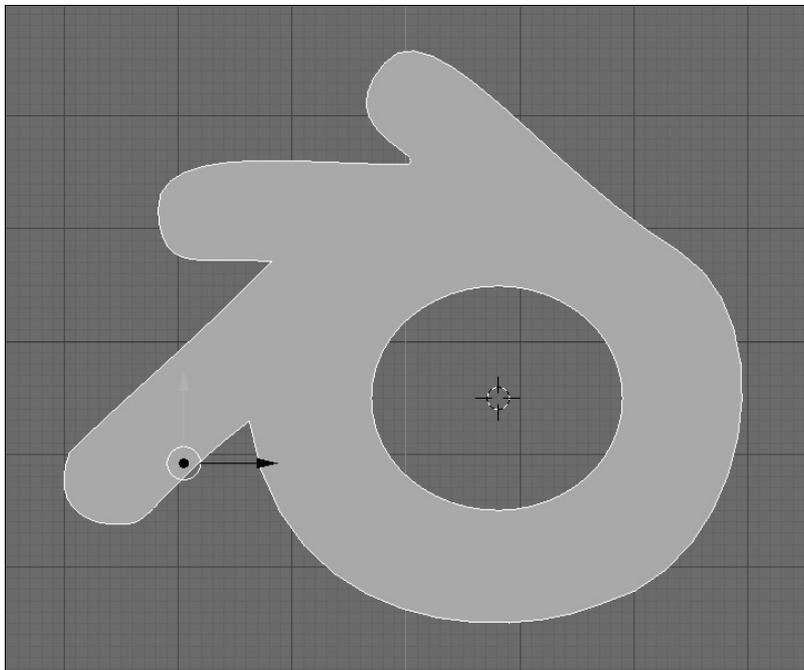


Рис. 3.9. Программа умеет правильно определять место под отверстия

По условию наша модель должна иметь некоторую толщину и закругленные края. Для подобных операций Blender предлагает опции, расположенные на панели **Curve and Surface** (Кривые и поверхности) окна **Buttons Windows** (горячая клавиша <F9>).

Некоторые из нужных опций мы сейчас рассмотрим (рис. 3.10):

- ◆ **DefResolU**. Редактируя это поле, можно добиться более плавных изгибов. Фактически происходит увеличение разрешения кривой;
- ◆ кнопки **Back** (Задний), **Front** (Передний), **3D** (Трехмерный). Управляют заливкой объекта;

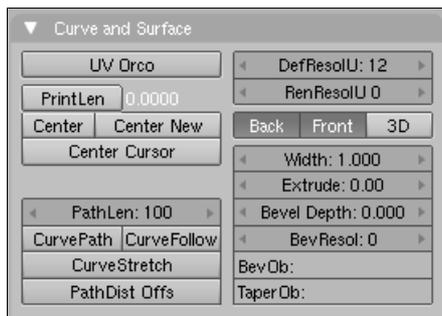


Рис. 3.10. Панель настроек кривой

- ◆ **Extrude** (Выдавливание). Функция прибавления объемности методом вытягивания по координате Z;
- ◆ **Bevel Depth** (Глубина края). Параметр, отвечающий за создание скосов на краях объекта;
- ◆ **BevResol**. Разрешение скоса.

Для окончательной доработки логотипа установите следующие параметры:

- ◆ **DefResolU** = 24;
- ◆ **Extrude** = 0.10;
- ◆ **Bevel Depth** = 0.080;
- ◆ **BevResol** = 1.

Не забудьте включить сглаживание (рис. 3.11)!

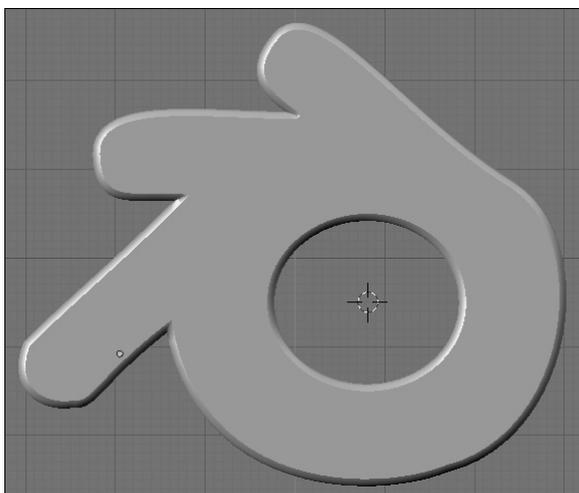


Рис. 3.11. Законченная модель логотипа Blender

Совет

Для просмотра окончательного результата используйте клавишу <F12>. Программа обработает сцену и выведет картинку в отдельном окне с параметрами по умолчанию. Рендер происходит всегда с точки обзора активной камеры.

3.3. Поверхности NURBS

Изучая кривые в предыдущем разделе, вы, наверное, задумывались о том, как с помощью их можно создавать объемные модели. Ведь операция *Extrude* производит только выдавливание формы конечного объекта по определенной оси, а редактирование самих сплайнов возможно лишь в двумерной плоскости. В этом и есть главное отличие кривых от поверхностей, которые предлагают полноценную работу с трехмерными координатами. Все поверхности основаны на сплайнах NURBS с добавлением второго измерения. Чтобы стало понятно, о чем идет речь, посмотрите на рис. 3.12.

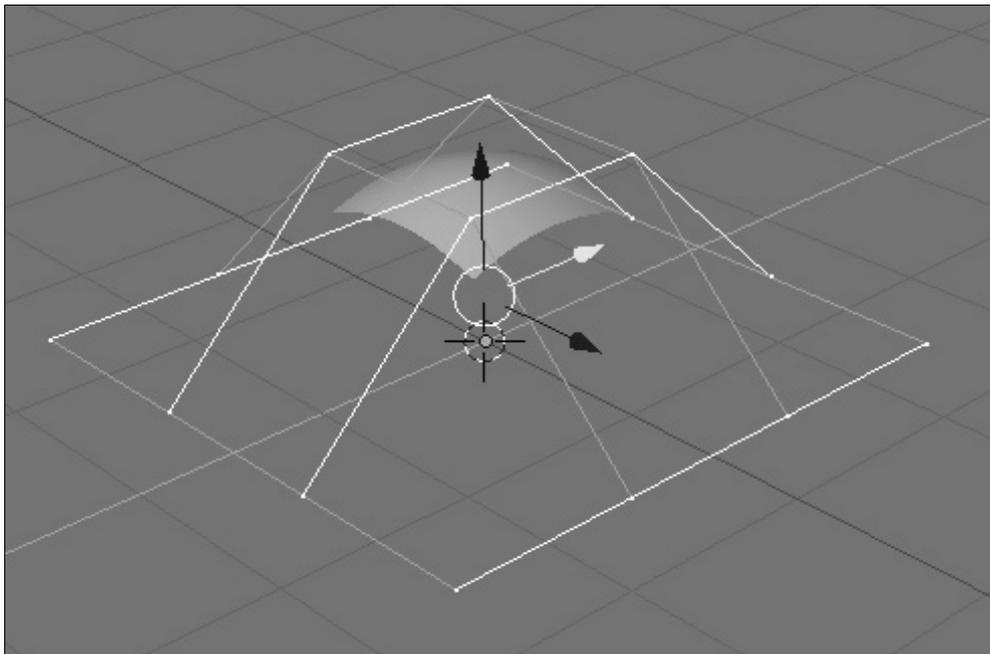


Рис. 3.12. NURBS Surface в режиме редактирования

Обычные кривые, будь то NURBS или Безье, имеют одно измерение, которое отмечается буквой U. В его пределах они могут располагаться, как заблагорассудится пользователю. Для поверхностей было введено второе измере-

ние — **V** (розовые линии в режиме редактирования), позволяющее размещать части объекта в дополнительных координатах. При этом сама поверхность остается двухмерным объектом, т. к. не имеет объема. Попробуйте создать два независимых объекта **Bezier Circle**, расположенных на разных уровнях по координате **Z** (рис. 3.13, а), и объединить их в единый объект командой **Join** (**<Ctrl>+<J>**). При этом произойдет совмещение обоих примитивов на одной плоскости по **Z** (рис. 3.13, б).

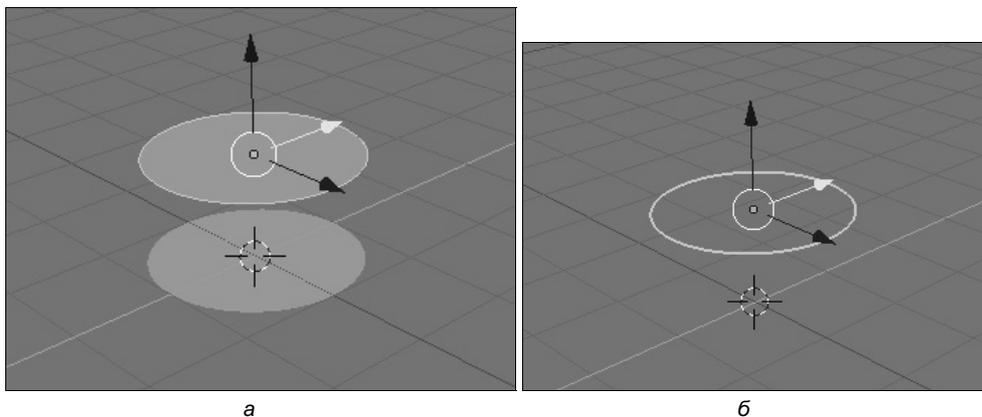


Рис. 3.13. Объекты **Bezier Circle** до слияния (а) и после (б)

Совет

Если по каким-либо причинам вам необходимо редактировать кривые в трехмерном пространстве, то включите кнопку **3D** на панели **Curve and Surface** текущего объекта. Но при этом не используйте функцию **Extrude** во избежание нежелательных искажений.

Работу с поверхностями рассмотрим на примере создания модели кувшина. Принцип моделирования предлагается следующий. Создается несколько примитивов **NURBS**, которые описывают форму будущего объекта. Затем они объединяются в единый объект со сплошной поверхностью.

В качестве базового примитива будем использовать **NURBS Circle** (Окружность) из меню **Add | Surface | NURBS Circle**. Для изменения формы объекта на вспомогательных рычагах имеются контрольные точки. Попробуйте переместить любую из них. Легко заметить, что они работают в трехмерном пространстве (рис. 3.14).

Обратите внимание на то, что вы не можете непосредственно управлять самой поверхностью, только лишь с помощью рычагов. По отношению к ним имеется возможность добавления новой контрольной точки. Для этого нужно

просто выделить две крайние вершины и командой **Specials | Subdivide** (<W>) добавить новую между ними.

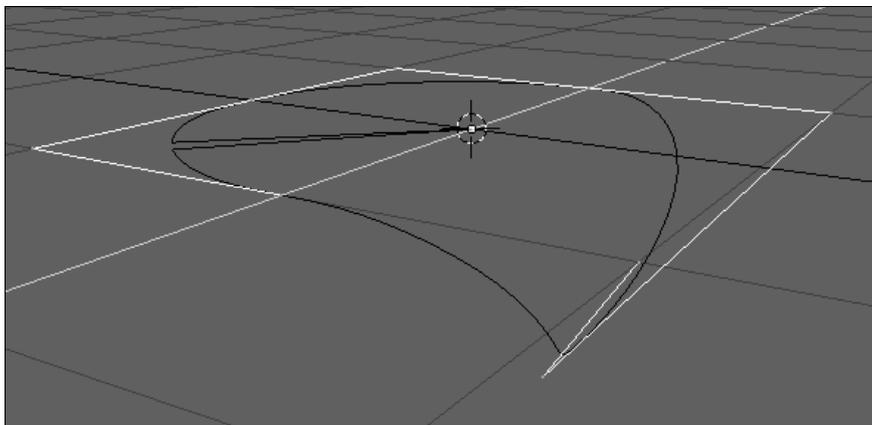


Рис. 3.14. Кривая NURBS в трехмерном исполнении

Создайте объект NURBS Circle и командой <Shift>+<D> размножьте его, как на рис. 3.15.

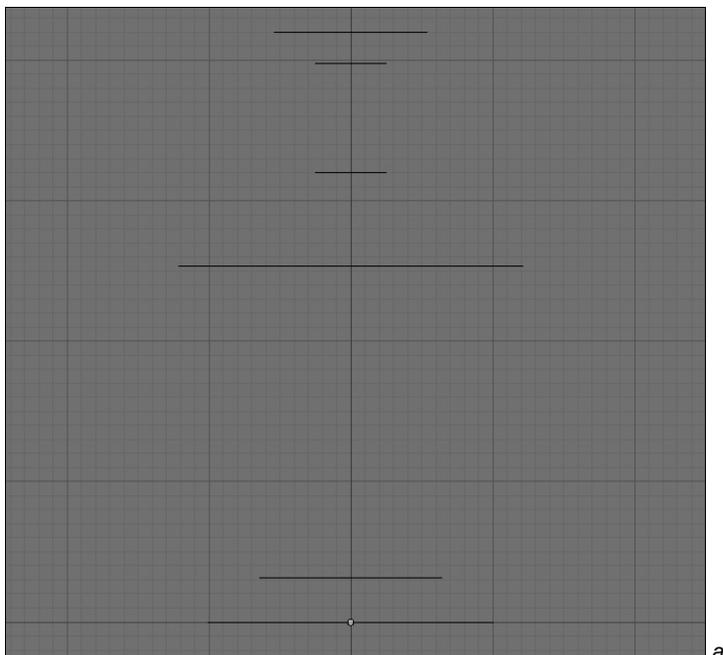


Рис. 3.15. Заготовка формы для модели кувшина: Front View (a)

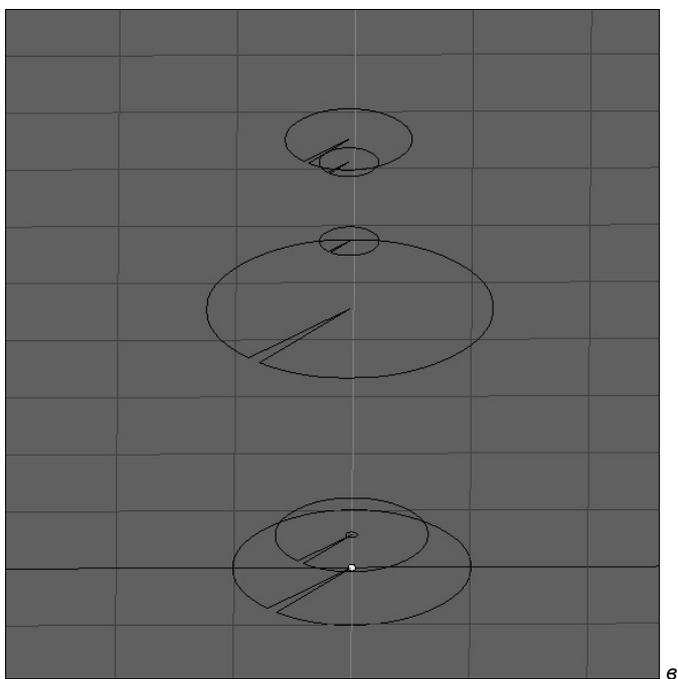
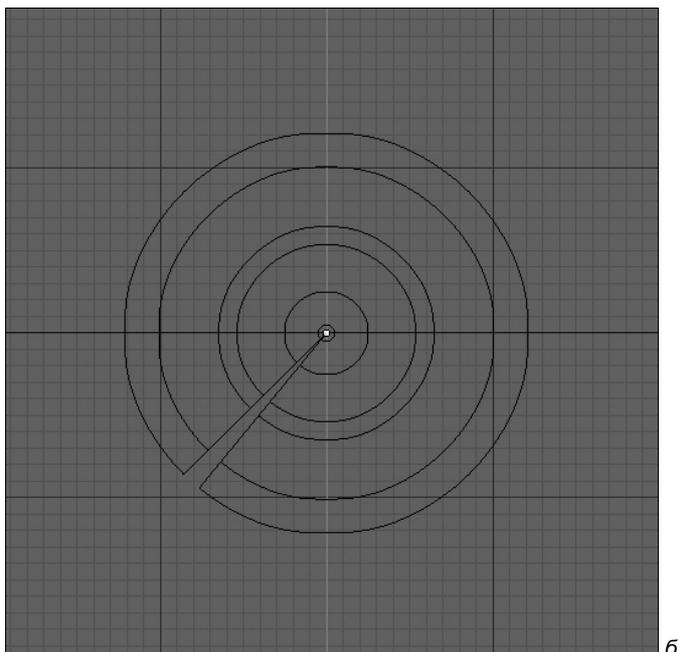


Рис. 3.15. Заготовка формы для модели кувшина: Top View (б), User View (в)

Отметьте все объекты с помощью клавиши <Shift> или рамкой выделения (). Произведите слияние командой Join (<Ctrl>+<J>). Для создания поверхности нужно перейти в режим редактирования, выделить все элементы (<A>) и нажать клавишу <F> (меню **Surface | Make Segment**).

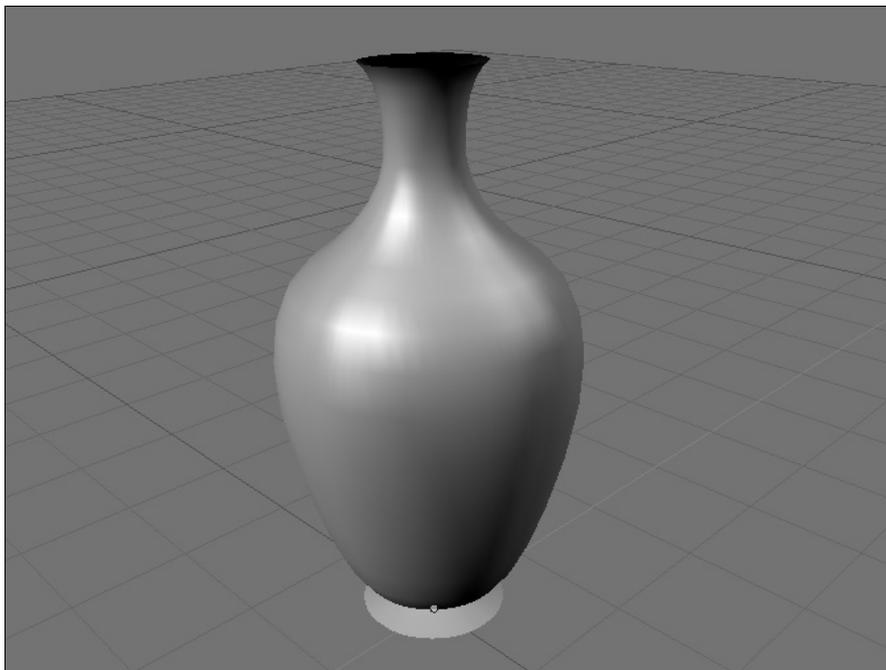


Рис. 3.16. Модель кувшина

3.4. Особенности работы с текстом

Программа Blender, как и любой другой мощный 3D-редактор, имеет возможность работы с текстом. Причем текст в понимании программы является обычным объектом, который может подвергаться манипуляциям, редактированию, модификации и многому другому. Несмотря на некоторые схожие черты работы с обычными текстовыми редакторами, встроенный модуль имеет свои специфические особенности и правила использования.

Этот текстовый объект находится в меню **Add | Text**. При его выборе программа создаст демонстрационный объект в окне просмотра с недвусмысленным содержанием — **Text**. Для работы с текстом служит стандартный режим редактирования, вызываемый клавишей <Tab>. При этом появляется характерный вертикальный курсор (рис. 3.17).

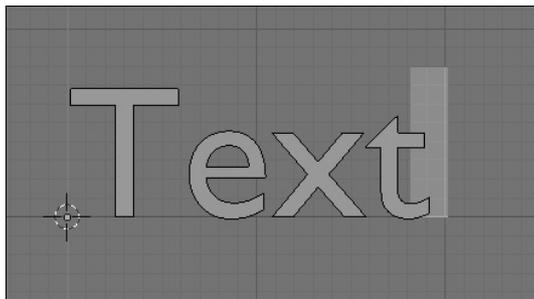


Рис. 3.17. Режим редактирования текста

Работа в этом режиме практически ничем не отличается от работы в том же Блокноте. Можно набирать или редактировать уже имеющийся текст, использовать привычные клавиши для копирования и вставки (<Ctrl>+<C>, <Ctrl>+<V>), выбирать нужные варианты шрифтов и т. д. Все необходимые функции расположены на панели **Font** (Шрифт) окна **Buttons Window** (рис. 3.18).

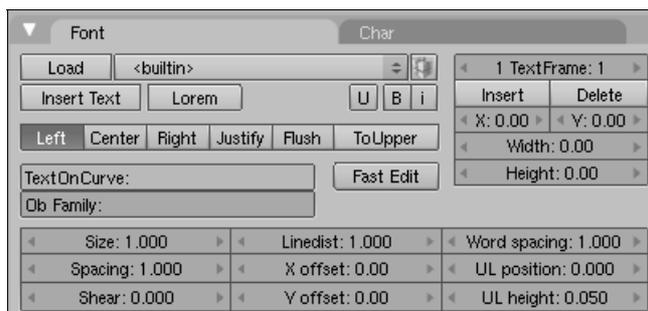


Рис. 3.18. Панель Font

Blender имеет встроенный шрифт, который по умолчанию используется для вывода начального текста. В отличие от обычных текстовых редакторов программа не умеет "на лету" работать с разными начертаниями шрифтов. Для каждого вида необходимо персонально указывать конкретный файл шрифта. Это кажется неудобным, если не принимать во внимание тот факт, что Blender — программа кроссплатформенная и ориентирована на несколько операционных систем. Независимо от того, какую ОС использует пользователь, принцип работы со шрифтами всегда один.

Для добавления и управления загруженными шрифтами служит кнопка **Load** (Загрузить) и расположенное рядом с ней выпадающее меню **Fonts** (рис. 3.19). При нажатии этой кнопки откроется окно файлового браузера, где можно выбрать желаемый файл. Не стоит надеяться, что программа предос-

тывает список имеющихся в системе шрифтов. Придется вручную указывать путь к папке, их содержащей. Это опять-таки связано с тем, что каждая ОС хранит свои шрифты в разных директориях. Так, для Linux характерны следующие директории:

- ◆ `usr/lib/X11/fonts;`
- ◆ `usr/local/share.`

Для семейства Windows используется путь `Windows\Fonts`. Ничто не мешает указать произвольную директорию, содержащую незарегистрированные в системе шрифты — на работе Blender это никак не отразится. К сожалению, файлы придется загружать по одному. Как бы то ни было, после загрузки нужных шрифтов необходимо присвоить их конкретным клавишам стиля.

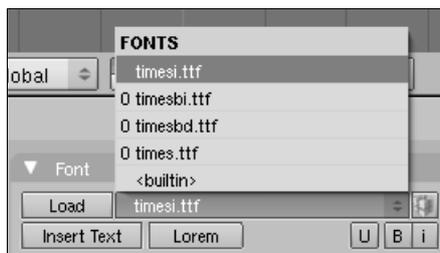


Рис. 3.19. Элементы для управления шрифтами

На рис. 3.19 в качестве демонстрации выбраны стандартные шрифты Times New Roman разного начертания:

- ◆ `times.ttf` — обычный шрифт;
- ◆ `timesi.ttf` — курсивный шрифт;
- ◆ `timesbd.ttf` — полужирный шрифт;
- ◆ `timesbi.ttf` — полужирный курсивный шрифт.

Операция присвоения выполняется путем нажатия кнопки **В** или **і** и выбора в меню нужного шрифта. В случае с `timesbi.ttf` должны быть нажаты обе эти кнопки. Для подчеркивания (**U**) в присвоении нет необходимости. Стоит заметить, что для каждого отдельного объекта `Text` можно назначить свои шрифты.

Совет

Blender безукоризненно работает с шрифтами латинского алфавита, чего не скажешь про кириллицу. Решить эту проблему можно двумя способами. Энтузиастами созданы модифицированные шрифты под Blender, которые при желании можно найти в Интернете. Второй вариант — использовать таблицу Unicode, доступ к которой открывается после выбора режима редактирования

в дополнительной вкладке **Char**. Для начала нужно загрузить подходящий шрифт с кириллицей, наподобие того же Times New Roman, и выбрать пункт **Cyrillic** в меню кнопки **Unicode Table**.

Основная работа с текстом происходит в режиме редактирования. Можно просто вводить текст, использовать буфер или вставлять из отдельного текстового файла. В последнем случае для загрузки файла служит кнопка **Insert Text** (Вставить текст). Программа позволяет произвольно менять начертание и шрифт для конкретного слова или буквы в тексте. Естественно, их предварительно нужно выделить. Для выделения можно использовать только клавиатуру. Работа с ней практически ничем не отличается от работы в простом текстовом редакторе. Необходимо переместить курсорными клавишами указатель на нужную позицию и, удерживая нажатой клавишу **<Shift>**, производить выделение текста. Также можно использовать следующие клавиши для редактирования текста:

- ◆ **<Ctrl>+<курсорные клавиши>** — перемещение в нужном направлении по словам;
- ◆ **<Ctrl>+<Shift>+<курсорные клавиши>** — выделение по словам;
- ◆ **<Home>** и **<End>** — перемещение курсора в конец или начало строки;
- ◆ **<PgUp>** и **<PgDn>** — перемещение курсора в начало или конец текста;
- ◆ использование **<Shift>** совместно с **<Home>**, **<End>**, **<PgUp>** или **<PgDn>** приведет к выделению соответствующего фрагмента текста.

Некоторые функции являются действительными только для всего массива текста и не реагируют на существующее выделение. К таким относятся различные выравнивания по имеющимся границам и функция **ToUpper**. Последняя позволяет изменять прописные буквы на строчные и наоборот.

Blender имеет возможность использования так называемых текстовых рамок. Пользователь может создать в пределах одного объекта несколько таких рамок с уникальными координатами и размерами. Данная функция годится для создания колонок.

По умолчанию при создании объект **Text** уже имеет текстовую рамку, которая просто невидима для пользователя. Для управления границами необходимо перейти в режим редактирования.

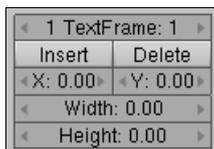


Рис. 3.20. Панель управления рамками

Создайте новый текст с содержанием "Frame I" и введите следующие значения в поля (рис. 3.20):

- ◆ **Width** (Ширина) — 4.00;
- ◆ **Height** (Высота) — 1.50.

Эти параметры отвечают за размеры активной рамки (рис. 3.21).

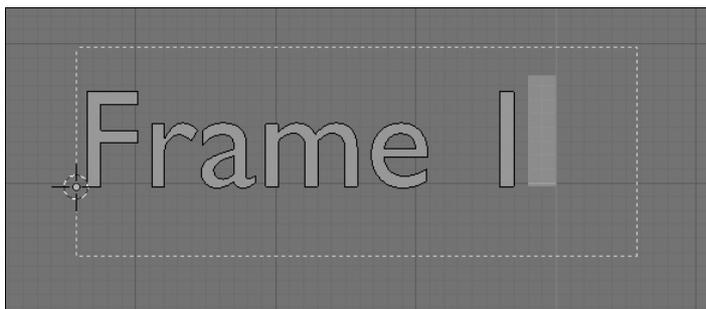


Рис. 3.21. Пример текстовой рамки

Для добавления новой рамки имеется кнопка **Insert** (Вставить). Расположенная рядом кнопка **Delete** удаляет активную область. Добавьте вторую рамку и введите следующие значения в поля панели:

- ◆ **X** — 3.20;
- ◆ **Y** — -2.00.

Если теперь продолжать ввод текста, то он будет располагаться в следующей области. Таким образом, текстовые рамки в программе являются сквозными. Любые операции выравнивания действительны для всех рамок объекта (рис. 3.22).

В отличие от обычных тестовых редакторов, Blender имеет более широкие возможности по управлению расположением букв. Все эти функции находятся в нижней части панели **Font** (рис. 3.23).

Пользователь может изменять расстояние между буквами, словами, строками. Управлять наклоном или подчеркиванием. Любые изменения в этой части панели действительны для всего объекта. Причем пользоваться ими можно как при редактировании, так и в основном режиме просмотра.

Описание функций управления текстом:

- ◆ **Size**. Глобальное изменение размера букв. Максимальное значение не больше 10 единиц. Программа очень аккуратно подходит к начертанию шрифта как при очень маленьком размере, так и при максимальном увеличении;

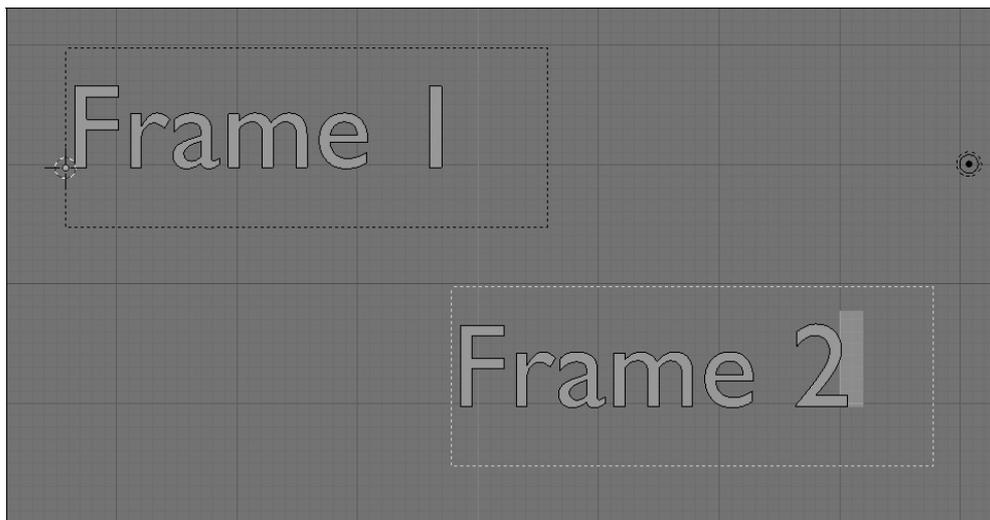


Рис. 3.22. Пример нескольких рамок

◀ Size: 1.000 ▶	◀ Linedist: 1.000 ▶	◀ Word spacing: 1.000 ▶
◀ Spacing: 1.000 ▶	◀ X offset: 0.00 ▶	◀ UL position: 0.000 ▶
◀ Shear: 0.000 ▶	◀ Y offset: 0.00 ▶	◀ UL height: 0.050 ▶

Рис. 3.23. Элементы для управления расположением букв

- ◆ **Linedist** (Line Distention). Расстояние между строками;
- ◆ **Word spacing**. Расстояние между словами или символами, разделенными пробелами;
- ◆ **Spacing**. Глобальное расстояние между символами;
- ◆ **Shear**. Управление наклоном букв. Причем имеется возможность изменения наклона в обоих направлениях;
- ◆ **X offset** и **Y offset**. Горизонтальное и вертикальное перемещение текста относительно его центра;
- ◆ **UL position**. Позиция вывода линии подчеркивания. Изменение происходит по вертикали. При соответствующей настройке можно добиться эффекта перечеркивания букв;
- ◆ **UL height**. Толщина линии подчеркивания.

Совет

Иногда бывает необходимо вертикальное расположение текста. Самый простой способ сделать это — воспользоваться рамкой. Настройте ее так, чтобы по горизонтали она была чуть больше, нежели первый символ. Теперь при наборе текста Blender будет вынужден переносить буквы на следующие строки.

3.5. Эффекты для текста

Несмотря на приличный встроенный редактор, Blender не имеет большой палитры спецэффектов для текста, но кое-что все же можно сделать. Так как объект `Text` основан на использовании техники сплайнов, то ему доступны некоторые функции, характерные для кривых. Все они расположены на панели **Curve and Surface** (см. разд. 3.2). Кроме того, имеется простая возможность создания фигурного текста, правда, только в двухмерном пространстве.

Создайте текстовый объект следующего содержания: Blender Foundation. Снимите выделение клавишей `<A>` и добавьте сплайн из меню **Add | Curve | Path**. Как уже говорилось ранее, этот объект является немного модифицированной версией обычной кривой и предназначен, в первую очередь, для вспомогательных целей. Впрочем, ничто не мешает использовать любой другой объект из этой группы.

Именно сплайн будет служить той формой, которую примет текстовый объект. Для этого необходимо подключить его к надписи. Обратите внимание на панель **Font** текстового объекта (см. рис. 3.18). Чуть ниже кнопок выравнивания расположено поле **TextOnCurve** (Текст по кривой). Введите здесь имя вспомогательного сплайна. Если все было правильно сделано — надпись сместится в центр кривой. При этом ее содержание сильно исказится. Чтобы избавиться от ненужного искажения, необходимо растянуть сплайн по размеру текста. Теперь вы можете приступить к приданию желаемой формы. Для этого в режиме редактирования кривой измените ее форму нужным образом (рис. 3.24).

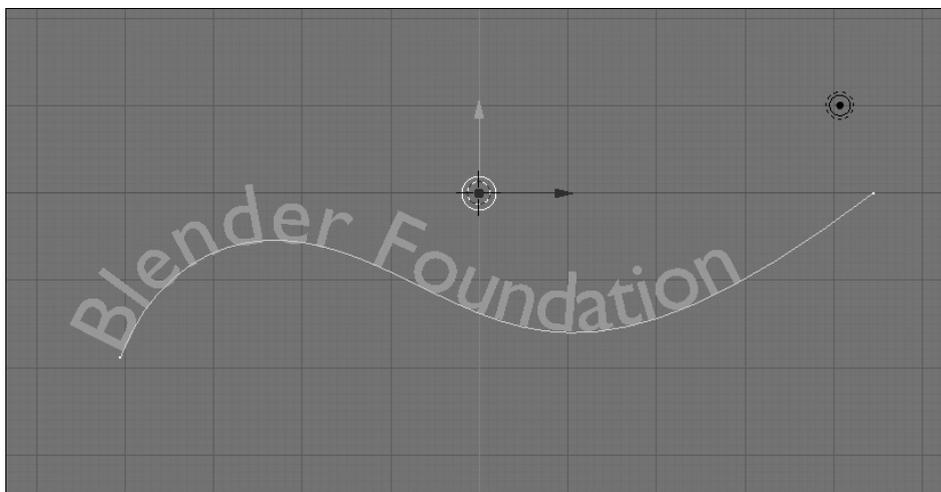


Рис. 3.24. Изменение формы текста с помощью Path

Полученный результат хорош, но что же делать, если необходимо трехмерное искажение? Существует небольшая хитрость, которую мы сейчас рассмотрим. В наборе модификаторов программы Blender имеется один с названием Curve. Назначение его в привязке формы объекта к вспомогательной кривой. Однако с форматом объекта Text он работает только в двухмерном пространстве. Но если его переконвертировать в Mesh или полноценный сплайн, то проблема будет решена.

Откройте новый проект и создайте текстовый объект с тем же самым содержанием, что и ранее. Добавьте кривую Path. Теперь выделите слово Text и нажмите клавиши <Alt>+<C>. Программа предложит выбрать формат для конвертирования. Выберите **Curve**.

Добавьте к объекту модификатор Curve и в поле **Ob:** введите имя кривой (рис. 3.25).

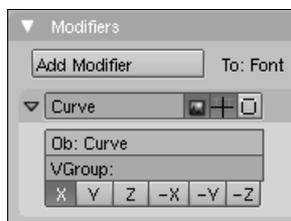


Рис. 3.25. Панель модификатора Curve

Здесь также можно указать, по какой оси производить деформацию объекта. Нам необходимо по X. Надпись должна принять форму кривой, однако с такими же искажениями, что и в предыдущем примере. Избавиться можно, если растянуть Path до нужного размера и передвинуть надпись в соответствии с центром кривой. Подготовительный этап закончен. Теперь вы можете крутить кривую как заблагорассудится (рис. 3.26).



Рис. 3.26. Деформация объекта с помощью Curve

Совет

Программа каждому создаваемому объекту автоматически присваивает имя, которое состоит из названия и числового номера по количеству имеющихся подобных в сцене. Ничто не мешает изменить это имя на нечто более подходящее. Для этого нужно отредактировать поле **Ob**: панели **Link and Materials**.

Рассмотрим еще один интересный эффект, напоминающий начертание логотипа известной американской киностудии. Заключается он в вытягивании задней стороны букв так, чтобы они постепенно сходились в центре надписи.

Удалите из предыдущего проекта кривую и модификатор *Curve*. Пусть останется одна лишь надпись. Теперь увеличьте параметр **Extrude** в панели **Curve and Surface** до 1 и переконвертируйте полученный объект в *Mesh* (<Alt>+<C>). Дело в том, что необходимый нам модификатор *SimpleDeform* (Простое деформирование) корректно работает только с *Mesh*.

Выберите в меню модификатора пункт **Taper** и измените степень сжатия в параметре **Factor** (Фактор) на единицу. Получившейся перекосяк в одну сторону объясняется тем, что модификатор стягивает полигоны к центру объекта, который в данный момент находится сбоку. Для автоматического расчета нового центра объекта нажмите кнопку **Center New** в панели **Mesh**.

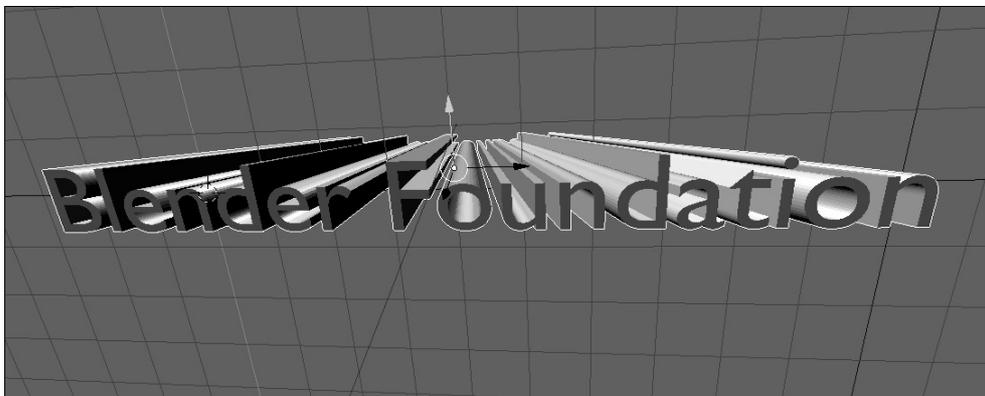


Рис. 3.27. Результат работы модификатора *SimpleDeform*

3.6. Дополнительный инструментарий

Предположим, вам поручили создать модель обыкновенного водяного шланга. На протяжении двух глав немало рассказывалось об объектах и способах моделирования, но задача эта отнюдь не простая. Можно, конечно, воспользоваться *Mesh*-объектом *Tube*, сделать на основе его заготовки разной формы и

с помощью модификатора `Array` размножить их до нужного количества. А потом просто объединить все в единый объект. Вот только полученная модель окажется монолитной и последующему редактированию с трудом поддается. Да и добиться от нее идеальных округлых форм на изгибах будет весьма сложно. На самом деле решить эту задачу очень легко, если воспользоваться сплайнами и предлагаемым редактором способом выдавливания объекта по пути. Исключительной особенностью такого моделирования является легкость изменения формы конечного результата, ведь в его основе лежит привычный сплайн.

Для моделирования этим способом понадобятся два примитива — заготовка, имеющая желаемую форму, и кривая, по которой эта форма будет выдавливаться. Причем при редактировании начального образца в реальном времени будет меняться и конечная модель.

Создайте объекты `Bezier Circle` и `Path` (рис. 3.28).

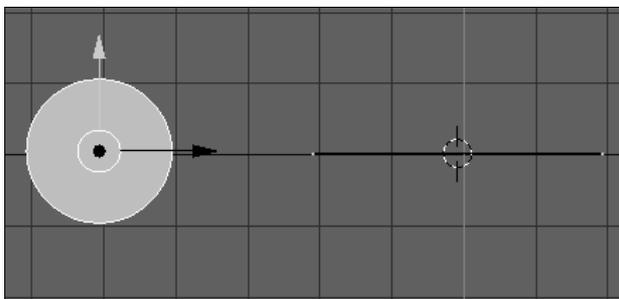


Рис. 3.28. Заготовки для выдавливания с `Curve Bevel`

В качестве траектории для выдавливания был выбран объект `Path` в силу его наибольшей приспособленности к подобным задачам, но вы можете использовать любой другой тип кривой.

Придайте произвольную форму пути. Для привязки заготовки к пути есть специальный инструмент `Curve Bevel`, имеющий поле **BevOb** на панели **Curve and Surface** (см. рис. 3.10). Выделите объект `Path` и введите имя объекта `Circle` в это поле.

После этого действия программа создаст сплошной объект, основанный на формах двух начальных примитивов. Попробуйте изменить форму заготовки `Circle` и вы увидите, как послушно изменяется основная модель. Точно так же, с легкостью, можно создать сколь угодно сложные переплетения для модели шланга (рис. 3.29).

Рассмотренный инструмент действительно очень удобен для выдавливания формы по указанной траектории, вот только полученный объект имеет оди-

наковую ширину по всей своей длине. В некоторых случаях бывает необходимо изменить в определенных местах объем модели, предположим, при создании щупальца осьминога. Для произвольного изменения формы и ширины предназначен инструмент *Curve Taper* (не путать с одноименным модификатором!). Пользоваться им так же просто, как и предыдущим.

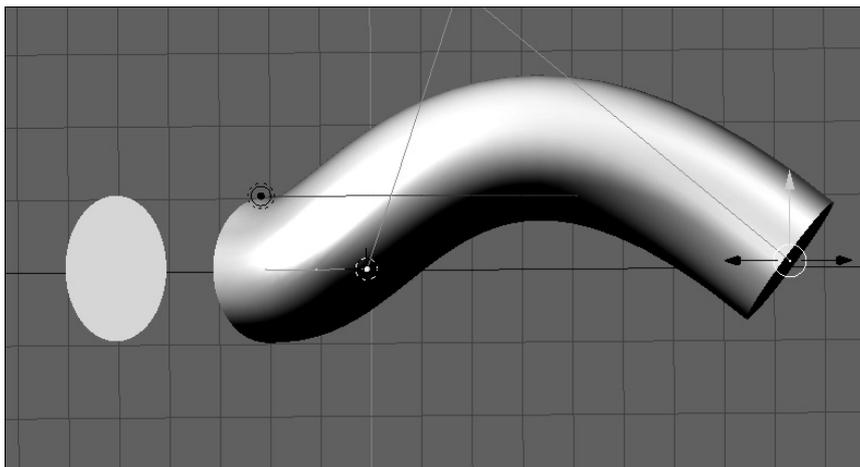


Рис. 3.29. Модель шланга

Создайте вспомогательную кривую чуть ниже основного объекта и назовите ее *Taper* (рис. 3.30).

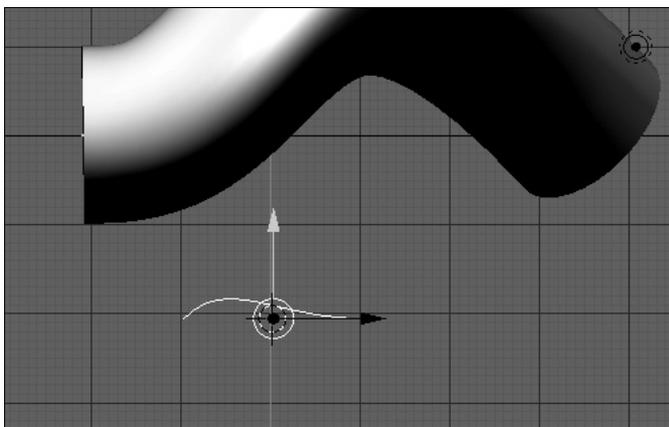


Рис. 3.30. Кривая Taper

Теперь введите имя вспомогательной кривой в поле **TaperOb**. Принцип работы заключается в управлении ее вершинами или рычагами, влияя тем самым

на основной объект. Так, приближая любую из точек к центру кривой, вы получите уменьшение объема в этой области. Естественно, при необходимости более детального изменения формы объекта можно добавлять контрольные точки на вспомогательном сплайне. Для создания щупальца просто немного приподнимите левую точку (рис. 3.31).

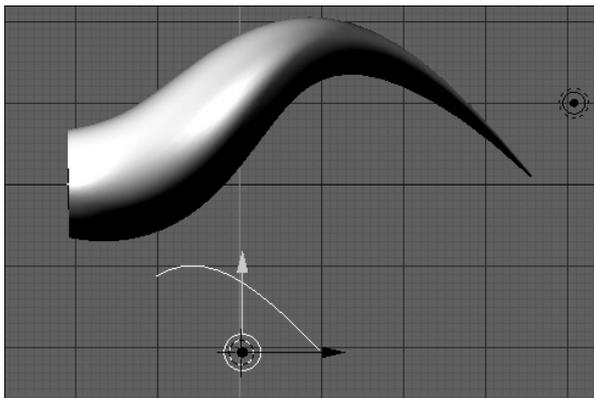


Рис. 3.31. Результат работы инструмента Curve Taper

Правила использования Curve Taper:

- ◆ формирование объекта идет слева направо;
- ◆ не допускайте перекрещивания на вспомогательной кривой.

3.7. Свойства отображения объектов

При работе с несколькими объектами зачастую приходится прокручивать сцену так, чтобы добраться до нужного элемента. Как правило, проблема эта решается путем переключения в подходящий режим отображения, такой как **Wireframe**. Однако имеется и другой способ, более удобный. Blender позволяет настроить определенные свойства отображения каждого из объектов. Так как эти параметры общие для всех видов примитивов, логично рассказать об этом здесь, в заключении глав об основах моделирования.

Основная особенность этих настроек в том, что они независимы от текущего режима отображения сцены. Для управления ими нажмите кнопку **Object** или горячую клавишу <F7>. Найти эти настройки можно на панели **Draw**. Для удобства использования элементы интерфейса разбиты на три глобальные группы (рис. 3.32):

- ◆ **Layers** (Слои). Дополнительная панель для управления привязки объекта к определенному слою;

- ◆ **Drawtype** (Способ прорисовки). Основной способ отображения объекта;
- ◆ **Draw Extra** (Дополнительная прорисовка). Вторичные параметры, которые присоединяются к выбранным в **Drawtype**.

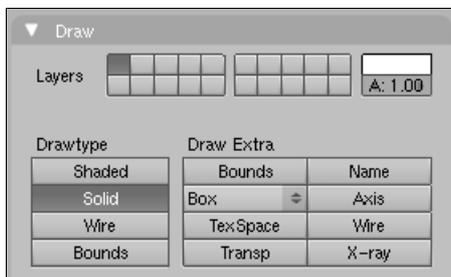


Рис. 3.32. Панель Draw

Кнопки области **Drawtype**:

- ◆ **Shaded** (Затененный);
- ◆ **Solid** (Сплошной);
- ◆ **Wire** (Каркас);
- ◆ **Bounds** (Границы). Для выбора бокса, наиболее подходящего объекту, в области **Draw Extra** имеется меню с видами примитивов.

Кнопки области **Draw Extra**:

- ◆ **Bounds** (Границы);
- ◆ **TexSpace** (Текстурное пространство);
- ◆ **Transp** (Прозрачность). Включение прозрачности для материалов, если они имеют соответствующие настройки (см. разд. 4.2);
- ◆ **Name** (Имя). Отображение рядом с объектом его имени;
- ◆ **Axis** (Оси). Показываются центр и оси объекта;
- ◆ **Wire** (Каркас);
- ◆ **X-ray** (Просвечивание). При включении этой кнопки объект всегда будет виден, даже если он заслонен другими объектами.

ГЛАВА 4



Материалы и текстуры

Оглянитесь вокруг — мир, окружающий нас, богат не только разнообразными формами живого и неживого, но и буйным обилием красок. Перенести все это в бездушную машину очень нелегко, а вот приблизиться к оригиналу вполне даже возможно. Для создания реалистичного трехмерного изображения важно научиться правильно работать с материалами, знать их структуру и возможности. В этой главе вы познакомитесь с понятием материала, его главными особенностями, научитесь работать с обычными и процедурными текстурами, а также с картами рельефа и смещения. Особое внимание уделяется работе с разверткой текстуры.

4.1. Что такое материал

В мире трехмерных редакторов часто используется понятие *материала*. Возьмите в руку обычный карандаш и внимательно рассмотрите его. На протяжении первых трех глав разговор шел исключительно о способах создания формы предмета. Но для реалистичного отображения оригинала этого мало. Карандаш, помимо формы, имеет много отличительных свойств, таких как цвет, царапины на поверхности, небольшие неровности окраски, надписи производителя и т. д. Все это в совокупности, влияя на наше восприятие, уверяет нас, что это действительно карандаш. И если провести аналогию настоящего предмета с его трехмерным двойником, то материалом можно назвать те визуальные свойства, которые отвечают за его правильное восприятие.

Любой объект в сцене может иметь уникальный материал, который, в свою очередь, состоит из нескольких слоев, определяемых пользователем. Рассмотрим, к примеру, структуру материала трехмерного карандаша. Во-первых, его деревянная основа имеет свой характерный цвет и рисунок. Да-

лее следует слой краски, сквозь которую может просвечиваться основа. Третий слой содержит потертости и царапины краски. Ну а следующий может нести какую-либо текстовую информацию, например марку карандаша.

По умолчанию начальный примитив уже имеет свой материал серого цвета, обладающий некоторыми свойствами, такими, к примеру, как рассеянное отражение от имеющегося источника света. Подобное "своевольничество" программы весьма продуманно, т. к. дает основу для настройки материала "под себя" и правильного отображения модели в сцене со светотенями.

Blender умеет прорисовывать сцену с разным качеством изображения. По умолчанию включен режим *Solid*, но ничто не мешает вам поменять его на более подходящий (см. разд. 1.2). При этом не забывайте, что максимального качества просмотра можно добиться лишь после обработки (рендера) сцены. Однако вначале давайте разберемся, что же все-таки представляет собой материал объекта.

4.2. Базовый цвет

Для работы с материалами существует особый набор панелей, которые можно вызывать нажатием клавиши <F5> или соответствующей кнопки заголовка окна **Buttons Window** (рис. 4.1).



Рис. 4.1. Основные и вспомогательные кнопки для работы с материалом

При этом раскрывается дополнительная панель со вспомогательными кнопками, где можно настроить освещение, глобальные параметры сцены, выбрать текстуры. Однако давайте остановимся пока именно на функциях материалов.

Любой вновь создаваемый объект не имеет присвоенного материала, хотя и окрашен по умолчанию в серый цвет. Для выбора имеющегося или создания нового материала существует панель **Links and Pipeline** (Связи и конвейер). Достаточно нажать кнопку **Add New** (Добавить новый), расположенную на панели **Link and Pipeline**, и пустынное до этого окно заполнится большим количеством панелей. Равноценно можно выбрать уже имеющийся материал из списка, расположенного рядом (см. рис. 4.3).

Внимание!

Один материал может быть присвоен сразу нескольким объектам. При его редактировании нужно помнить, что любые изменения проявятся во всех родственных объектах.

Панель **Links and Pipeline** содержит глобальные параметры материала. Оно разбито на две области:

- ◆ **Link to Object** (Привязать к объекту). Создание, присвоение или удаление материала объекта, управление мультиматериалами (см. разд. 4.5);
- ◆ **Render Pipeline** (Обработка конвейера). Настройки для обработки конечного результата.

Любой материал имеет основной сплошной цвет (Solid), который можно настроить в панели **Material** (рис. 4.2).



Рис. 4.2. Панель **Material**

Но не все так просто! На самом деле основной цвет объекта складывается из трех составляющих, которые рассчитываются в зависимости от положения камеры и источников света:

- ◆ **Diffuse Color** (Цвет диффузии) — кнопка **Col** панели **Material**. В основе его лежит явление диффузии, когда падающий на поверхность свет равномерно отражается во всех направлениях. Благодаря этому камера может видеть отраженный свет независимо от точки просмотра. Запомните, цвет диффузии является основным, который замечает глаз;
- ◆ **Specular Color** (Бликовое отражение) — кнопка **Spe**. Эффект, обратный явлению диффузии. Зеркальное отражение зависит от точки просмотра. С помощью его можно добиться отблесков, которые появляются на глянцево-й поверхности. Как правило, его следует ставить чисто белым;
- ◆ **Mirror Color** (Цвет зеркала) — кнопка **Mir**. Этот цвет используется для расчета полного отражения, как в случае с реальным зеркалом.

Подобные расчеты являются довольно сложными и обычно выполняются с применением шейдеров. Причем алгоритмов выполнения обоих вариантов имеется несколько (см. разд. 4.3).

Для лучшего понимания особенностей настройки цветов рассмотрим следующий пример. Создайте два объекта `Sphere` (**Add | Mesh | UVsphere**) и

присвойте им *разные* материалы. Пусть один примитив будет иметь цвет Diffuse синего оттенка, а второй — синий Specular. Как и в случае с объектами, имеется возможность сменить имя материала на более подходящее (рис. 4.3).

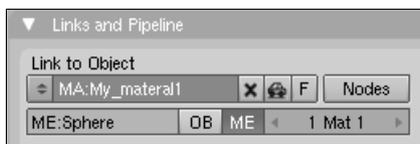


Рис. 4.3. Название материала можно с легкостью изменить

Программа предлагает несколько способов изменения цвета. Самое простое — это воспользоваться специальным окном, которое появляется, если щелкнуть левой кнопкой мыши на образце цвета соответствующей составляющей, где первая полоска содержит Diffuse, вторая — Specular, а третья, соответственно, Mirror (рис. 4.4).

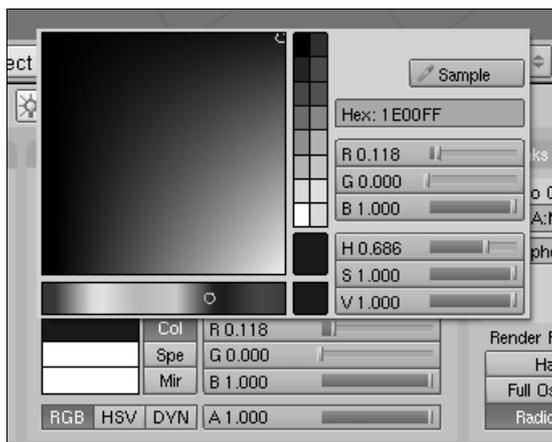


Рис. 4.4. Окно для интерактивного выбора цвета

Подобрать подходящий цвет можно в нижней палитре появившегося окна, а затем оттенок в его центре. Также имеется возможность использования инструмента *Sample* (Образец), который позволяет взять образец цвета с любой точки рабочего пространства программы. Просто нажмите кнопку **Sample** и щелкните в нужном месте экрана (рис. 4.5).



Рис. 4.5. Инструмент *Sample*

Для более точного выставления цвета можно использовать цифровые поля, расположенные рядом с кнопками **Col**, **Spe** и **Mir**. В этом случае нужно знать, что программа умеет работать с несколькими цветовыми форматами. По умолчанию включен RGB (Red, Green, Blue), при котором конечный цвет получается путем смешивания красной, зеленой и синей составляющей. Для использования другого типа, HSV (Hue, Saturation, Value), нужно просто нажать соответствующую кнопку (см. рис. 4.2). Кроме того, отдельно имеется параметр Alpha (кнопка **A**), который отвечает за прозрачность материала. Какой способ использовать — выбирайте сами.

Итак, для первого объекта параметры цвета должны быть следующие:

- ◆ Color — синий;
- ◆ Specular — белый;
- ◆ Mirror — белый.

Цвета второго объекта:

- ◆ Color — белый;
- ◆ Specular — синий;
- ◆ Mirror — белый.

Нажмите клавишу <F12> для обработки результата (рис. 4.6).

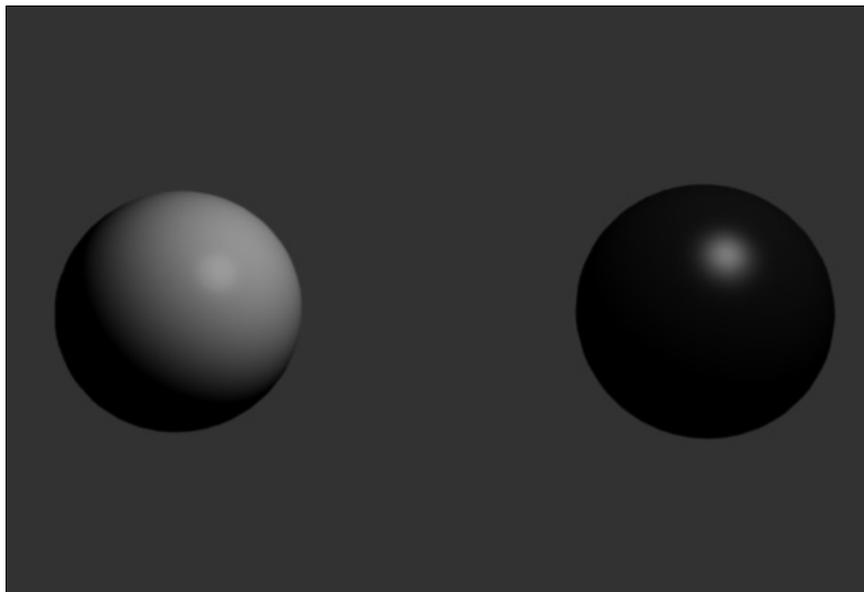


Рис. 4.6. Два одинаковых объекта с разными материалами

Обратите внимание на сферу слева. Отчетливо видны темные блики (на самом деле они синие) на светлом фоне. Именно так отразился свет при измененном параметре `Specular`. Для правильного отображения бликов на поверхности обычно оставляют параметр `Specular` без изменений, т. е. белым. Путем изменения его на иной можно добиться интересных результатов.

Для удобства работы с материалом программа Blender предлагает специальную панель **Preview** (Предварительный просмотр). Возможно, у вас возник вопрос о ее целесообразности, ведь программа отображает результат сразу в 3D-окне. В некоторых случаях эта возможность приходится очень даже кстати. Окно **3D View** для ускорения работы выводит изображение со значительным ухудшением качества. Можно, конечно, выбрать один из подходящих режимов прорисовки в меню **Drawtype** или окончательно обработать сцену. Однако при большом количестве объектов это сильно замедлит работу. Кроме того, иногда бывает нужно создать материал без применения его к текущему объекту (рис. 4.7).

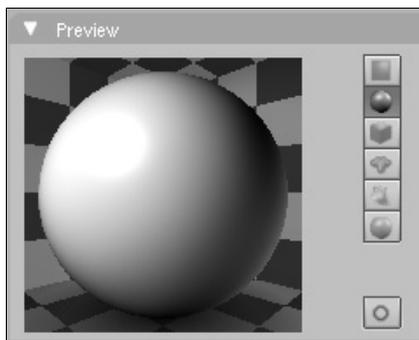


Рис. 4.7. Панель **Preview** для просмотра материала

Уникальным свойством предварительного просмотра является возможность выбора формы объекта для демонстрации материала. Просто нажмите кнопку с соответствующим изображением примитива.

Даже в этом случае картинка имеет несколько упрощенный вид. Для максимального соответствия окончательному результату можно последней кнопкой панели (см. рис. 4.7) включить режим сглаживания OSA (Oversampling).

Итак, благодаря материалам мир Blender заиграл яркими красками. Однако при обработке приведенного примера можно заметить, что полученный результат уж больно выглядит ненастоящим, игрушечным. Все правильно, ведь мы рассмотрели лишь самый первый слой в материале, а именно `Solid`. Но уже сейчас можно добиться значительных результатов, если поиграть с настройками шейдеров.

4.3. Встроенные шейдеры

При слове "шейдеры" у многих может возникнуть ассоциация с крутыми возможностями видеокарт. Отчасти это правильно, ведь обработка шейдеров производится в большей степени именно ими. Но начать следует все же с того, что шейдерами называют программы, написанные с помощью специальных шейдерных языков и обрабатываемые графическими процессорами. Обычно на долю шейдеров приходится вычисления, связанные со светом, преломлениями, иногда анимацией, в общем, те задачи, которые требуют больших вычислений. Ведь современные видеокарты умеют выполнять подобные операции гораздо быстрее любого центрального процессора. В случае с Blender разговор пойдет о так называемых встроенных шейдерах, связанных в основном с обработкой света и теней. Впрочем, программа умеет работать и с одним из популярных шейдерных языков — GLSL (OpenGL Language Shaders, язык программирования шейдеров OpenGL).

Встроенные шейдеры поделены на две большие группы — диффузные шейдеры и отражающие. Уже из названий понятно, какой из них на что влияет. Все функции управления вынесены в специальную панель **Shaders** (Шейдеры), где в верхнем меню можно выбрать диффузный, а в нижнем, соответственно, отражающий шейдер (рис. 4.8).

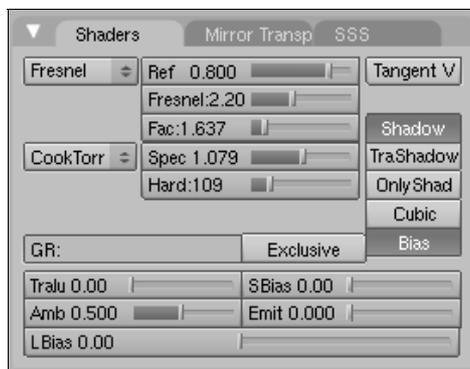


Рис. 4.8. Панель Shaders

Любой из этих типов имеет несколько алгоритмов вычисления, каждый из которых подходит для выполнения определенного круга задач. Рассмотрим сначала диффузные шейдеры.

По умолчанию при создании материала выбран алгоритм Lambert, названный в честь своего создателя (почти все используемые в программе шейдерные алгоритмы получили свои названия по имени авторов). Это самый простой и первый шейдер, появившийся в Blender. Его главная функция — вычисление

степени отражения диффузного света. Единственный параметр `Reflection` (Отражение) позволяет изменять отражающее свойство поверхности. При минимальном значении объект будет выглядеть абсолютно черным (рис. 4.9).

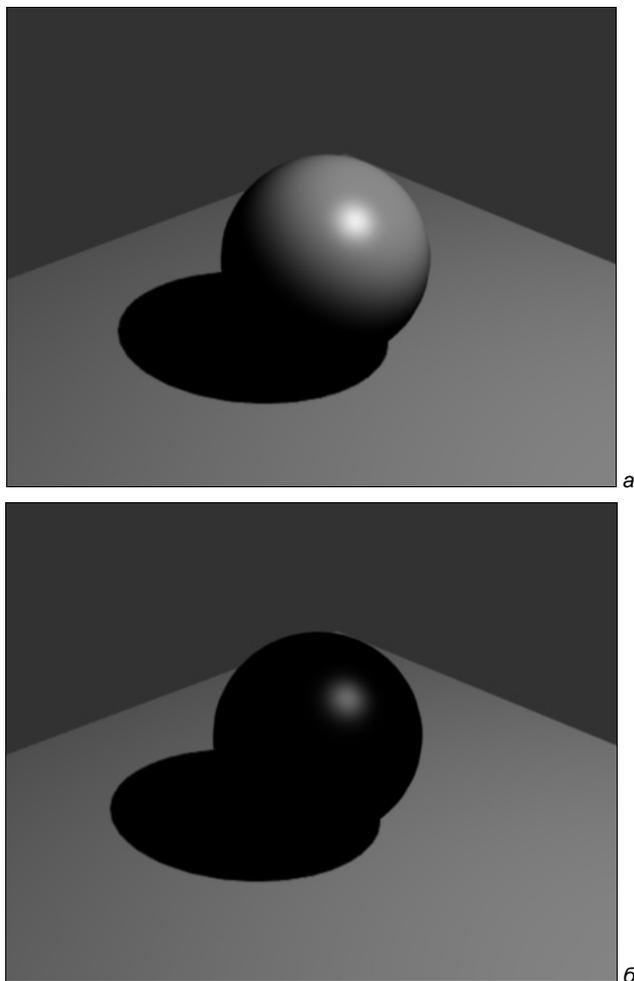


Рис. 4.9. Шейдер Lambert со стандартными настройками (а) и минимальными (б)

Этот пример отчетливо показывает разницу между диффузным (см. рис. 4.9, а) и отражающим (см. рис. 4.9, б) светом.

Следующий шейдер, Oren-Nayar, является логичным продолжением Lambert и имеет несколько более физический подход к просчету освещения. В дополнение к параметру `Reflection` есть свойство `Roughness` (Шероховатость), которое учитывает неровность поверхности при обработке.

Более практичным выглядит третий из рассматриваемых шейдеров — Minnaert. С его помощью можно добавить материалу объекта бархатистость. Особенностью работы его алгоритма является более четкое разделение на темные и светлые области объекта. Помимо уже стандартного свойства Reflection имеется новый параметр — Dark (Темнота). При уменьшении значения этого параметра произойдет осветление краев объекта, что и придаст необходимый эффект. Так, для объекта на рис. 4.10 параметр Reflection был установлен в значение 0.5, а Dark — 0.2.

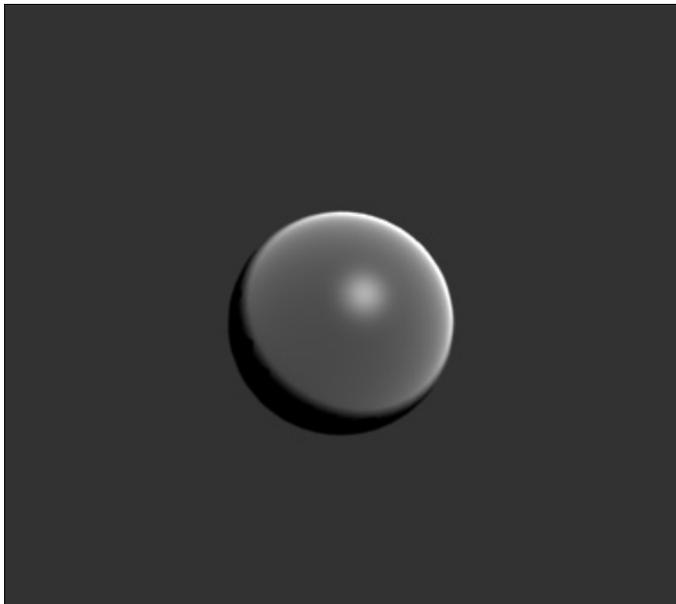


Рис. 4.10. Шейдер Minnaert может придать объекту бархатистость

Результат работы следующего шейдера, Fresnel, выглядит немного непривычно (рис. 4.11). С точки зрения физики и других шейдеров, сторона, обращенная к источнику света, будет выглядеть ярче остальных. Fresnel понимает это с точностью до наоборот. Благодаря этой уникальности можно добиться интересных эффектов. Здесь пользователя поджидают два новых параметра: Fresnel (Мощность эффекта) и Factor (Качество смешивания).

Попробуйте, к примеру, следующие параметры (рис. 4.11):

- ◆ Reflection — 0.8;
- ◆ Fresnel — 2.2;
- ◆ Factor — 1.4.



Рис. 4.11. Результат работы алгоритма Fresnel

Совет

Иногда при использовании некоторых шейдеров результат рендера абсолютно не похож на образец в окне предпросмотра. Чаще всего появляются артефакты в виде зубчиков на материале объекта. Просто нажмите кнопку **Bias** (Смещение) на панели **Shaders**, и искажения исчезнут.

Среди набора шейдеров, предназначенных для естественной имитации игры светотеней, имеется один, создающий эффект мультяшности. Происходит это благодаря четкому разделению областей света и теней, а также равномерно-сти их окрашивания. Называется этот шейдер Toon (рис. 4.12). Для настройки имеется два параметра:

- ◆ **Size** (Размер). Область света;
- ◆ **Smooth** (Сглаживание). Переход между тенями и светом.

На этом обзор диффузных шейдеров закончен, однако нас сейчас ожидают шейдеры отражающие. Для работы программа предлагает всего пять алгоритмов. Обратите внимание на то, что зеркальные шейдеры задумывались как дополнительные к уже имеющимся диффузным и, как правило, работают в паре с определенными из них. Впрочем, ничто не мешает комбинировать разные виды алгоритмов на свое усмотрение.

Первые два отражающих шейдера, CookTorr и Phong, логично рассматривать одновременно, т. к. они имеют одинаковые параметры и незначительно

различаются в результатах. Использование этих шейдеров в совокупности с диффузным Lambert позволяет добиться от материала эффекта пластика. Для регулировки имеются два параметра:

- ◆ *Specular* (Отражение). Интенсивность излучения. Общий параметр для шейдеров этой группы;
- ◆ *Hard* (Жесткость). Указание размера отблеска.

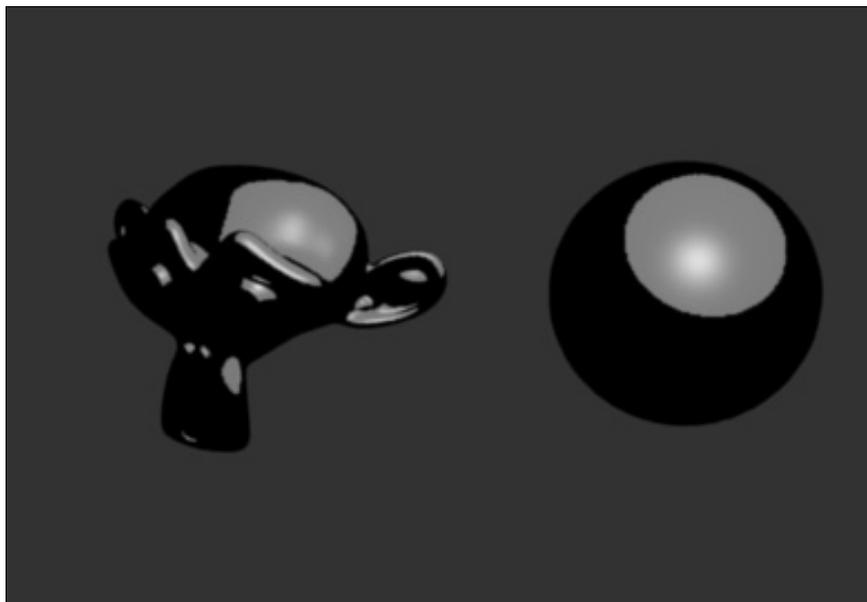


Рис. 4.12. Результат работы шейдера Toon, где *Size* = 0.7, а *Smooth* = 0.0

Следующий шейдер, *Blinn*, обычно используется в паре с *Oren-Nayar*. Обладает более продвинутой системой просчета физической модели. По сравнению с двумя предыдущими имеется дополнительный параметр *Refraction* (Преломление), который позволяет точнее просчитывать интенсивность отражения.

Wardiso — шейдер, наиболее подходящий для создания металлических поверхностей. Обладает достаточно жестким и ярким излучением. Регулировать область отражения можно параметром *RMS* (рис. 4.13).

Последний шейдер из этой группы, *Toon*, предназначен для работы с диффузным собратом *Toon* (рис. 4.14, а). Параметры обоих абсолютно одинаковые. Использование этой пары позволит достичь более правдоподобного эффекта двухмерного рисунка. Оптимально для отражающего шейдера выставлять параметры более мягкие, нежели для диффузного (рис. 4.14, б).

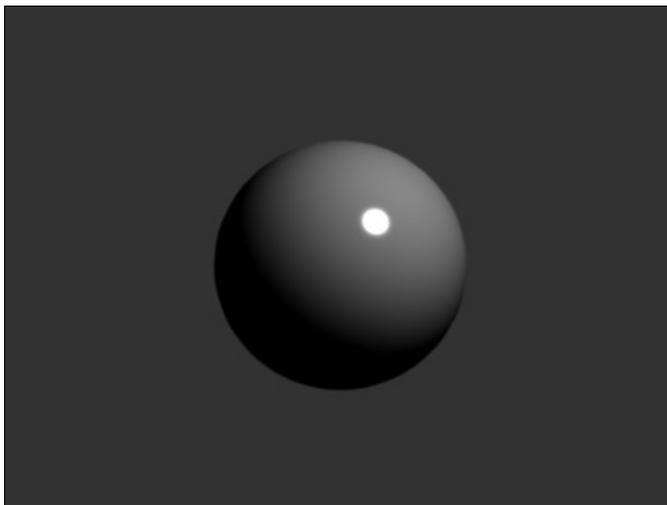


Рис. 4.13. Зеркальный шейдер Wardiso оптимально подходит для металла

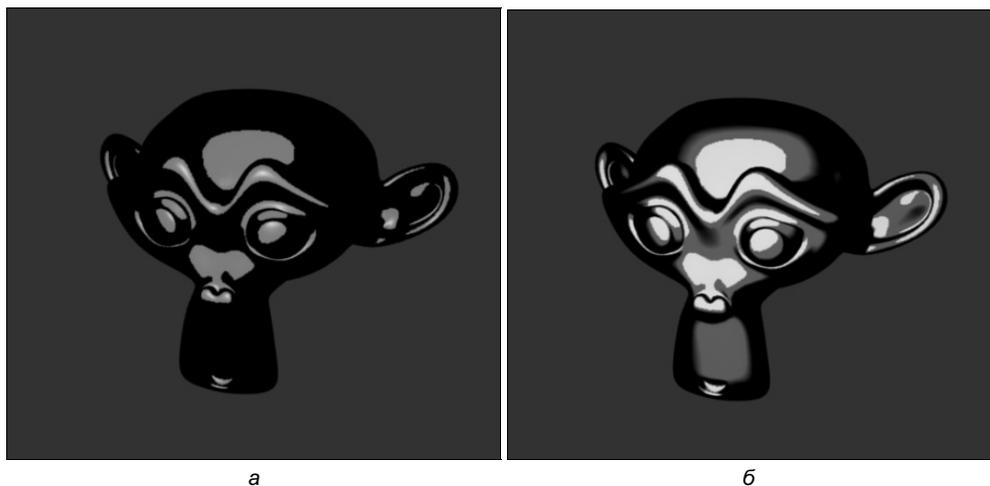


Рис. 4.14. Шейдер Toon без зеркального собрата (а) и вместе с ним (б)

4.4. Отражение и преломление

Рассмотренные в предыдущем разделе шейдеры позволяют настроить отражающие свойства материала, но для создания действительно прозрачных или зеркальных поверхностей их использовать нельзя. Для этой цели разработчиками предусмотрен метод трассировки лучей. Принцип работы достаточно

простой. Представьте, что от камеры исходит луч, который путешествует по сцене и при столкновении с объектом принимает цвет в зависимости от настроек его материала. Если встреченная поверхность имеет зеркальные свойства, то происходит отражение и блуждание луча по сцене до тех пор, пока не встретится неотражающий объект. В этом случае первый объект наследует все цвета окружающей среды.

Все функции управления трассировкой расположены на панели **Mirror Transp** (Прозрачность и зеркало) окна **Buttons Window** (рис. 4.15).



Рис. 4.15. Функции трассировщика

Эти функции поделены на две глобальные группы, которые активируются включением соответствующей кнопки:

- ◆ Ray Mirror (Зеркальный луч). Настройки отражения;
- ◆ Ray Transp (Прозрачный луч). Настройки прозрачности.

Естественно, их можно использовать и одновременно. Однако сам процесс просчета сцены с трассировкой лучей довольно трудоемкий и занимает немало времени. Поэтому используйте этот метод только при необходимости.

Рассмотрим сначала работу с зеркальными поверхностями. Создайте сцену, состоящую из четырех примитивов и персональных материалов для каждого. Пусть все объекты будут разного цвета (рис. 4.16).

Теперь измените все составляющие цвета материала объекта Plane на белый и включите кнопку **Ray Mirror** (Зеркальный луч). За любыми изменениями материала можно следить в окне **Preview**, не прибегая к обработке. В случае с трассировщиком это является несомненным удобством.

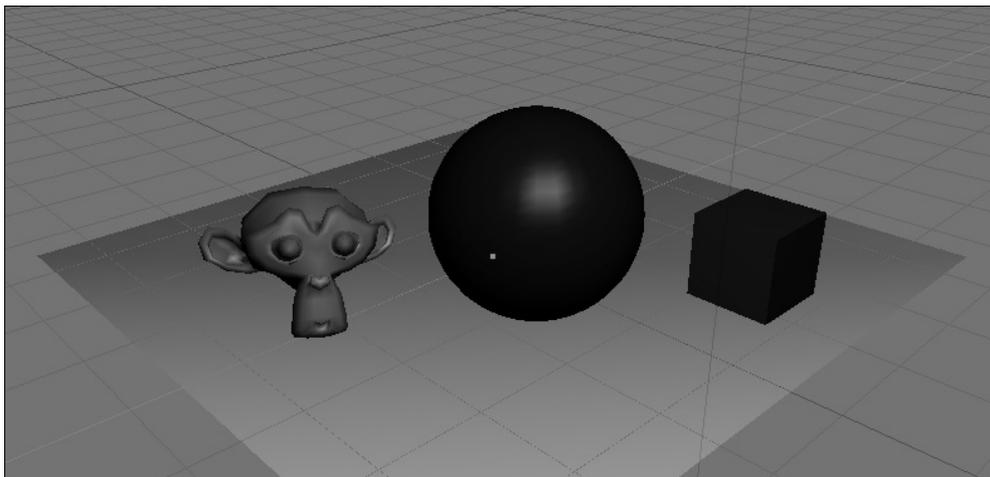


Рис. 4.16. Начальная сцена для Ray Mirror

Для управления чувствительностью отражения существует бегунок **RayMir**. В минимальном положении объект будет прорисовываться со своим цветом **Diffuse** и не сможет отражать другие элементы. Установите значение **RayMir** на 0.5 единиц и обработайте сцену (рис. 4.17).

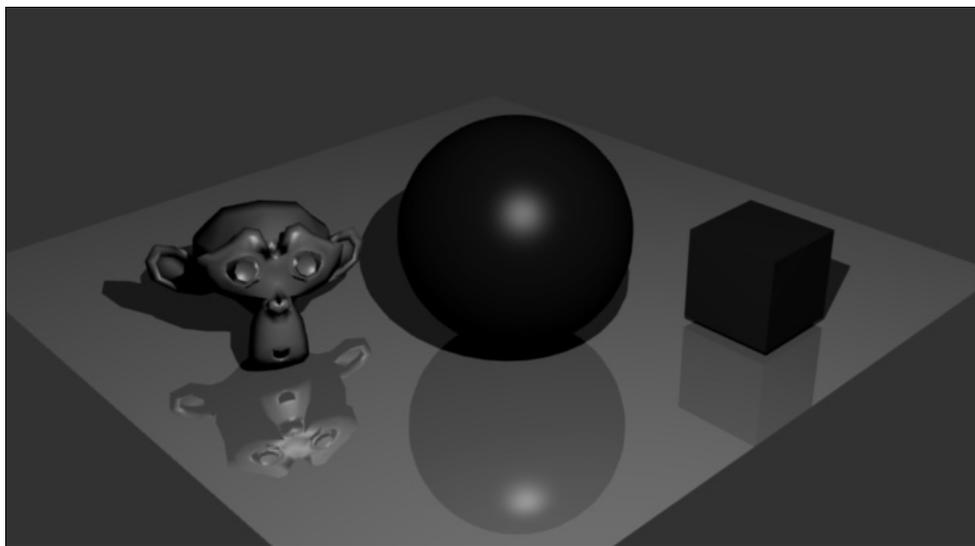


Рис. 4.17. Результат обработки с параметром **RayMir** = 0.5

Конечный цвет отражающей поверхности вычисляется на основе значений **Diffuse** и **Mirror**. Попробуйте установить максимальное значение параметра

RayMir и обработать сцену. Полученный результат будет иметь цвет **Mirror**. Этим свойством удобно пользоваться, чтобы придать отраженному объекту желаемый оттенок (рис. 4.18).

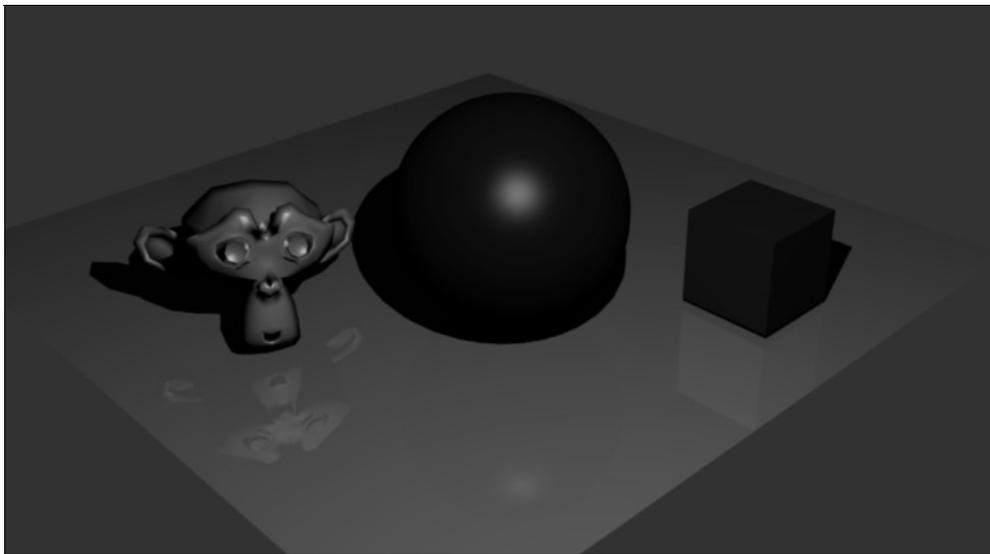


Рис. 4.18. Параметр **Mirror** влияет на конечный цвет отражения

Не всегда отражение в реальном мире столь четко копирует оригинал. К примеру, органическое стекло дает характерное матовое отражение, когда можно различить лишь контуры объекта и то приблизительно. Для создания такого эффекта можно воспользоваться функцией **Gloss** (Блеск). Установите значение 0.8 в свойства **Gloss** и **RayMir** и обработайте сцену (рис. 4.19).

Теперь настала пора разобраться с настройками прозрачности. Проведите небольшой эксперимент. Налейте воду в стакан и поместите туда ложку. Если посмотреть на ложку сбоку сквозь стакан, то видно будет искажение ее формы. Подобный эффект называется преломлением, а сила, с которой это происходит, — коэффициентом преломления. Каждый мало-мальски прозрачный объект имеет свой коэффициент преломления, и для реального его моделирования это значение необходимо учитывать.

Разместите объекты **Sphere** и **Suzanne** так, чтобы первый закрывал собою второй. Включите кнопку **Ray Transp** для сферы. За степень прозрачности отвечает параметр **Fresnel**. Установите для него значение 3.0 и обработайте (рис. 4.20).

В Интернете и специальной литературе можно найти коэффициенты практически для всех природных и искусственных материалов. Так, для льда

коэффициент будет равен значению 1.309. Для изменения его в программе имеется специальный параметр `IOR` (Индекс преломления).

Подобно размытию отражения имеется возможность влияния и на результат прозрачности. Для этого также служит параметр `Gloss` (рис. 4.22).

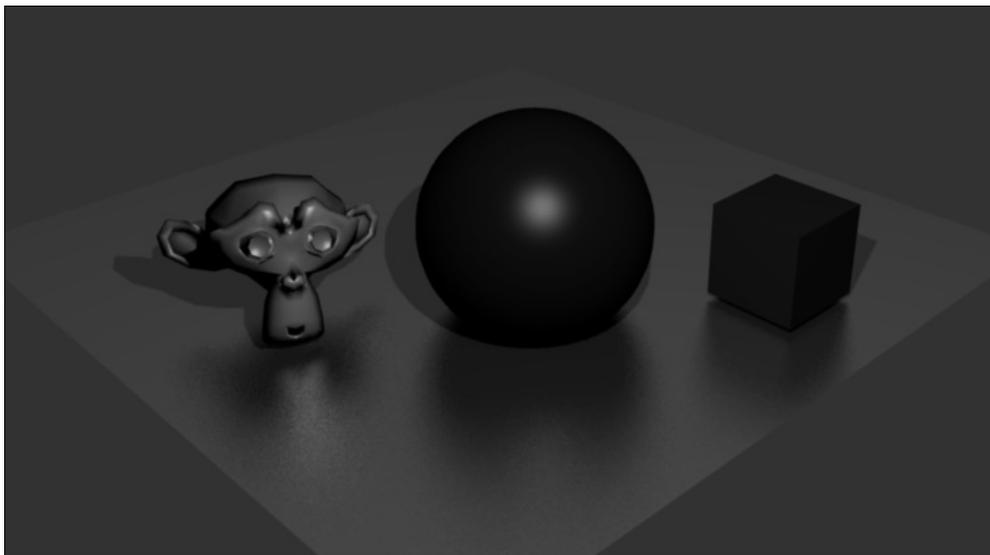


Рис. 4.19. Размытие отражения функцией `Gloss`

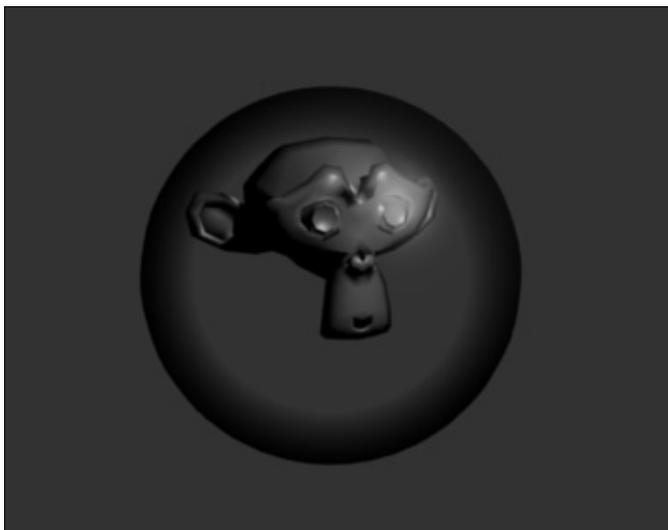


Рис. 4.20. Сделать прозрачным? Легко!

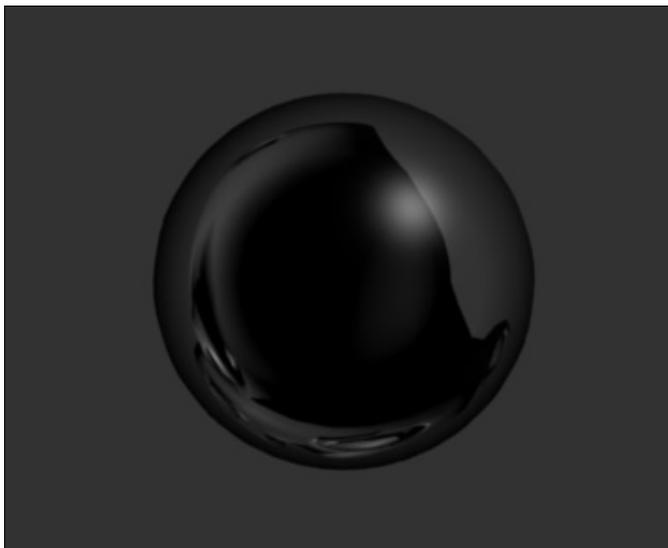


Рис. 4.21. Примерно так искажает предметы обычный лед

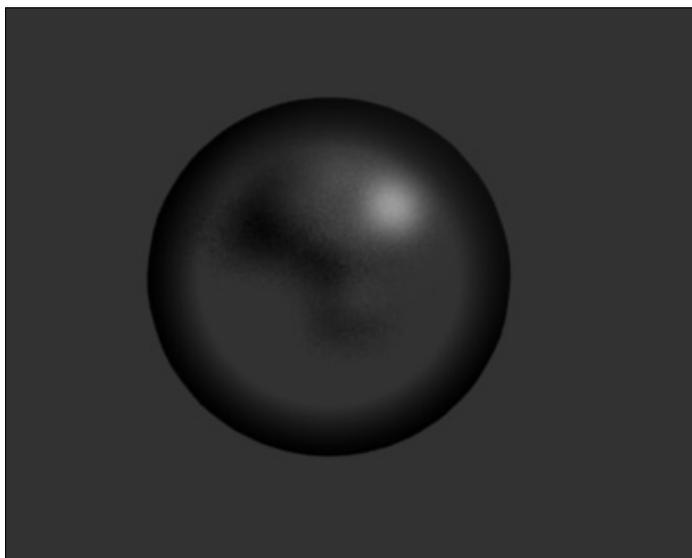


Рис. 4.22. Эффект размытия при преломлении

Внимание!

Если при обработке сцены не видно результата трассировки, то, скорее всего, эта функция отключена в настройках рендера. Для ее включения нажмите клавишу <F10> и в панели **Render** активируйте кнопку **Ray** (Луч).

4.5. Мультиматериалы

На этапе раскраски объекта очень часто возникает необходимость использования разных материалов. Можно, конечно, создавать модель, состоящую из нескольких элементов, но не всегда такое бывает целесообразно. Blender предлагает свой подход к решению этой задачи.

Принцип работы заключается в группировании отдельных элементов объекта и присвоении им отдельного материала. Рассмотрим это на следующем примере.

Создайте сферу и материал для нее белого цвета. Чтобы не путаться в дальнейшем, переименуйте его в нечто более подходящее. Перейдите в режим редактирования и выделите половину объекта любым удобным способом. Нажмите клавишу <F9> для перехода в панель **Link and Materials** (Связи и материалы). Именно здесь сосредоточены необходимые функции (рис. 4.23).

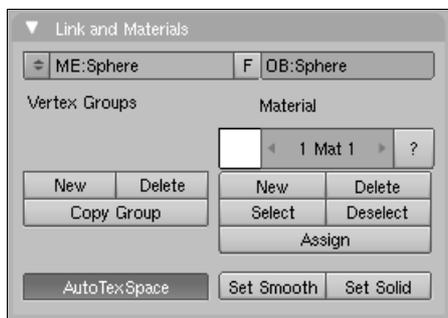


Рис. 4.23. Панель **Link and Materials**

Панель поделена на две половины:

- ◆ **Vertex Groups** (Группы вершин). Управление группами вершин;
- ◆ **Material** (Материал). Управление материалами.

Для создания новой группы необходимо нажать кнопку **New** (Новый), а затем **Assign** (Присоединение). При желании группу также можно переименовать (рис. 4.24).

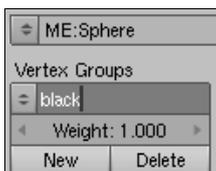


Рис. 4.24. Переименование группы вершин

Теперь осталось выбрать или создать новый материал в соседней колонке панели. Точно так же необходимо нажать кнопку **New**, а затем **Assign**. После этого можно переходить в окно материала (<F5>) и настраивать материал под свою задачу (рис. 4.25).

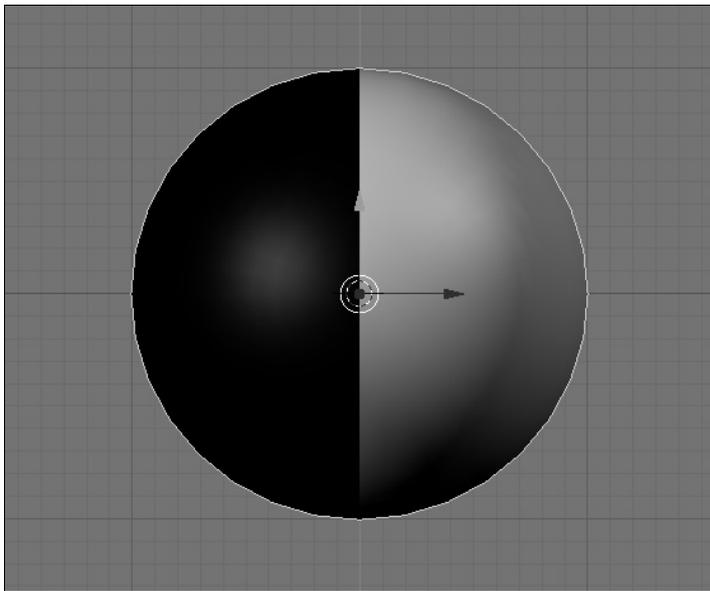


Рис. 4.25. Объект с двумя материалами

Для удаления группы или материала из списка объекта предназначены кнопки **Delete** (Удалить), расположенные в соответствующих колонках. Кроме того, имеется возможность просмотра выделенных областей для каждой группы. Сделать это можно только в режиме редактирования. Просто нажмите кнопку **Select** (Выделить), и программа отметит групповые элементы. Для снятия выделения служит кнопка **Desel** (Снять выделение).

Иногда бывает нужно изменить содержимое в уже существующей группе. Сделать это очень и очень просто. В режиме редактирования выбирается нужная группа. Затем выделяется необходимый элемент и кнопкой **Assign** присоединяется к активной группе. И наоборот, для удаления элементов из группы нужно использовать кнопку **Remove** (Удалить).

Совет

Материал, используемый несколькими объектами, можно сделать индивидуальным, если щелкнуть на цифре, находящейся справа от его названия. Эта цифра показывает количество объектов, присоединенных к данному материалу.

4.6. Эффекты *Halo*

Начинающие пользователи часто недоумевают: почему имеющийся источник света в сцене при обработке не виден? Дело в том, что объект `Light` или ему подобный всего лишь механизм для позиционирования источника света и при рендере он остается невидимым. Если нужно создать реальное свечение объекта, то используют специальные настройки материала. К примеру, для реализации обычной лампочки, как минимум, нужна ее модель, источник света, помещенный внутрь и специальный объект, который, собственно, и будет светиться.

Для создания световых, видимых эффектов используется материал `Halo`. Обратите внимание на то, что хотя такой объект излучает свет, на самом деле он не является источником и не может освещать другие объекты.

Материал `Halo` может быть применен к любому примитиву группы `Mesh` или системе частиц. Однако если применить его к целому объекту, скажем, к `Circle`, то при обработке можно увидеть россыпь светящихся элементов (рис. 4.26).



Рис. 4.26. Примитив `Circle Mesh` и материал `Halo`

Дело в том, что `Halo` применяется конкретно к каждой вершине, входящей в структуру объекта. Поэтому для создания одиночного источника свечения нужна всего лишь одна точка. В палитре примитивов `Mesh` отдельной точки

как объекта не существует. Тем не менее имеется два способа ее создания на основе других примитивов. Можно добавить в сцену любой объект из этой группы, выделить все точки, кроме одной, и удалить. Второй вариант — использовать объект *Empty Mesh*, который представляет собой просто виртуальный каркас без наличия элементов. После его выделения и перехода в режим редактирования можно добавить новую точку, щелкнув левой кнопкой мыши в нужном месте с удерживаемой клавишей <Ctrl>.

Совет

После создания отдельной вершины бывает необходимо совместить ее с центром объекта. Самое простое — выделить ее и воспользоваться инструментом привязки **Mesh | Snap** (Меш | Привязка).

Для включения режима *Halo* на панели **Links and Pipeline** расположена одноименная кнопка **Halo**. После ее нажатия содержимое панели **Material** изменится (рис. 4.27).

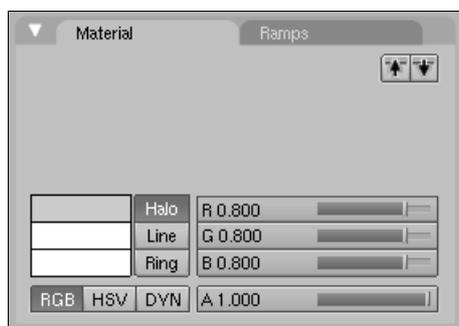


Рис. 4.27. Настройки цвета для Halo

Всего имеется четыре вида свечения, которые можно использовать с этим материалом. Все они расположены на панели **Shaders** (рис. 4.28).



Рис. 4.28. Заготовки и настройки для Halo

Эффекты подключаются путем выбора необходимых кнопок на этой панели. Естественно, можно использовать сразу несколько вариантов. Обратите внимание на то, что настройки для типов **Rings** (Кольца), **Lines** (Линии), **Star** (Звезда) заранее вынесены на панель — в отличие от **Flare** (Вспышка), которая добавляет свои параметры. По умолчанию при активации **Halo** выбранному элементу сразу придается простое свечение (см. рис. 4.26). Настройки, расположенные в верхней части панели **Shaders**, позволяют регулировать параметры, которые действительны для всех перечисленных ранее типов:

- ◆ **HaloSize**. Размер свечения;
- ◆ **Hard**. Степень жесткости;
- ◆ **Add**. Интенсивность свечения.

Особенностью материала **Halo** является то, что можно индивидуально настраивать цвета для каждого из перечисленных эффектов. Как обычно, выбрать окраску можно на панели **Material** (см. рис. 4.27).

Имеется возможность выбора количества элементов для типов **Rings**, **Lines**, **Star** (рис. 4.29). Эти опции расположены чуть ниже главных настроек **Halo**. Кроме того, для генерации случайных координат лучей **Lines** и размера колец **Rings** можно использовать опцию **Seed**.

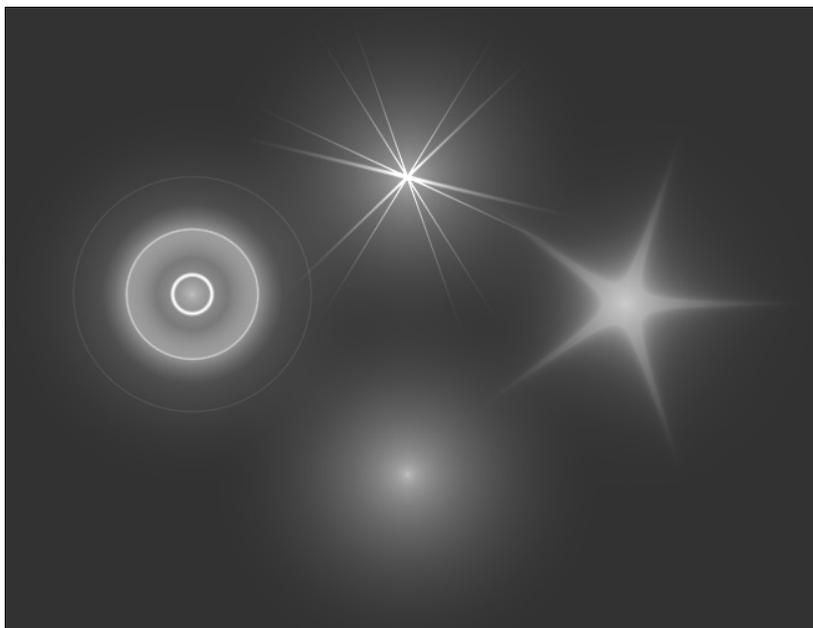


Рис. 4.29. Эффекты Halo

Особое место занимает эффект `Flare`, который имитирует линзовые блики камеры. Обычно его используют для придания естественности действия. Особенно он зрелищен при анимации камеры (рис. 4.30).



Рис. 4.30. Линзовый эффект `Flare`

При включении кнопки **Flare** панель **Shaders** дополняется новыми элементами:

- ◆ **Boost**. Используйте этот параметр для усиления зрительного эффекта;
- ◆ **Fl.seed**. Позволяет выбрать вид эффекта из нескольких имеющихся;
- ◆ **Flares**. Количество демонстрационных элементов.

4.7. Рамповые шейдеры

Немного ранее уже рассказывалось о шейдерах, с помощью которых можно настраивать реакцию материала на освещение. Однако при этом изменяется только яркость и не затрагивается основной цвет. Рассматриваемый инструмент позволяет создавать цветные области на этапе шейдерной обработки. Дело в том, что система рендеринга материала обрабатывает сначала основной цвет или текстуру и лишь потом освещение. Благодаря этому имеется возможность полной замены первоначального цвета, а если воспользоваться

специальными опциями смешения, то можно получить удивительные эффекты. Главной особенностью инструмента **Ramps** является цветовая обработка результата работы обычных шейдеров. Цвета можно настраивать как для диффузного шейдера, так и для отражающего.

Рассмотрим простейший пример использования цветных шейдеров. Пусть имеется обычная сфера серого цвета. Заставим ее сверкать цветами радуги в области обработки диффузного шейдера. Таким образом, при перемещении источника света будет перемещаться и радужная область.

Создайте новый материал для сферы и откройте панель **Ramps** в окне материала (рис. 4.31).

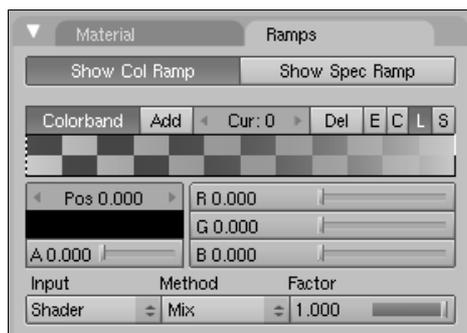


Рис. 4.31. Панель настроек рамповых шейдеров

Нажмите кнопку **Show Col Ramp**, а затем **Colorband** для работы с диффузным шейдером. Рассмотрим панель **Ramps** подробнее:

- ◆ **Colorband**. Включение рампового шейдера;
- ◆ **Add** (Добавить). Добавление нового цветового ключа. По умолчанию он помещается в центр канала и имеет серый цвет;
- ◆ поле **Cur** (Курсор). Переключение активного ключа;
- ◆ **Del** (Удалить). Удаление активного ключа;
- ◆ кнопки **E**, **C**, **L**, **S**. Тип интерполяции между цветом рампового шейдера и основного материала;
- ◆ поле **Pos** (Позиция). Перемещение активного ключа по цветовому полю;
- ◆ бегунки **RGB**. Выбор цвета для активного ключа. Можно также использовать образец, расположенный слева;
- ◆ бегунок **Alpha** (Альфа-канал). Настройка прозрачности. Позволяет регулировать соотношение между основным цветом материала и рамповым для конкретного ключа;

- ◆ **Input** (Ввод). Режим работы рампового шейдера. Выпадающее меню содержит следующие пункты:
 - **Shader** (Шейдер). Используется простой алгоритм, который принимает в расчет только направление света, но не его силу;
 - **Energy** (Энергия). Изменение цвета в зависимости от количества падающего света;
 - **Normal** (Нормаль). Использование нормали поверхности по отношению к камере;
 - **Result** (Результат). Позволяет достичь максимального эффекта, т. к. вычисление происходит после шейдерной обработки. Таким образом, тип используемого шейдера никак не ограничивается;
- ◆ **Method** (Метод). Глобальный способ микширования результата работы рампового шейдера с основным цветом (текстурой) материала. Выпадающее меню содержит большое количество функций смешения;
- ◆ **Factor** (Фактор). Сила эффекта.

Принцип работы заключается в создании и настройке необходимого числа ключей, расстановки их по цветовому каналу, выбора функций смешения. Для выполнения поставленной задачи добавьте еще четыре цветовых ключа, используя кнопку **Add**. Нумерация ключей всегда начинается с 0, а располагаются они слева направо. Если изменить порядок ключей, то автоматически изменится и их нумерация.

Чтобы выбрать ключ, можно воспользоваться полем **Cur** или просто щелкнуть на нужном маркере левой кнопкой мыши. Выделенный ключ отображается пунктирной линией. Затем выбирается цвет, свойство канала прозрачности и размещение ключа на цветовом канале. Перемещать можно как вручную, ухватившись за маркер, так и с помощью поля **Pos**.

Присвойте перечисленным номерам ключей следующие цвета:

- ◆ 0 — синий;
- ◆ 1 и 2 — зеленый;
- ◆ 3 и 4 — красный;
- ◆ 5 и 6 — желтый.

Расставьте маркеры, как на рис. 4.32.

Обратите внимание на то, что от местоположения ключей зависит не только расположение цветов, но и способ перехода между ключевыми позициями. Чем ближе находятся крайние ключи, тем четче граница между цветами. В приведенном примере именно так и сделано (рис. 4.33).

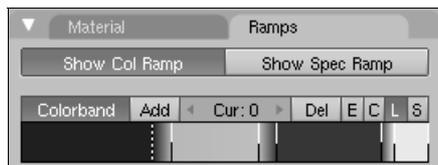


Рис. 4.32. Расположение ключей для примера

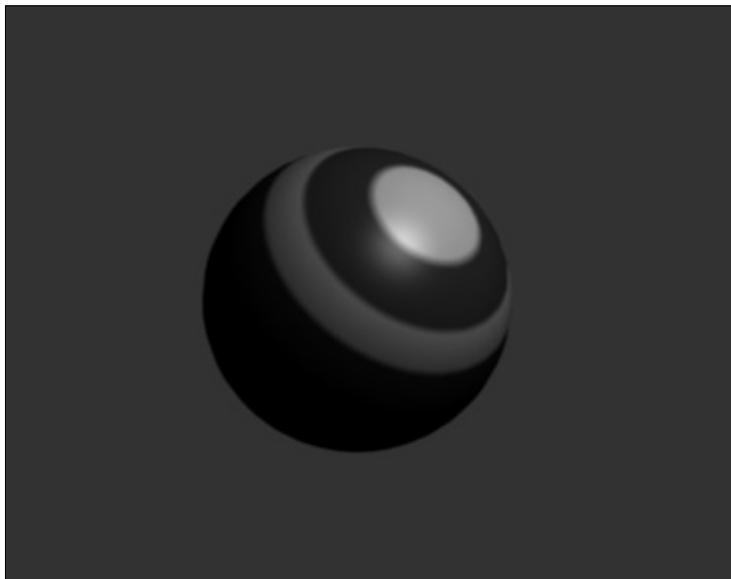


Рис. 4.33. Рамповые шейдеры в действии

4.8. Использование текстур

Обычно под словом "текстура" понимается красивая картинка, облегающая форму модели, наподобие фантика конфеты. С одной стороны, верное суждение, но слишком уж однобокое. В терминологии Blender текстурами называют промежуточные слои между основным цветом материала и шейдерной обработкой. А вот уже в роли текстуры может выступать графический файл. Всего в одном материале может содержаться до десяти слоев, или, как их еще называют, *текстурных каналов*. Все они взаимодействуют между собой, определенным образом перемешиваются, и в итоге получается необходимая картинка, если, конечно, уметь этим пользоваться.

В настройках материала имеется специальная панель **Texture** (Текстура) для управления слоями (рис. 4.34).

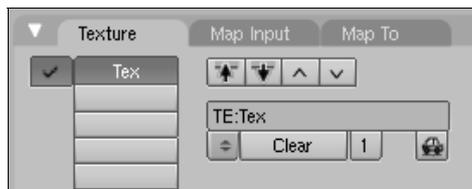


Рис. 4.34. Панель для управления слоями

По умолчанию при создании материала один из каналов уже заполнен текстурой с именем **Tex**. На самом деле, это всего лишь пустая оболочка и реальной текстуры она не имеет. Дело в том, что данная панель предназначена только для управления слоями, а выбор и настройка конкретной текстуры происходит в другом окне. В этом можно убедиться, если нажать клавишу <F6> или выбрать соответствующую кнопку на заголовке **Buttons Window** (рис. 4.35).



Рис. 4.35. Эта кнопка служит для перехода в панель настроек

При отсутствии выбранной текстуры перед вами предстанут всего две панели: **Preview** (Предварительный просмотр) и **Texture**. С первой панелью все понятно — она предназначена для просмотра активного канала. Но тут есть одна тонкость: в окне **Material** (<F5>) имеется такая же панель предварительного просмотра. Вот только показывает она полностью настроенный материал со всеми существующими слоями.

Для присвоения текстуры конкретному каналу имеется меню **Texture Type** (Тип текстуры). Всего программа предлагает 14 заготовок, большинство из которых — *процедурные* текстуры. В отличие от обычных, двумерных, картинок, процедурные текстуры создаются на основе вычислений математических функций и обладают трехмерными координатами. Использование таких текстур позволяет точнее и правильнее подгонять их к трехмерному объекту. К числу дополнительных достоинств можно отнести гибкость настроек и бесшовность.

Процесс размещения текстуры на объекте называется *мэптингом*, или *проецированием*. Используя определенные опции, можно регулировать расположение, масштаб, режимы смешивания и многое другое. Давайте поэкспериментируем на двумерной картинке.

Создайте куб и материал для него. Выберите из списка **Texture Type** заготовку **Image** (Картинка). Именно она служит для загрузки и настройки графических файлов (можно использовать видео- или анимационные последователь-

ности). Дополнительно откроются еще две панели, где для непосредственной загрузки служит панель **Image**. Загрузите любой графический файл. При удачной операции окно **Preview** отобразит загруженный рисунок. Нажмите клавишу <F5> для перехода в основное окно настройки материала.

Как уже говорилось ранее, панель **Texture** служит для управления слоями материала. Чтобы создать новый слой, нужно активировать пустой слот и нажать кнопку **Add New** (Добавить новый) или выбрать уже имеющийся из списка рядом. При желании активную текстуру можно переименовать в поле **TE:** (см. рис. 4.34). Для удаления служит кнопка **Clear** (Очистка).

Большое значение имеет порядок расположения текстур. Изменить его можно, используя кнопки перемещения активного слоя (рис. 4.36).



Рис. 4.36. Кнопки перемещения канала

Обратите внимание на галочки, находящиеся слева от заполненных слотов. При снятии их соответствующие слои отображаться не будут.

Итак, выделите канал с картинкой и откройте вкладку **Map Input** (Мэп-вход). Эта панель предназначена для настройки проецирования текстуры на объект (рис. 4.37).

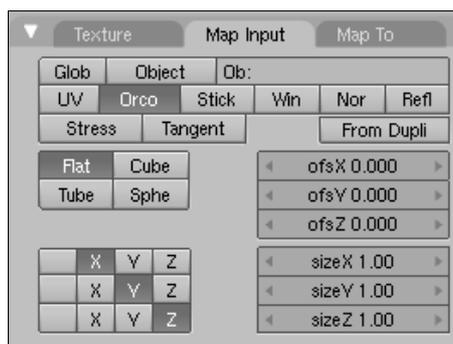


Рис. 4.37. Панель Map Input

Некоторые из этих кнопок отвечают за способ просчета координат текстуры:

- ◆ **Glob.** Зависимость от глобальной системы координат программы;
- ◆ **Object.** Вычисление в соответствии с координатами другого объекта. Имя вспомогательного объекта заносится в поле **Ob:**. Как правило, используется объект Empty;

- ◆ **UV.** Использование второго способа проецирования текстуры под названием UV-Mapping (см. разд. 4.11);
- ◆ **Orco.** Использование локальных координат объекта;
- ◆ **Stick.** Работа с постоянными координатами объекта;
- ◆ **Win.** Использование координат окна рендеринга.

По умолчанию должна быть включена кнопка **Orco**, если нет, то активируйте ее. На следующем шаге нужно выбрать, как двухмерная текстура будет накладываться на трехмерный объект. Самое простое решение — воспользоваться кнопками проецирования (см. рис. 4.37):

- ◆ **Flat** (Плоскость);
- ◆ **Cube** (Куб);
- ◆ **Tube** (Труба);
- ◆ **Sphere** (Сфера).

Для данного примера оптимально использовать **Cube** (рис. 4.38).

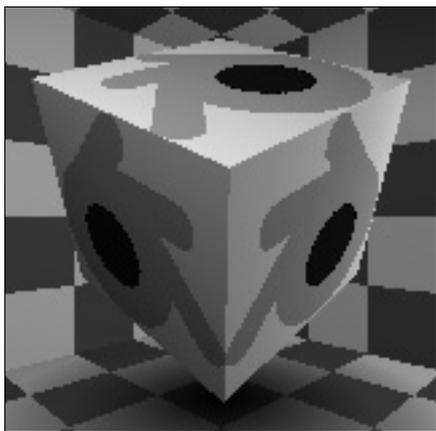


Рис. 4.38. Кубическое проецирование на объект



Рис. 4.39. Результат увеличения масштаба текстуры

Для более точной подстройки проецирования текстуры используйте блок переключения координат, а также функции масштабирования и смещения текстуры, которые расположены на этой же панели. Путем нехитрых настроек можно добиться, к примеру, многократного размножения текстуры (рис. 4.39).

Совет

Окно **3D View** умеет отображать текстурированные объекты, стоит только выбрать специальный режим в меню **Draw Type | Textured**. Вот только качество

просмотра желает быть лучше. Однако имеется способ автоматического рендеринга области окна с выходным качеством, который происходит при любом изменении сцены. Нажмите <Shift>+<P> для вызова плавающего окна рендера. По желанию его можно перемещать или масштабировать.

Для управления способами наложения и микширования слоев имеется панель **Map To** (Мэп-выход). Именно в ней происходит настройка взаимодействия конкретной текстуры с другими слоями. Рассмотрим особенности работы с ней на примере создания мраморной плиты.

Сожмите по координате Z имеющийся куб для получения модели плиты. Пусть основным цвет материала будет белым, а прожилки серыми. Измените цвет Diffuse на белый и выберите текстуру Marble для первого слота материала. Настройте текстуру в соответствии с рис. 4.40.

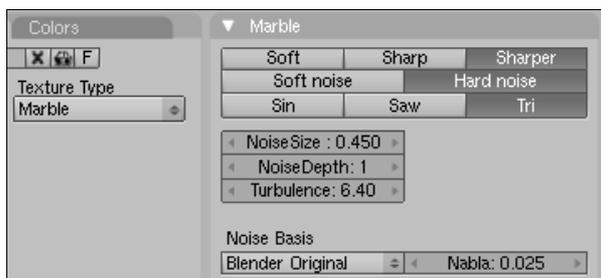


Рис. 4.40. Параметры текстуры Marble для мрамора

Теперь перейдите на панель **Map Input** и измените масштаб текстуры (**Size**) на 2.0 по всем координатам. Это необходимо для создания более плотного рисунка мрамора. Все остальные параметры оставьте без изменений.

Откройте вкладку **Map To** для окончательной настройки текстуры (рис. 4.41).



Рис. 4.41. Панель Map To

Верхние кнопки отвечают за характеристики материала после применения к нему текстуры. Обратите внимание на то, что некоторые из них при нажатии имеют три положения, которые индицируются изменением цвета надписей:

- ◆ **Col.** При ее активации используется цвет текстуры, а не материала;
- ◆ **Nor.** Изменение локальных нормалей. Эта функция используется при рельефном текстурировании (см. разд. 4.9);
- ◆ **Disp.** Используется для карт смещения;
- ◆ кнопки **Cmir, Ref, Spec, Alpha** и др. активируют соответствующие свойства материала.

Следующие кнопки определяют способ использования слоя:

- ◆ **Stencil** (Графарет). Текстура будет использоваться как маска для следующего слоя;
- ◆ **Neg** (Негатив). Переключение цвета текстуры на негативное;
- ◆ **No RGB** (Без RGB). Черно-белое изображение.

Желаемый цвет текстуры выставляется стандартными бегунками **RGB** или в окне, вызываемом щелчком мыши по образцу. Так как в рассматриваемом примере вторичным цветом является серый, то установите его удобным для вас способом. Все кнопки, кроме **Col**, должны быть неактивными.

Blender предлагает несколько вариантов смешивания цветов для граничащих слоев. Расположены они в специальном выпадающем меню в правой половине панели (рис. 4.42).

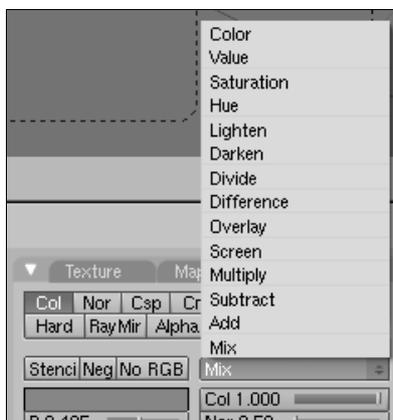
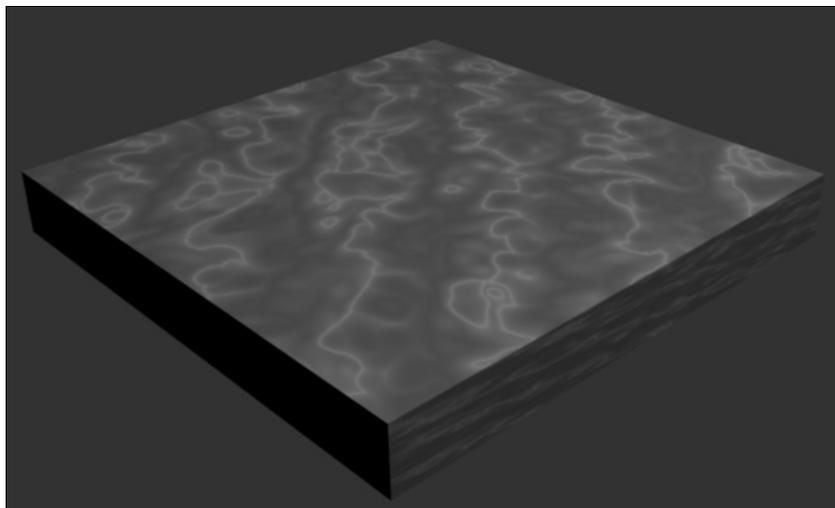


Рис. 4.42. Способы смешивания слоев

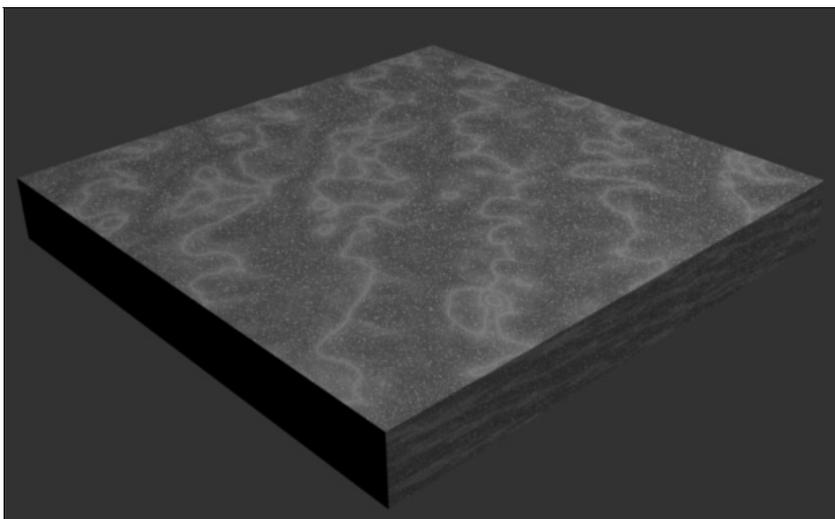
Как правило, по умолчанию используется вариант **Mix**. Но ничто не мешает выбрать иной. Просто попробуйте их в действии.

Последние нерассмотренные элементы панели — это поля **Col**, **Nor**, **Var**, **Disp**. Все они служат для более точной подстройки соответствующих режимов, выбранных в верхней части панели. Так, для активной кнопки **Col** бегунок **Col** обеспечивает чувствительность при микшинге цветов.

Если обработать полученную модель, то первое, что бросится в глаза, — это слишком "компьютерная" гладкость поверхности (рис. 4.43, *а*). В реальности кусок мрамора, как и любой камень, имеет шероховатости. Давайте исправим этот недостаток.



а



б

Рис. 4.43. Обработка объекта с одним текстурным слоем (а) и двумя (б)

Добавьте новый слой для объекта и выберите текстуру `Noise` (Шум). Эта заготовка настолько простая, что никаких настроек ее не предлагается. Однако для нашего мрамора зернистость `Noise` слишком крупная. Для ее уменьшения измените параметры `Size` панели **Map Input** на 2.0. Теперь выберите подходящий цвет для зерен, к примеру белый, и обработайте результат (рис. 4.43, б).

4.9. Рельефные карты

Если присмотреться к последним новинкам из мира компьютерных игр, то можно отметить, какие качественные и детализированные модели используются в них. Однако при загрузке такой модели в трехмерный редактор взору предстанет, скорее всего, весьма простенький с небольшим количеством полигонов объект. Так в чем же фокус, откуда в игре берутся мелкие детали при низкополигональном источнике?

Давным-давно учеными мужами был придуман способ имитации сложных поверхностей — использование обычной текстуры как хранилища данных о неровностях на ней. В настоящее время существует несколько алгоритмов, которые реалистично реализуют рельеф, такие текстуры стали называться *рельефными картами*. Обратите внимание на то, что эти способы не изменяют геометрии объекта, а "играют" со светотенями. Благодаря этому, появилась возможность использования низкополигональных моделей с достаточно хорошим выходным качеством. Однако рельефные карты применяются не только в играх. Представьте на минуту, что вам понадобилась модель обычного апельсина. Сделать ее не сложно: взять примитив (сферу) и натянуть на нее оранжевый материал. Только вот беда, настоящий апельсин имеет характерную пупырчатую поверхность. Выделять через раз точки и масштабировать? Нет и еще раз нет! Нужно использовать соответствующую рельефную карту.

Blender умеет работать с двумя видами рельефных карт. Самый простой и один из первых способов реализации этого эффекта получил название *бампинг* (`Bump Map`). В основе его использования лежит черно-белая текстура, где градациями серого обозначаются неровности поверхности. Чем светлее пиксел, тем выше будет выглядеть область поверхности и наоборот. Такую текстуру не сложно нарисовать вручную, к примеру, в программе GIMP (рис. 4.44).

Пользоваться ею тоже очень просто. Выберите для материала объекта текстуру `Image` для загрузки графического файла. Затем в панели **Map To** нужно нажать кнопки **Nor** и **No RGB**. Силу эффекта можно регулировать бегунком **Nor** (рис. 4.45).



Рис. 4.44. Типичная карта Bump



Рис. 4.45. Использование Bump

Обратите внимание на то, что кнопка **Nor** имеет три положения, которые обозначаются переключением цвета надписи — это включение и отключение эффекта, а также инвертирование его.

Благодаря легкости создания бамповой карты этот способ очень широко распространен. Однако он крайне неэффективен для создания высокоточных рельефов.

В последние несколько лет набирает обороты технология использования так называемых карт нормалей (*Normal Map*), которые обеспечивают гораздо лучший результат.

Нормалью называется вектор, расположенный перпендикулярно поверхности. Любой примитив состоит из некоторого количества полигонов, и каждый из них обладает своей нормалью. Blender умеет графически отображать нормали объекта (рис. 4.46). Для этого выделите объект, перейдите в режим редактирования (<Tab>) и в панели **Mesh Tools More** (доступна в режиме редактирования при нажатии клавиши <F9>) нажмите кнопки **Draw Normals** (Показывать нормали) и **Draw VNormals** (Показывать нормали вершин). Для регулирования размера прямых можно использовать поле **NSize**.

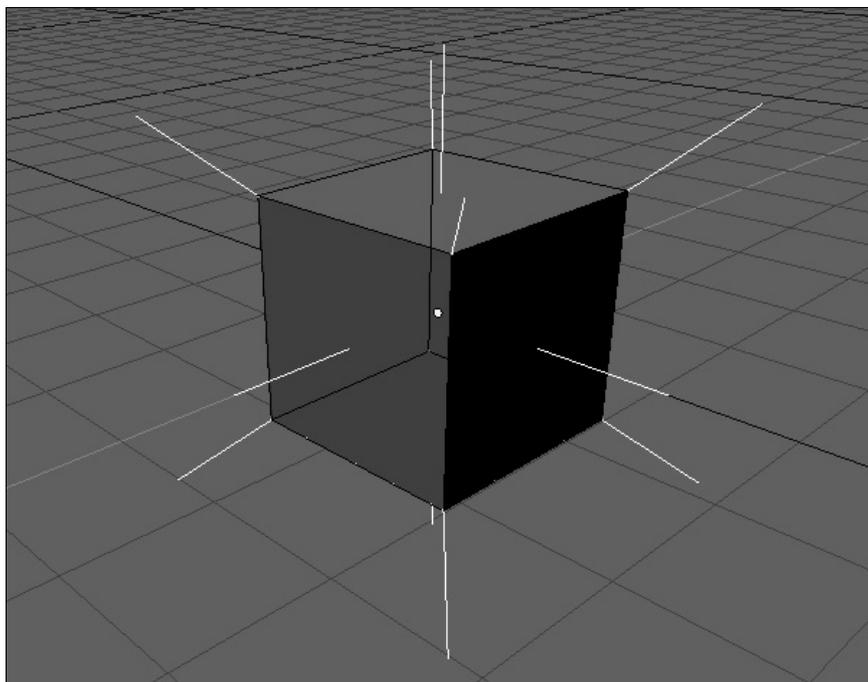


Рис. 4.46. Нормали объекта

Карты *Normal Map* представляют собой цветные текстуры RGB, где каждый цвет несет информацию о движении нормали по определенной координате:

- ◆ Red (Красный). Отвечает за координату X;
- ◆ Green (Зеленый). Работает с координатой Y;
- ◆ Blue (Синий). Данные по координате Z.

Значения в каждом канале могут располагаться в интервале от 0 до 255 (рис. 4.47).

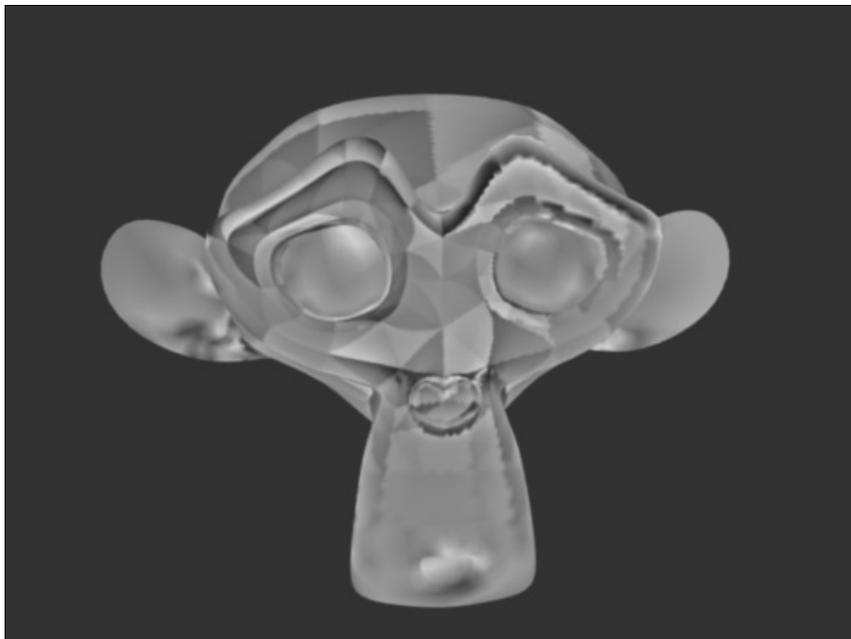


Рис. 4.47. Suzanne с текстурой *Normal Map*

Эффективнее всего использовать *Normal Map* для имитации высокодетализированных моделей. Всего программа поддерживает четыре вида карт, которые различаются способом проецирования и использования:

- ◆ Camera (Камера). Размещение карты с точки зрения камеры. Самый первый способ реализации *Normal Map*. Объект должен использоваться как статичный без перемещения или деформации;
- ◆ World (Мир). Могут использоваться координаты, остальное остается неизменным;
- ◆ Object (Объект). Расчет координат с точки зрения самого объекта. При этом его можно перемещать;

◆ **Tangent** (Тангенс). Последнее нововведение. Оптимально использовать при анимации или деформации.

Прежде чем рассматривать использование карт нормалей, нужно научиться их создавать. Процесс создания разительно отличается от рисования **Bump**. До недавнего времени пользователи были вынуждены использовать сторонние программы для генерации **Normal Map**. В текущей реализации программа **Blender** умеет создавать эти текстуры на основе имеющихся моделей, причем для всех видов. Процесс этот называется *выпечкой*. В основе его находится метод сравнения высокодетализированной модели и низкополигональной. Рассмотрим следующий пример.

Добавьте примитив **Plane** в проект и назовите его **Low**. Он будет представлять собой низкополигональную модель. Создайте его копию (<Shift>+<D>) и, используя инструмент **Subdivide**, подразделите структуру несколько раз. Для дополнительного детализирования можно использовать **Multires**.

Теперь в режиме **Sculpt Mode** изобразите что-нибудь на поверхности и назовите полученный объект именем **High**. Итак, вы имеете готовую высокополигональную модель (рис. 4.48).

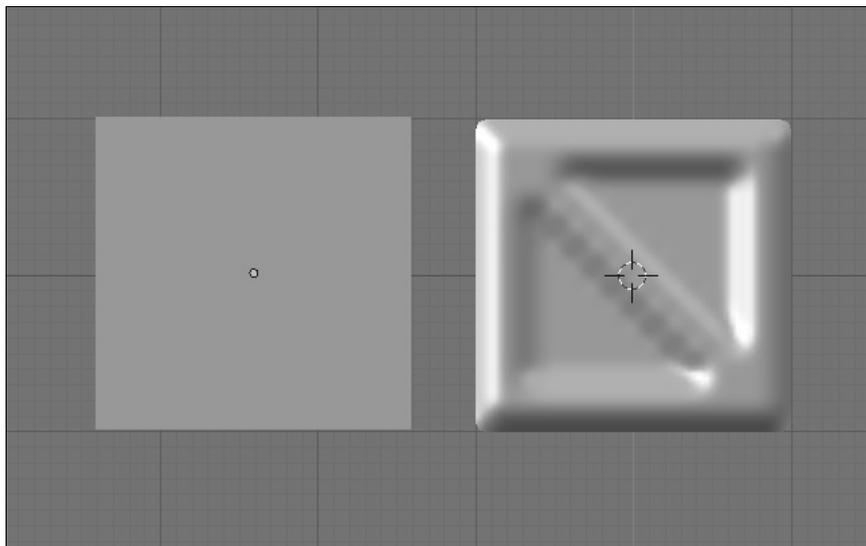


Рис. 4.48. Модели для **Normal Map**

Совет

Если **Multires** не обеспечивает необходимую плавность формы модели, можно использовать обычное сглаживание. При выпечке программа это учитывает.

Для генерации *Normal Map* нужно выполнить следующие шаги:

1. Создать текстурную заготовку желаемого размера.
2. Проецировать текстуру на низкополигональную модель с использованием UV-координат.
3. Настроить систему выпечки и обработать.
4. Сохранить карту нормалей в текстурном канале или в отдельном файле.

Подробно работа с UV-мэппингом будет рассматриваться в *разд. 4.11*. Здесь остановимся только на самом необходимом.

Для манипуляций с текстурами служит специальное окно программы **UV/Image Editor**, которое можно вызвать через глобальное меню любого окна. Выделите объект *Low* и нажмите клавишу *<Tab>* для редактирования. Теперь в окне **UV/Image** создайте новую текстуру, используя меню **Image | New** (Картинка | Новая). Установите параметры в появившемся окне, как на *рис. 4.49*.

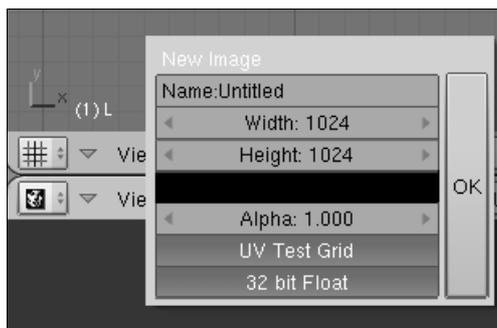


Рис. 4.49. Окно с параметрами новой текстуры

При указанных настройках программа создаст текстуру с разрешением 1024×1024 , глубиной 32 бита и демонстрационным рисунком решетки. Для создания и привязки UV-координат в режиме редактирования объекта нажмите клавишу *<U>* для вызова меню **UV Calculation** (Расчет UV). Выберите пункт **UnWrap**. Если все сделано правильно, то на текстуре в окне **UV/Image** отобразится развертка модели.

Чтобы программа смогла правильно обработать карту нормалей, оба объекта нужно совместить. Проще и точнее это можно сделать, если у моделей сбросить координаты расположения. Сделать это можно, нажав сочетание клавиш *<Alt>+<G>* для расположения и *<Alt>+<R>* — ротации.

Итак, подготовительный этап завершен. Для расчета карты нормалей или иных эффектов имеется специальная панель **Bake**, которая вызывается нажа-

тием клавиши <F10> в окне **Buttons Window**. Включите кнопки **Normals**, **Clear**, **Selected to Active**. В меню выбора пространства для расчета найдите пункт **Tangent**. Настройка панели **Bake** закончена.

Прежде чем начать выпечку, необходимо программе указать объекты и порядок следования. Выделите высокополигональный объект **High** и, удерживая клавишу <Shift>, добавьте объект **Low**. Нажмите кнопку **BAKE** для обработки.

Несколько секунд ожидания, и карта нормалей готова. Для просмотра результата откройте окно **UV/Image** (рис. 4.50).

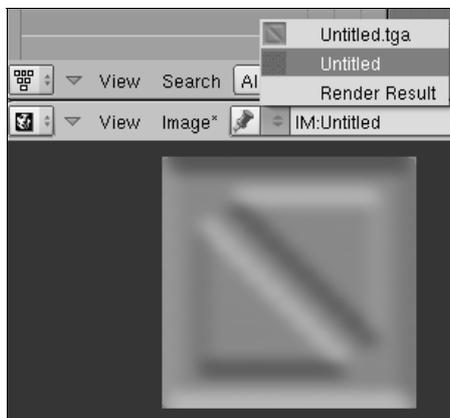


Рис. 4.50. "Свежевыпеченная" карта нормалей

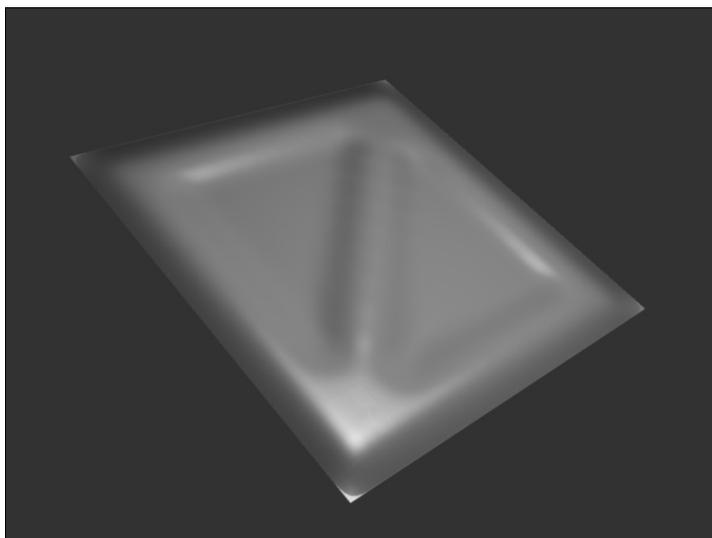


Рис. 4.51. Низкополигональная модель с картой Normal Map

Полученную текстуру можно сохранить в отдельном файле командой **Image | Save As** (Картинка | Сохранить как) окна **UV/Image**.

Настройка материала для использования **Normal Map** особенно не отличается от ранее рассмотренного **Bump**. Выделите объект **Low**, добавьте материал и текстуру типа **Image**. Загрузите карту нормали или выберите из списка, расположенного слева. На панели **Map Image** активируйте кнопку **Normal Map** и пространство **Tangent**. Вернитесь в окно **Map Input**. Так как карта нормали рассчитывалась с учетом UV-координат модели, то нажмите кнопку **UV**. В панели **Map To** выберите кнопку **Nor**. Чувствительность выдавливания можно регулировать одноименным бегунком (рис. 4.51).

4.10. Карты смещения

Рассмотренные ранее карты **Normal Map** представляют собой мощный инструмент для доработки низкополигональной модели. Путем небольших усилий можно добиться прекрасного результата при минимуме затрат. Однако эта технология имеет один существенный недостаток — рельефные карты лишь имитируют изменение геометрии. Если посмотреть на такой объект сбоку, то можно заметить, что это игра все тех же светотеней.

Displacement Map (Карта смещения) реально влияет на геометрию объекта, что позволяет создавать полноценные трехмерные модели со всеми возможностями. При этом она все же остается искажающим фактором, который проявляется лишь при обработке сцены. Этими картами очень удобно пользоваться для создания горных ландшафтов.

В качестве текстуры для карт смещения могут участвовать как обычные графические файлы, так и процедурные текстуры. Причем одновременно можно использовать с одной текстурой технологии **Normal Map** и **Displacement Map**, что позволяет точнее воплотить задуманное. Следует учитывать одно важное условие: объект должен иметь достаточное количество полигонов. Рассмотрим следующий пример.

Создайте плоскость и подразделите ее несколько раз. При желании можно использовать инструмент **Multires** и общее сглаживание. Добавьте в первый слот материала текстуру **Marble**. В панели **Map To** включите кнопки **Nor** и **Disp**. Для управления эффектами смещения и выдавливания используйте бегунки с соответствующими названиями (рис. 4.52).

Совет

Для качественной визуализации карты смещения требуют высокодетализированную поверхность, что весьма негативно сказывается на производительности обработки. Используйте обычные **Normal Map** для удаленных от камеры объектов.

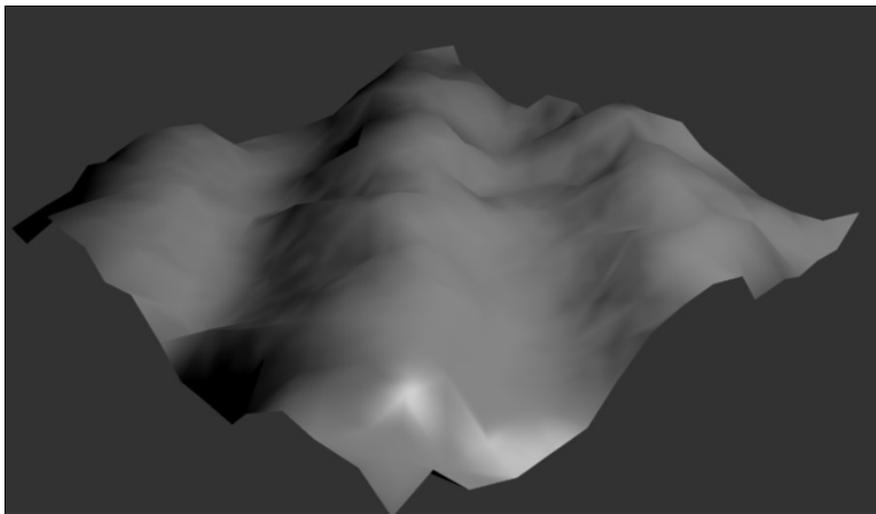


Рис. 4.52. Использование карты смещения

4.11. Работаем с UV/Image Editor

Рассмотренные инструменты панелей материала позволяют быстро и качественно настроить текстуру для объекта. Однако годится этот способ лишь тогда, когда необходимо обработать объект целиком. Для текстурирования отдельных полигонов имеются другие, гораздо более мощные инструменты.

В предыдущем разделе уже упоминалось о редакторе UV/Image Editor. Именно в нем выбирается, редактируется привязка картинка к выделенному полигону. Причем можно использовать как весь объект целиком, так и отдельную его часть.

Чтобы картинка могла накладываться на модель, ей необходимо сообщить координаты для привязки. Так как картинка является плоским объектом, то и координаты используются двухмерные. Для обозначения текстурных координат в Blender имеется сочетание UV, что по сути дела равноценно привычным X и Y.

Первоначально, после создания, примитив не обладает UV-координатами. Их можно присвоить позже в режиме редактирования. Алгоритм работы очень простой:

- ◆ выделение части или всего объекта;
- ◆ вызов меню **UV Calculation**, которое содержит заготовки для проецирования на текстуру (<U>);
- ◆ подгонка проекции под текстуру в UV-редакторе.

Самый простой и эффективный способ создания UV-координат — выбрать в меню **UV Calculation** пункт **Unwrap** (Развертка). При этом программа для каждого полигона создаст координатную сетку так, чтобы она оптимальным способом покрывала загруженную текстуру в редакторе (рис. 4.53). Естественно, развертка зависит от вида полигона.

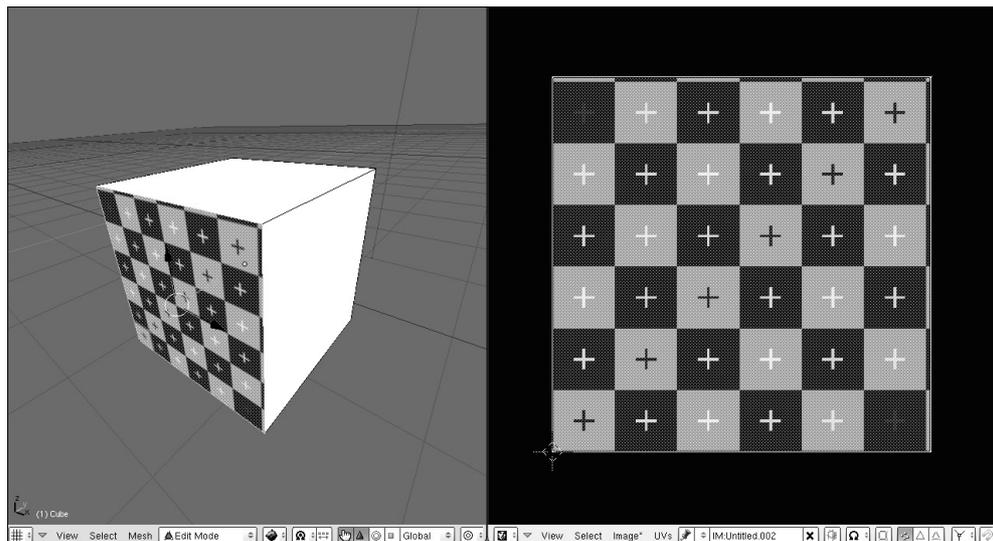


Рис. 4.53. Применение Unwrap к отдельному полигону

Работа с UV-редактором не отличается сложностью и частично похожа на редактирование в 3D-окне. Можно перемещать, масштабировать, вращать как всю сетку целиком, так и отдельный ее элемент. Горячие клавиши манипулирования эффективны и здесь. При необходимости можно управлять отображением самой текстуры в окне, перемещая или масштабировая ее. Для удобства работы желательно включить режим *Textured* окна **3D View** (рис. 4.54).

Помимо **Unwrap** меню **UV Calculation** предлагает и другие способы проецирования. Выбирать следует заготовку, наиболее подходящую для формы модели. Так, для сферического объекта больше подойдет пункт **Sphere from View**.

Особо нужно отметить способы проецирования в зависимости от положения объекта:

- ◆ **Project from View**. Координатная сетка развернется в соответствии с расположением, ротацией и масштабированием объекта;
- ◆ **Project from View (Bounds)**. В отличие от первого способа, программа постарается расположить сетку в пределах активной текстуры.

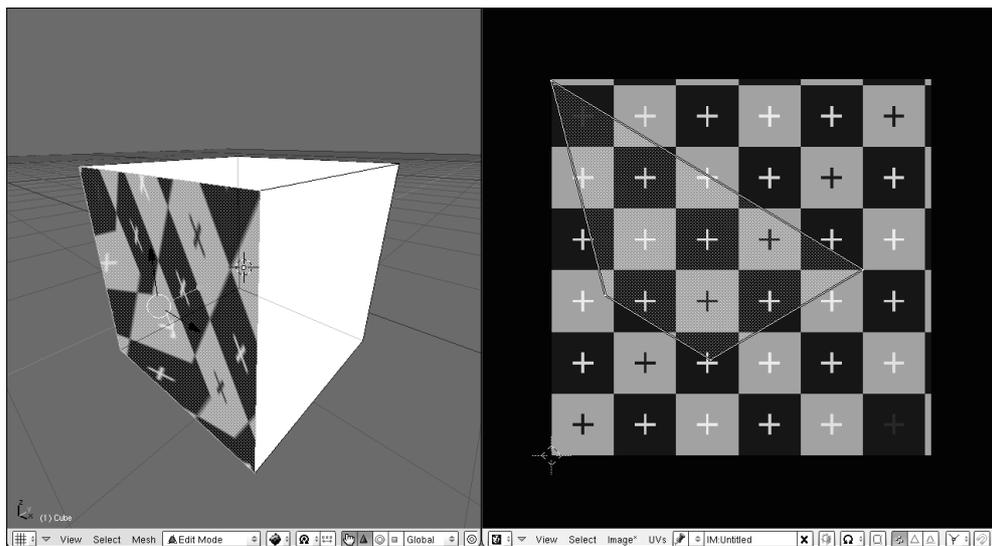


Рис. 4.54. Редактирование сетки

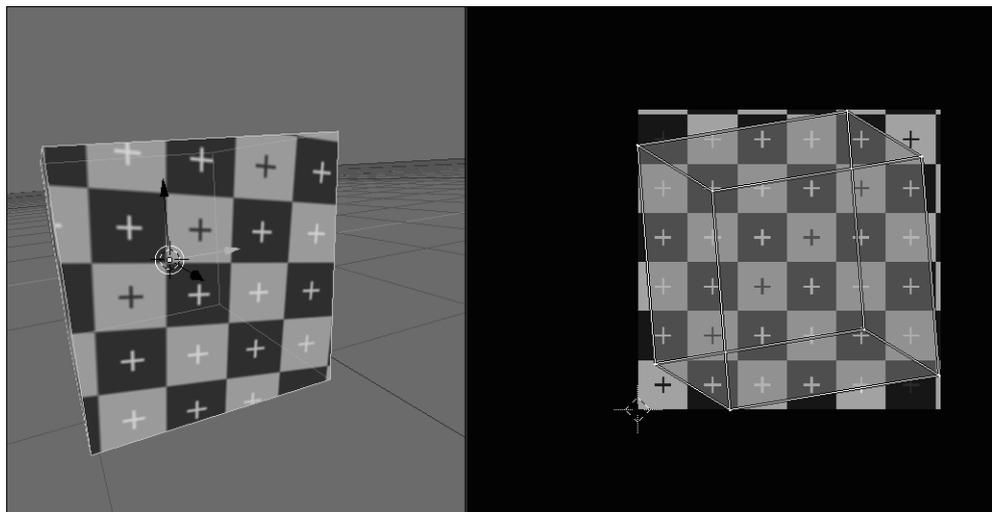


Рис. 4.55. Развертывание куба

После создания развертки можно загружать подходящую текстуру в окне **UV/Image Editor** командой **Image | Open** (Картинка | Открыть) или выбрать уже имеющуюся на заголовке окна (рис. 4.56).

Рассмотренный способ текстурирования подходит для относительно простых моделей. Для более точного создания развертки нужно использовать инстру-

мент *Seams* (Швы). Это название очень хорошо отражает суть его работы. Поверхность объекта размечается таким образом, чтобы удобно было в дальнейшем совмещать развертку с имеющейся текстурой. Представьте, что вы разрезаете и раскатываете модель, как заготовку ткани для последующего шитья. Рассмотрим пример работы с моделью гриба.

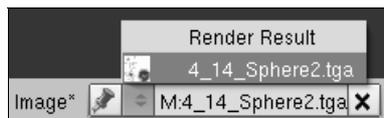


Рис. 4.56. Список загруженных файлов

Сначала нужно определить ключевые зоны рабочей модели. Так, для модели гриба такими областями будут: верхняя часть шляпки, нижняя часть и ножка. Все они должны иметь разные текстуры. Установка шва происходит путем выделения нужных ребер и выбора в меню **Mesh | Edges** (Каркас | Ребра) пункта **Mark Seam** (горячий вызов — <Ctrl>+<E>). Сделанный шов отмечается оранжевым цветом (рис. 4.57).

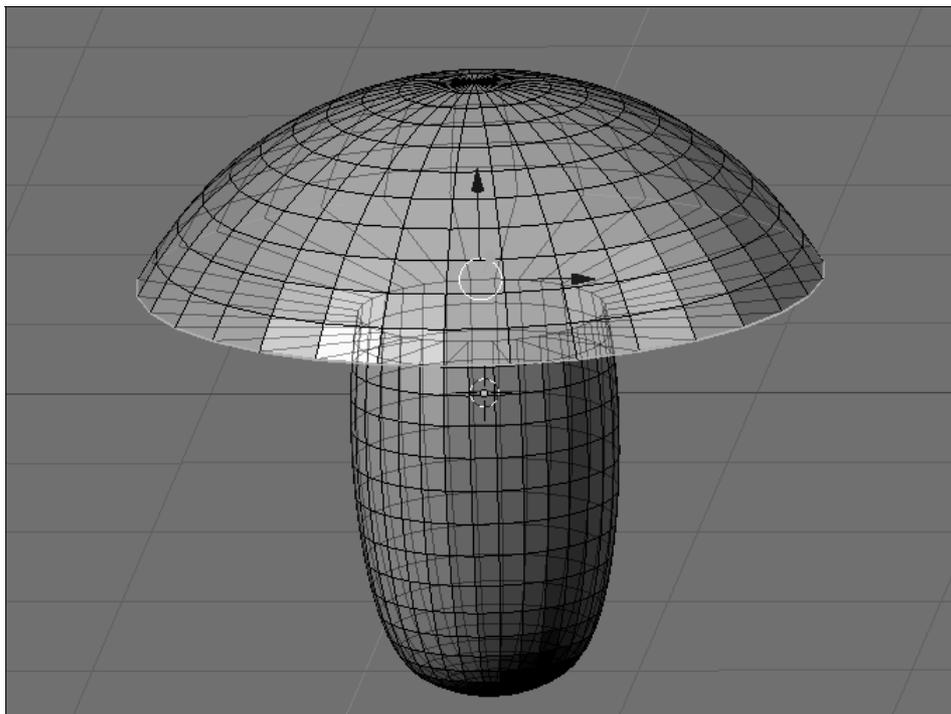


Рис. 4.57. Выделение краев шляпки

Совет

Программа позволяет выделять ребра по замкнутой окружности одним щелчком. Используйте сочетание клавиши <Alt> и правой кнопки мыши.

Для просмотра полученного разбиения нужно выделить объект целиком и установить UV-координаты. Как правило, хороший результат дает использование обычного Unwrap (рис. 4.58).

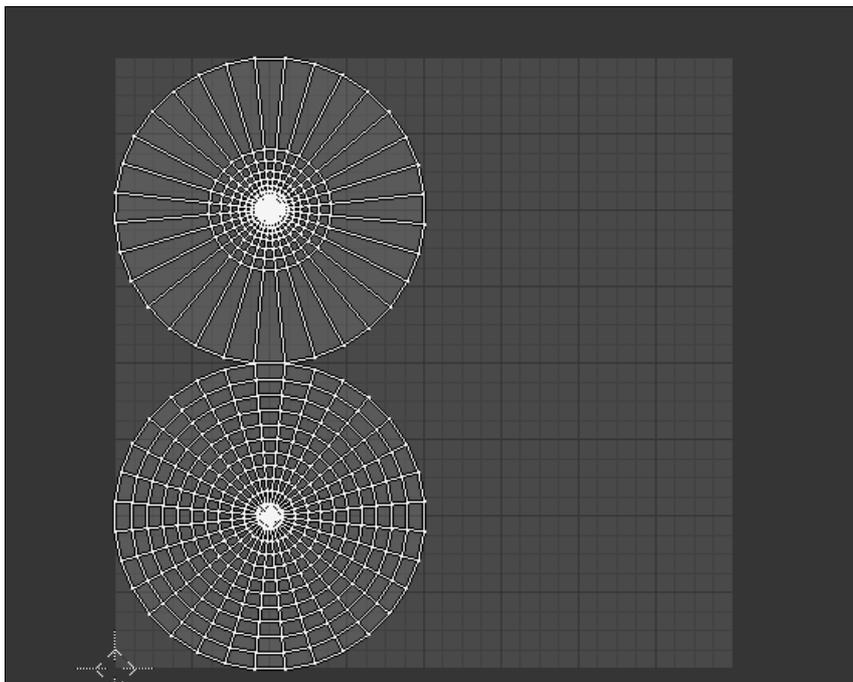


Рис. 4.58. Развертка с использованием шва

На этом рисунке верхняя окружность принадлежит ножке, а нижняя — шляпке гриба (раздельные элементы развертки называются *островками*). Для создания области нижней части шляпки нужно выделить место ее стыковки с ножкой. Повторный перерасчет координат приведет к новой развертке (рис. 4.59).

Обратите внимание на то, что старые элементы развертки переместились на новое место. Если это нежелательно и определенный островок устраивает, его можно заморозить клавишей <P> (меню UVs | Pin). При этом он выделяется оранжевым цветом. Последующие перерасчеты никак не затронут такие элементы. Естественно, перед этим островок должен быть выделен. Для обратной операции служит сочетание клавиш <Alt>+<P>.

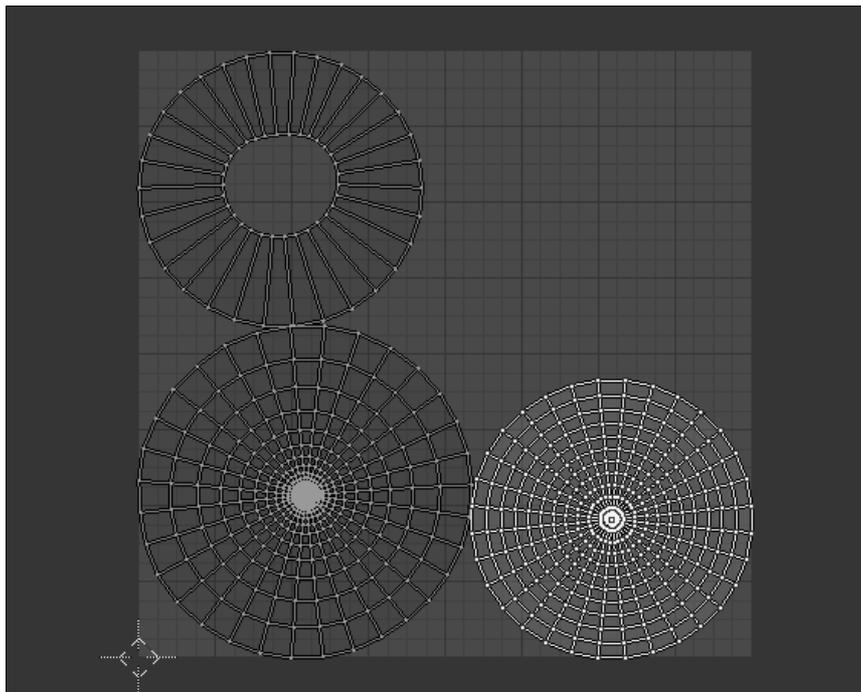


Рис. 4.59. Окончательная развертка для модели гриба

Что же делать дальше? Программа позволяет вывести на диск сделанную развертку в виде графического файла. В дальнейшем его можно будет отретрировать в любом графическом редакторе, например в GIMP. Для этого имеется специальный скрипт **Save UV Face Layout**, расположенный в меню **UVs | Scripts (UVs | Скрипты)**. После создания текстуру можно загрузить в программу. При наличии погрешностей наложения их легко устранить путем масштабирования или перемещения развертки в окне **UV/Image Editor** (рис. 4.60).

Внимание!

По умолчанию программа не будет обрабатывать текстуры, наложенные этим способом. Включите кнопку **TexFace** на панели **Material** и наслаждайтесь (см. рис. 4.2).

Не зря окно **UV/Image Editor** имеет двойное название. Дело в том, что оно предназначено еще и для ручной раскраски текстуры. Так давайте сделаем гриб похожим на мухомор!

Для включения режима рисования служит кнопка с рисунком карандаша на заголовке окна **UV/Image Editor** (рис. 4.61).



Рис. 4.60. Модель гриба с наложенной текстурой



Рис. 4.61. Активация режима рисования

Основные инструменты расположены на специальной панели, которую можно вызвать, нажав клавишу <C> (рис. 4.62).

Режимы работы кисти:

- ◆ **Draw** (Рисовать). Основной режим рисования;
- ◆ **Soften** (Смягчить). Используется для размытия изображения;
- ◆ **Smear** (Размазать). Эффект тягучести изображения;
- ◆ **Clone** (Клонировать). Многократное размножение кусочка рисунка.

Настройки кисти:

- ◆ **Color** (Цвет). Щелкните образец для выбора цвета;
- ◆ **Opacity** (Прозрачность). Установите значение 1 для максимальной эффективности;
- ◆ **Size** (Размер). Размер кисти;
- ◆ **Falloff** (Спад). Настройка жесткости краев кисти;

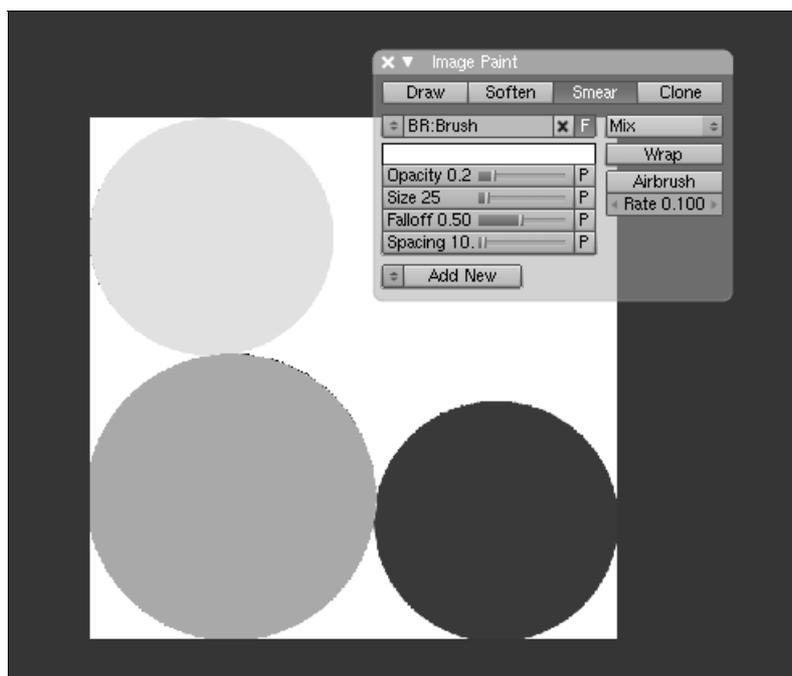


Рис. 4.62. Панель Image Paint



Рис. 4.63. Шляпка гриба раскрашена вручную

- ◆ **Spacing** (Интервал). Blender для создания сплошной линии использует последовательность изображений определенного размера. Этот параметр указывает расстояние между такими шаблонами.

Если раскрашивание текстуры в **UV/Image Editor** по каким-либо причинам сопряжено со сложностями, то можно выполнять это непосредственно в окне **3D View**. Для этого имеется специальный режим **Texture Paint**, который находится в меню **Mode** заголовка окна. Нажмите клавишу <F9> для открытия панелей редактирования, где имеется нужная панель **Paint**. Инструменты и настройки кистей обоих вариантов практически ничем не отличаются (при рисовании в окне **3D View** нельзя использовать клонирование).

ГЛАВА 5



Анимация

Когда-то, на заре мультипликации, художники были вынуждены тратить сотни, тысячи часов на прорисовку всех движений героев и деталей сцены. Сейчас существует немало профессиональных анимационных пакетов, способных облегчить этот поистине адский труд. Однако совсем другие перспективы открываются для аниматоров при использовании трехмерной графики. Вспомните нашумевшие хиты прошлых лет, такие как серии про Шрека, "Тачки", "Сезон охоты". Красочная графика, "настоящая" физика, объемное изображение — все это вывело трехмерные мультфильмы на совершенно другой качественный уровень. А ведь анимационное 3D способно на большее. Так, в последнее время стало популярным использование возможностей компьютерной графики для создания максимально точных "двойников" природных и искусственных объектов или явлений. Чего стоят только замечательные документальные фильмы BBC про эпоху динозавров, в которых все выглядит очень и очень натурально.

Эта глава посвящена основам анимации в Blender, знакомству с богатым набором инструментов и возможностей программы.

5.1. Основы анимации

Blender, как и другие мощные профессиональные пакеты трехмерной графики, позволяет не только создавать высококачественные модели, но и успешно работать с анимацией. Лучшим подтверждением этому служит презентационный короткометражный мультфильм Big Buck Bunny, который был полностью создан в Blender (найти его можно на диске, прилагаемом к книге). Именно этот анимационный фильм заявил о Blender, как о полноценном трехмерном пакете, способном тягаться с известными коммерческими программами (рис. 5.1).



Рис. 5.1. Кадр из мультфильма Big Buck Bunny

Условно анимационные возможности Blender можно разделить на четыре большие группы, каждая из которых имеет свои уникальные инструменты для реализации:

- ◆ *основная анимация объектов в пространстве и во времени.* Под этим подразумевается перемещение, ротация и масштабирование объектов в сцене;
- ◆ *деформация начальной структуры объекта.* Поистине потрясающая возможность изменения самой формы модели как заблагорассудится;
- ◆ *анимация модели с использованием системы костей.* Особый вид, который чаще всего используется в работе с персонажами. Для ее реализации создается скелет, состоящий из составных частей (костей), который привязывается к модели. В итоге анимация осуществляется путем управления этих костей. Именно этот способ обеспечивает наиболее точную имитацию движений живых существ;
- ◆ *анимация физических явлений.* Благодаря встроенному физическому движку Bullet, программа позволяет имитировать такие природные явления, как ветер, текучесть, гравитация и многие другие.

Естественно, эти способы не используются поодиночке и пользователь волен комбинировать их по своему усмотрению.

Краеугольным камнем в работе с анимацией является понятие *ключевой кадр*. Под этим подразумевается уникальное положение объекта в пространстве и во времени. Представьте, что вам нужно передвинуть объект из точки А

в точку Б за определенное время. До появления специальных программ художники были вынуждены создавать все промежуточные рисунки для имитации движения. В наше время этим занимается компьютер. Достаточно указать ключевые положения в пространстве, и программа сгенерирует все необходимые кадры. Давайте попробуем создать простейшее анимационное действие.

Экспериментировать будем с обычным кубом, который появляется при каждом запуске программы. Условием задачи является перемещение его по одной из координат с одновременным изменением масштаба за 1 секунду времени.

Основной алгоритм работы заключается в следующем:

- ◆ выбор необходимого значения времени (кадра), которое будет являться ключевым;
- ◆ собственно изменение положения, ротации или масштабирования объекта;
- ◆ создание ключа для выполненного действия.

На практике все это выглядит очень даже просто. Пусть исходное положение куба будет являться начальным ключевым кадром. Так как по умолчанию при запуске программы отсчет кадров анимации начинается с единицы, то нам остается лишь установить ключ для этого положения. Сделать это можно, выбрав из специального меню **Insert Key** (Вставить ключ) необходимый ключ. Выделите объект и нажмите клавишу <I>. В появившемся меню выберите пункт **Loc** (рис. 5.2).

Обратите внимание на первые три пункта. Именно они отвечают за создание ключей для основных манипуляций с объектом. В нашем случае пункт **Loc** закрепляет имеющееся расположение примитива.

Для создания следующего ключевого кадра необходимо выбрать ему другое время выполнения. На заголовке окна **Buttons Window** имеется специальное окно, которое позволяет переключать текущий кадр сцены (рис. 5.3).

Введите в него значение 25. Именно такое количество кадров приходится на одну секунду времени (по умолчанию Blender настроен на выполнение анимации со скоростью 25 кадров в секунду). Передвиньте объект на нужное место и установите очередной ключ **Loc**. Вот таким простейшим способом вы создали свою первую анимацию.

Чтобы просмотреть полученный результат, достаточно установить начальный кадр сцены (см. рис. 5.3) и нажать клавиши <Alt>+<A> (клавиша <Esc> отменит выполнение). Анимация при этом будет проигрываться только для активного окна.

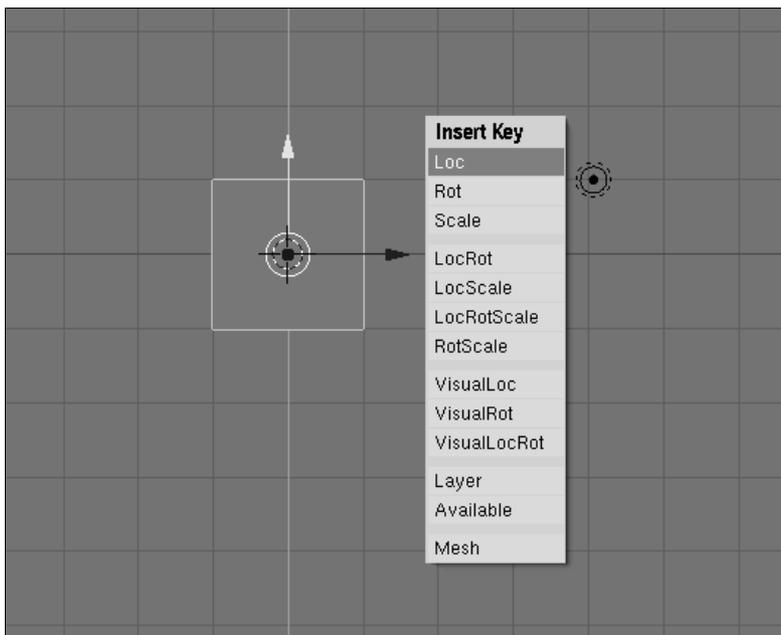


Рис. 5.2. Меню установки ключа



Рис. 5.3. Окно для изменения текущего кадра

Совет

При необходимости можно просматривать анимацию по кадрам. Для этого используйте стрелки в окне выбора текущего кадра (см. рис. 5.3).

5.2. Простое управление с *Timeline*

Рассмотренные способы проигрывания анимации являются самыми удобными и эффективными, если поставленная задача заключается только в этом. Действительно, нажав всего лишь пару клавиш, можно оживить содержимое активного окна. Но вот просмотреть созданные ключи или даже отредактировать их таким способом не удастся. Куда более мощными возможностями обладает специальный режим программы — *Timeline*. Особенность его в том, что в специальном окне в графическом виде выводится информация об анимационных ключах выделенного объекта.

В отличие от большинства популярных программ 3D-моделирования, режим **Timeline** программы **Blender** позволяет только просматривать созданную анимацию, последовательность ключей и имеет некоторые полезные особенности, но перемещать ключевые кадры здесь невозможно. Для этого существуют особые режимы, которые будут рассмотрены позже.

Панель **Timeline** можно открыть в любом подходящем окне, выбрав из глобального меню опцию с соответствующим именем (рис. 5.4).

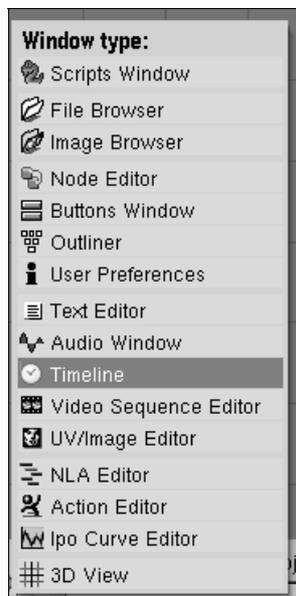


Рис. 5.4. Меню выбора режима окна — **Timeline**

Основное пространство окна **Timeline** занимает графическая область, где линиями желтого цвета отображаются имеющиеся анимационные ключи выделенного объекта. Единственная полоска зеленого цвета является своего рода курсором, передвигая который можно проигрывать соответствующие кадры (рис. 5.5).

Выполняется это очень просто. Щелкните левой кнопкой мыши в нужном месте **Timeline**, и линия-курсор переместится туда, с одновременным обновлением экрана. Также можно, не отпуская кнопку мыши, перемещать курсор в нужном направлении и с удобной скоростью просматривать анимацию. Обратите внимание на то, что перерисовываться будут все окна программы. В некоторых случаях это не нужно и излишне загружает систему. Для управления отображением при проигрывании имеется специальное меню **Playback** (Проигрывание), расположенное на заголовке окна. Снимите флажок **Play-**

back | All 3D Windows (Проигрывание | Все 3D-окна) и установите флажок **Playback | Top-Left 3D Window** (Проигрывание | Верхнее левое 3D-окно). В этом случае во время перемещения курсора анимация будет демонстрироваться только в верхнем окне.

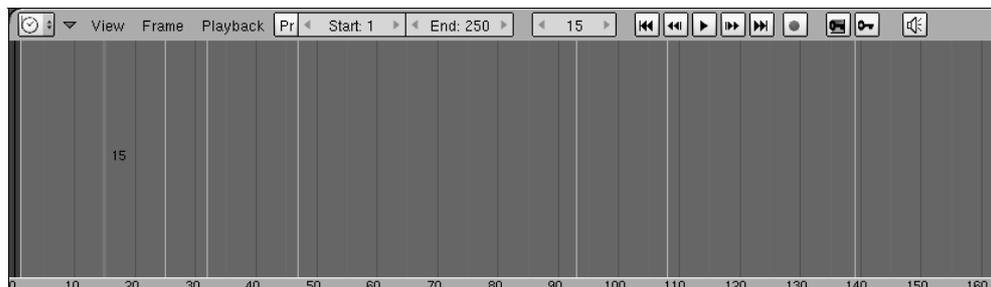


Рис. 5.5. Основная область окна Timeline

Основные кнопки управления режимом Timeline расположены на заголовке окна (рис. 5.6).

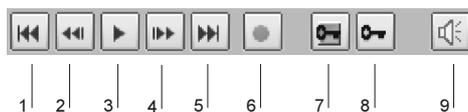


Рис. 5.6. Основные кнопки режима Timeline

В целом они характерны для большинства медиапроигрывателей, но некоторые из них представляют особый интерес:

1. Перемотка в начало анимации.
2. Переход к предыдущему ключевому кадру.
3. Проигрывание и остановка действия.
4. Переход к следующему ключевому кадру.
5. Перемотка в конец анимации.
6. Включение и отключение режима автоматической записи.
7. Установка нового ключевого кадра. При этом появляется контекстное меню для выбора типа ключа (**Loc**, **Rot**, **Scale** и т. д.).
8. Удаление активного ключевого кадра (для выделения можно воспользоваться кнопками перемещения по ключам). Также предлагается выбрать в меню нужный тип ключа.
9. Синхронизация со звуком.

Одной из замечательных особенностей режима `Timeline` является возможность автоматического создания ключевых кадров. Нужно просто нажать соответствующую кнопку на панели, и программа будет создавать ключи для любых манипуляций с объектом. Этим удобно пользоваться при работе с относительно простой анимацией. Давайте рассмотрим приблизительный алгоритм использования автоматической записи:

- ◆ активация режима записи. При этом появляется меню с выбором режима записи:
 - **Replace Keys** (Переписать ключ);
 - **Add/Replace Keys** (Добавить/Переписать ключ);
- ◆ первоначальная ручная установка ключевого кадра с помощью меню **Insert Keys** (<I>). Дело в том, что после включения записи программа не создает ключей для текущих параметров объекта, и это понятно, ведь она не знает, какой тип необходим. Можно также переместить (масштабировать, развернуть) объект. В этом случае ключ будет создан;
- ◆ подготовка к созданию следующего ключа. Здесь нужно выбрать промежуток времени, в течение которого программа будет обрабатывать переход от одного состояния к другому. Это достигается переносом линии-курсора на новое место;
- ◆ необходимые манипуляции с объектом;
- ◆ отключение режима записи.

Совет

По умолчанию длительность анимации в программе установлена в 250 единиц (при 25 кадров в секунду — это 10 секунд). При необходимости эти настройки легко изменить, поменяв значения в полях **Start** и **End** на заголовке окна **Timeline**.

Интересной и полезной возможностью режима `Timeline` является работа с маркерами. Они служат всего лишь в демонстрационных целях и не влияют на саму анимацию. Этим удобно пользоваться для отметки границ при наличии сложных последовательностей. Причем маркеры являются объектами глобальными и не зависят от выделенного объекта (рис. 5.7).

Все функции работы с маркерами находятся в меню **Frame** (Кадр). Чтобы создать новый маркер, нужно установить линию-курсор в подходящем месте и нажать клавишу <M> (**Frame** | **Add Marker**). Полученный указатель легко переместить в новое место, ухватившись мышью за него и удерживая нажатой ее правую кнопку. Для закрепления результата просто щелкните левой кнопкой мыши (повторное нажатие правой отменит выполненное действие). Естественно, можно присвоить выделенному маркеру произвольное имя. Для

этого служит пункт меню **Frame | Name Marker** или горячие клавиши **<Ctrl>+<M>**.

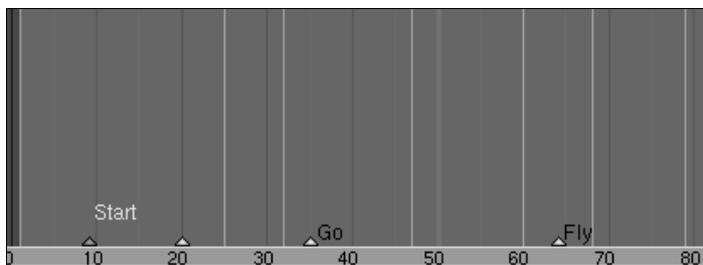


Рис. 5.7. Пример установленных маркеров

Совет

Европейским стандартом телевизионной картинки является система PAL и ее разновидности с соотношением 25 кадров в секунду (в России и СНГ чаще всего используется SECAM с таким же количеством кадров). Однако при необходимости это соотношение легко изменить путем выбора опции **Playback | Set Frames/Sec** (Проигрывание | Установка кадров в секунду).

5.3. Использование кривых IPO

Впервые увидев окно *редактора IPO*, новичок задается вопросом: как это можно использовать для управления анимацией? В действительности это мощный инструмент, который позволяет настраивать анимацию точно под свою задачу (рис. 5.8).

Сначала нужно разобраться, что это такое. Сокращение IPO произошло от английского слова *InterPOLation* (интерполяция). По-простому, этим термином обозначается расчет промежуточных кадров при обработке анимации. Графическим выражением математических формул здесь являются хорошо знакомые нам кривые. По сути дела IPO и ключевые кадры являются одним хорошо отлаженным механизмом. При создании анимации объекту присваивается свой модуль IPO. Причем их может быть сразу несколько. Бывают и такие ситуации, когда один модуль IPO обслуживает более одного объекта. Использовать этот механизм можно не только для анимации объекта, но и для изменения свойств материалов, окружения, физики.

Окно **Ipo Curve Editor** (Редактор кривых IPO) подчиняется тем же правилам, что и другие окна программы. Рабочую область можно масштабировать, перемещать, изменять ее размер. Работа с кривыми IPO ничем не отличается от тех же кривых Безье (см. разд. 3.2).

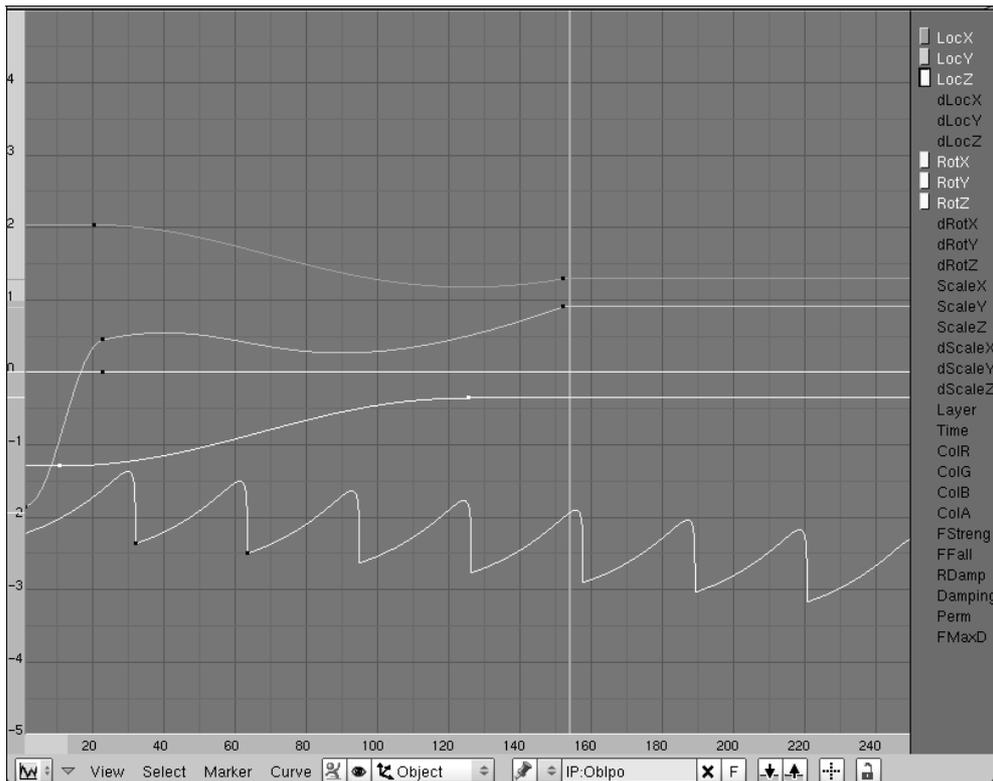


Рис. 5.8. Типичное окно с кривыми IPO

Давайте вернемся к первому эксперименту с анимацией куба. Если присмотреться внимательно к анимации, то можно заметить, что перемещение объекта происходит неравномерно. Сначала объект плавно разгоняется, доходит до некой серединной точки и точно так же плавно гасит скорость. Выглядит, конечно, очень эффектно, но не всегда такое бывает нужно. Предположим, для некоторого действия необходима постоянная скорость объекта от точки до точки. Давайте откроем окно редактора IPO, а проще всего это сделать — выбрать заготовку *Animation*, нажав <Ctrl>+<Влево> (рис. 5.9).

Любой модуль IPO имеет некоторое количество каналов, характерных для выбранного режима. Так, при создании перемещения куба были созданы три кривые *Loc* с заполнением соответствующих каналов (рис. 5.10).

Каждый из каналов имеет свой индивидуальный цвет, который присваивается и для созданных кривых. Помимо функции индикации, столбик с каналами используется для отображения и выделения кривых. К примеру, если анимация объекта чрезвычайно сложная и кривые представляют собой нечто похожее на паутину, легко убрать лишнее, сняв выделение с названий каналов.

В свою очередь, цветовые эскизы каналов служат для выделения самих кривых.

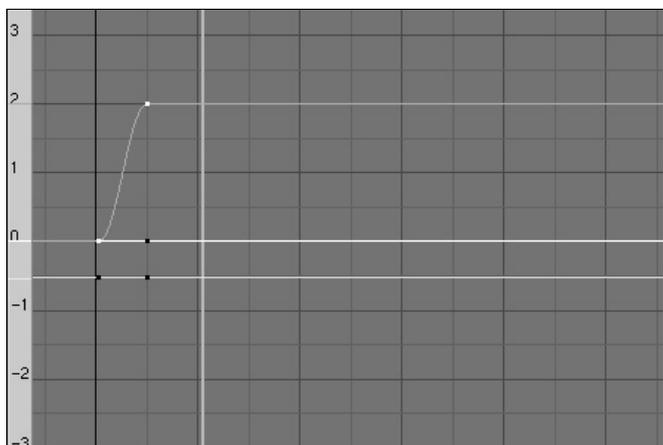


Рис. 5.9. Созданные кривые IPO для примера с анимацией куба

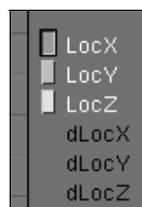


Рис. 5.10. Каналы `Loc` предназначены для управления перемещением объекта

Окно редактора IPO имеет две шкалы, где горизонтальная обозначает время (кадры), а вертикальная — качество. Таким образом, перемещая вертикальную линию-курсор (зеленого цвета), можно просматривать созданную анимацию. Если приглядеться к кривым куба, то можно на них увидеть по две точки, которые обозначают созданные ранее ключевые кадры. В этом примере (см. рис. 5.9) хорошо видно, что анимация осуществляется по одной лишь координате X, т. к. две другие кривые имеют для обеих ключевых позиций одинаковое значение качества. Обратите внимание, что кривая `LocX` не является прямой и это объясняет неравномерное перемещение между имеющимися ключами. Достаточно выровнять ее для обеспечения постоянной скорости перемещения объекта. Просто выделите кривую `LocX`, нажмите `<Tab>` и отредактируйте ее в соответствии с правилами работы с кривыми (см. разд. 3.2).

Совет

Для управления самими кривыми, а не их элементами можно воспользоваться стандартными горячими клавишами-манипуляторами (`<G>`, `<R>`, `<S>`). Однако есть более удобный способ перемещения кривой с помощью мыши. Для этого нужно выделить кривую правой кнопкой мыши и, удерживая кнопку нажатой, переместить объект в желаемое место. При этом нажатие левой кнопки мыши зафиксирует результат, нажатие правой — отменит.

Рассмотренный пример очень просто и наглядно объясняет работу с IPO. Но давайте немного усложним задачу. Предположим, необходимо придать постоянное вращение объекту по одной из осей. Первое, что может прийти в голову, — это создание двух ключевых кадров канала Rot с соответствующим градусом поворота. Этот способ действительно придаст вращение объекту, но только в пределах имеющихся ключей. Для постоянного движения придется создавать новые или копировать уже имеющиеся ключи. Поэтому гораздо удобнее воспользоваться некоторыми возможностями редактора IPO.

Существует два способа создания ключевых кадров и, соответственно, каналов IPO. Первый способ мы рассматривали ранее, когда использовали меню **Insert Key** (<I>). Откройте новый проект и установите ключевой кадр Rot для имеющегося куба (**Insert Key | Rot**). Обратите внимание, что в действительности создается сразу три кривые Rot для каждой координатной оси. В некоторых случаях это просто не подходит.

Второй способ связан непосредственно с самим редактором IPO. Отмените сделанные изменения и откройте редактор IPO. Чтобы создать кривую, сначала необходимо включить нужный канал путем выделения его в имеющемся списке (рис. 5.11).

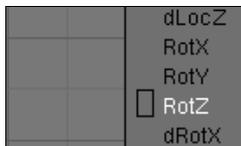


Рис. 5.11. Выделение канала для последующего его редактирования

Теперь, удерживая нажатой клавишу <Ctrl>, щелкните левой кнопкой мыши в окне редактора. Тем самым будет создана кривая IPO с начальной контрольной точкой. Для добавления следующего ключевого кадра передвиньте курсор на новое место и, точно так же, удерживая <Ctrl>, щелкните левой кнопкой мыши (рис. 5.12).

Попробуйте запустить полученную анимацию. Если все сделано правильно, объект совершит поворот по оси Z и остановится после прохождения второго ключа. В нашем случае это не является окончательным решением поставленной задачи.

Существует возможность придания кривой особых метасвойств. Возможные значения находятся в меню **Ipo Extend Mode** (Расширенный режим), которое вызывается в окне **Ipo Curve Editor** нажатием клавиши <E>. Выделите имеющуюся кривую и нажмите клавишу <E>. Затем выберите пункт **Ipo**

Extend Mode | Extrapolation (Расширенный режим | Экстраполяция). Кривая примет следующий вид (рис. 5.13).

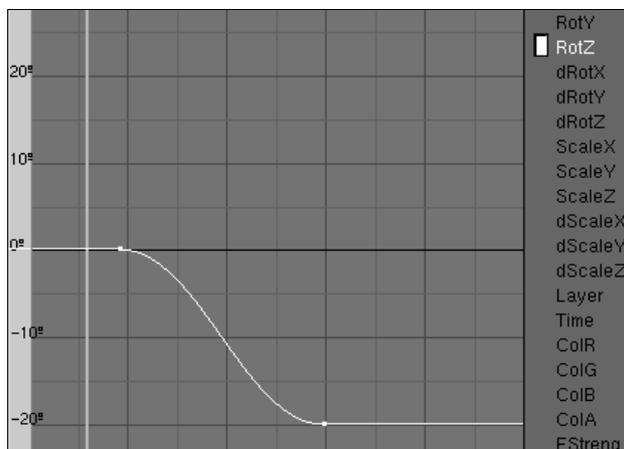


Рис. 5.12. Кривая IPO для канала RotZ с двумя ключевыми кадрами

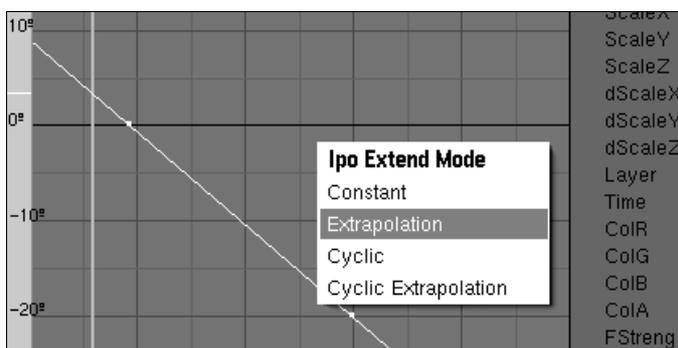


Рис. 5.13. Меню Ipo Extend Mode и результат работы функции Extrapolation

Теперь при запуске анимации объект будет неизменно вращаться на протяжении имеющейся временной линейки.

Возможности модуля IPO не ограничиваются манипуляциями с расположением объекта. Имеется возможность управления свойствами окружения, физики, текстур и материалов. Давайте рассмотрим небольшой пример работы с материалом.

Основные режимы IPO сгруппированы по смыслу и расположены в меню **Show IPO type** (Типы IPO). Именно здесь можно выбрать группу каналов для решения своей задачи (рис. 5.14).

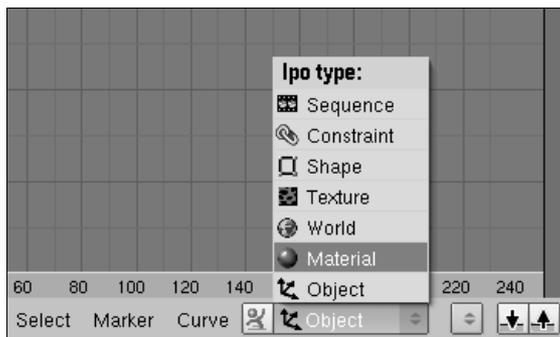


Рис. 5.14. Меню выбора групп каналов IPO

Внимание!

Состав меню групп каналов зависит не только от объекта, который в данный момент выделен, но и от его ведущих свойств. К примеру, если такому объекту не присвоен материал, то и в меню соответствующий пункт будет отсутствовать.

Создайте новый объект и присвойте ему материал с нулевым значением основного цвета RGB (0, 0, 0). Задачей этого примера ставится изменение цвета объекта по каналу *Red* (красный) в течение некоторого времени. Перейдите в окно редактора IPO и выберите в меню **Show IPO type** пункт **Material** (см. рис. 5.14). Обратите внимание, что состав возможных каналов IPO изменился.

Выделите канал *Red* (обозначение в списке — **R**) и создайте кривую в точке 0.0 (вертикальная шкала). Теперь добавьте новую точку со значением 1.0 и смещением на пару десятков кадров. Попробуйте запустить анимацию, и вы увидите, что объект постепенно меняет свой цвет с черного на красный (рис. 5.15).

Работа с редактором IPO не отличается большой сложностью и имеет немало интересных возможностей. Одна из самых полезных — это использование режима IPO-ключей. Им удобно пользоваться для выстраивания анимации по времени. Для перехода в этот режим необходимо в окне редактора IPO нажать клавишу <K>. При этом становится невозможным перемещение кривых по вертикали. Для изменения времени выполнения ключевого кадра достаточно ухватиться за него мышью и перетащить в нужное место (рис. 5.16).

Дополнительно с этим режимом можно использовать особый способ прорисовки объекта в 3D-окне — *Draw Keys* (Показать ключи). В этом случае программа отобразит ключевые позиции анимации в виде "призрачных" копий

объекта (рис. 5.17). Для включения этого режима достаточно нажать клавишу <K> в окне **3D View**.

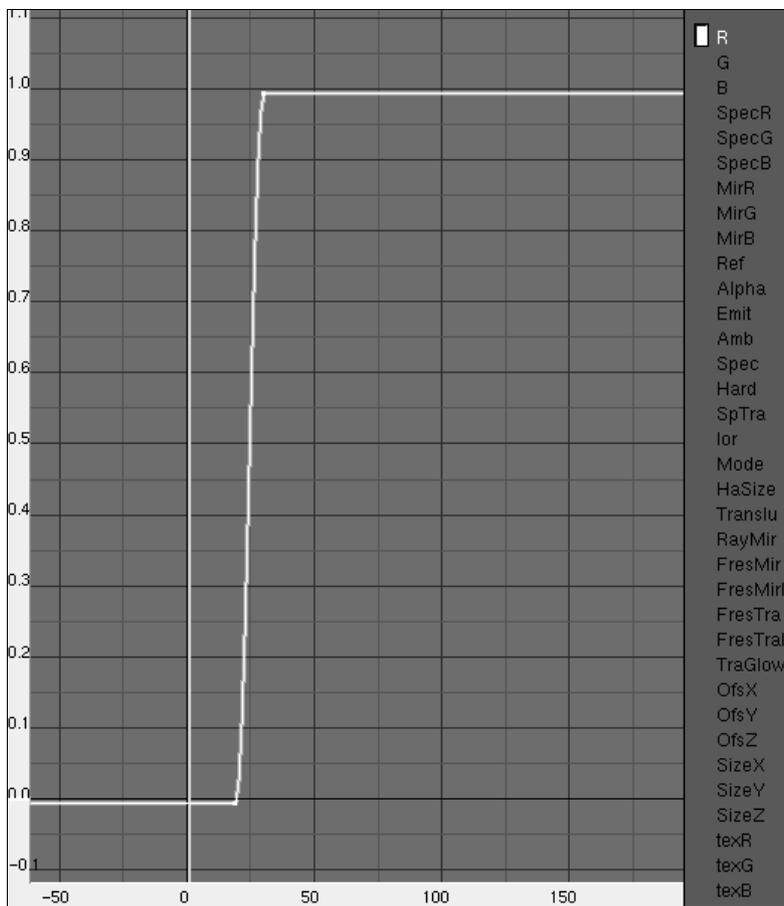


Рис. 5.15. Кривая и ключевые кадры для канала Red

В начале этой темы говорилось о том, что модулей IPO для одного объекта может быть несколько. Причем эта возможность имеется не только для основных его групп, но и в пределах какой-нибудь отдельно взятой группы. Важно запомнить, что созданный модуль является глобальным и его вполне можно использовать для другого объекта. По сути дела, работа с модулями аналогична работе с материалами, которые уже были рассмотрены ранее (см. разд. 4.2).

Для использования этой возможности имеется специальное меню, расположенное на заголовке окна **Ipo Curve Editor** (рис. 5.18).

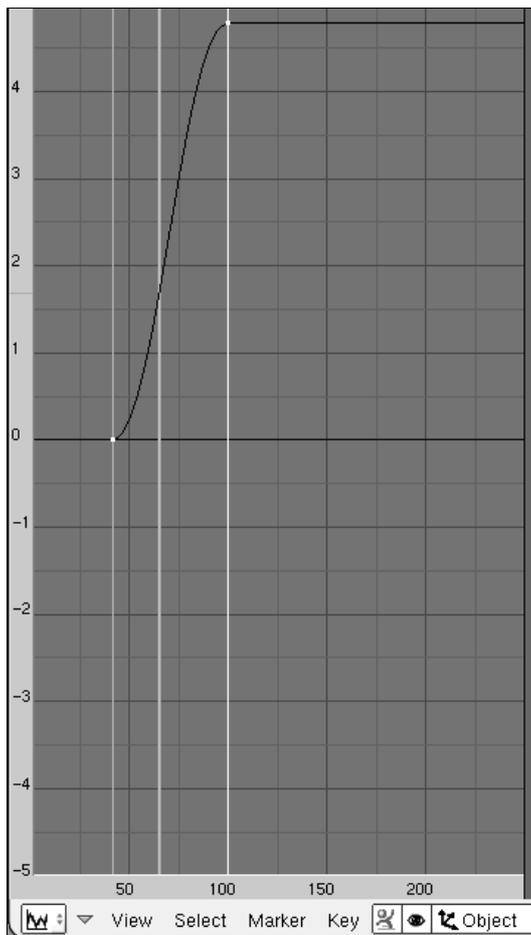


Рис. 5.16. В режиме Show Keys все ключи отображаются вертикальными линиями

Работа с уже созданными блоками заключается в следующем:

- ◆ отметить нужный объект для присоединения;
- ◆ выбрать нужную группу (см. рис. 5.14);
- ◆ выбрать имеющийся модуль из списка или создать новый (см. рис. 5.18).

Модуль, выбранный для нескольких объектов, является общим, и при его редактировании изменения отразятся для всех объектов. В некоторых случаях такое просто не подходит. К счастью, программа позволяет создать независимую копию имеющегося блока. Обратите внимание на кнопку **F**, которая расположена рядом с названием IPO (см. рис. 5.18). Она необходима для сохранения активного модуля, даже если он не используется ни одним объек-

том в сцене. При ее нажатии программа отобразит имеющиеся связи в виде числа по количеству присоединенных объектов. Если нажать кнопку с изображением этого числа, то редактор отсоединит этот модуль и сделает его уникальным по отношению к выбранному объекту.

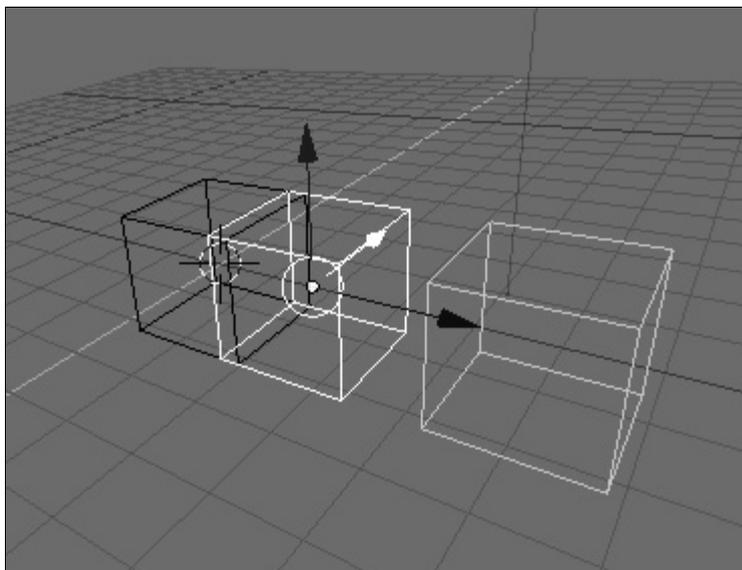


Рис. 5.17. Включение Draw Keys для объекта позволит точнее настроить анимацию



Рис. 5.18. Меню для выбора или создания модуля IPO

5.4. Как заставить объект следовать по траектории

Обычно, работая с движением объекта, вполне хватает тех возможностей, которые предоставляют стандартные средства анимации, но некоторые задачи требуют совсем иного подхода. К примеру, полет спутника вокруг планеты почти невозможно воссоздать при помощи ключей. Куда удобнее выглядит предлагаемый программой способ использования кривой для обозначения траектории движения.

В качестве пути могут выступать любые типы кривых, которые доступны в меню добавления объектов. Однако лучше использовать специальный объект `Path`, который представляет собой не что иное, как привычный NURBS, но уже специально настроенный для этой цели (рис. 5.19).

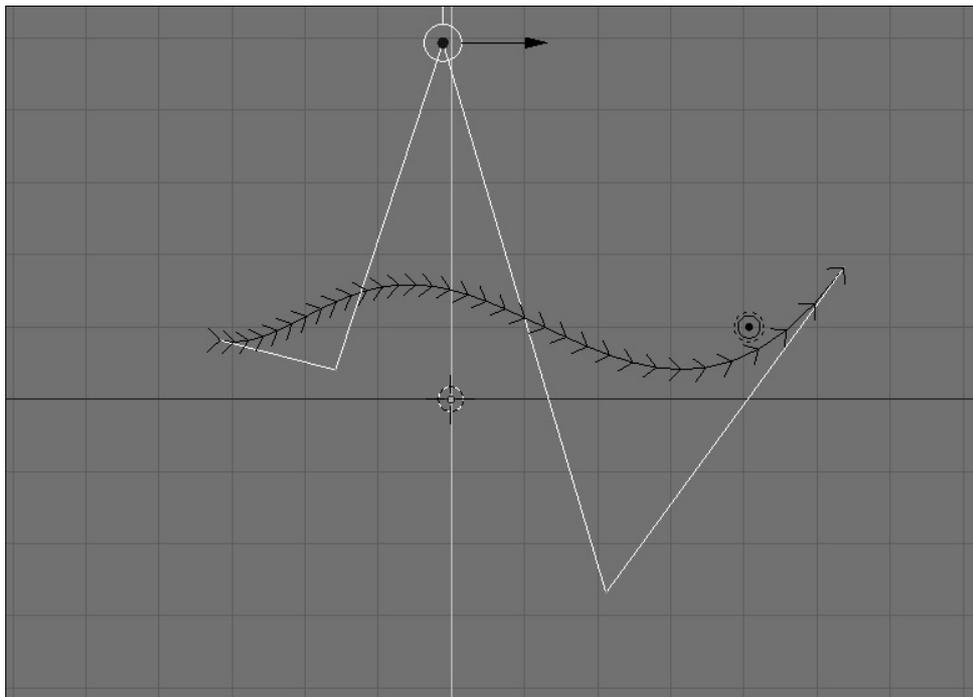


Рис. 5.19. Кривая `Path` используется для создания траектории движения

Редактирование `Path` ничем не отличается от уже рассмотренных в *главе 3* вариантов. Поэтому обратим более пристальное внимание на способ привязки кривой к необходимому объекту. Основой тут служит механизм `Constraints` (Ограничения) — особые правила для настройки анимации, но и не только.

Программа предлагает достаточно большой набор *привязок*, которые расположены на кнопочных панелях **Object** (<F7>) на вкладке **Constraints** (рис. 5.20).

Для настройки анимации по траектории служит привязка **Follow Path** (Следовать траектории). Естественно, привязка создается для объекта, который будет анимироваться (не для `Path`!). В появившемся окне нужно указать в поле **Ob**: название кривой (рис. 5.21).

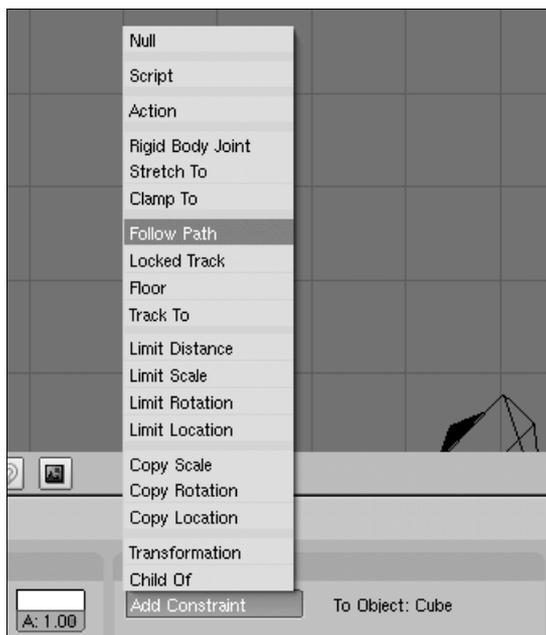


Рис. 5.20. Меню выбора привязки

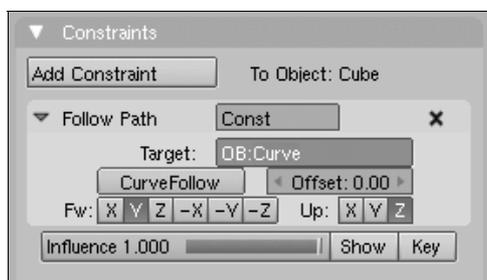


Рис. 5.21. Панель настроек Follow Path

Имеется два варианта движения объекта по пути:

- ◆ следование по траектории без изменения ориентации объекта;
- ◆ следование по траектории с изменением ориентации объекта по направлению кривой.

По умолчанию активен именно первый вариант (рис. 5.22).

Для второго варианта необходимо включить кнопку **CurveFollow** на панели **Follow Path** (рис. 5.23).

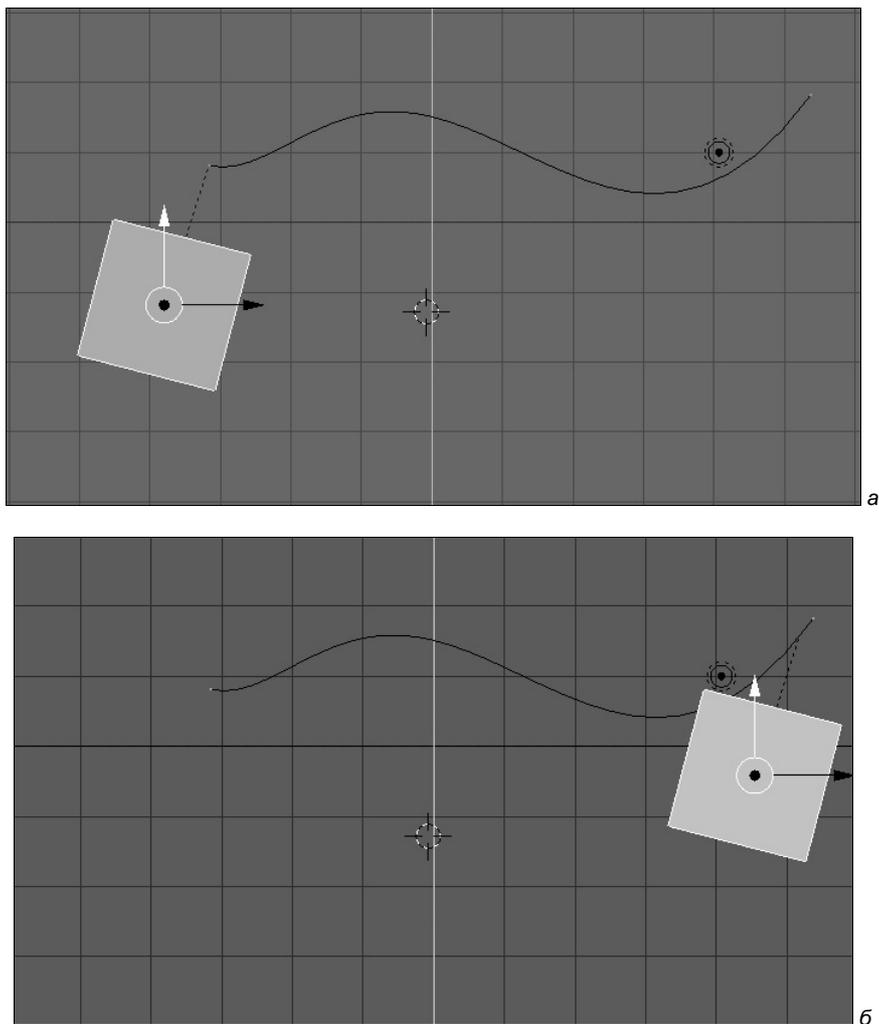


Рис. 5.22. Следование траектории без изменения ориентации: начальная точка (а) и конечная (б)

Панель **Follow Path** позволяет выбрать оси объекта, которые будут являться ведущими при движении по траектории. Кнопки группы **Fw**: (**X**, **Y**, **Z**, **-X**, **-Y**, **-Z**) предоставляют выбор оси объекта, которая будет направлена вдоль пути. Кнопки группы **Up**: (**X**, **Y**, **Z**) отвечают за вертикальное расположение объекта, т. е. какая из осей будет смотреть вверх.

Длина пути объекта **Path** при создании объекта равна 100 кадрам. Именно за это время ведомый объект должен пробежать всю траекторию. Изменить значение времени можно в настройках кривой на панели **Curve and Surface**

(<F9>) в поле **PathLen** (Длина пути). Однако есть более гибкий способ настройки времени движения — воспользоваться редактором IPO.

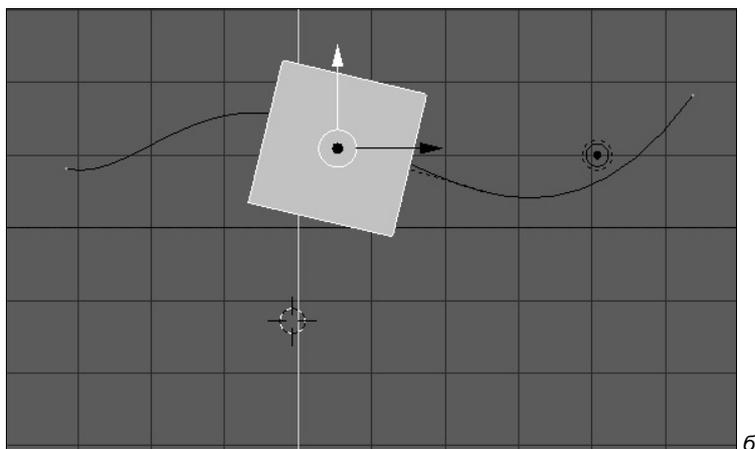
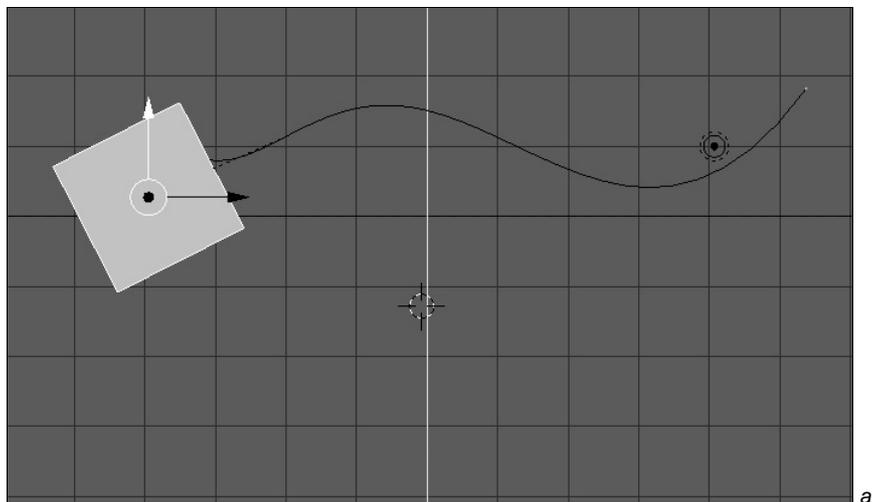


Рис. 5.23. Следование траектории с изменением ориентации (с включением **CurveFollow**): начальная точка (а) и конечная (б)

Если выделить кривую и выбрать в меню **Show IPO type** тип **Path**, то вашему вниманию предстанет единственный канал **Speed** (Скорость), который и отвечает за настройку времени движения объекта по этому пути (рис. 5.24).

Перемещением контрольных вершин по горизонтали можно изменять время движения объекта, но вот вертикальные значения лучше оставлять как есть. Таким образом появляется дополнительная возможность инвертирования

движения и управление линейностью. Запомните, что использование модуля IPO отключает соответствующие настройки самой кривой Path.

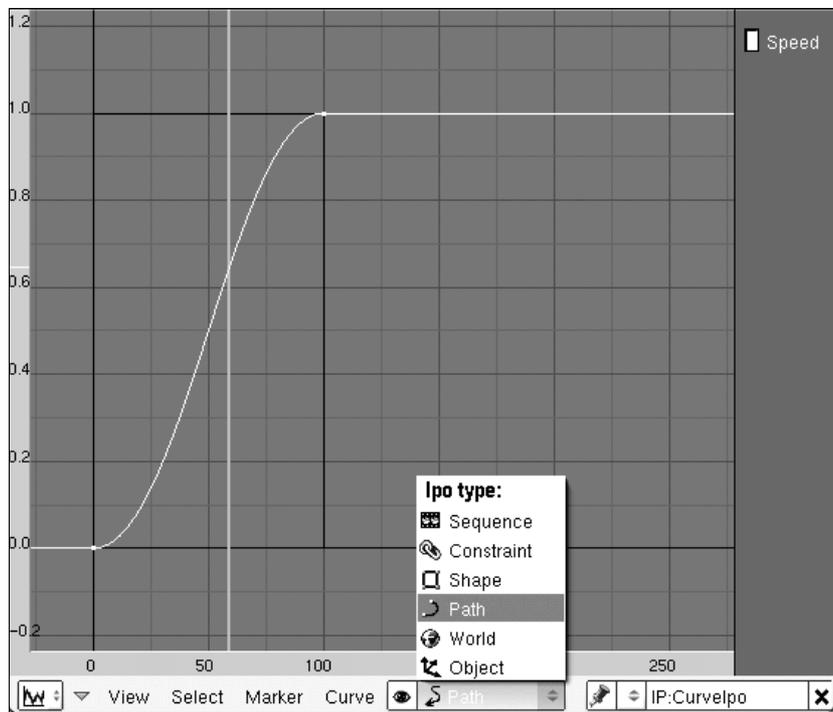


Рис. 5.24. Окно Ipo Curve Editor и кривая Speed объекта Path

5.5. Деформация объекта

Поддержка анимации объекта на уровне его структуры открывает поистине потрясающие возможности в реализации трехмерного мира. Мимика лица, колыхание травы, движение "живых" существ — все это возможно благодаря чудодейственным инструментам, предлагаемым редактором.

Под деформацией объекта понимается изменение его формы путем перемещения элементов структуры в пространстве и во времени. Имеется несколько способов, доступных пользователю:

- ◆ работа с абсолютными и относительными ключами;
- ◆ использование вспомогательных инструментов типа Lattice;
- ◆ создание скелета и анимации на его основе (см. разд. 5.6).

Начнем с того, что редактируемый объект может иметь тысячи вершин. Было бы неразумно для каждой из них создавать свою уникальную кривую IPO, и поэтому используется обычная система ключей. Впрочем, окно редактора IPO принимает самое живое участие в процессе анимации. Рассмотрим подробно первый способ анимации.

Откройте файл `5_shape1_1.blend`, который содержит модель образа сердца. Попробуем придать ему анимацию биения с помощью абсолютных ключей. Удобнее работать с деформацией, имея рядом открытое окно **Ipo Curve Editor** (используйте клавиши `<Ctrl>+<Влево>`).

Имеется два способа создания ключей. Первый, вам хорошо знакомый, — использование контекстного меню **Insert Key**, вызываемого клавишей `<I>`. Другим вариантом является работа со специальной панелью для управления ключами (рис. 5.25).

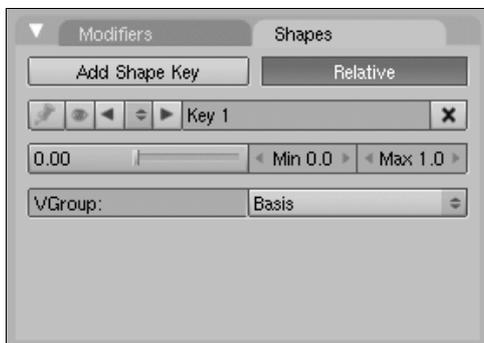


Рис. 5.25. Панель **Shapes** для управления ключами деформации (`<F9>`)

В любом случае алгоритм создания анимации выглядит следующим образом:

- ◆ создание базового ключа;
- ◆ перемещение на нужное количество кадров вперед;
- ◆ создание обычного ключа;
- ◆ деформация объекта;
- ◆ настройка времени выполнения.

Начальным пунктом этого списка стоит создание *базового ключа*. Именно от него начинается отсчет последующей анимации. Естественно, для рабочего объекта он является единственным. Важно запомнить, что базовый ключ определяет координаты текстур, для изменения которых он должен быть активным.

Выделите модель и нажмите клавишу `<F9>` для открытия панелей редактирования. Найдите вкладку с именем **Shapes** (Формы) и нажмите кнопку **Add**

Shape Key (Добавить ключ). Отключите в этой панели кнопку **Relative**. Так как до этого для этой модели не создавались ключи деформации, то он является первым и, соответственно, базовым. Просмотреть результат можно в окне **Ipo Curve Editor**, предварительно включив режим отображения **Shape** (рис. 5.26).

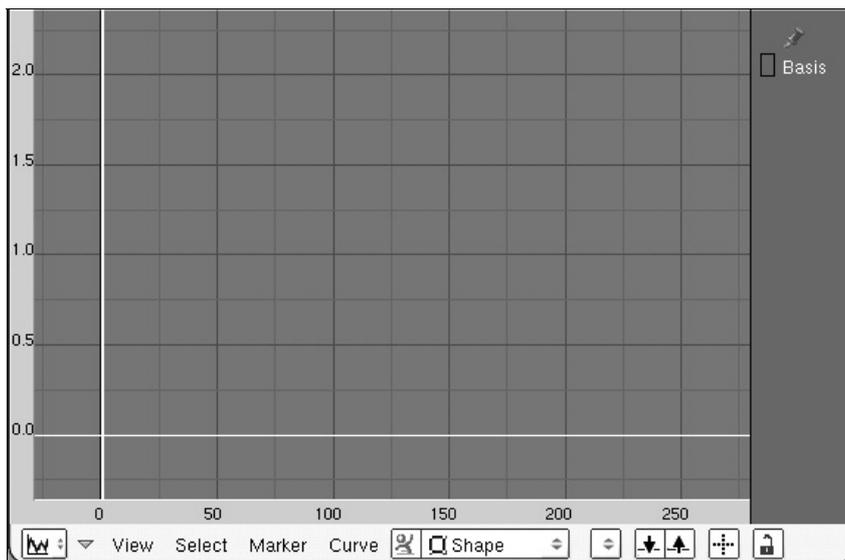


Рис. 5.26. Отображение ключей деформации в окне Ipo Curve Editor

Обратите внимание, что созданный базовый ключ отображается горизонтальной полосой желтого цвета. Все последующие ключи уже будут синего цвета.

Передвиньте курсор в окне **Timeline** на некоторое количество кадров вперед. Так как Blender позволяет после создания ключа сразу производить редактирование объекта, то нажмите клавишу <Tab>. В принципе, ничто не мешает вам сначала создать все ключи, а потом редактировать объект.

Создайте новый ключ и выделите центральные полигоны вокруг модели (рис. 5.27).

Совет

Для быстрого выделения замкнутого круга элементов используйте правую кнопку мыши совместно с клавишей <Alt>.

Включите режим пропорционального редактирования (<O>) и отмасштабируйте выделенные полигоны, придав форме объекта набухший вид (рис. 5.28).

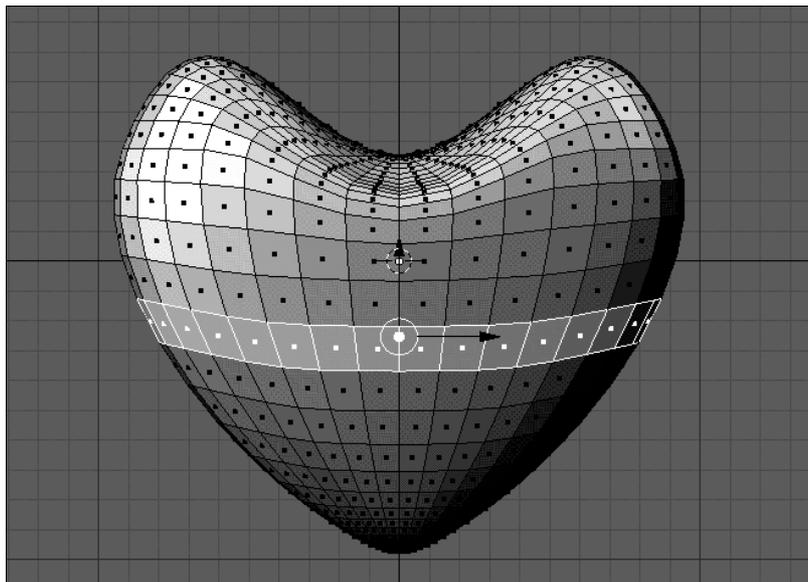


Рис. 5.27. Подготовка модели для деформации

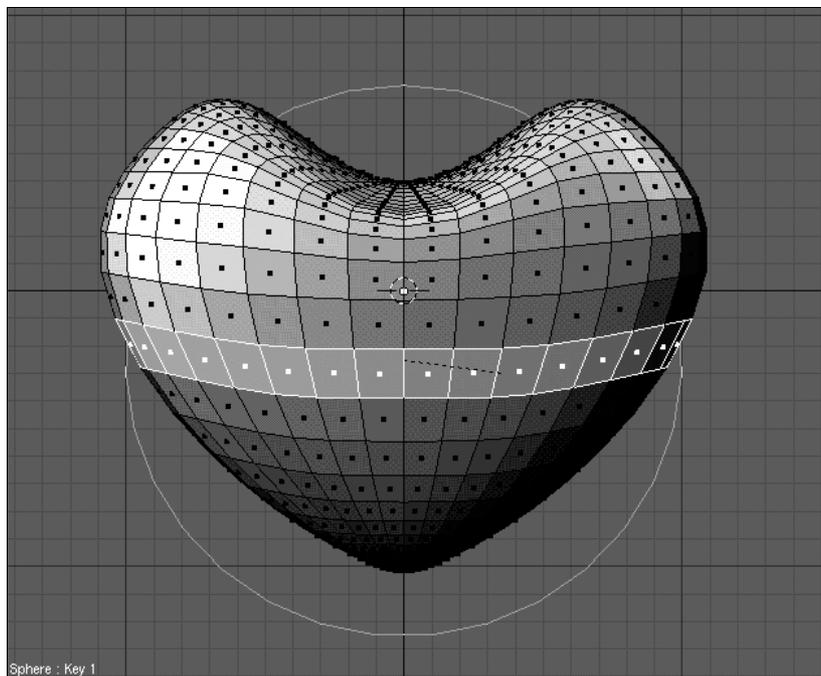


Рис. 5.28. Изменение формы для второго ключа

Просмотреть результат работы обоих ключей можно либо выделяя соответствующие горизонтальные полосы в окне **Ipo Curve Editor**, либо переключая клавиши со стрелками на панели **Shapes** (см. рис. 5.25).

Внимание!

Просмотр результата настроек ключей в режиме редактирования объекта невозможен. При попытке это сделать программа выдаст соответствующее предупреждение.

Если вы думаете, что на этом создание анимации закончено, то попробуйте проиграть ее, нажав сочетание клавиш <Alt>+<A>. Скорее всего, ничего из этого не выйдет. Дело в том, что программе нужен дополнительный параметр в виде кривой **Speed**, отвечающей за время выполнения.

Выделите в окне **Ipo Curve Editor** базовый ключ. При этом надпись **Basis** в информационной области должна стать белого цвета. Как вы, наверное, помните, кривая может быть создана путем удерживания клавиши <Ctrl> и щелчка левой кнопкой мыши в нужной области экрана. Создайте две вершины новой кривой таким образом, чтобы они лежали на обоих ключах (рис. 5.29). Теперь при запуске анимации модель придет в движение.

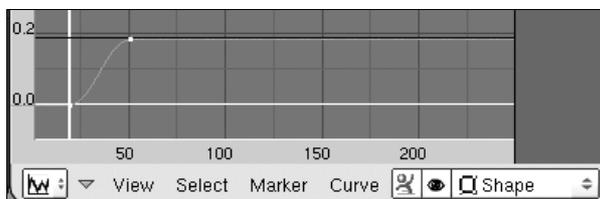


Рис. 5.29. Создание кривой, ответственной за настройку анимации

Для окончательного завершения работы над анимацией необходимо добиться постоянной пульсации сердца. Добавьте новую вершину кривой **Speed** немного со смещением на прямой ключа **Basis** для возврата анимации в исходное положение. Теперь осталось только применить режим **Cyclic** из меню **Ipo Extend Mode** (<E>) для циклического выполнения анимации (рис. 5.30).

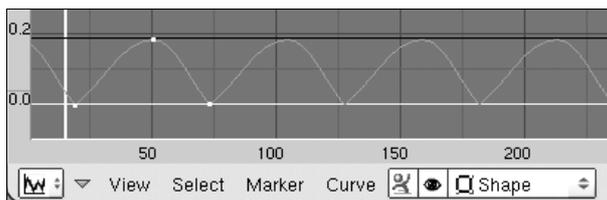


Рис. 5.30. Окончательная настройка анимации

Совет

Для создания эффекта ровности проигрывания анимации нет нужды вручную выравнивать кривую скорости. Достаточно изменить тип кривой на *Linear*, выбрав соответствующий пункт в меню **Ipo Type** (<T>).

Так как работа с относительными ключами и вспомогательным объектом *Lattice* базируется на технике, описанной ранее, то будут приведены лишь рекомендации и советы.

Относительные ключи вершин содержат информацию о разнице между базовым и последующим состоянием объекта. Это позволяет одновременно использовать несколько таких ключей, микшируя их для получения необходимого результата, что с абсолютными ключами невозможно. Принцип создания не отличается от уже рассмотренного. Использовать этот способ можно, включив кнопку **Relative** на панели **Shapes** (см. рис. 5.25).

Ранее рассматривался вспомогательный инструмент для моделирования *Lattice* (см. разд. 2.6). В некоторых случаях его можно использовать и для анимации объекта с помощью абсолютных или относительных ключей вершин. Достаточно сделать этот объект родительским по отношению к модели и производить манипуляции уже с ним.

5.6. Создание и редактирование скелета

Одной из самых сложных и эффективных анимационных техник является *скелетная* анимация. В основе ее лежит использование вспомогательных объектов, реализующих те или иные правила действий. Такие объекты называют *звеньями*, или *костями*, а объект целиком — *арматурой*, или *скелетом*. Именно арматура отвечает за анимацию модели. В дальнейшем модель привязывается особым способом к полученному скелету, и настраиваются места сгибов.

Так как кость является обычным объектом Blender, то к ней можно применять те же самые способы редактирования, которые рассматривались ранее.

Откройте новый проект и удалите куб. Для создания арматуры выберите пункт меню **Add | Armature**. Так как по умолчанию программа запускается с режимом просмотра окна **Top** (Сверху), то переключитесь в режим **Front** (Спереди), нажав клавишу <1> на добавочной клавиатуре (рис. 5.31).

После выполнения этой операции в проекте появляется арматурный объект с одной-единственной косточкой. В свою очередь, если рассматривать снизу вверх, то кость состоит из главной части (*Root*), тела (*Body*) и наконечника (*Tip*). В режиме редактирования (<Tab>) эти части можно выделять по отдельности.

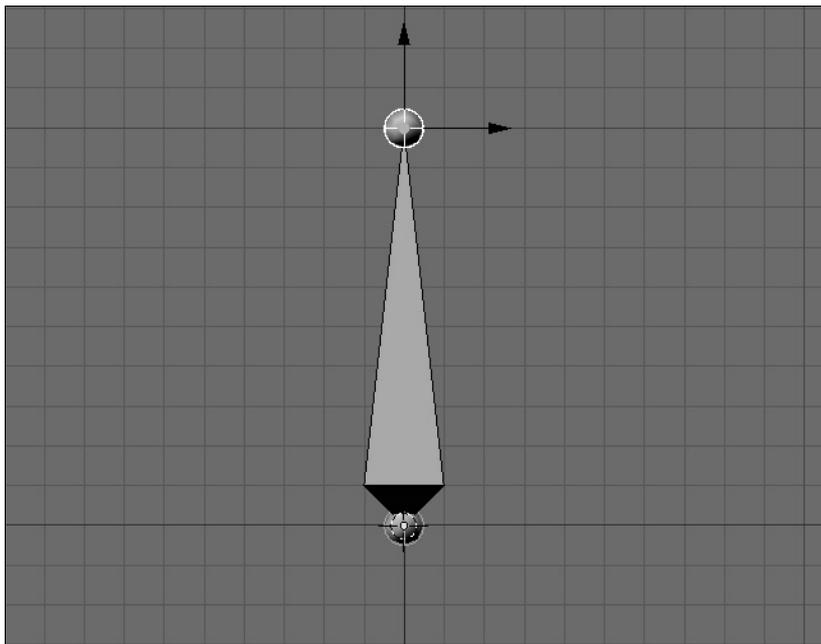


Рис. 5.31. Кость с параметрами по умолчанию

Для создания следующего звена достаточно отметить нужную часть (Root или Tip) исходной кости и произвести операцию выдавливания (<E>). Важно знать, что в зависимости от выбранного источника новая кость либо станет подчиненной родительскому объекту, либо будет независимой. Попробуйте создать и передвинуть кость от части Root. Как видите, она двигается свободно и не зависит от первой кости. В случае с выдавливанием от наконечника кость жестко привязана к родителю. Однако в обоих вариантах полученные звенья являются частями одной и той же арматуры.

Кости ведут себя по-разному при манипуляциях в зависимости от выделенной части. Создайте еще пару звеньев для первой кости (рис. 5.32).

Попробуйте, ухватившись за наконечник последнего звена, перетащить его на новое место (<G>). Хотя кость не отрывается от основной цепочки, она свободно вращается и масштабируется. Если выделить ее центральную часть, то перемещение уже затрагивает соседнее звено. На этом этапе важно запомнить, что режим редактирования существует только для подгонки созданного скелета и не влияет на конечную анимацию. Вне зависимости от того, как при анимации кости будут сдвигаться, переход в режим редактирования вернет арматуру в исходное состояние.

◆ Каждое звено имеет свое уникальное название, просмотреть и отредактировать которое можно на панели **Armature Bones** (Кости арматуры)

в кнопках редактирования (<F9>). Здесь же перенастраиваются иерархические связи. Чтобы изменить родителя выделенного звена, достаточно выбрать нужную кость из списка **child of** (рис. 5.33).

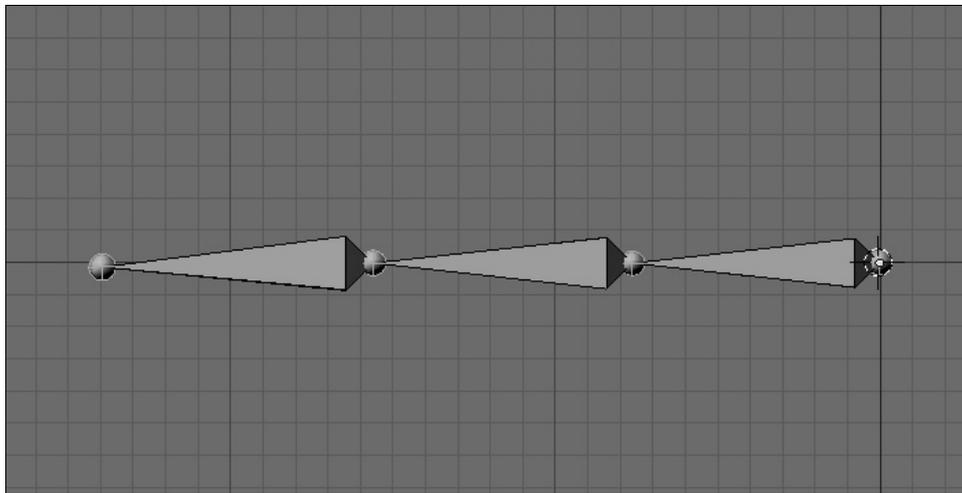


Рис. 5.32. Объект, состоящий из костей со связями, называется цепочкой

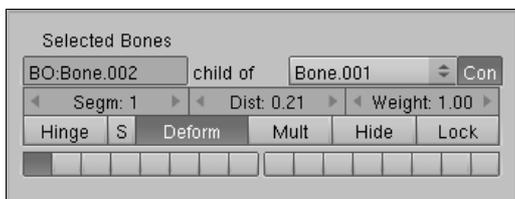


Рис. 5.33. Панель свойств кости

- ◆ По умолчанию программа первую кость называет Bone, а все последующие просто инкрементирует. В некоторой степени это неудобно. Так как готовая арматура может состоять из большого количества элементов, то важно давать им понятные названия. Это впоследствии пригодится при настройке анимации. Для переименования служит поле **ВО**: (рис. 5.33).

Существуют определенные правила наименования:

- ◆ слово может состоять из латиницы, цифр и некоторых знаков (точка, подчеркивание, тире). Однако первым символом обязательно должна быть буква;
- ◆ если скелет состоит из симметрично расположенных элементов, то к названиям костей должны быть прибавлены буквы R и L (от слов *right* и

left — правый и левый). Делать это желательно, т. к. программа имеет некоторые автоматические функции, значительно облегчающие жизнь при разработке скелета. Естественно, стиль наименования должен выдерживаться одинаково для всех. Пример написания: L_head, R_head, Head.L, Head.R.

Совет

Полезной функцией является возможность отображения названий костей прямо в 3D-окне рядом с их хозяевами. Для этого просто активируйте кнопку **Names** (рис. 5.34).

Настройки арматуры содержат большое количество опций, которые могут помочь при создании анимации. Все они расположены на панели **Armature** окна **Buttons Window** (рис. 5.34).

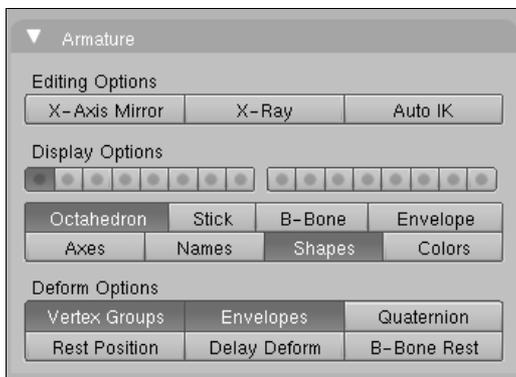


Рис. 5.34. Панель настроек арматуры

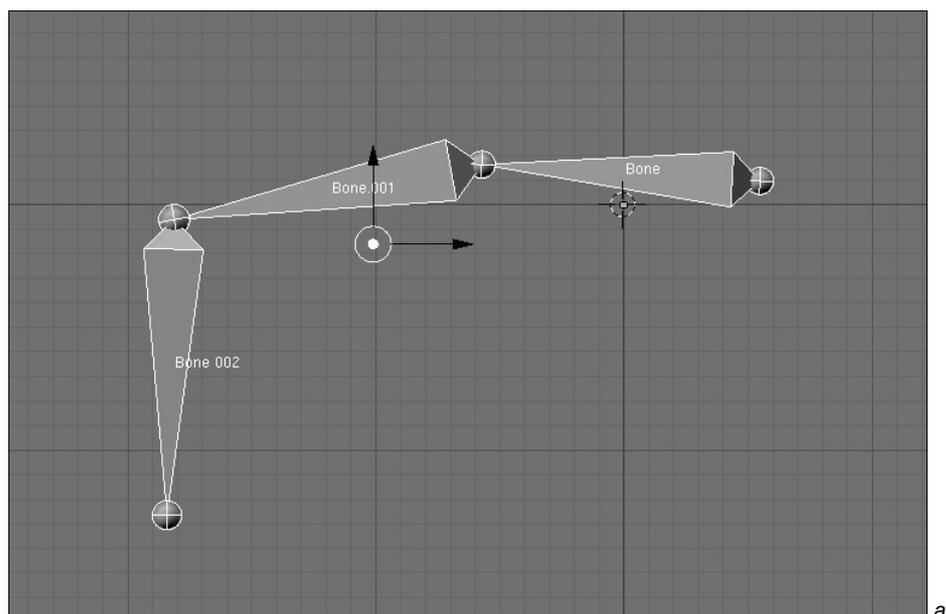
Обратите внимание на группу **Display Options** (Настройки отображения). Здесь можно придать желаемый вид начертанию костей (рис. 5.35). На самую анимацию эти параметры влияния не оказывают, но в некоторых случаях эта возможность совсем не лишняя.

5.7. Обычная и инверсная кинематика

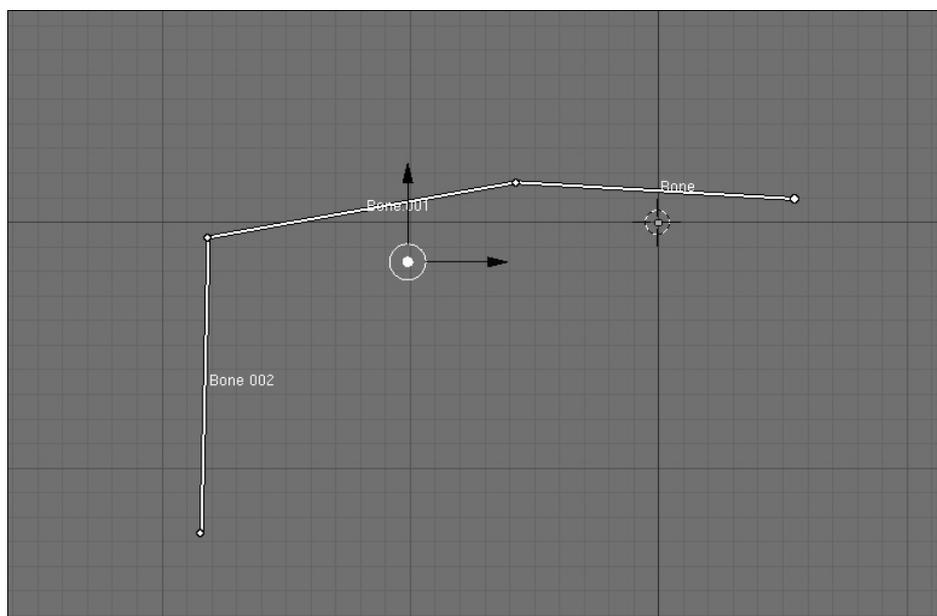
Арматура, в отличие от других объектов Blender, имеет три режима работы. Как и положено, их можно выбрать в меню **Mode** окна **3D View** (рис. 5.36).

Каждый из режимов имеет свои характерные особенности и предназначение:

- ◆ **Object Mode** (Режим объекта). Основные функции манипуляции с объектом в пределах сцены;



а



б

Рис. 5.35. Виды костей: Octahedron (а), Stick (б)

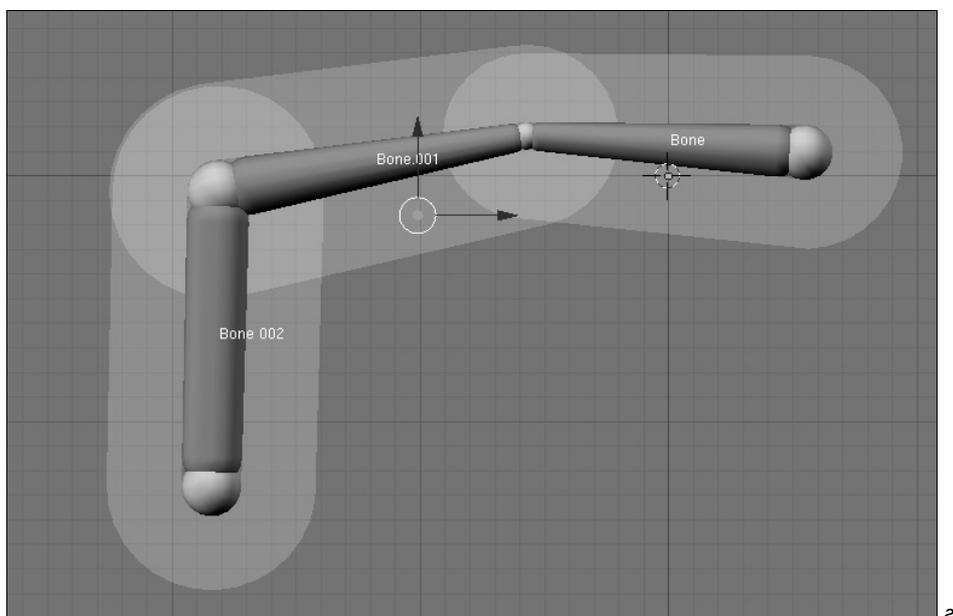
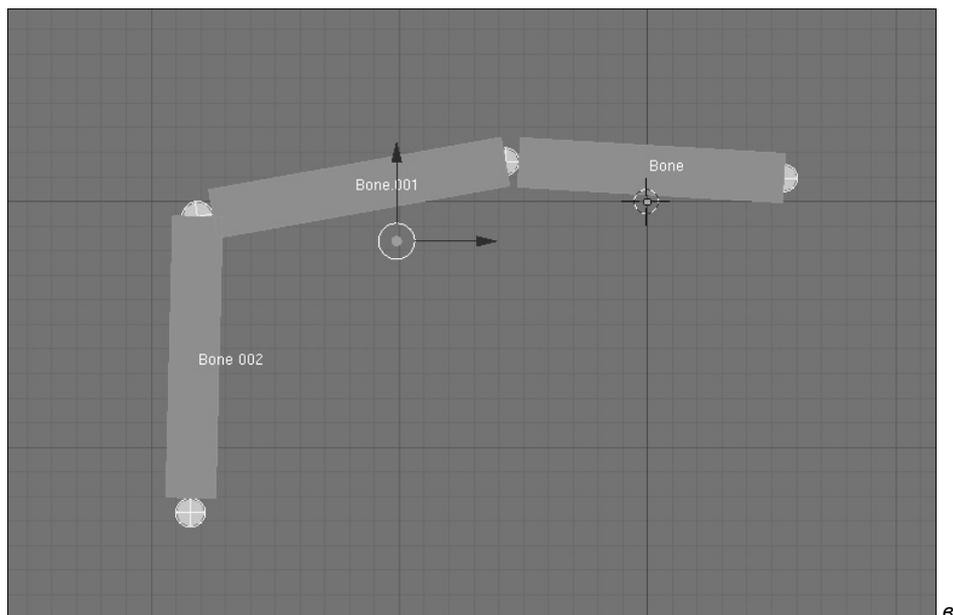


Рис. 5.35. Виды костей: B-Bone (а), Envelope (б)

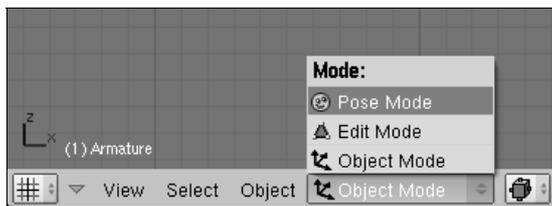


Рис. 5.36. Возможные режимы для скелета

- ◆ **Edit Mode** (Режим редактирования). Создание, удаление звеньев в арматуре, настройка связей и свойств. Придание формы скелету, которая будет являться базовой для анимации. Привязка модели к арматуре;
- ◆ **Pose Mode** (Режим позы). Основной режим для настройки кинематики объекта и построения анимации.

Обратим более пристальное внимание на режим позы. Главное отличие анимации арматурного объекта от любых других заключается в том, что она может совершаться только в этом режиме. Способы ее создания рассмотрим чуть позже, а пока важно разобраться с кинематикой.

Создайте арматурный объект, как на рис. 5.37, и настройте его иерархию в соответствии с рис. 5.38 (см. на диске файл *5_armature1_1.blend*).

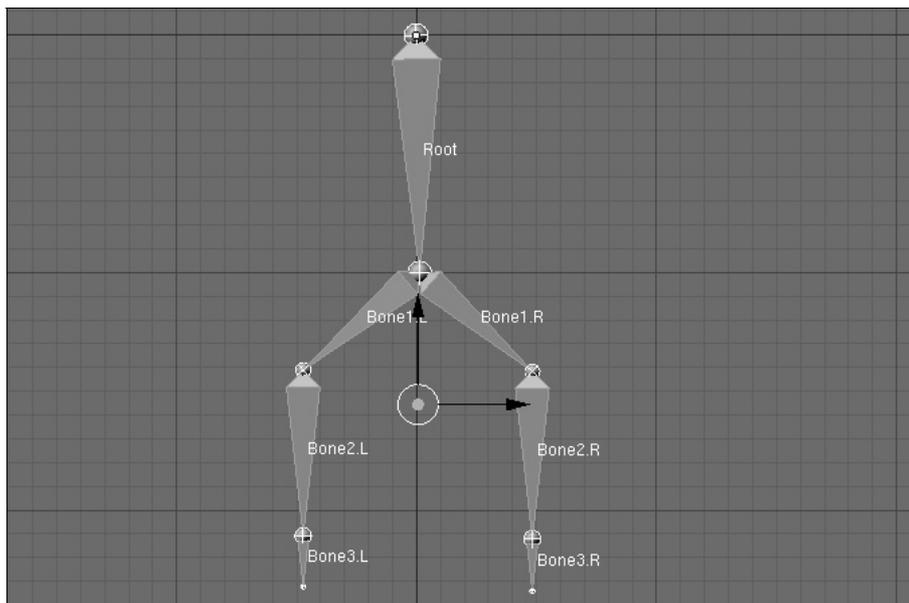


Рис. 5.37. Арматура для кинематики

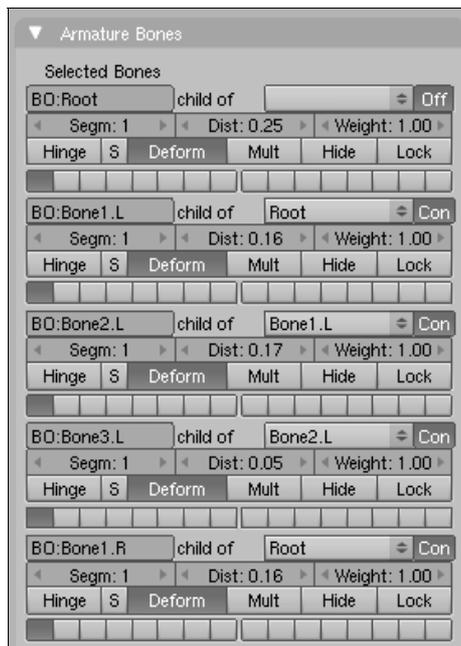


Рис. 5.38. Иерархия костей

Совет

Blender умеет автоматически переименовывать звенья в соответствии со своими требованиями и симметрией скелета. Для этого достаточно дать названия для левой половины арматуры и выбрать в 3D-окне функцию **Armature | AutoName Left-Right**. При этом должен быть включен режим редактирования.

Перейдите в режим позы и попробуйте подвигать разными звеньями объекта. Как видите, старшие по иерархии кости остаются неподвижными, а все присоединенные двигаются вместе со своим родителем. Подобное явление называется *прямой кинематикой*.

Программа имеет несколько полезных функций для управления костями. Иногда бывает необходимо зафиксировать ориентацию определенных звеньев в сцене. Выделите главную кость арматуры Root и нажмите клавишу <R> для ее ротации. Как видите, вся арматура послушно вращается вместе с ней (рис. 5.39, а). Снимите выделение и отметьте кость Bone1.R. Найдите в панели редактирования <F9> вкладку **Armature Bones** и кнопку **Hinge** (Шарнир). Если включить ее и повернуть объект снова, то можно увидеть, что отмеченная кость, равно как и ее звенья, не изменяет свою ориентацию (рис. 5.39, б).

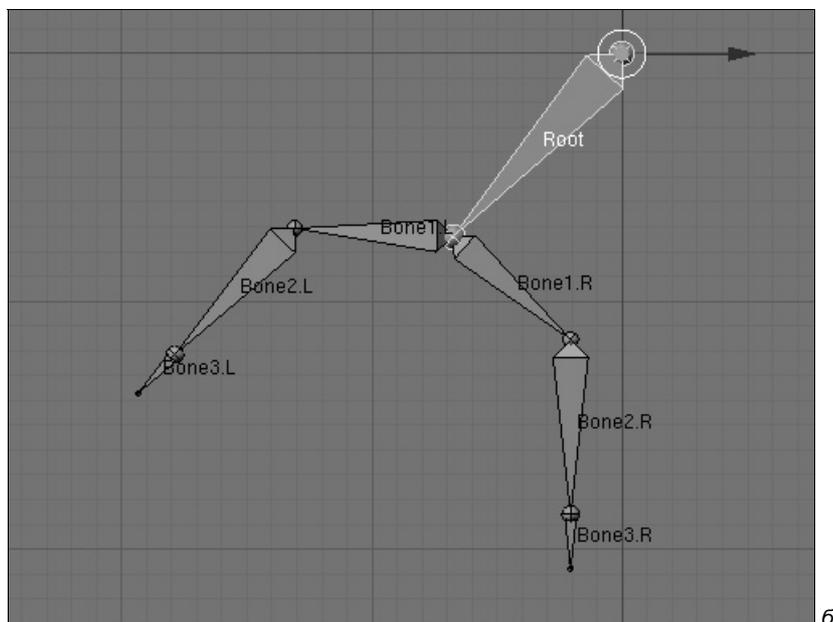
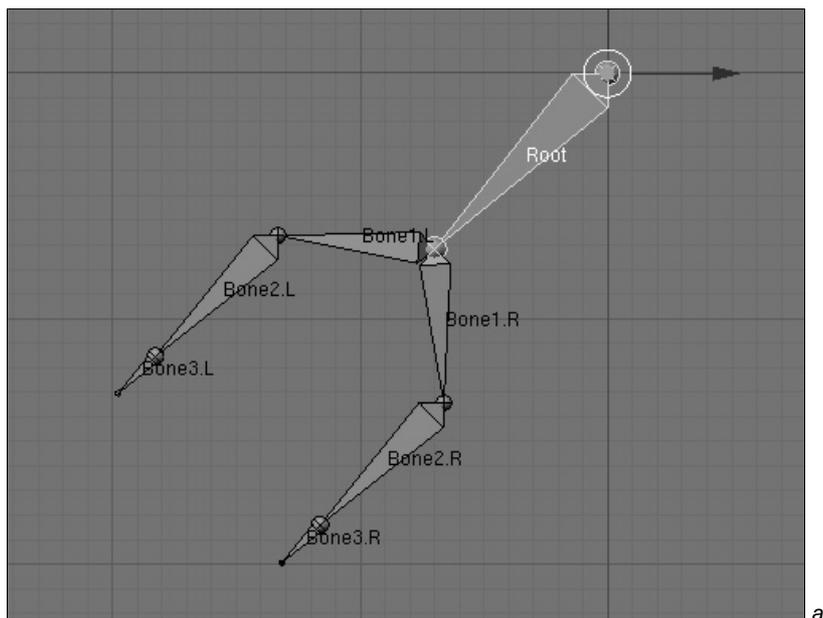


Рис. 5.39. Результат вращения объекта: настройка по умолчанию (а), включение функции Hinge (б)

Совет

Перед настройкой анимации необходимо ориентировать по оси Z все звенья арматуры во избежание неправильной работы некоторых функций редактора. Сделать это можно, выделив арматуру в режиме редактирования и нажав сочетание клавиш <Ctrl>+<N>. В появившемся меню нужно выбрать пункт **Clear Roll** (Очистить вращение).

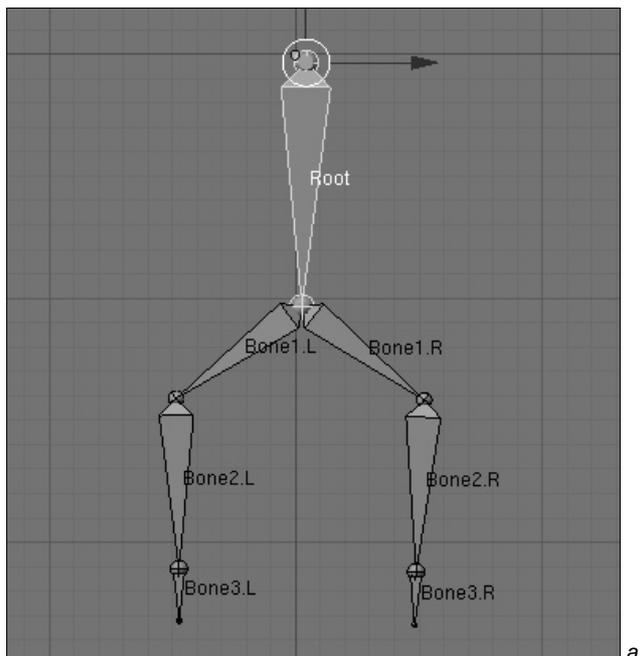
В некоторых случаях обычный режим поведения костей для решения задачи не подходит. На помощь может прийти *инверсная* кинематика. Чтобы лучше понять, для чего она нужна, проведем небольшой эксперимент. Поднимите свою руку и проследите за поведением ее частей. Движение начинается от плеча, при этом все остальное послушно двигается за ним. В понятии программы плечо является родительским объектом, а все остальные его потомками. Таким образом, налицо обычная кинематика. Совсем другое поведение происходит при рукопожатии. В этом случае, колебательные движения начинаются от кисти и передаются выше. Все звенья одной цепочки так или иначе участвуют в движении. При этом основная иерархия не нарушается. Такое явление называется инверсной кинематикой.

Существует несколько способов создания инверсной кинематики. Самый простой — воспользоваться функцией автоматической настройки кинематики. Войдите в режим **Pose** и на панели **Armature** (см. рис. 5.34) нажмите кнопку **Auto IK** (Автоматическая инверсная кинематика). Вот и все. Программа послушно подберет оптимальный вариант для данного скелета. Попробуйте, ухватившись за кость **Root**, перемещать объект. По мановению мыши скелет пустится в пляс (рис. 5.40).

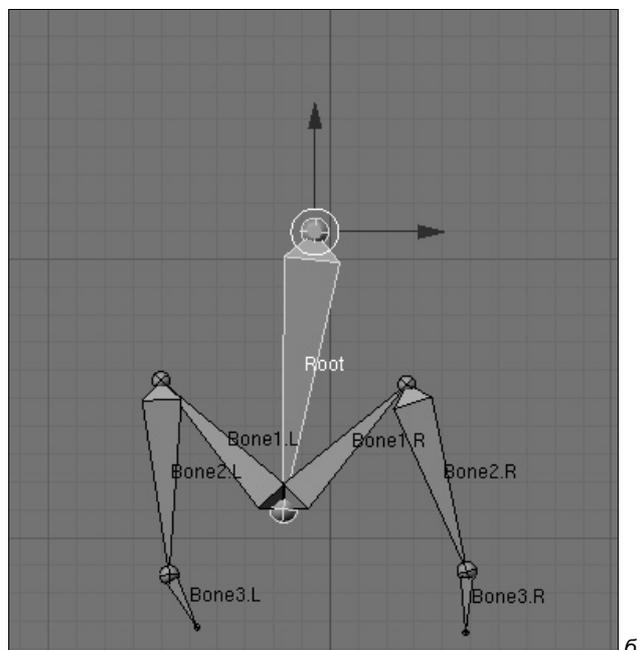
Иногда результат работы автоматики не приносит должного эффекта. В этом случае можно настроить кинематику вручную.

Для этого сначала нужно определиться с костью, которая будет выступать в качестве IK-привода. Пусть таким звеном будет являться кость **Bone3.R**. Выделите ее, нажмите сочетание клавиш <Shift>+<I> и в появившемся меню выберите пункт **Without Target** (Без назначения). Программа создаст для этой кости привязку **IK Solver**. Однако для правильного просчета редактору нужен объект, который будет являться последним звеном в IK-цепочке. По умолчанию таким звеном становится корневая кость. При этом конечные кости соединяются пунктирной линией оранжевого цвета, сигнализирующей об имеющейся инверсной цепи (рис. 5.41).

Если вы попытаете переместить звено **Bone3.R**, то заметите, что поведение цепочки ничем не отличается от поведения в автоматическом режиме. Однако, поменяв конечную кость на другую, можно получить совсем другие результаты. Сделать это можно в настройках имеющейся привязки (рис. 5.42).



a



б

Рис. 5.40. Инверсная кинематика в действии:
исходное положение (а), положение 2 (б)

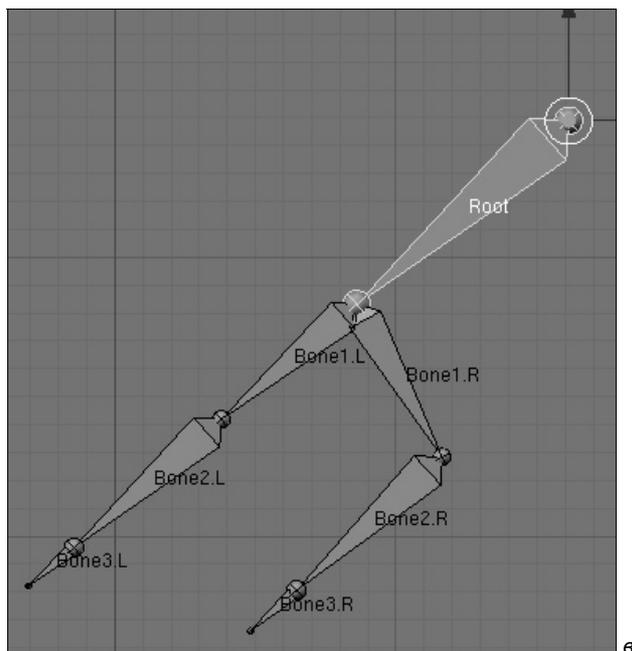


Рис. 5.40. Инверсная кинематика в действии: положение 3 (е)

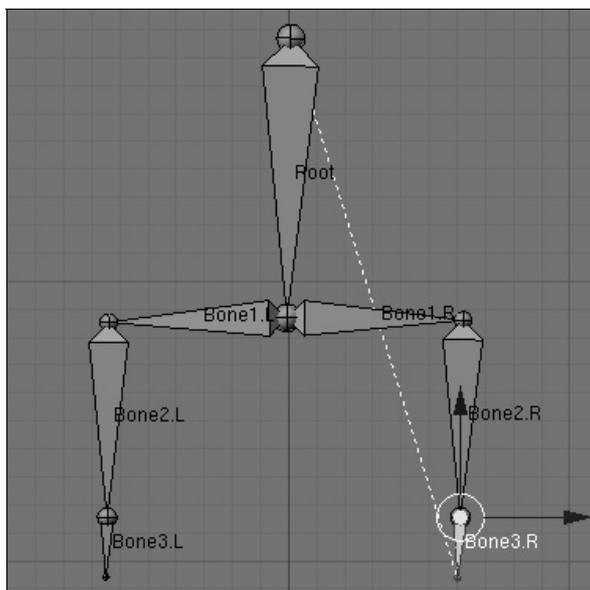


Рис. 5.41. Кость с присвоенной привязкой IK Solver выделяется оранжевым цветом

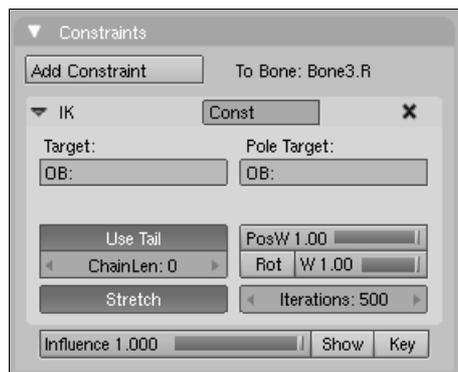


Рис. 5.42. Панель привязки IK Solver

Обратите внимание на поле **ChainLen** (Изменить длину). Именно здесь можно выбрать необходимый номер конечного звена. Нулевое значение обозначает привязку к главному объекту вне зависимости от количества промежуточных элементов. Введите 3 для выбора Bone1.R в качестве конечного звена (рис. 5.43).

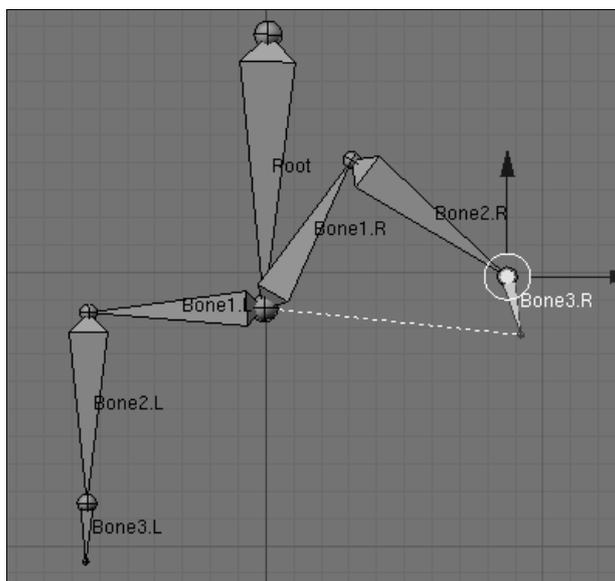


Рис. 5.43. Поведение арматуры при изменении конечного звена

Несложно заметить, что при такой настройке цепи невозможно добиться "пляшущего" результата, как при использовании автоматике. Решить эту проблему можно, если привязать выбранные кости не к элементам арматуры,

а к отдельным объектам, которые располагаются в нужных местах. В качестве такого объекта обычно используют *Empty*.

Создайте *Empty* в месте предполагаемой привязки (рис. 5.44).

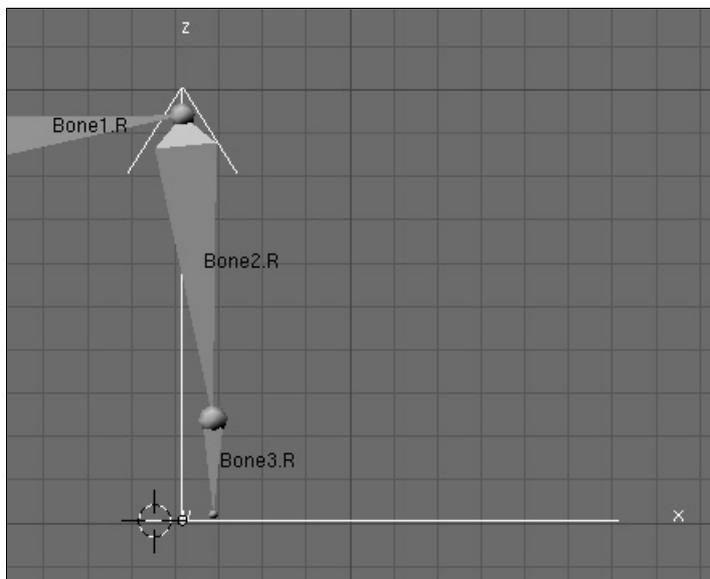


Рис. 5.44. Объект *Empty* будет использоваться в качестве конечного звена

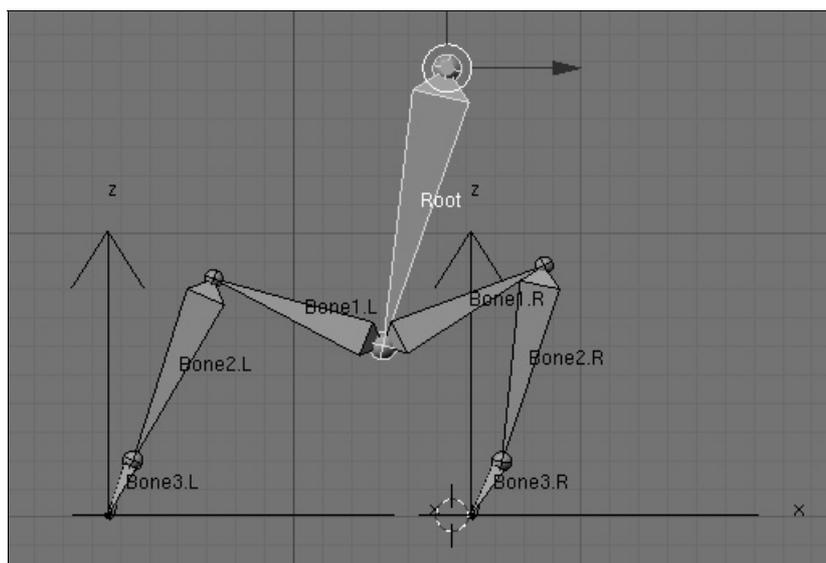


Рис. 5.45. Движение арматуры с привязками к *Empty*

Введите его название (по умолчанию — `Empty`) в поле **Target** (Назначение) звена `Bone3.R`. Заметьте, что при этом звено окрашивается в желтый цвет и переносится в центр вспомогательного объекта.

Теперь выделите кость `Bone3.L` и добавьте к ней привязку **IK Solver** из меню **Add Constraint** (Добавить привязку). Создайте новый объект `Empty` и разместите его напротив первого объекта. Для завершения осталось только изменить содержимое поля **Target** (рис. 5.45).

Совет

Создание привязки кости к вспомогательному объекту можно сделать и горячими клавишами `<Shift>+<I>`. При этом в появившемся меню следует выбрать пункт **To New Empty Object** (Привязать к новому объекту `Empty`), и программа создаст новый объект с уже настроенной привязкой.

5.8. Нарращиваем "мясо"

Итак, модель создана, арматура настроена — пора объединить все в одно целое. Такой процесс в мире трехмерной графики называется *одеванием* (*skinning*). Смысл его заключается в том, что арматура объявляется предком модели, что позволяет переносить ее анимацию на сам объект. Делается это просто. Выделяется модель, затем скелет и производится привязка. Рассмотрим этот процесс на простом примере.

Откройте новый проект и создайте арматуру, состоящую из трех звеньев. Затем в качестве тестового объекта растяните имеющий куб по размеру цепочки (рис. 5.46).

Пусть объект у нас будет представлять собой высокополигональную модель, поэтому разбейте его структуру несколько раз функцией `Subdivide`.

Поместите скелет в центр объекта и нажмите на панели **Armature** кнопку **X-Ray**. Последнее действие позволяет видеть арматуру, даже если она полностью перекрыта моделью.

Для внедрения скелета в иерархию объекта выделите модель, а затем арматуру, удерживая нажатой клавишу `<Shift>`. Нажмите `<Ctrl>+<P>` и в появившемся меню выберите пункт **Armature**. Это действие откроет еще одно подменю, в котором нужно выбрать опцию **Name Groups** (Группы имен).

Итак, что же мы сделали? Во-первых, внедрили арматуру в иерархию модели. Для проверки можно просмотреть структуру сцены в окне **Outliner** или просто попробовать перетащить скелет на другое место (модель должна двигаться вместе с ним). Во-вторых, команда **Name Groups** автоматически создала для куба группы вершин по количеству звеньев в арматуре. При этом имена созданных групп соответствуют именам костей.

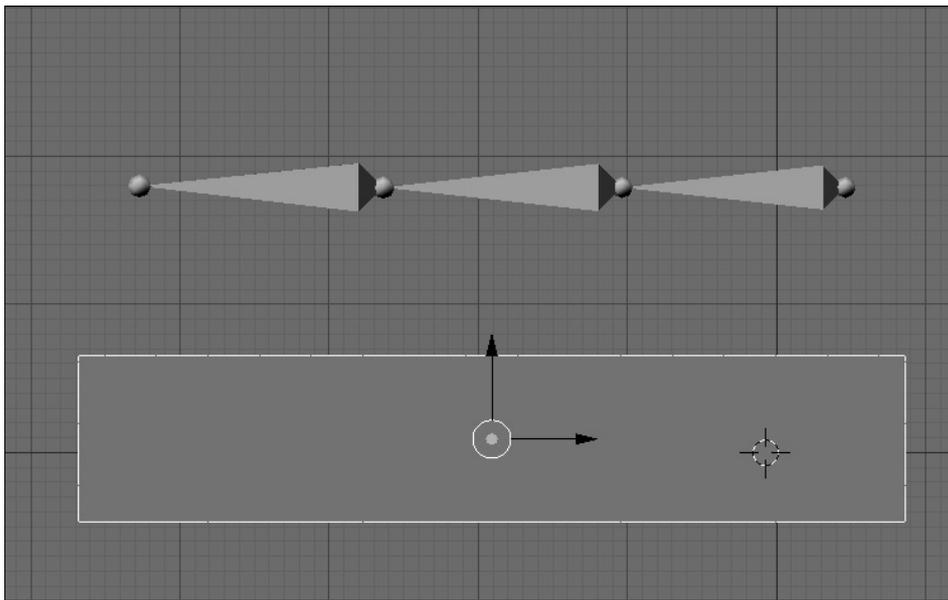


Рис. 5.46. Арматура и скелет для skinning

А теперь давайте разберемся, что такое группы вершин и для чего они нужны. Хотя скелет уже находится в иерархии объекта, тем не менее кости "не знают", для каких частей модели они предназначены. Ведь при перемещении звена должна двигаться определенная часть объекта и деформироваться место сгиба. Использование групп вершин позволяет назначить каждой кости часть модели, на которую они (кости) будут влиять.

Выделите модель, нажмите клавишу <Tab> для перехода в режим редактирования. На панели **Link and Material** расположена область **Vertex Groups**, которая и отвечает за работу с группами вершин (рис. 5.47).

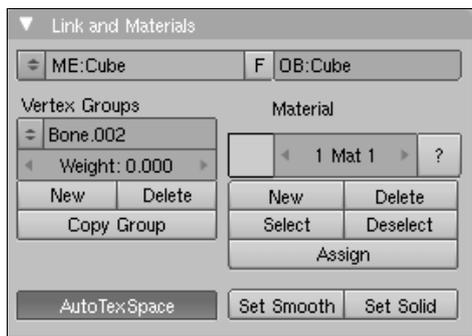


Рис. 5.47. Панель **Vertex Groups** для управления группами

Выполненная ранее команда **Name Groups** создала необходимые группы, но не присвоила им какие-либо вершины. Присвоим вручную.

Для выбора группы в этой панели рядом с названием имеются курсорные кнопки. В этом же поле можно и переименовать активную группу.

Выберите группу **Bone** и отметьте рамкой выделения () область объекта, соответствующую для этой кости (рис. 5.48).

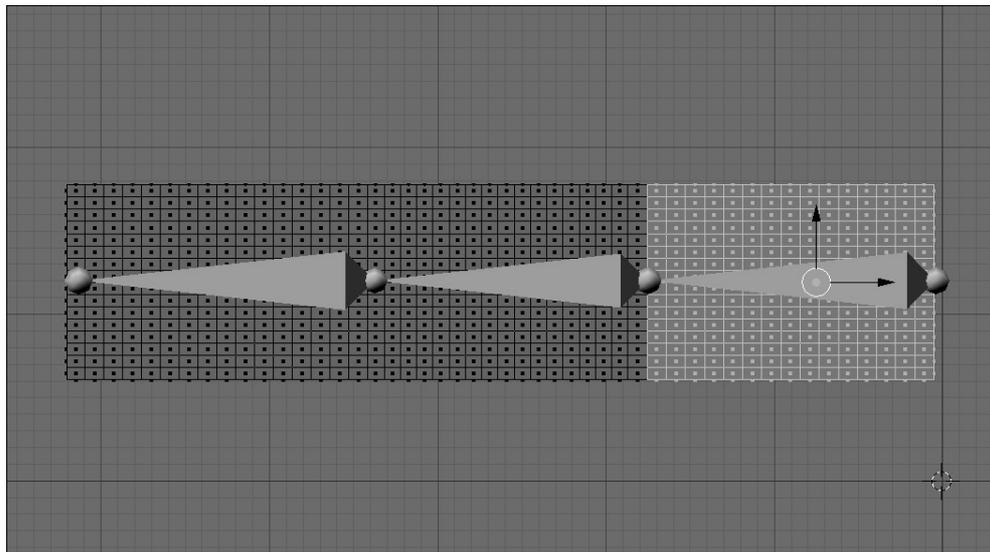


Рис. 5.48. Выделенная группа вершин для кости **Bone**

Для закрепления этой области за активной группой нажмите кнопку **Assign** (Присоединить). Чтобы проверить правильность привязки, снимите выделение (<A>) и нажмите кнопку **Select** (Выделить). Эта кнопка показывает вершины, закрепленные за данной группой. Прodelайте подобную операцию для оставшихся костей.

Совет

Иногда бывает нужно добавить вершины в уже имеющийся массив группы. Просто выделите старые кнопкой **Select**, отметьте дополнительные и заново присвойте их группе командой **Assign**.

Теперь при перемещении костей в режиме **Pose Mode** части модели будут послушно двигаться за ними (рис. 5.49).

Обратите внимание на параметр **Weight**, который отвечает за вес выделенных вершин. Значение веса влияет за уровень зависимости от движения звена арматуры. Чем меньше вес, тем хуже вершина реагирует на перемещение. При

нулевым значении группа вообще не участвует в анимации. Таким образом можно контролировать качество деформации части объекта.

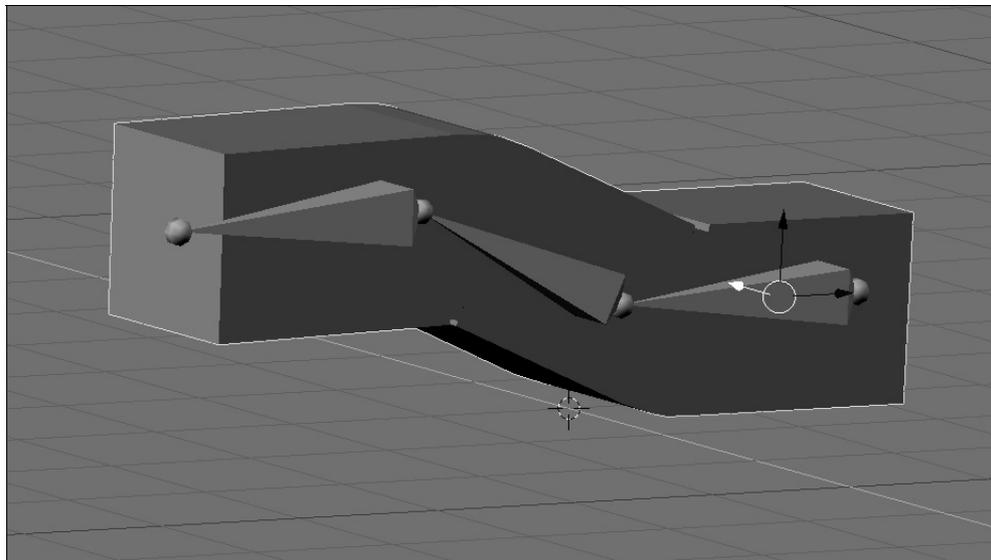


Рис. 5.49. Выделенная группа вершин для кости Bone

Имеется и альтернативный графический способ назначения веса. Его главное достоинство в наглядности происходящего и точности настройки.

Переключите арматуру в режим позы и выделите модель. Найдите в списке режимов пункт **Weight Paint** (Рисование веса). В зависимости от выделенной кости и настройки веса группы вершин объект предстанет окрашенным в разные цвета (рис. 5.50).

Градация цветов обозначает уровень зависимости вершин от конкретной выделенной кости. Чем ярче цвет, тем больше значение веса. Минимальному значению соответствует темно-синий цвет, а максимальному — красный. Таким образом, рисуя определенным цветом на поверхности объекта, можно задать нужный вес вершин.

Основные настройки этого режима находятся в панели **Paint** (Рисование). Здесь можно выбрать размер кисти, тип смешивания и нужный вес (рис. 5.51).

Внимание!

При рисовании на поверхности объекта следует помнить, что окраске подвергаются только видимые вершины. Не забывайте прокручивать и просматривать весь объект.

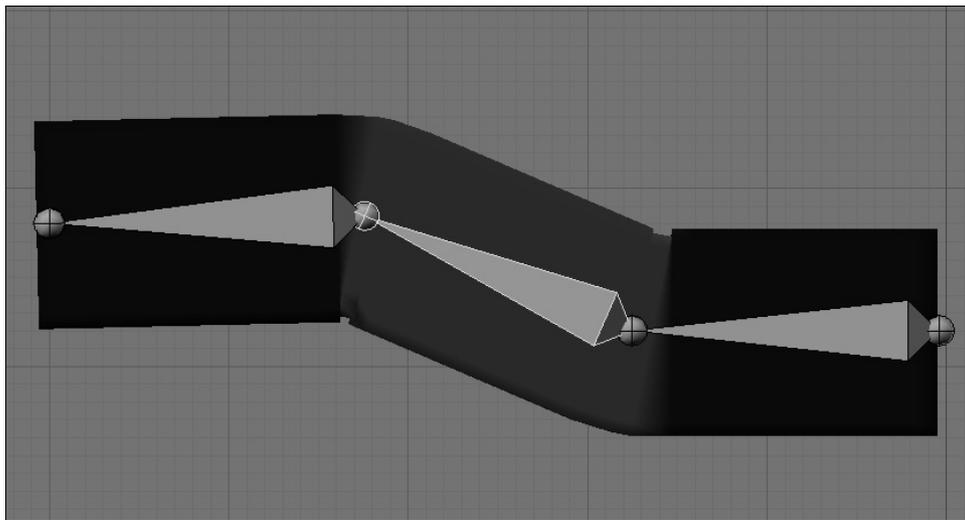


Рис. 5.50. Работа в режиме Weight Paint

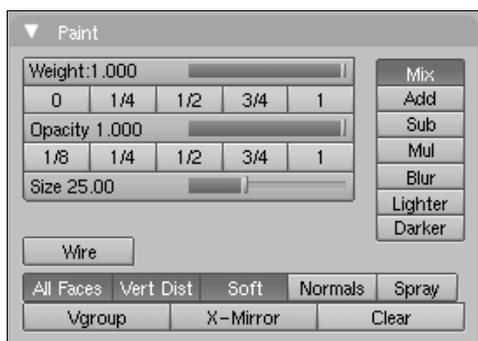


Рис. 5.51. Настройки работы кисти

Режим `Weight Paint` хорош тем, что позволяет сразу оценить результат работы. Одновременно с окраской объекта происходит перерасчет деформации и вывод на экран. Также можно не выходя из этого режима управлять звеньями арматуры.

5.9. Редактор действий для арматуры

Работа со скелетной анимацией имеет свои тонкости и особенности из-за сложности самого объекта. Ведь в составе арматуры могут иметься десятки элементов в совокупности с привязками. Хотя основа такой анимации все так же базируется на использовании ключей, работать с редактором IPO уже ста-

новится затруднительно. Дело в том, что звенья арматурной цепочки с обратной кинематикой будут иметь отдельные модули IPO для каждого звена. В этом случае редактирование превращается в кошмар. Нужно также учитывать, что более или менее сложная анимация имеет не одно действие, а целый комплекс.

В предыдущем предложении появилось слово "действие". Именно так называется предлагаемый программой инструмент для организации и тонкой настройки скелетной анимации.

Action Editor, или, как его еще называют, *редактор действий*, представляет собой очередной уровень структуризации над привычным IPO. В его функции входит не только редактирование отдельных моментов анимации, но и создание законченных по смыслу действий. Какой же смысл вкладывается в это слово? Возьмем для примера анимацию человека. Он может сидеть, бежать, обедать или спать, в общем, все, что входит в план анимационного проекта и может быть разбито на логичные модули.

Теперь настала пора попробовать создать свою анимацию. Откройте файл 5_action1.blend. Он содержит простую арматуру с инверсной кинематикой (рис. 5.52).

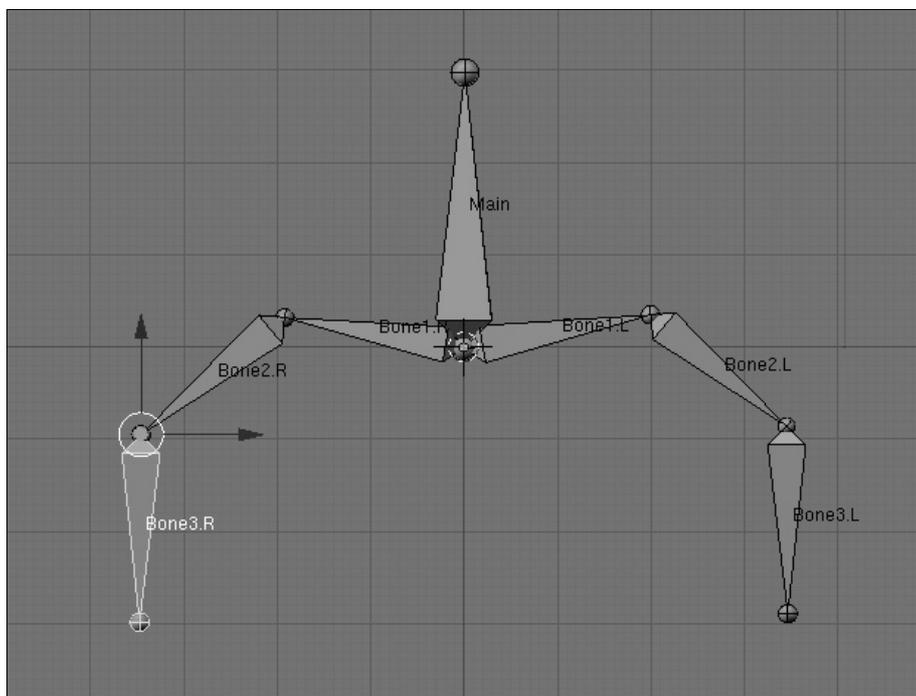


Рис. 5.52. Арматура для примера

Для удобства работы лучше воспользоваться оконной заготовкой *Animation* (<Ctrl>+<Влево>). Так как окно **Buttons Window** сейчас без надобности, то откройте в нем *Action Editor*, выбрав его из глобального меню или нажав сочетание клавиш <Shift>+<F12>.

План учебной анимации заключается в следующем. Пусть в течение нескольких секунд происходит выпрямление сначала левой стороны арматуры, а затем и правой. Для создания ее выполните следующие действия:

- ◆ включите кнопку записи в окне **Timeline** и проследите, чтобы рядом в списке режимов был включен **Add/Replace Keys** (см. разд. 5.2);
- ◆ выберите режим **Pose Mode** для окон **3D View** и **Ipo Curve Editor**;
- ◆ выделите звено *Bone3.R* и установите первый ключ для имеющейся позиции (для быстрого создания необходимых ключей можно просто подвигать это звено);
- ◆ передвиньте курсор в окне **Timeline** на секунду вперед и поднимите *Bone3.R* на максимальную высоту;
- ◆ для следующей секунды анимации опустите выпрямленную цепочку перпендикулярно звену *Main*.

Продолжите далее эту анимацию с использованием правой стороны скелета. В итоге должно получиться последовательное выпрямление цепей арматуры.

Обратите внимание на окно редактора IPO. Помимо привычных каналов *Loc* и *Scale* появились новые: *QuatW*, *QuatX*, *QuatY*, *QuatZ*. Все они описывают так называемые кривые IPO действия, используемые для элементов арматуры. Причем для расчета используются кватернионы, которые обеспечивают лучшую интерполяцию между позами. Но не будем вдаваться далее в математику. Нужно только запомнить, что добиться интуитивного управления звеном с помощью каналов кватерниона очень сложно и лучше использовать непосредственное изменение местоположения звена в пространстве.

Теперь рассмотрим окно **Action Editor**. Здесь представлена информация об анимации для каждого из звена арматуры в виде персональных полос, которые называются каналами действия. В совокупности они образуют собой законченное действие (рис. 5.53).

В окне действия отображаются маркеры, соответствующие ключевым кадрам анимации. Редактирование заключается в выделении канала и соответствующего маркера, а затем переноса его на нужное место. Для выделения можно использовать правую кнопку мыши. Если нужно отметить группу маркеров, то лучше использовать рамку выделения (). Чтобы перенести отмеченный блок, достаточно нажать клавишу <G>.

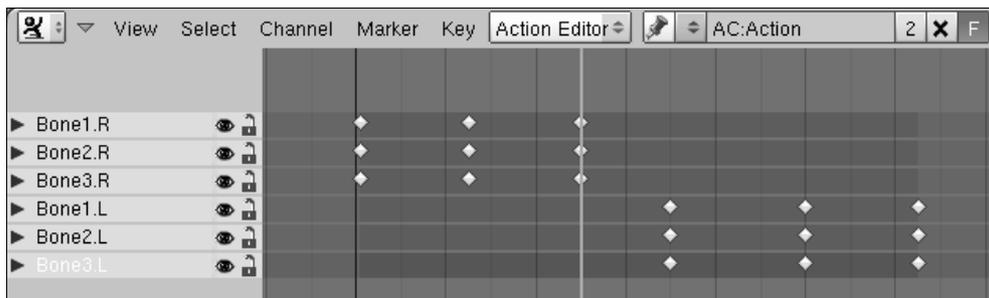


Рис. 5.53. Окно Action Editor

С левой стороны редактора находятся названия каналов, соответствующие именам звеньев в арматуре. У каждого канала имеется треугольная кнопка, раскрывающая его иерархию. Здесь можно более точно отредактировать нужный канал IPO (рис. 5.54).

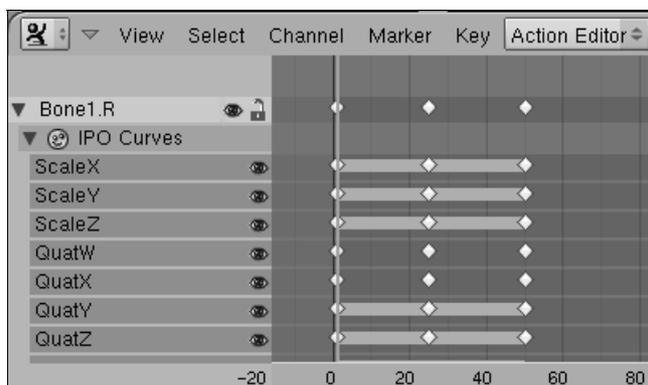


Рис. 5.54. Свернутые данные канала действия

Редактор действий умеет работать с группами каналов. Эта возможность используется для объединения звеньев в логические блоки по их функциональности. Если взять все тот же скелет человека, то можно выделить такие группы, как руки, ноги, голова. Для рабочей арматуры, которая имеет две цепочки звеньев, можно создать группы "левая рука" и "правая рука". В дальнейшем это отразится на эффективности настройки анимации, ведь уже известно, какая группа и за что отвечает.

Итак, выделите каналы Bone1.R, Bone2.R, Bone3.R. Это можно сделать, удерживая нажатой клавишу <Shift> и щелкая правой кнопкой мыши по названиям каналов. Теперь выберите пункт меню **Channel | Grouping | Add to New Group** (Каналы | Группировка | Добавить в новую группу).

Название новой группы можно поменять, если щелкнуть по ней правой кнопкой мыши и удерживать при этом клавишу <Ctrl> (рис. 5.55).



Рис. 5.55. Окно настройки параметров группы

В появившемся окне можно не только переименовать группу, но и защитить ее от редактирования. Для этого нужно просто активировать кнопку **Protected** (Защищено). При этом у названия канала появится характерный рисунок закрытого замочка. Теперь сделайте то же самое для оставшихся звеньев правой цепочки и назовите новую группу *Left_Hand* (рис. 5.56).



Рис. 5.56. Группировка звеньев по смыслу

Action Editor умеет создавать новое действие на основе уже имеющегося. Для этого выберите пункт меню **ADD NEW** (Добавить новое) и переименуйте действие в Action2 (рис. 5.57).

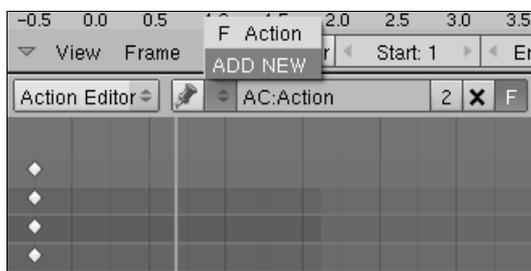
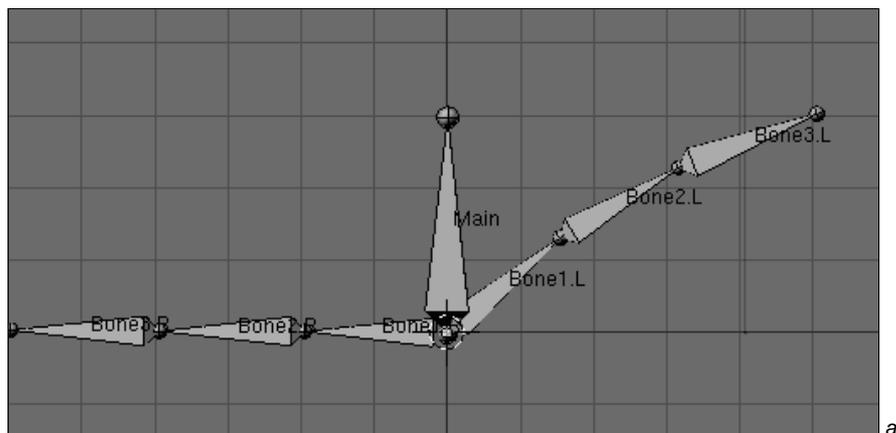
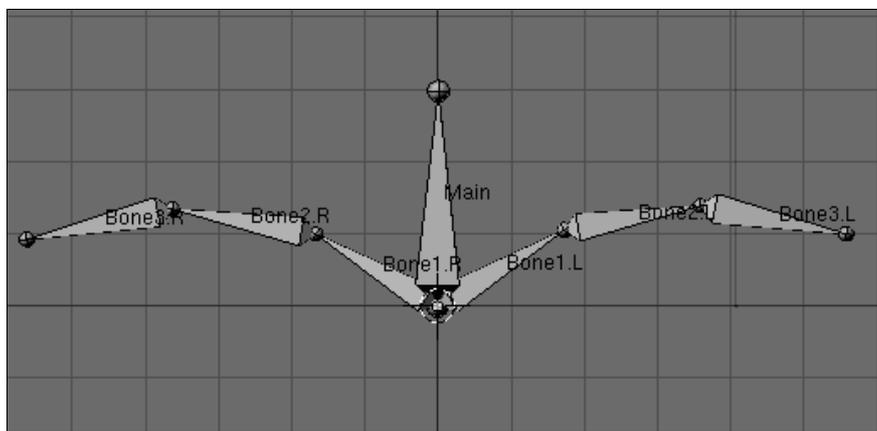


Рис. 5.57. Меню для выбора или создания действия

Рассмотрим простой пример изменения анимации для второго действия. Если в первом варианте арматура расправляла свои "руки" последовательно, то в этот раз пусть это действие произойдет одновременно. Для этого выделите маркеры группы Left_Hand и передвиньте их точно под маркеры группы Right_Hand. При необходимости отстройте каждый маркер вручную (рис. 5.58, б).



a



б

Рис. 5.58. Анимация арматуры: первое действие (а), второе действие (б)

Совет

Помимо перемещения маркеров можно использовать масштабирование канала, что позволяет равномерно изменять время проигрывания анимации (<S>).

5.10. Нелинейная анимация

В сложных проектах в анимации могут принимать участие десятки объектов. При этом используются различные виды анимации от простого перемещения модели до морфинга ее структуры. Понятное дело, что разработка большого количества разнообразных действий сопряжено со значительными сложностями в плане их состыковки между собой. Однако программа Blender имеет уникальный инструмент, представляющий собой последний уровень глобализации анимационных процессов. Именно он позволяет следить и редактировать поведение всех анимационных объектов в сцене. Итак, встречаем — Non Linear Animation Editor (Редактор нелинейной анимации), или просто NLA Editor.

Рассмотрим работу с этим редактором на примере предыдущего проекта (файл 5_action2.blend). Как вы помните, арматура этого файла имеет два созданных в Action Editor действия. Откройте в любом окне программы NLA Editor, выбрав его из глобального меню (рис. 5.59).

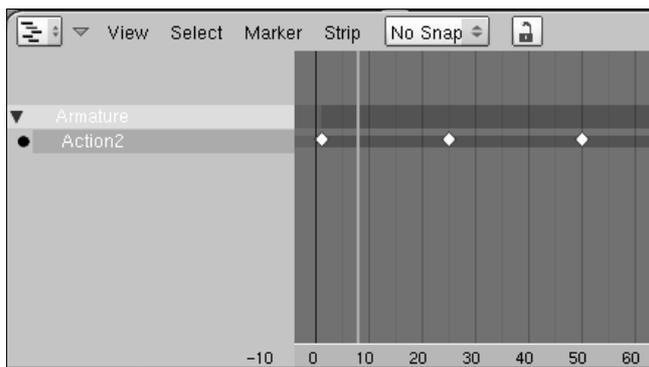


Рис. 5.59. Окно редактора нелинейной анимации

Так как в проекте имеется лишь один анимационный объект, то и окно **NLA Editor** показывает только его. Обратите внимание на то, что, хотя анимация арматуры состоит из двух действий, нелинейный редактор содержит всего одно. Это связано с тем, что по умолчанию для арматуры показывается то действие, которое в данный момент является активным в окне **Action Editor**. А что же делать, если необходимо использовать сразу два действия? Все просто. Стоит только добавить нужное действие в NLA Editor, и оно будет доступно для редактирования. Чтобы это сделать, выделите правой кнопкой мыши название арматуры в окне нелинейного редактора и нажмите сочетание клавиш <Shift>+<A>. Появившееся меню отображает возможные анимации, имеющиеся для данного объекта. Выберите сначала **Action**, а потом и **Action2** (рис. 5.60).

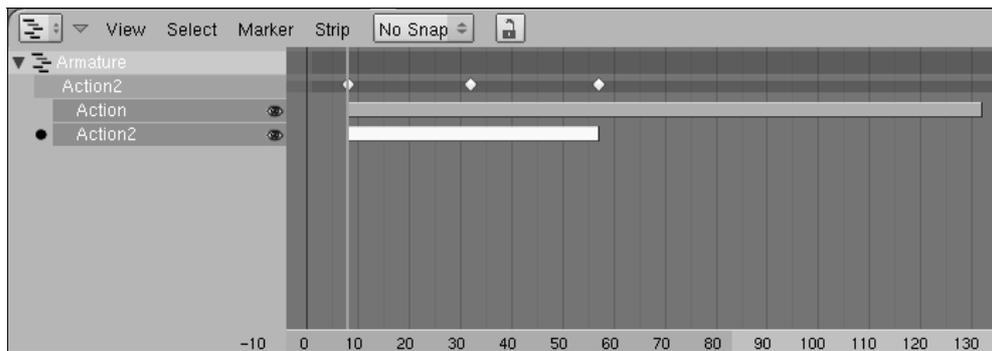


Рис. 5.60. Каждая анимация имеет свою полосу редактирования

Работа с этими полосами ничем не отличается от таких же в Action Editor. Попробуйте, к примеру, выделить канал **Action** и перетащить его чуть дальше середины действия **Action2** (рис. 5.61).

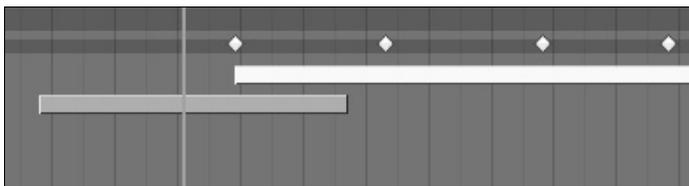


Рис. 5.61. Расположение полос влияет на очередность проигрывания анимации

Совет

Для перемещения полосы в редакторах действия и нелинейной анимации можно использовать не только горячую клавишу <G>, но и мышь. Нажмите и удерживайте несколько секунд правую кнопку мыши на нужной полосе. Теперь при перемещении манипулятора будет двигаться и полоса. Щелчок левой кнопки мыши закрепит результат, а правой (или клавиша <Esc>) — отменит его.

Обработка имеющихся полос в редакторе происходит снизу вверх. Таким образом самая нижняя полоса будет перекрывать более высокие. Чтобы изменить порядок полос, нужно выделить желаемую и выбрать пункт меню **Strip | Move Up** (Полоса | Передвинуть вверх) или **Strip | Move Down** (Полоса | Передвинуть вниз).

Помимо управления порядком анимации и скорости ее проигрывания (маркеры на полосах обозначают ключевые кадры) можно настраивать поведение каналов. Основные параметры активной полосы доступны при нажатии клавиши <N> (рис. 5.62).

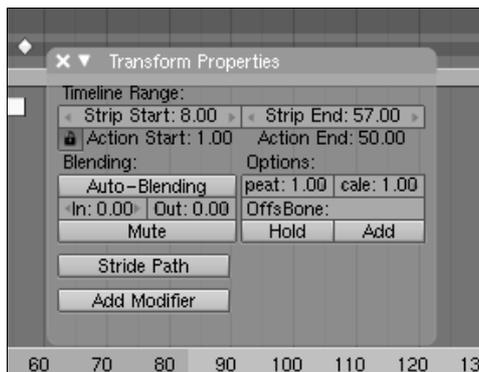


Рис. 5.62. Плавающая панель настроек полосы

Здесь вы можете отключить проигрывание полосы кнопкой **Mute**, заморозить кадр под курсором кнопкой **Hold**, но куда интереснее возможность редактора создавать плавные переходы от действия к действию. Выделите, к примеру, полосу **Action2** и нажмите кнопку **Auto-Blending** (Автоматическое смешивание). Теперь при проигрывании анимации программа с указанного места произведет плавный переход на другое действие (рис. 5.63).



Рис. 5.63. Переход имеет свое графическое отображение в окне

ГЛАВА 6



Физика в Blender

Одной из самых сложных и трудоемких задач при разработке трехмерного мира всегда была имитация физических законов. К примеру, создание развевающегося флага, несмотря на свою кажущуюся легкость, может потребовать немало усилий, по крайней мере теми средствами, о которых вы уже читали. Однако Blender обладает мощными и легкими в использовании инструментами, с помощью которых за считанные минуты можно воплотить любой дерзкий замысел. Ветер, капли дождя, разбивающиеся о лужу, одинокий прохожий, бегущий за своей шляпой...

Думаете нереально создать такую сцену в компьютерной программе? Эта глава книги посвящена решению именно подобных задач.

6.1. Создание и использование частиц

Оглянитесь вокруг. Как вы думаете, что можно смоделировать с помощью частиц? Правильно. Снег, облака, дым, огонь. А теперь, к примеру, прибавьте сюда такие объекты, как мыльные пузыри, листья деревьев, волосы, и вы получите начальное представление об открывающихся возможностях при использовании частиц.

В понимании Blender, *частицы* — это особые объекты, обладающие набором специальных параметров, позволяющих организовать их поведение. Такими параметрами могут быть: время жизни частицы, направление движения, элемент случайности и т. д.

Частицы не могут возникать из пустого пространства и должны базироваться на каком-либо объекте. Такой объект называется *эмиттером*. В качестве эмиттера могут быть все объекты Blender: мешы, сплайны и даже лампы. На данный момент программа предлагает очень широкие возможности для соз-

дания частиц. Все элементы управления расположены на панели настроек объекта (<F7>).

По умолчанию система частиц является неактивной. Чтобы перейти в панель управления эффектом, нужно активировать кнопку **Particle buttons** (Кнопки частиц), которая расположена на заголовке окна **Buttons Window** (рис. 6.1).



Рис. 6.1. Выбор панелей частиц

Для подключения частиц к объекту необходимо нажать кнопку **Add New** (Добавить новый) или выбрать уже имеющиеся из списка (рис. 6.2).

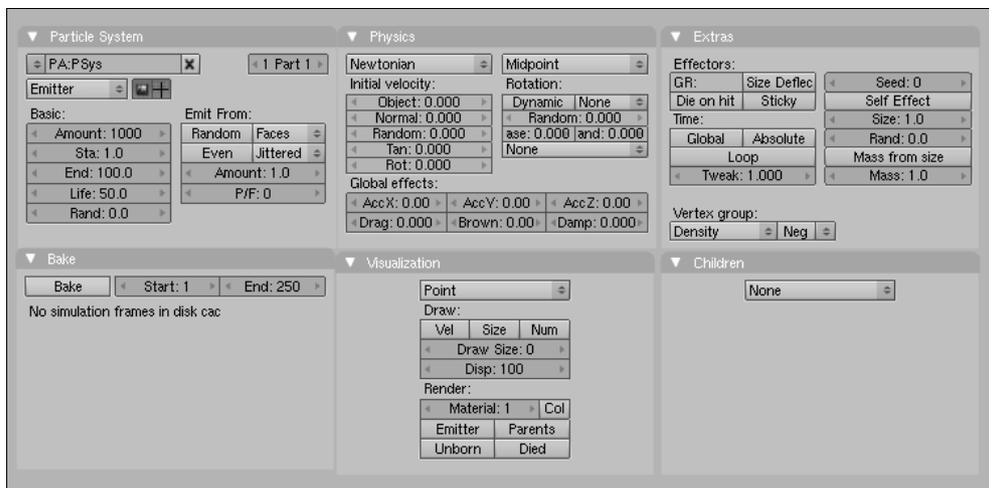


Рис. 6.2. Набор функций по умолчанию

С этого момента, с точки зрения Blender, такой объект превращается в эмиттер и при обработке не будет отображаться.

Основные параметры генерации частиц расположены на панели **Particle System** (Система частиц):

- ◆ **Amount** (Количество). Максимальное количество частиц, которые будут участвовать в действии;
- ◆ **Sta** (Начало). Номер кадра для начала генерации;
- ◆ **End** (Конец). Номер кадра для окончания генерации;
- ◆ **Life** (Жизнь). Продолжительность существования отдельной частицы;

◆ **Rand** (Случайность). Параметр, связанный с **Life**, который приносит некоторое разнообразие по времени жизни частицы.

Совет

Для предварительного просмотра результата без обработки достаточно выбрать желаемый кадр сцены или просто нажать сочетание клавиш <Alt>+<A>.

Давайте поработаем с частицами на основе примитива `Plane`. Создайте объект и подключите к нему систему частиц, т. к. было описано ранее. Для контроля получившегося установите активным 30-й кадр (рис. 6.3).

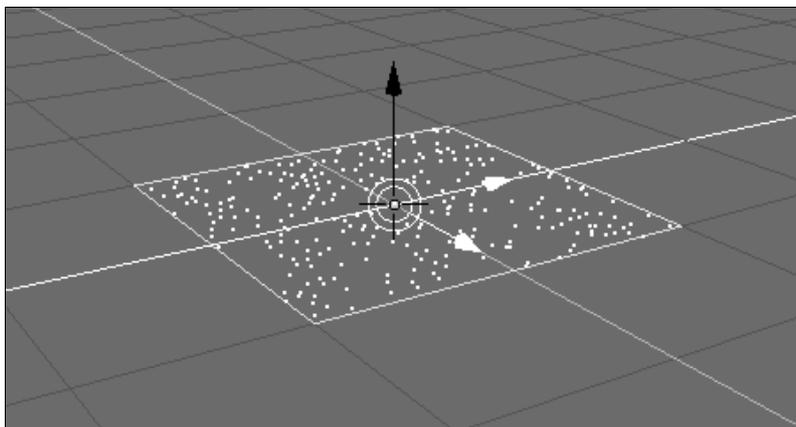


Рис. 6.3. По умолчанию частицы располагаются на плоскости объекта

Как видите, вся плоскость примитива служит для генерации частиц. Однако имеется возможность использовать в качестве эмиттера вершины объекта, стоит только выбрать пункт **Verts** в меню **Emit From**.

Для управления перемещением частиц служит панель **Physics** (см. рис. 6.2). Здесь можно установить координатную ось излучения, скорость перемещения или ротации. К примеру, изменение параметра **AccZ** заставит частицы перемещаться по координате **Z** (рис. 6.4).

Внимание!

После редактирования любых параметров частиц необходимо установить начальный кадр сцены и проиграть анимацию. Это действие принудительно заставит Blender пересчитать результат работы системы частиц. В противном случае может наблюдаться неправильная работа генератора.

Обратите внимание, что частицы перемещаются строго по оси **Z**. Можно придать некоторый фактор случайности направлениям движения, и даже вращению частиц. За это отвечают функции **Random** областей **Initial Velocity**

(Начальное движение) и **Rotation** (Вращение). Установите для первого пункта значение 1.0 (рис. 6.5).

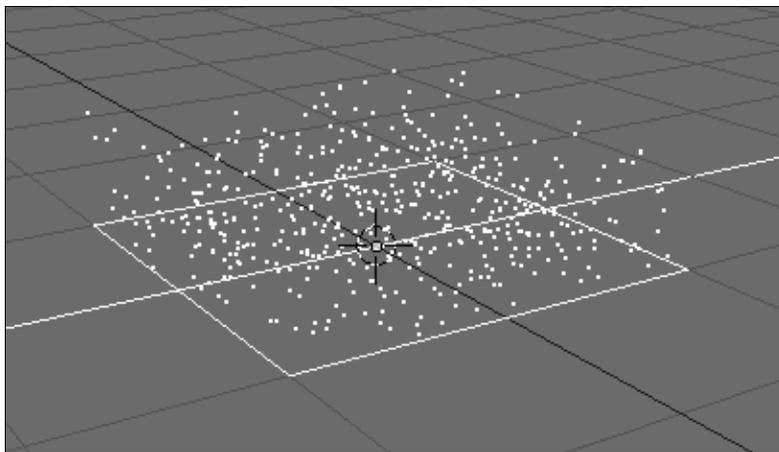


Рис. 6.4. Вертикальное движение частиц

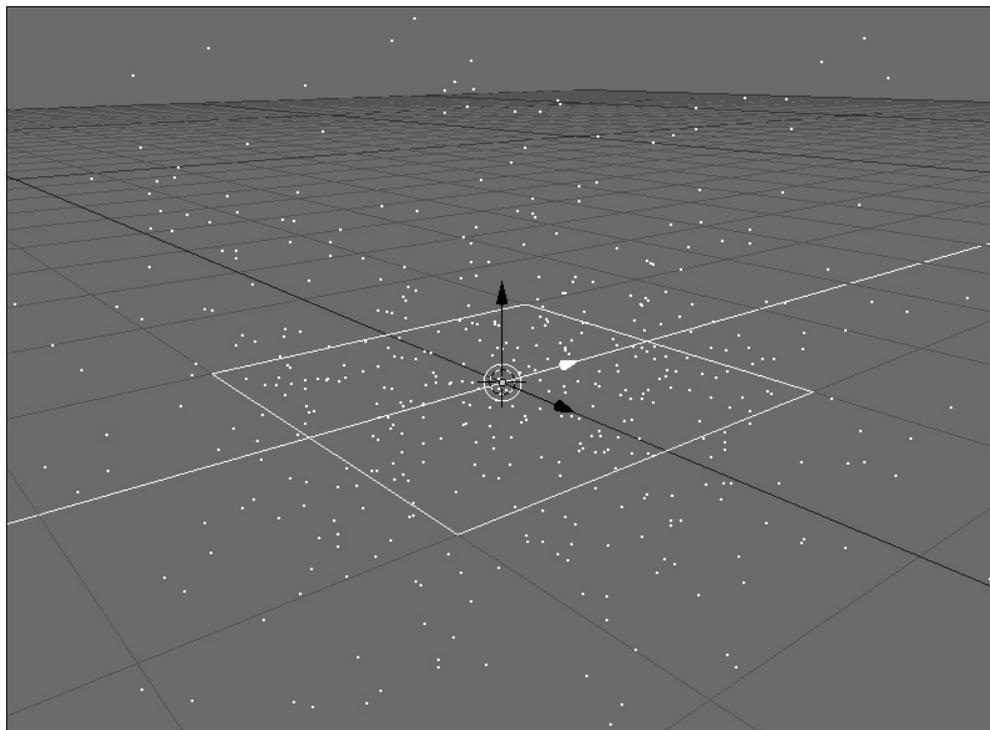


Рис. 6.5. Несмотря на "случайность", движение частиц в основном вертикальное

Следующая панель, **Visualization** (см. рис. 6.2), предназначена для настройки отображения частиц как в окне редактора, так и при конечной обработке. Здесь можно выбрать тип объекта, отличный от точек (*points*), включить вспомогательные элементы для 3D-окна, настроить некоторые функции рендера.

Набор элементов, предлагаемых программой для генерации частиц, весьма широк и варьируется от простейшей точки до целых групп. Все это богатство заключено в меню визуализации. В зависимости от выбранного типа объекта в этой же панели появляются дополнительные функции его настройки. Попробуем в качестве частиц использовать обычный примитив *Cube*.

Создайте требуемый объект и уменьшите его размер в соответствии с масштабом сцены. Теперь в панели **Visualization** эмиттера выберите пункт **Object**. В появившемся окне **Ob:** введите название куба. После этого программа создаст ссылку на куб. Обратите внимание на то, что редактирование базового объекта влияет на его ссылку в системе частиц (рис. 6.6).

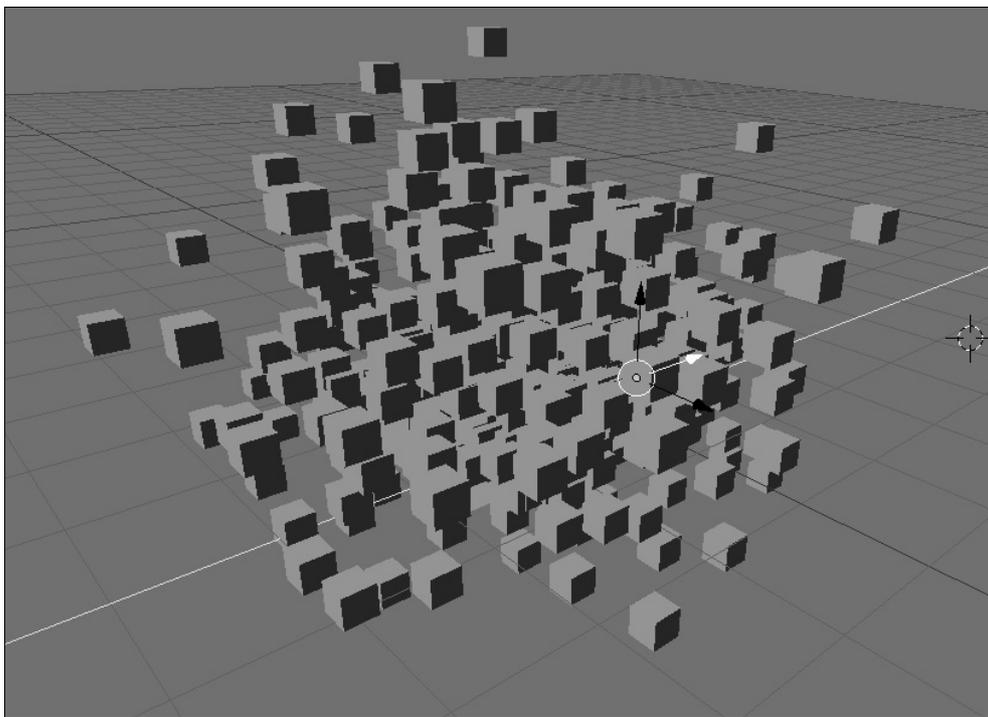


Рис. 6.6. В качестве частиц могут быть примитивы

Так как жизнь частицы имеет ограничение, то по истечении отведенного ей срока она уничтожается. В некоторых случаях бывает нужно оставить на

месте частицы, уже отработавшие свой лимит. Для этого служит кнопка **Dead** (Умерший), которая замораживает элементы на последнем шаге их жизни. Есть и обратная функция — **Unborn** (Будущие), демонстрирующая все частицы в начале их пути.

Элементам системы частиц могут быть присвоены материалы и текстуры. По сути дела, материал назначается эмиттеру. Рассмотрим следующий пример.

Создайте новый материал белого цвета и включите кнопку **Halo**. Теперь в панели **Shaders** выберите тип **Rings** (см. разд. 4.6). Перейдите в панель настроек частиц и введите в поле **Amount** значение 100. В качестве объекта визуализации выберите **Points** (рис. 6.7).

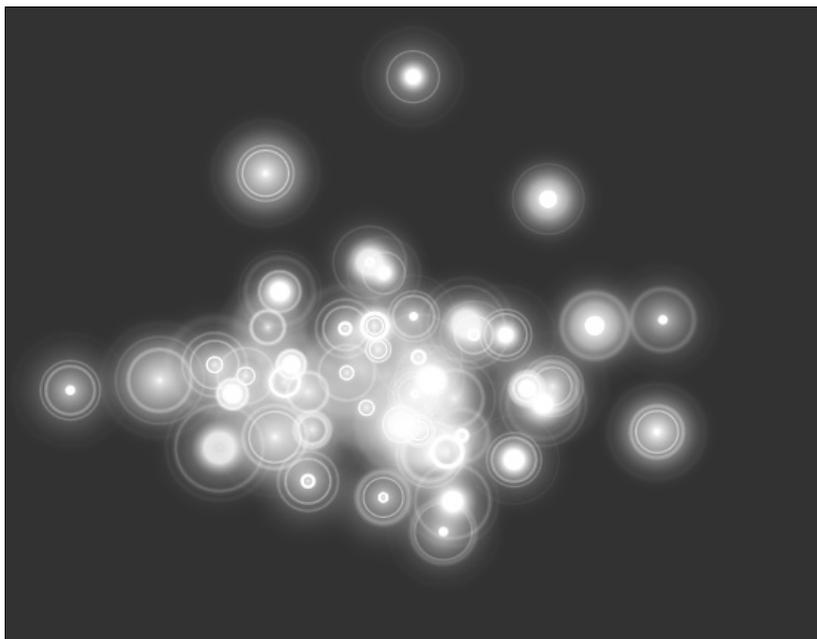


Рис. 6.7. Система частиц с эффектом Halo

Внимание!

Для ускорения расчетов частиц Blender сохраняет промежуточные значения на диск. По умолчанию кэш располагается в той же папке, где и проект. В зависимости от сложности эффекта размер этого кэша может составлять сотни мегабайт.

Одной из замечательных возможностей программы Blender является работа со статичными частицами. Использовать их можно, к примеру, для создания волос, меха. Рассмотрим следующий пример.

Откройте новый проект и создайте в нем сферу `UVSphere`, которая будет служить эмиттером частиц. Нажмите клавишу `<F7>`, затем в панели **Particle System** кнопку **Add New**. По умолчанию программа создаст динамические частицы. Для включения статичных достаточно выбрать пункт **Hair** (Волосы) в меню **Type** (рис. 6.8).

Чтобы увидеть созданные частицы, введите в поле **Normal** панели **Physics** значение `0.1`. При желании можно выбрать в качестве эмиттера как полигоны, так и вершины. Для увеличения максимального количества частиц введите значение `5000` в поле **Amount** (рис. 6.9).



Рис. 6.8. Выбор режима частиц

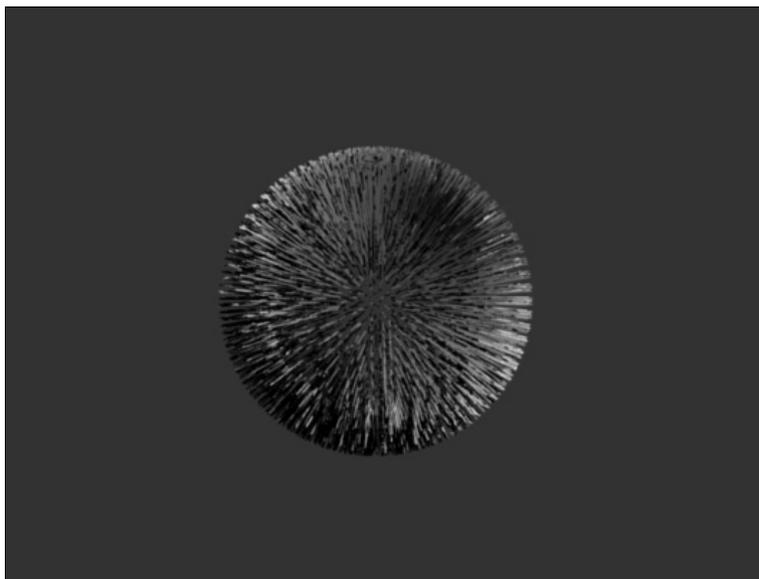


Рис. 6.9. Результат обработки статичных частиц

Совет

Увеличение количества частиц негативно сказывается на скорости работы программы и удобстве редактирования в окне **3D View**. Для решения этой проблемы можно уменьшить значение поля **Disp** (Экран) панели **Visualization**. Эта функция отвечает за процентное соотношение частиц окна редактирования к конечному результату. На саму обработку параметр **Disp** не влияет.

Ранее уже говорилось о том, как изменить направление испускания частиц. Те же самые функции доступны и для статичных частиц. Но есть более удобный способ манипулирования потоком — с помощью направляющей кривой. Добавьте в проект объект **Bezier Curve** и нажмите кнопку **Physics buttons**, которая открывает панели настроек физики (рис. 6.10).

На панели **Fields** (Поля) имеется единственное, в данный момент неактивное, меню. Выберите в нем пункт **Curve Guide** (Направляющая кривая). Обратите внимание на то, что частицы послушно приняли форму самой кривой. Теперь путем редактирования **Path** можно добиться любой формы образования частиц (рис. 6.11).



Рис. 6.10. Эта кнопка открывает настройки физики Blender

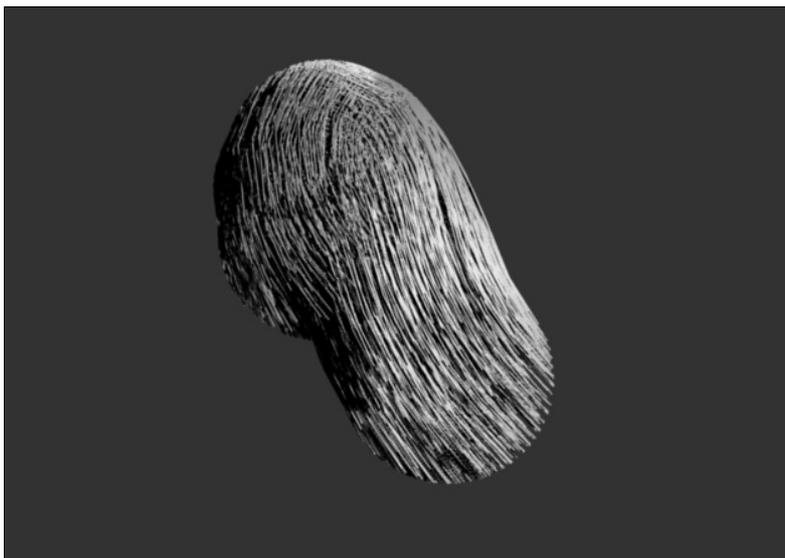


Рис. 6.11. Результат редактирования кривой

Совет

Для одного эмиттера можно использовать несколько направляющих кривых. Это позволит точнее расположить частицы.

6.2. Работа с тканями

Все параметры и способы настройки физики расположены на панелях, которые доступны при нажатии клавиши <F7> и выбора соответствующей кнопки **Physics buttons** (см. рис. 6.10). Вашему вниманию предстанут несколько панелей, вобравшие в себя всю мощь физического мира Blender. В данном случае нас интересует панель **Cloth** (Ткань).

Итак, создайте новый проект, удалите куб и добавьте обычную сферу. В этом примере сфера будет тем объектом, вокруг которого должна оборачиваться ткань. В качестве ткани используем `Plane` (рис. 6.12). Для качественной анимации необходимо увеличить количество полигонов ткани. Используйте инструмент `Subdivide` для разбиения плоскости несколько раз.

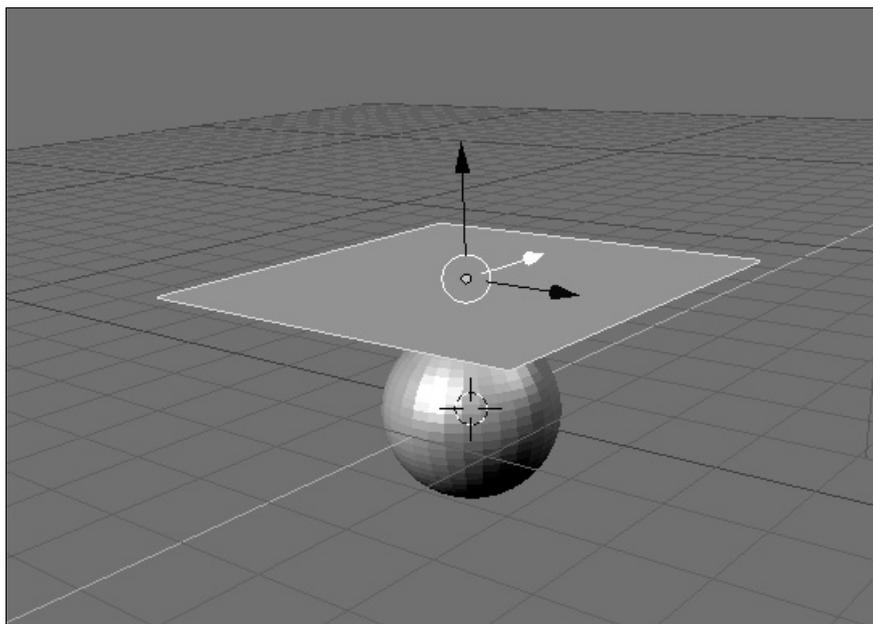


Рис. 6.12. Установите плоскость над сферой

Чтобы имитация нормально работала, необходимо установить физические параметры для обоих взаимодействующих объектов. В случае со сферой, ей достаточно придать свойство **Collision** (Столкновение). Это свойство обеспе-

чивает реагирование объекта на другие при их столкновении. Для этого выделите сферу, откройте панель **Collision** и нажмите кнопку с одноименным названием (рис. 6.13).

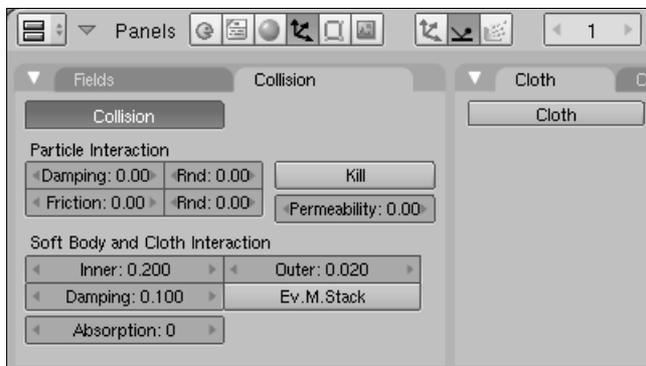


Рис. 6.13. Настройки панели **Collision**

Функции панели **Collision**:

- ◆ **Particle Interaction** (Взаимодействие частиц). Переменные для работы с частицами:
 - **Damping** (Упругость);
 - **Friction** (Трение);
- ◆ **Soft Body and Cloth Interaction** (Взаимодействие системы мягких тел и тканей):
 - **Inner** (Внутренний);
 - **Outer** (Внешний);
 - **Damping** (Упругость);
 - **Absorption** (Поглощение).

Трогать здесь ничего не нужно, т. к. параметры по умолчанию вполне удовлетворительные для этого примера.

Выделите плоскость и включите кнопку **Cloth** (Ткань) на панели **Cloth** (рис. 6.14).

Blender предлагает несколько уже настроенных заготовок тканей, которые можно использовать в своей анимации. Расположены они в меню **Material Preset** (Заготовки материала):

- ◆ **Custom** (Выборочно). Этот пункт меню выбирается автоматически, если пользователь производил изменения в полях панели;

- ◆ **Leather** (Кожа);
- ◆ **Denim** (Джинсовая ткань);
- ◆ **Rubber** (Каучук);
- ◆ **Cotton** (Хлопок);
- ◆ **Silk** (Шелк).



Рис. 6.14. Настройки физики ткани

Выберите из приведенного списка, к примеру, шелк. Помимо этого, также необходимо включить проверку столкновений у плоскости. Обратите внимание на то, что на панелях физических свойств имеется несколько вкладок **Collision**. В данном случае нужна именно та, которая находится на панели **Cloth** (рис. 6.15).

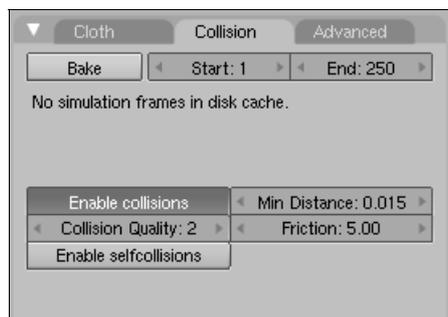


Рис. 6.15. Настройки Collision применительно к Cloth

Для просмотра результата достаточно просто проиграть анимацию (рис. 6.16). Проведите еще один эксперимент для полного понимания возможностей, которые открывает физика тканей в Blender. Поменяйте местами оба примитива. Пусть теперь происходит падение сферы на плоскость (рис. 6.17). Соответствующим образом измените и их настройки физики.

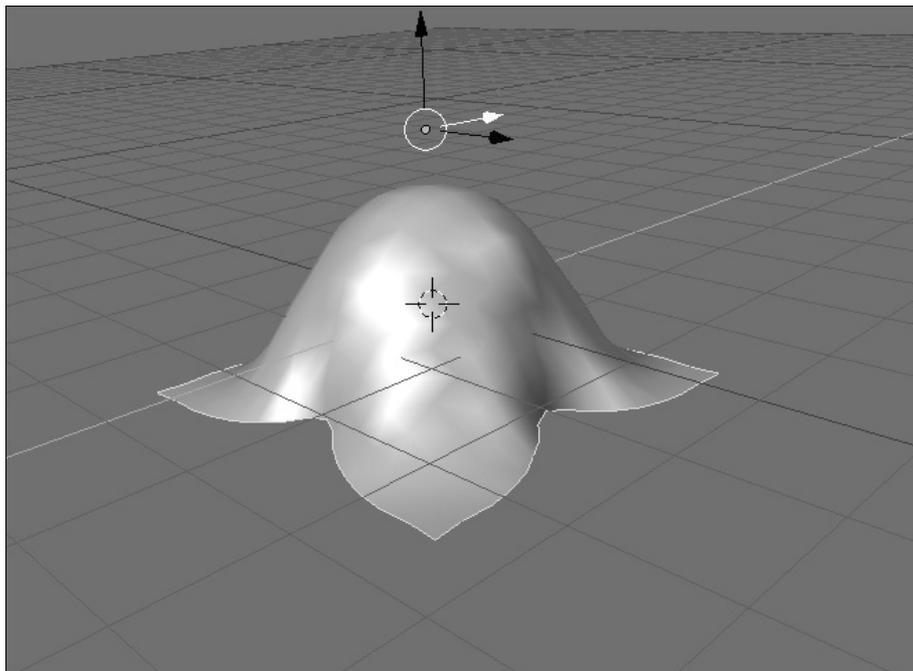


Рис. 6.16. Результат столкновения сферы и ткани

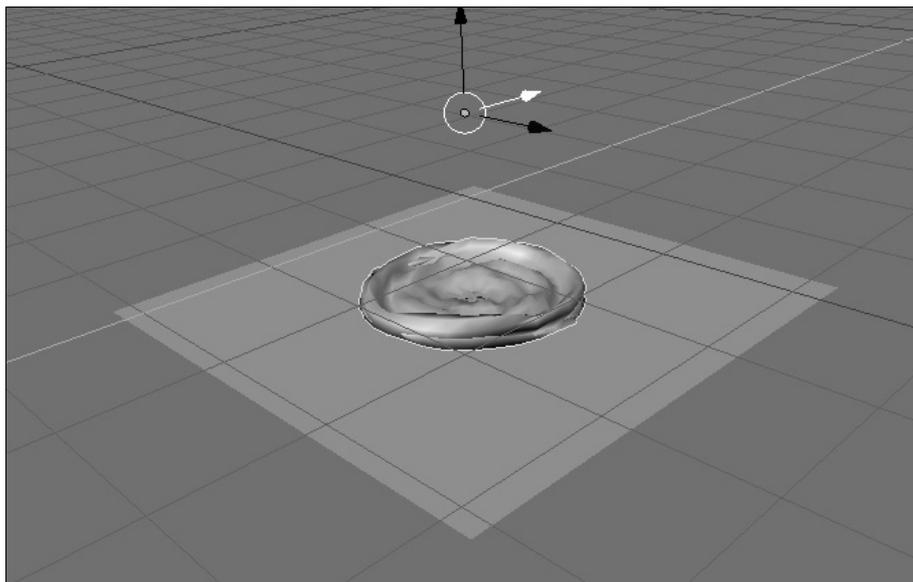


Рис. 6.17. Растекшийся круг — все, что осталось от сферы

6.3. Физика мягких тел

Работая с тканями, вы убедились, с какой легкостью можно создавать имитации сложных физических явлений буквально несколькими щелчками мыши. Однако ткани — это всего лишь часть сложной физической системы, которая имеет название **Soft Body** (Мягкие тела). Несмотря на свое название, система мягких тел позволяет создавать не только ткани, но и любые эластичные объекты. Рассмотрим ее работу на примере обычного спортивного батута.

Создайте два объекта: куб и плоскость. Так как в качестве батута будет выступать плоскость, то расположите ее чуть ниже куба. Включите проверку столкновений для батута (см. рис. 6.13), и на этом оставим пока его в покое.

Для нашей задачи необходима высокополигональная модель куба. Разбейте структуру куба несколько раз, используя инструмент **Subdivide**. Перейдите на панель физики и нажмите кнопку **Soft Body** (рис. 6.18).

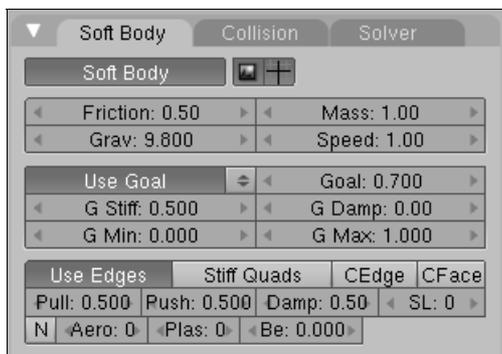


Рис. 6.18. Панель настроек системы мягких тел

Если сейчас запустить анимацию, то можно увидеть, что куб начинает качаться, как на резинке. Система мягких тел позволяет использовать группы вершин в качестве статичных привязок, на которые физика не распространяется. Этим можно пользоваться для закрепления объекта на другом объекте, к примеру ткани на флагштоке. В нашем случае эта возможность не нужна. Отожмите кнопку **Use Goal** (Использовать цель) на панели **Soft Body**. После этого куб свободно упадет на плоскость и буквально растечется (рис. 6.19).

Итак, что же произошло? Дело в том, что программа позволяет настраивать для структуры объекта такие свойства, как упругость, сопротивление, аэродинамику, и по умолчанию стоят довольно мягкие параметры.

Нажмите кнопки **Use Edges** (Использовать ребра) и **Stiff Quads** (Жесткие квадраты). Затем измените на максимальное значение параметры **Pull** (На-

пряжение), **Push** (Упругость), **Damp** (Демпинг). Запустите анимацию. В этот раз не происходит разрушения структуры куба и видны лишь небольшие искажения его формы.

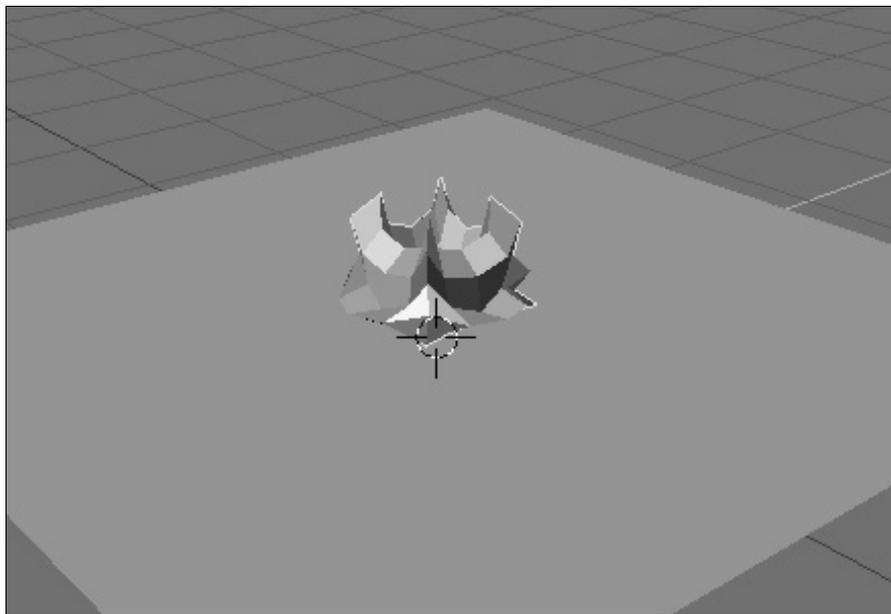


Рис. 6.19. Результат падения модели

Рассмотрим эти параметры подробнее:

- ◆ **Use Edges** (Использовать ребра). Кнопка **Use Edges** включает основные настройки физики для внутренней структуры объекта. Эта группа отвечает за время, в течение которого ребра смогут сохранять свой первоначальный размер;
- ◆ **Stiff Quads** (Жесткие квадраты). При нажатии кнопки **Stiff Quads** программа будет использовать диагональные ребра, что обеспечит дополнительную упругость основным граням;
- ◆ **Pull** (Напряжение). Устанавливает жесткость для ребер. При высоком значении ребра приобретают высокую инертность;
- ◆ **Push** (Упругость). Этим параметром можно настроить внутреннее сопротивление структуры объекта к сжатию. К примеру, высокое значение годится для надутых моделей, типа воздушного шарика;
- ◆ **Damp** (Демпинг). Сила трения для ребер. Максимальное значение снижает трение.

Придадим нашему кубу свойства желе — материала, обладающего небольшим коэффициентом деформации при столкновении и достаточной упругостью. Измените следующие параметры:

- ◆ Pull — 0.700;
- ◆ Push — 0.950;
- ◆ Damp — 0.00.

На этом настройку куба можно считать законченной (рис. 6.20).

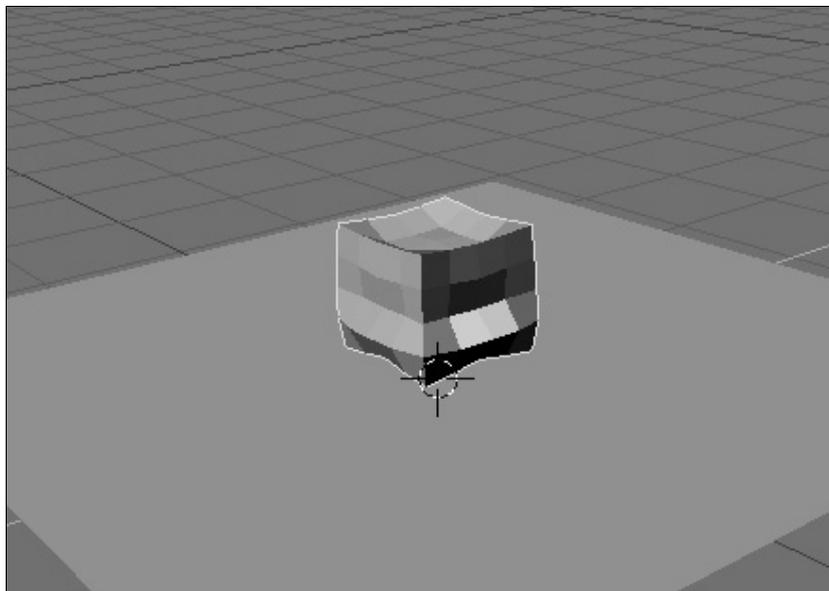


Рис. 6.20. С этими настройками куб сохраняет свою форму

Для работы с плоскостью необходимо сначала ее подготовить. Разбейте структуру примитива несколько раз и выделите крайние угловые вершины (рис. 6.21).

Создайте новую группу вершин в настройках панели **Link and Material** (см. разд. 5.8). Эти вершины будут использоваться в качестве булавок, "припиливающих" объект к сцене. Включите физику **Soft Body** и нажмите кнопку **Use Goal** (рис. 6.22).

Параметры **Use Goal** отвечают за настройку взаимодействия группы вершин с остальной частью объекта. Так, параметр **G Stiff** отвечает за упругость, **G Damp** — трение, **G Max** и **G Min** используются для точной настройки воздействия при использовании режима `Weight Paint`.

В нашем случае необходимы следующие значения:

- ◆ G Stiff — 0.120;
- ◆ G Damp — 0.80;
- ◆ G Min — 0.016;
- ◆ G Max — 1.000.

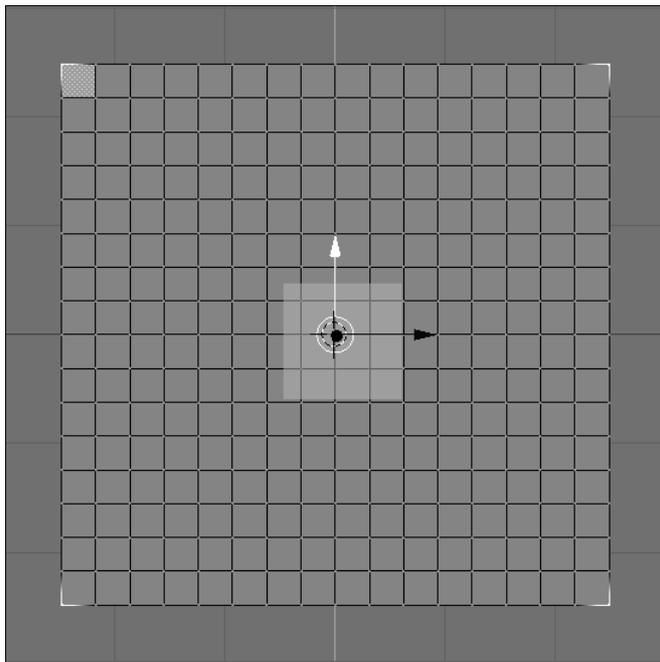


Рис. 6.21. Выделение вершин для введения ограничения воздействия физики

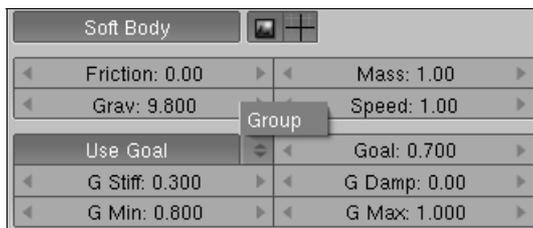


Рис. 6.22. Выбор нужной группы вершин объекта

Так как материал катушки должен обладать достаточной жесткостью и упругостью, то необходимо включить кнопки **Use Edges** и **Stiff Quads**.

Измените следующие параметры группы **Use Edges**:

- ◆ Pull — 0.950;
- ◆ Push — 0.999;
- ◆ Damp — 50.0.

Вот теперь при запуске анимации объекты ведут себя подобающим образом. Однако еще нет самого взаимодействия между ними. Если куб отскакивает от поверхности плоскости, то сам батут не принимает никакого участия в этом действии. В идеале он должен хотя бы прогнуться.

Поможет нам в этом следующая группа параметров, которые являются глобальными для физики объекта:

- ◆ Friction (Трение). Установка силы трения. Чем выше значение, тем легче гравитация (Grav);
- ◆ Grav (Гравитация). Сила тяготения, направленная вниз по отрицательной оси Z. По умолчанию имеет значение, равное тяготению Земли, т. е. 9,8;

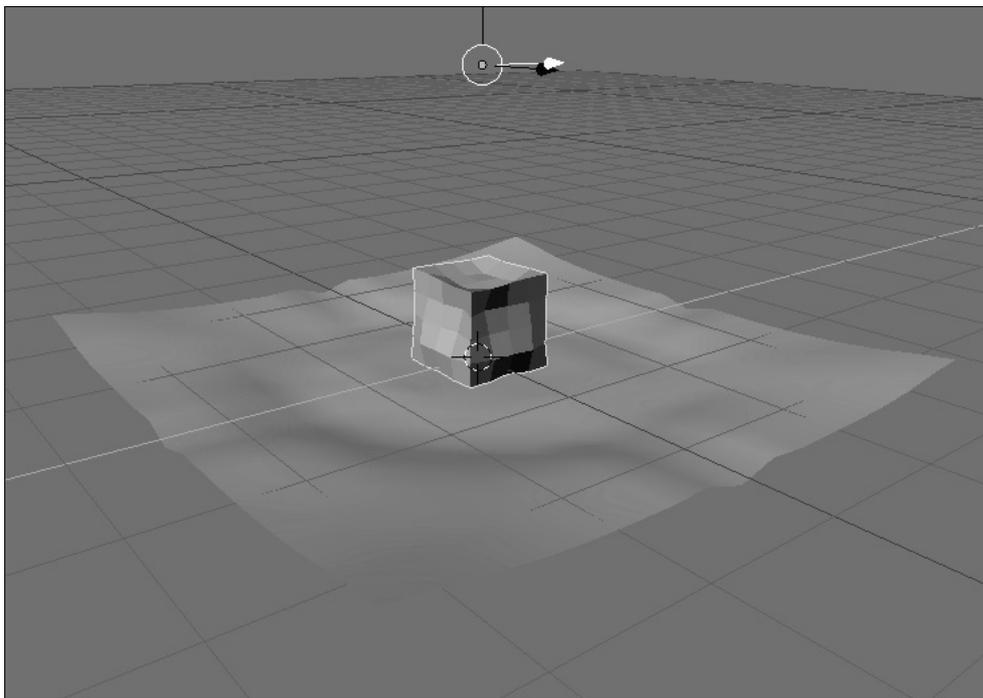


Рис. 6.23. При столкновении на плоскости видны искажения

- ◆ **Mass (Масса).** Масса объекта. Увеличение массы не влияет на гравитацию, но влияет на скорость;
- ◆ **Speed (Скорость).** Скорость объекта.

Установите следующие параметры для этой группы:

- ◆ **Friction** — 5.10;
- ◆ **Grav** — 9.8;
- ◆ **Mass** — 1.00;
- ◆ **Speed** — 1.00.

Вот теперь настройку физики можно считать законченной (рис. 6.23).

6.4. Силовые установки

Использование физики серьезно раздвигает границы реальности 3D-миров. Но, как и в живой природе, виртуальный мир не может обходиться без динамических сил, влияющих на окружающее. *Силовые установки* — это инструменты, моделирующие, к примеру, такое природное явление, как ветер. Всего программа предлагает восемь возможных вариантов силовых установок, некоторые из них мы сейчас рассмотрим.

Как и положено, инструменты управления динамикой находятся на панелях **Physics buttons** на вкладке **Fields** (Поля). Обычно в качестве носителя поля выступает вспомогательный объект `Empty`. В свою очередь поля могут влиять на объекты типа меш или системы частиц. Рассмотрим их на примере развивающегося флага.

Откройте новый проект и добавьте в сцену два объекта: плоскость и `Empty`. Разбейте структуру плоскости несколько раз инструментом `Subdivide`. Создайте группу из двух крайних вершин для закрепления полотнища на древке.

В качестве физической модели для плоскости будем использовать `Ткань`. Откройте соответствующую панель и включите кнопку **Cloth**. Для прищипливания полотнища к сцене в панели **Cloth** предназначена кнопка **Pinning of cloth** (Закрепление ткани). Нажмите ее и выберите сделанную группу из списка (рис. 6.24).

Расположите вспомогательный объект так, чтобы он находился позади закрепленной части флага, и откройте панель **Fields**. Сейчас там находится всего лишь одно выпадающее меню, которое позволяет выбрать тип силовой установки. Выберите в нем пункт **Wind** (рис. 6.25).

Настройки любых из возможных полей разделены на две части. С левой стороны показаны уникальные свойства для каждого типа, а с правой — вспомогательные опции визуализации в 3D-окне.

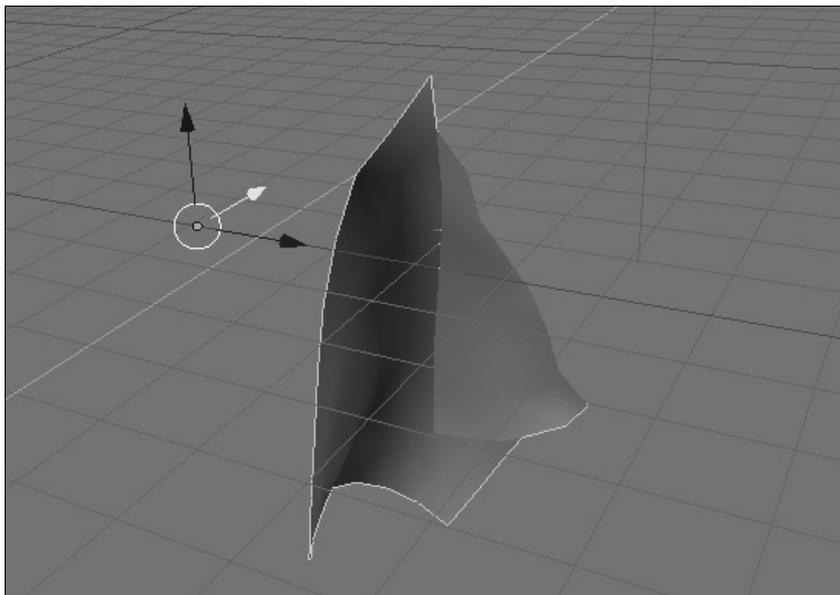


Рис. 6.24. Без ветра ткань просто провисает



Рис. 6.25. Настройки поля Wind

Обратите внимание, что при выборе поля **Wind** (Ветер) вокруг вспомогательного объекта появился круг, обозначающий не только направление ветра, но и его силу. Убедиться в этом можно, если изменить параметр **Strength** (Сила). По умолчанию воздействие силовой установки происходит по оси Z вспомогательного объекта и распространяется на всю сцену. В большинстве случаев в этом нет необходимости. Ограничить воздействие можно опциями **MinDist** (Минимальная дистанция) и **MaxDist** (Максимальная дистанция). Для активации этих опций служат кнопки **Use**, расположенные около них. Визуально ограниченная область обозначается окружностями. Однако при

необходимости тип границ можно изменить, выбрав подходящий пункт из меню в правой части панели (рис. 6.26).

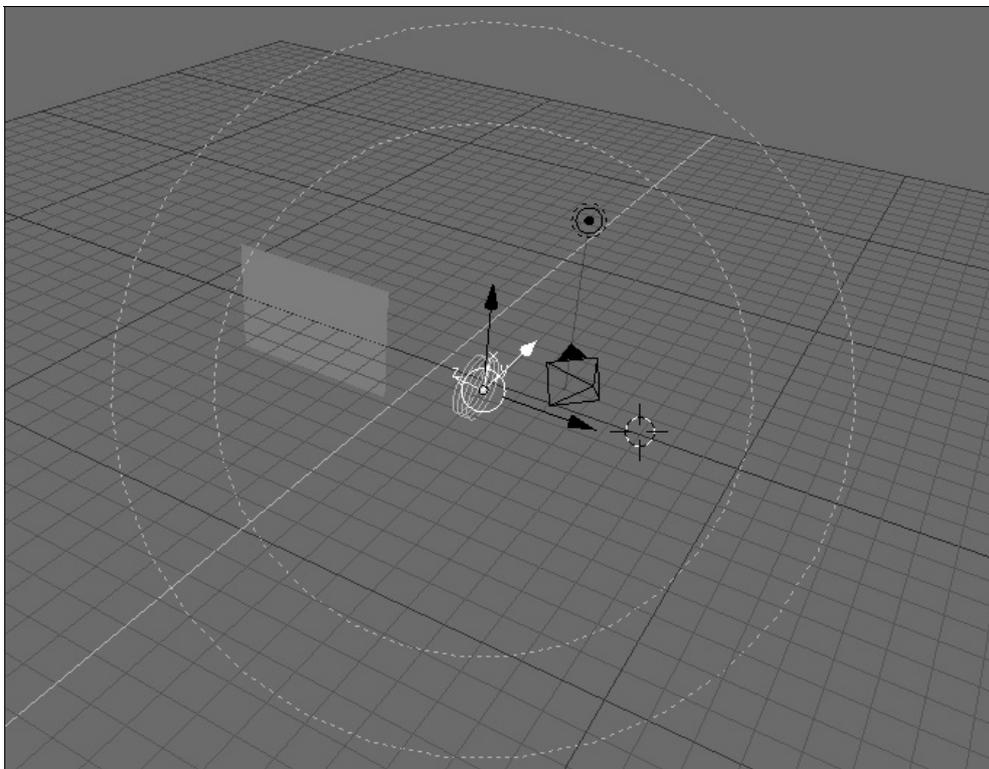


Рис. 6.26. Вспомогательные элементы поля **Wind**

Чтобы имитировать порывы ветра, нет необходимости в анимации `Empty`. Достаточно воспользоваться опцией **Noise** (Шум). Попробуйте установить максимальное значение 10 (рис. 6.27).

Как уже говорилось, установка `Wind` распространяет свое воздействие по одной определенной оси. В случае если необходимо воздействие по всем направлениями, можно использовать силовую установку `Spherical` (Сферический). Выберите в меню **Fields** пункт **Spherical**. Для демонстрации возможностей создайте копию флага (`<Shift>+<D>`) и переместите ее на другую сторону от оригинала (рис. 6.28).

Совет

Для создания физических моделей вихря, торнадо, смерчей используйте силовую установку `Vortex`. Параметры ее ничем не отличаются от уже рассмотренных систем.

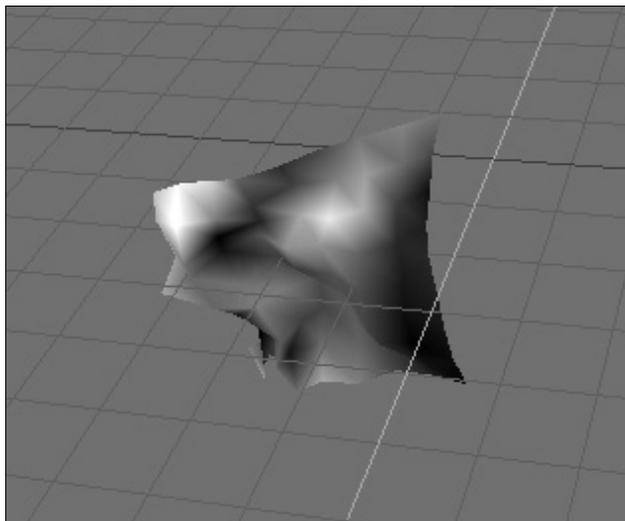


Рис. 6.27. Анимация плоскости при помощи Wind

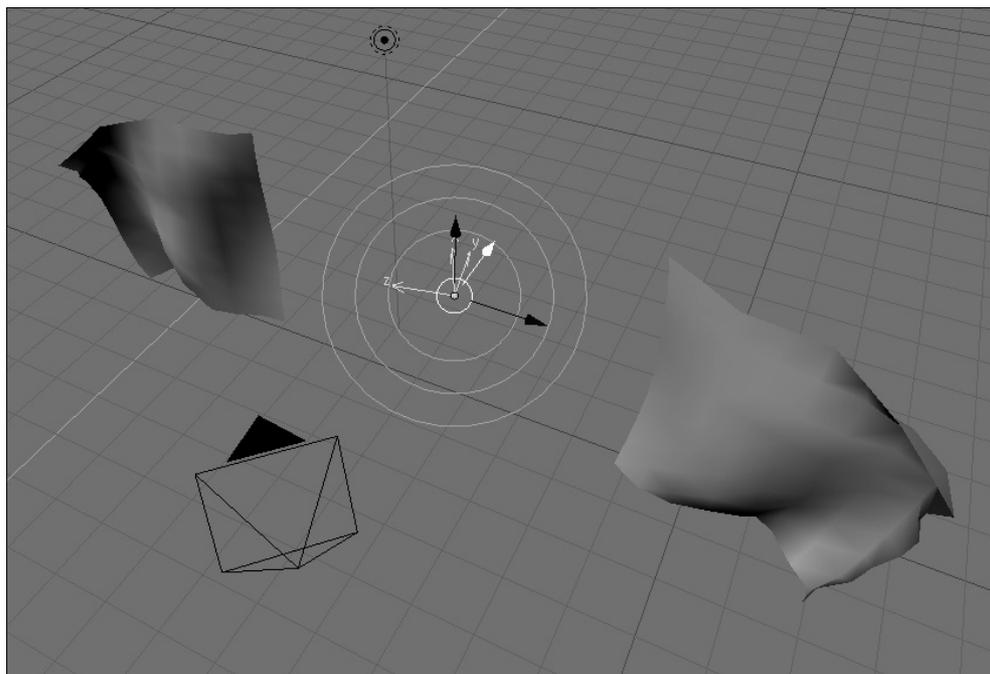


Рис. 6.28. Результат работы установки Spherical

6.5. Создание жидкостей

Новинкой последних версий Blender является возможность моделирования поведения жидкостей. Совокупность физической модели и развитой системой материалов позволяет создавать реалистичные трехмерные сцены с их участием. Стоит заметить, что это по-настоящему трудоемкая задача для системы и отнимает немалое время для обработки. Сам процесс моделирования и настройки достаточно простой.

В создании модели жидкости всегда участвуют по меньшей мере два объекта. Основополагающим является понятие *домен* (domain). Его функция заключается в ограничении пространства, для которого ведутся расчеты поведения жидкости. Физически домен представляет собой куб. Именно примитив Cube и используется для указания расположения домена. Вторым обязательным объектом является *источник* (fluid). В его роли может выступать примитив любой формы.

Смысл совместной работы обоих объектов заключается в следующем. Домен принимает форму источника и постепенно стекает на место своего расположения. Давайте рассмотрим это на простом примере.

Создайте новый проект Blender и добавьте примитивы Plane и Sphere к уже имеющемуся кубу. Плоскость будет служить своеобразным подносом для стекающей жидкости, а сфера — ее источником. Расположите сферу над поверхностью Plane. Куб является носителем домена, поэтому расположите его по центру плоскости так, чтобы нижняя часть полностью ее закрывала. Нижние части обоих объектов должны быть на одинаковой высоте по Z. Обратите внимание на то, чтобы сфера обязательно целиком находилась в пределах куба. Та часть объекта, которая не попала в домен, в расчетах принимать участие не будет (рис. 6.29).

Как и положено, настройки системы жидкости находятся на панелях **Physics buttons**. Выделите сферу и нажмите кнопку **Fluid** в одноименной панели (рис. 6.30).

Так как сфера в примере является источником, то нажмите соответствующую кнопку на панели **Fluid**.

Теперь подготовим соответствующим образом куб. Откройте для него панель **Fluid** и нажмите кнопку **Domain** (рис. 6.31).

Самая большая и главная кнопка — **BAKE** (Выпечь) — предназначена для конечного просчета настроенной имитации. Обратите внимание, что выпечка не является рендером всей сцены целиком, а лишь обрабатывает систему жидкостей. Просчитанный результат программа сохраняет на диск для дальнейшего использования. В поле **Path** нужно указать папку для кэша. После выпечки анимацию можно просмотреть стандартным способом (<Alt>+<A>).

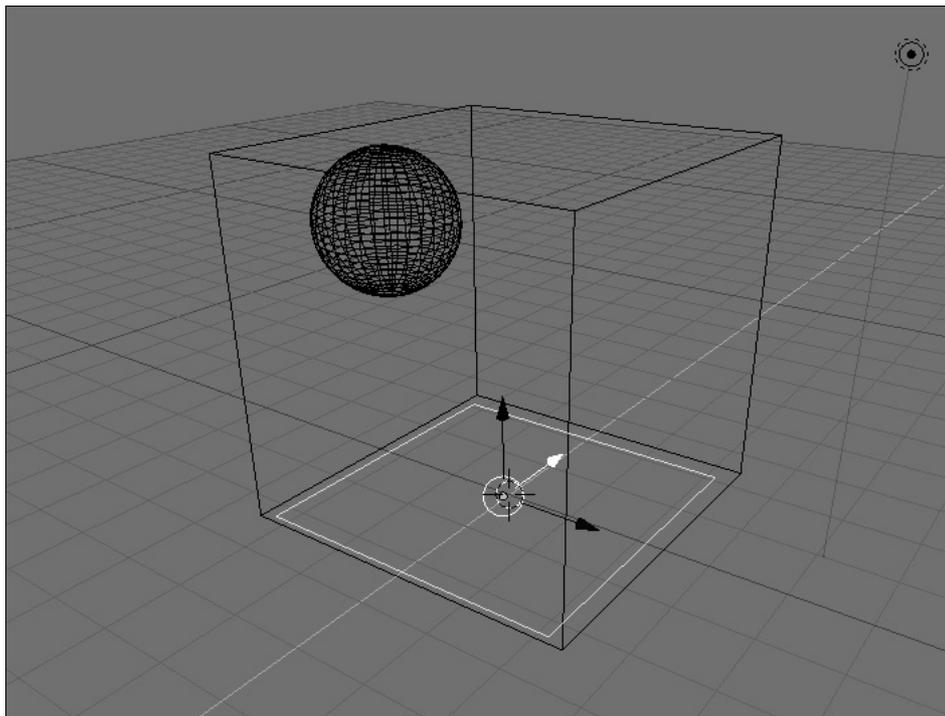


Рис. 6.29. Расположение объектов для примера

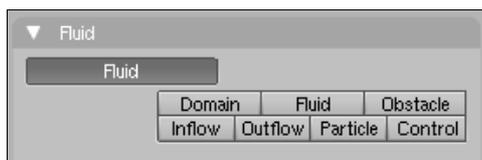


Рис. 6.30. Основные режимы для имитации

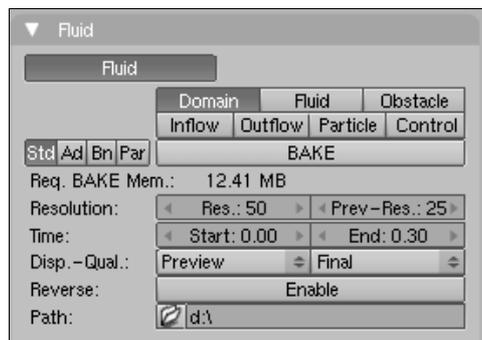


Рис. 6.31. Панель режима Domain

Рассмотрим остальные поля:

- ◆ **Req. BAKE Mem** (Необходимая память). Указывает приблизительный объем места, необходимый для расчета имитации;
- ◆ **Resolution** (Разрешение). Настройка разрешения конечного результата и предварительного просмотра по координатным осям. Увеличение значения приводит к значительному расходу памяти и отрицательно сказывается на обработке. Это поле можно использовать для создания мелких брызг;
- ◆ **Time** (Время). Установка начального и конечного кадров, на протяжении которых должна проигрываться анимация;
- ◆ **Disp.–Qual** (Качество). Качество, с которым происходит вывод изображения в окне программы и при окончательном рендеринге;
- ◆ **Reverse** (Обратно). При активации этой опции программа будет рассчитывать имитацию, как и обычно, но готовая анимация будет проигрываться с конца.

Система жидкостей по умолчанию имеет настройки имитации воды. Нажав кнопку **Advance** (Расширенные опции), можно выбрать другой тип жидкости из нескольких имеющихся или изменить данные вручную (рис. 6.32).

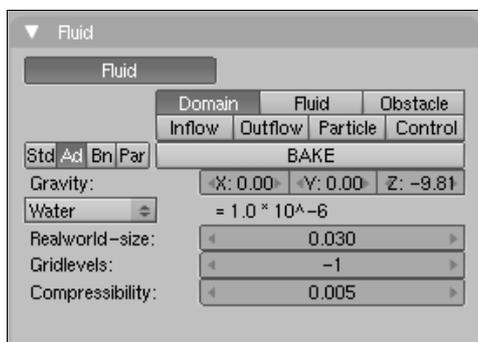


Рис. 6.32. Здесь можно выбрать необходимую жидкость

Теперь, после окончания всех настроек, осталось только выпечь анимацию. Нажмите кнопку **BAKE**, а затем сочетание клавиш <Alt>+<A> для просмотра результата (рис. 6.33).

Совет

К сожалению, Blender не имеет функции сброса просчитанной имитации, но если выбрать объект-домен и отключить, а затем включить главную кнопку **Fluid**, то система вернется в исходное состояние.

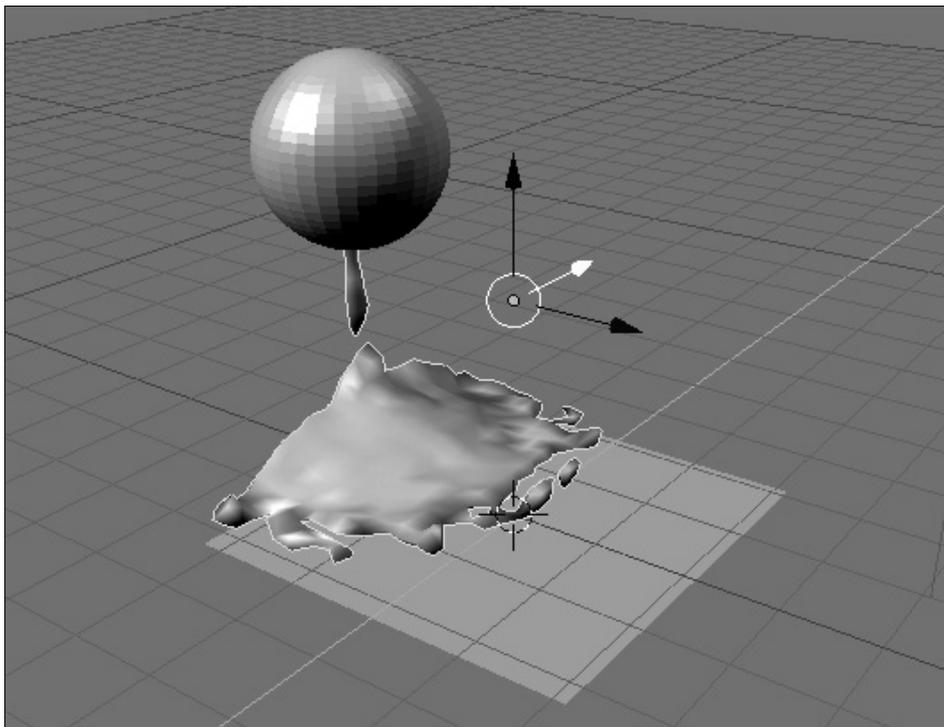


Рис. 6.33. Один из кадров полученной анимации

Как уже говорилось, в качестве домена может быть только куб, и, на первый взгляд, это накладывает серьезные ограничения на использование физики жидкостей. К счастью, программа имеет возможности для работы с любыми объектами. Рассмотрим этот способ.

Создайте еще один куб и поместите на середину плоскости. Теперь нужно сделать его полым. Самый простой способ — воспользоваться выдавливанием. Выделите четыре вершины верхней части объекта и нажмите клавишу <E>. В появившемся меню нужно выбрать пункт **Region** (Область) и сразу же нажать клавишу <Enter>, не сдвигая курсор. Затем, используя масштабирование, немного сожмите выделение и передвиньте по координате Z для получения полого пространства.

Внимание!

Система жидкостей неправильно работает с полыми объектами, если до этого производилось удаление их частей с помощью клавиши <X>. Используйте выдавливание или перерасчет нормалей в меню **Mesh | Normals | Recalculate** (Меш | Нормали | Перерасчет).

Для этого объекта включите в панели **Fluid** кнопку **Obstacle** (Препятствие) и обработайте анимацию (пример на диске содержит настроенные материалы для воды и стекла).

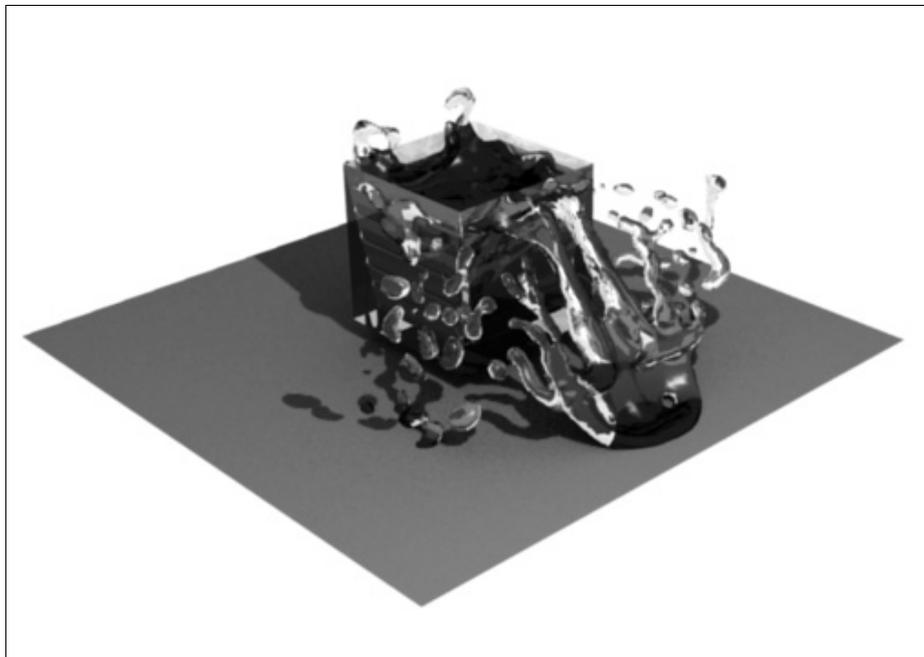


Рис. 6.34. Вода задерживается в емкости

Совет

По умолчанию жидкость стремится вниз по координате Z . Изменить направление можно в настройках объекта `Fluid`, если изменить значение полей **Initial Velocity** (Начальное направление).

ГЛАВА 7



Свет, камера, окружение

На завершающей стадии работы моделлеру приходится уделять большое внимание освещению, настройке камеры и другим, кажущимся малозначительными, задачам. Между тем, правильно настроенный свет и выбор оптимального ракурса оказывают большое влияние на восприятие сцены в целом. Поэтому в данной главе приводится простая и эффективная схема расстановки источников света для получения лучшего результата. Также рассматриваются: глобальное освещение, туман, свойства фона и генератор звезд.

7.1. Источники света и тени

При первом запуске программы вы, наверное, уже обратили внимание на то, что в сцене присутствуют как минимум три объекта: тестовый куб, камера, источник света. Если первый является всего лишь образцом и особой смысловой нагрузки не несет, то в случае удаления любого из оставшихся объектов программа функционировать нормально не сможет. Разговор о камере пойдет чуть дальше, здесь мы рассмотрим работу со светом и некоторые интересные эффекты, с ним связанные.

Источник света представляет собой обычный объект Blender, которым можно манипулировать, к примеру, как примитивом. Единственное исключение в том, что при обработке конечного результата он не будет виден. По сути дела, этот объект является лишь средством позиционирования (рис. 7.1).

Всего программа предлагает пять видов источников света. Каждый из них имеет свои характеристики и особенности:

- ◆ `Lamp`. Основная лампа, используемая по умолчанию при создании новой сцены. Является точечным источником, а излучаемый ею свет распространяется по всем направлениям;

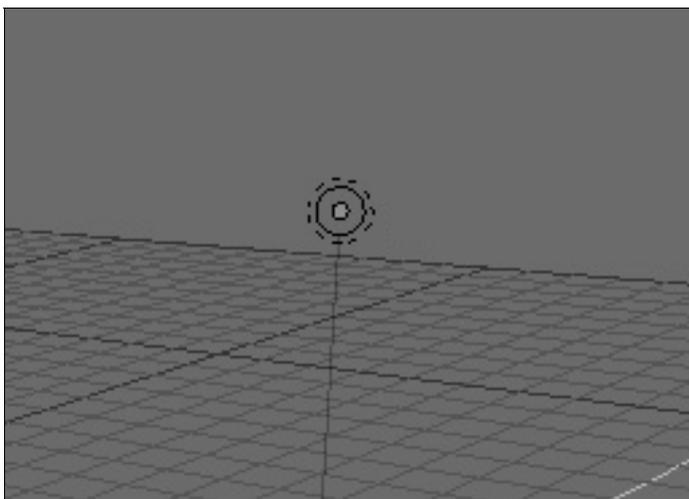


Рис. 7.1. Источник света Lamp

- ◆ **Area**. Направленная лампа, которая может применяться для освещения большой площади. Эмулирует несколько источников света и позволяет изменять форму освещенного участка;
- ◆ **Spot**. Точечный источник, или прожектор;
- ◆ **Sun**. Источник света одинаковой интенсивности вне зависимости от расположения объектов;
- ◆ **Hemi**. Направленный источник света, наподобие **Area**, но несколько более масштабный. Проще всего представить его в виде полушария, закрывающего сцену. Не отбрасывает тени.

Настройки ламп расположены, как обычно, в окне **Buttons Window** (рис. 7.2).



Рис. 7.2. Кнопки для вызова настроек света

Окно **Preview** позволяет не только просмотреть примерную область освещения, но и выбрать другой вид лампы (рис. 7.3).

Основные характеристики ламп доступны в панели **Lamp**, причем некоторые параметры одинаковы для всех видов ламп (рис. 7.4).

В первую очередь, здесь можно настроить мощность излучения (параметр **Energy**) и его окраску. По умолчанию предлагается белый цвет и, как правило, этого достаточно.



Рис. 7.3. Выбор и просмотр источника света

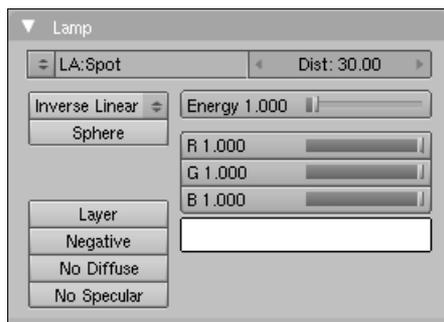


Рис. 7.4. Основные настройки света

Параметр *Dist* (Дистанция) отвечает за расстояние, на которое может распространиться поток света. Не путайте его с мощностью излучения!

Кроме того, все виды ламп позволяют выбрать тип цвета материала объектов, которые будут участвовать в освещении. За это отвечают кнопки с характерным названием: **No Diffuse** (Без диффузии) и **No Specular** (Без отражения).

Совет

Для удобства работы с освещением лучше включить режим прорисовки *Shaded*.

Как уже говорилось, каждый источник света имеет свое предназначение. Если обычный *Lamp* используется для простого освещения объектов, то с помощью *Spot* можно выделить определенное место в сцене. Его работа напоминает принцип действия прожектора, когда конусообразный луч точно направляется в нужную сторону (рис. 7.5).

Представленный на рисунке конус указывает направление излучения света и границы, за пределами которых освещения не будет. При этом внутренняя окружность очерчивает зону максимального освещения, а внешняя отвечает за границу минимального освещения. Таким образом, играя с их размерами, можно добиться как резко очерченного пятна, так и мягкого с плавными переходами. Для указания диаметра внешнего кольца служит параметр *SpotSi*, а для внутреннего — *SpotBl*. Кроме того, можно изменить форму области освещения. По умолчанию программа предлагает овал, но если нажать кнопку **Square**, то область примет форму прямоугольника (рис. 7.6).

Источник *Spot* позволяет добиться интересного эффекта объемного света, т. е. такого света, который как бы проходит сквозь пылевое облако. Для создания его достаточно нажать кнопку **Halo** и отстроить интенсивность в поле **HaloInt** (рис. 7.7).

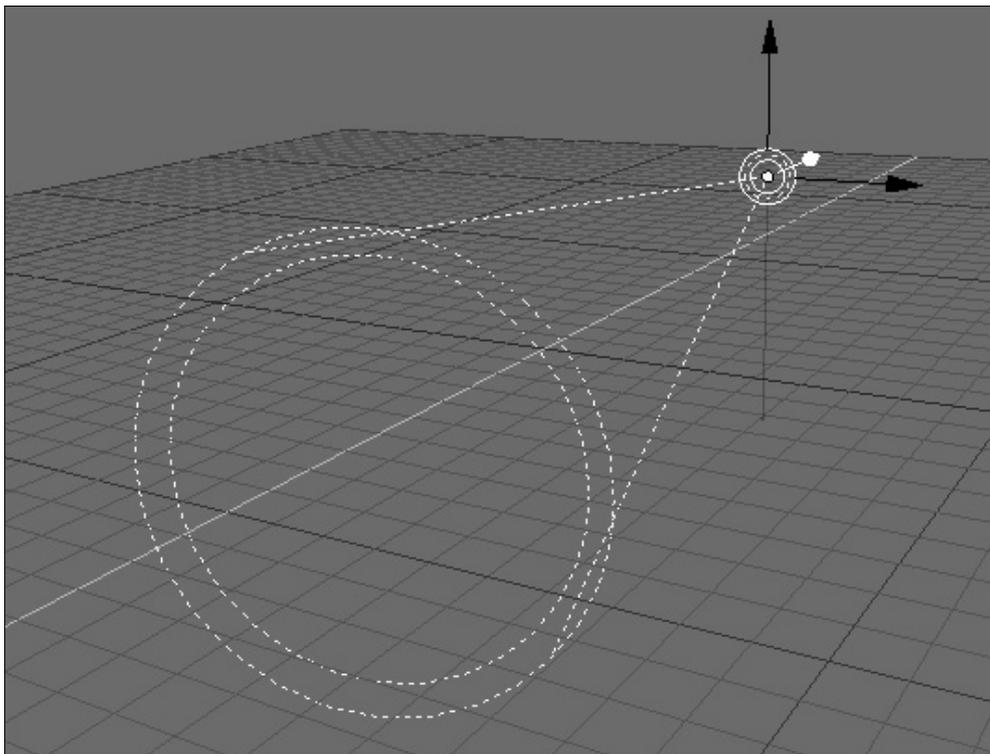


Рис. 7.5. Отображение лампы Spot в сцене

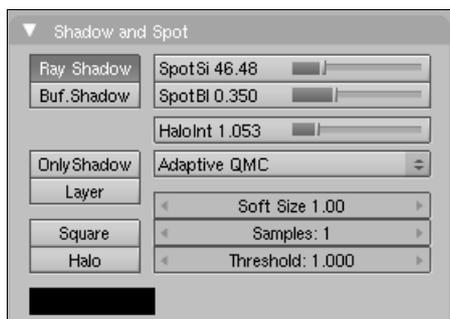


Рис. 7.6. Панель настроек Spot

Невозможно добиться реалистичности картинке без игры светотеней и правильной расстановки источников. Обычно начинающие моделлеры ограничиваются использованием в сцене всего лишь одной лампы, что является в корне неверным. Ведь один источник просто не в силах полноценно осветить модель со всех сторон.

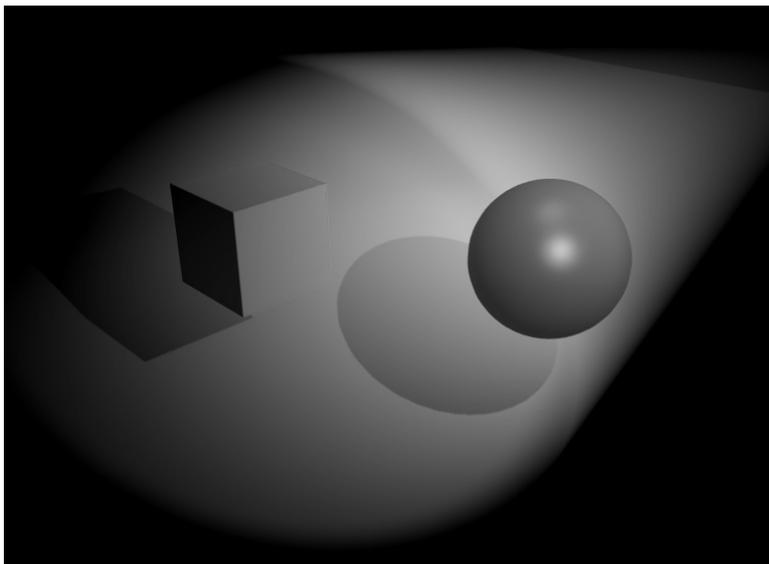


Рис. 7.7. Лампа Spot с активной кнопкой Halo

Чаще всего используется схема освещения с трех точек. На рис. 7.8 показан результат обработки сцены, состоящей из плоскости, модели обезьянки и точечного источника Spot.

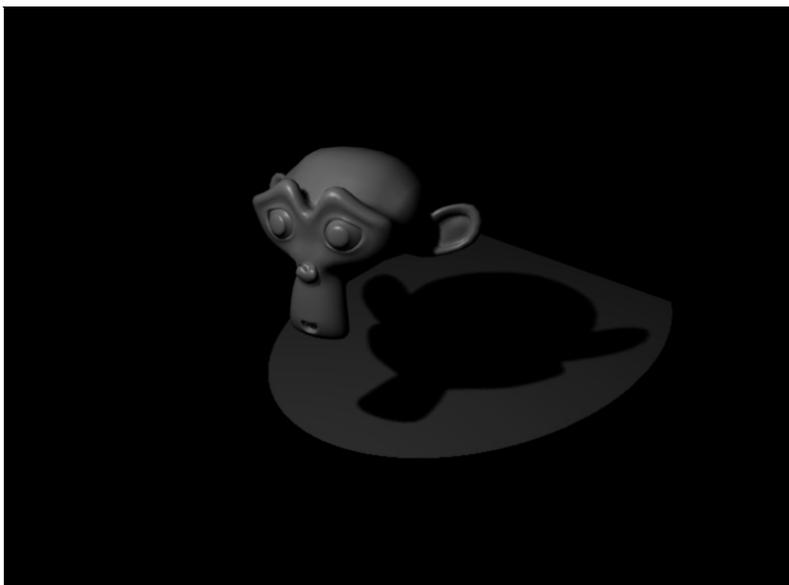


Рис. 7.8. Освещение модели одним источником света

Как видите, результат весьма плачевный. Хотя лицевая сторона модели выглядит достаточно ярко, все остальные видимые стороны просто сливаются с фоном. Да и резкие границы света и теней являются признаком дурного вкуса, если конечно это не первоначальная задумка. Необходимо добавить второй источник света для освещения тыльной стороны сцены. В качестве него нужно использовать лампу, отличную от `Spot`, и поэтому идеально подойдет `Lamp` (рис. 7.9).

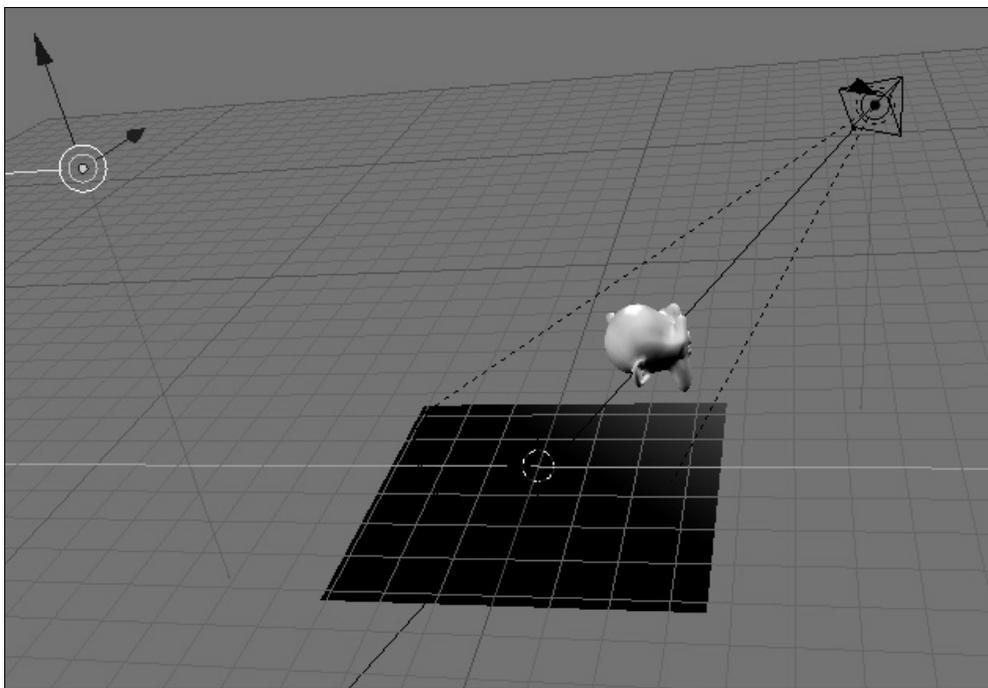


Рис. 7.9. Задняя лампа придает сцене глубину

Так как роль ее вспомогательная, то параметр `Energy` нужно установить ниже, чем у главного источника. Использование лампы заднего плана позволит придать глубину самой сцене и отделить модель от основного фона (рис. 7.10).

Теперь модель выглядит гораздо лучше, за исключением тех областей, куда падают тени от главной лампы. Последний источник в этой схеме носит название "заполняющая лампа" и располагается рядом с камерой. Его задача осветить и смягчить затененные участки на переднем плане. Тут также можно использовать `Lamp` с еще более низкими параметрами `Energy`, нежели у лампы заднего плана (рис. 7.11).

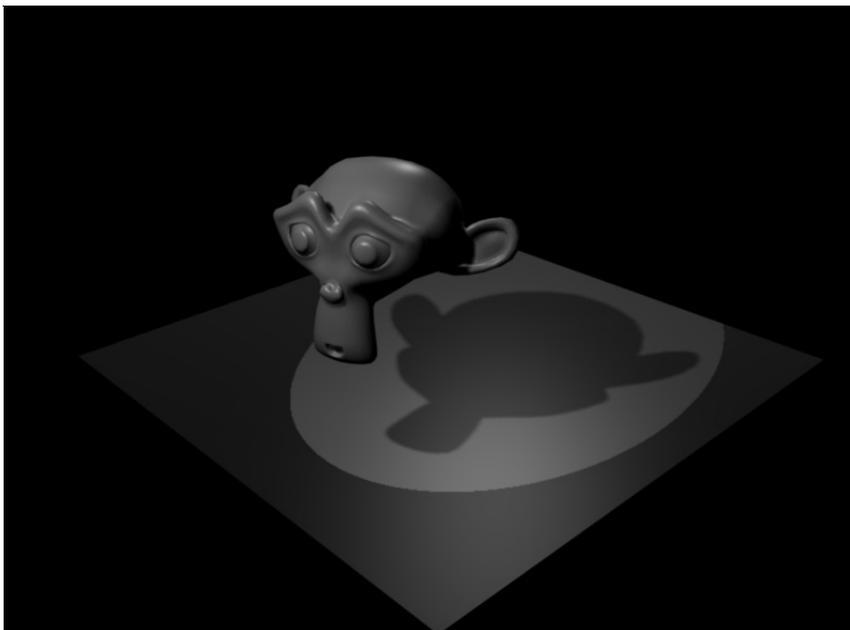


Рис. 7.10. Освещение модели двумя лампами

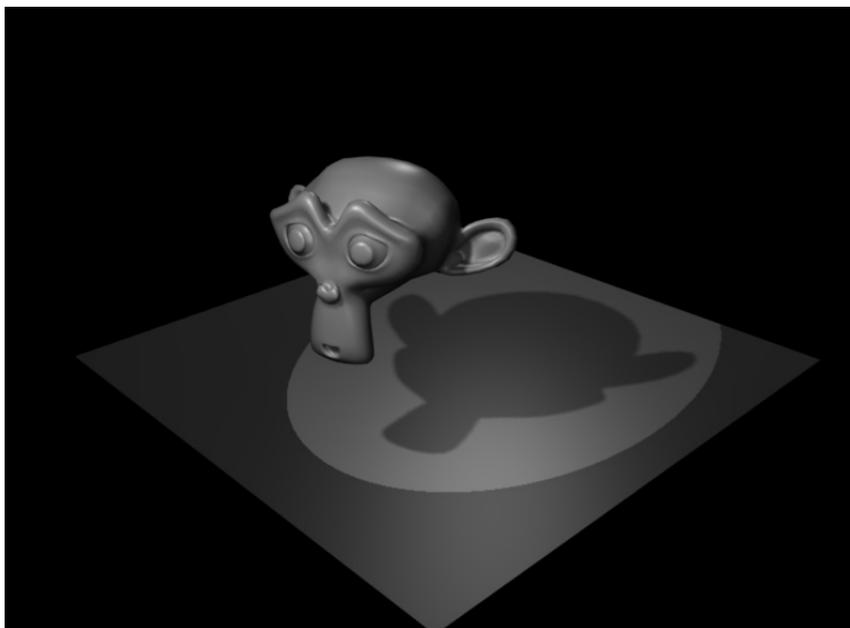


Рис. 7.11. Освещение модели тремя лампами

Теперь и сцена выглядит вполне пристойно. Но это еще не все. Посмотрите на световое пятно от главной лампы, которое имеет слишком резкие границы. Для смягчения можно поиграть параметрами *SpotSi* и *SpotBl*, которые отвечают за размеры колец светового конуса лампы.

Установив и настроив лампы, можно переходить к завершающему этапу — настройке теней. Ранее уже упоминалось, что из пяти источников света только один, *Hemi*, не может отбрасывать тени. Все остальные работают по технологии *ray tracing* (трассировка лучей), что позволяет достичь оптимальных результатов. Правда, есть одно "но": такие расчеты довольно сложны и трудоемки. Поэтому существует упрощенный способ создания теней через *теневого буфера*. Из четырех оставшихся ламп лишь *Spot* умеет работать с *Shadow Buffer*. Такой выбор обусловлен тем, что именно этот тип лампы чаще всего используется для главного освещения модели. Помимо быстроты вычислений *теневого буфера* позволяет работать с так называемыми объемными тенями (рис. 7.12).

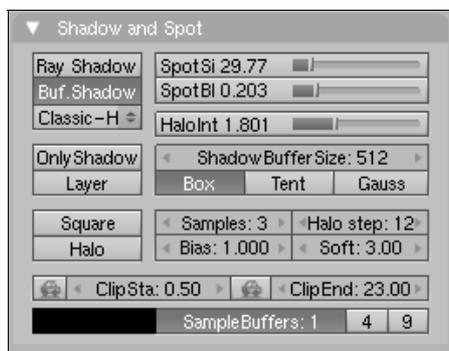


Рис. 7.12. Панель настроек лампы Spot

Все основные функции настроек *Spot*, равно как и теней, располагаются на панели **Shadow and Spot**. Для переключения режима расчета теней имеются кнопки **Ray Shadow** и **Buf.Shadow**. По умолчанию обычно включена вторая.

Для эффективной работы с *теневого буфера* имеются переменные *ClipStart* (Начальная точка) и *ClipEnd* (Конечная точка), которые обозначают промежуток, где объект будет отбрасывать тени. Визуально в окне **3D View** он отображается прямой, находящейся в конусе лампы. Важно знать, что при размещении объекта дальше, чем конечная точка, или ближе, чем начальная, такой объект не будет обрабатываться.

Настройка качества теней осуществляется следующими кнопками:

- ◆ **Soft.** Основной параметр, при увеличении которого происходит сглаживание *теневого* границы;

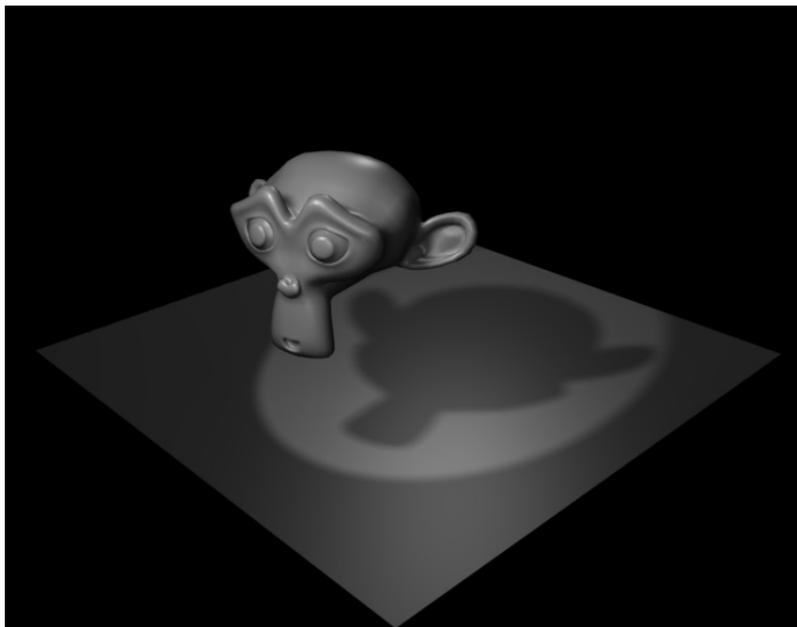


Рис. 7.13. Окончательный вариант настроенного освещения

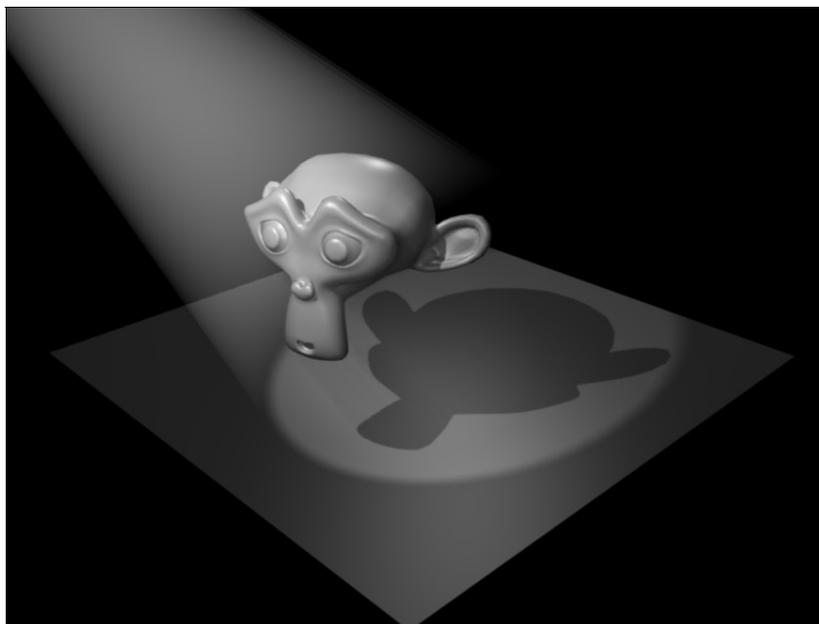


Рис. 7.14. Сцена с объемным светом и тенью

- ◆ **Samples.** Дополнительный алгоритм сглаживания. Увеличение значения улучшает результат, но повышает нагрузку на систему;
- ◆ **Bias.** Смещение, используемое для просчета теней.

Вы уже знаете, что для создания эффекта объемного света нужно включить кнопку **Halo** для лампы *Spot*. Однако если внимательно присмотреться к тени, отбрасываемой моделью при таком освещении, то можно увидеть, что свет проходит сквозь нее и реальной картинке нет (рис. 7.13). Получается, что при наличии объемного света остается обычная тень. Для исправления этой "оплошности" достаточно изменить значение поля **Halo Step** (Уровни свечения). Максимальное значение в 12 единиц добавит сцене реалистичности (рис. 7.14).

7.2. Работа с камерой

Вторым обязательным объектом в проекте Blender является *камера*. Именно с позиции камеры программа просчитывает конечный результат. Она представляет собой обычный объект Blender, который можно вращать, передвигать или масштабировать.

Внимание!

Изменение масштаба не влияет на параметры просмотра камерой и является только удобным способом подгонять ее размер под рабочую сцену.

Так как единственной функцией камеры является установка точки зрения для рендера программы, то имеется специальный режим просмотра в окне **3D View**, вызываемый клавишей <NumPad 0>. Удобнее всего настраивать вид камеры, имея активными несколько окон просмотра с разных позиций, включая и саму камеру. Можно настроить самому или выбрать подходящую заготовку из меню окна **User Preferences** (Пользовательские настройки).

Объект *Camera* представляет собой стилизованный примитив, где треугольник обозначает ее верхнюю точку. Если смотреть с точки зрения камеры, то можно увидеть, что имеются ограничительные рамки в количестве трех штук. Каждая из них несет определенную смысловую нагрузку. Первая, внешняя, является декоративной и просто демонстрирует вид из камеры. Вторая рамка указывает область, в пределах которой будет просчитываться сцена. А вот третья предназначена в качестве условных границ для использования текста. Конечно, можно игнорировать текстовую рамку, но делать это не желательно по одной простой причине: при выводе на телевизор готового анимационного ролика есть шансы, что картинка может обрезаться за границами рамки. Подобная практика безопасной текстовой зоны используется во всех профессиональных программах для монтажа видео (рис. 7.15).

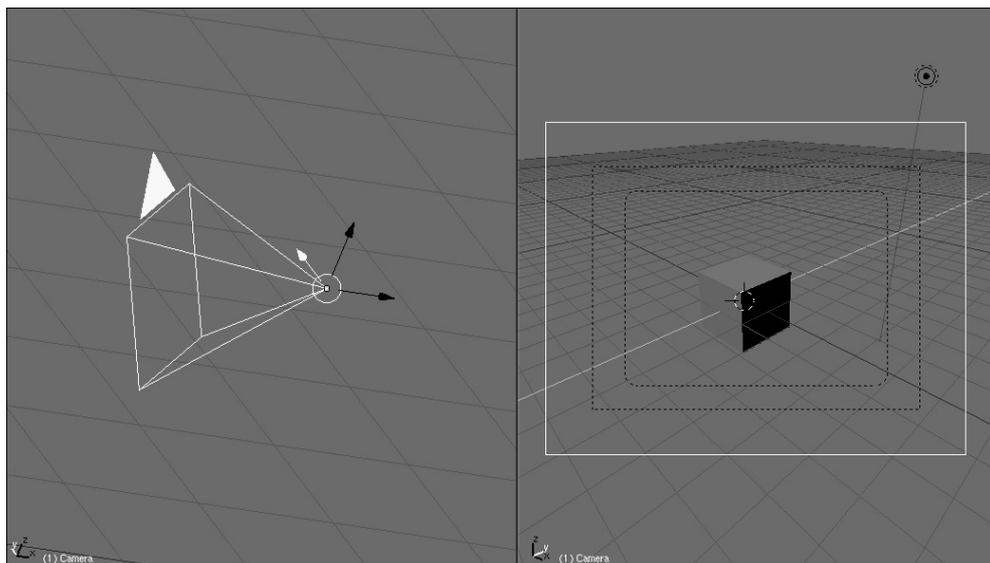


Рис. 7.15. Так выглядит камера и вид из нее

Настроек камера имеет не так уж много, и все они находятся на панели **Camera**, которая становится доступной при нажатии клавиши <F9> (рис. 7.16).

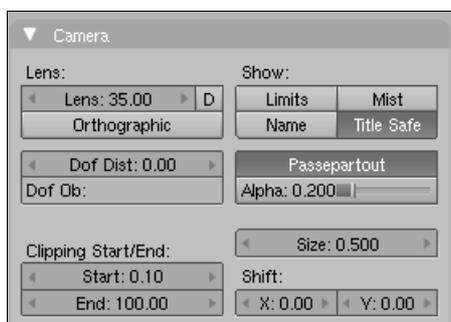


Рис. 7.16. Настройки камеры

Сначала нужно определиться с длиной линзы, за которую отвечает параметр **Lens** (Линзы). Лучше всего использовать значение 35 мм, стоящее по умолчанию. Параметры **Clipping Start/End** (Начало и конец обрезки) необходимы для установки зоны видимости камеры. Все, что находится за их пределами, в расчете участвовать не будет. Для перемещения картинка в кадре без реального перемещения объектов в сцене можно использовать настройки функции **Shift** (Смещение). Путем изменения координат по X или Y можно добиться необходимого смещения кадра (рис. 7.17).

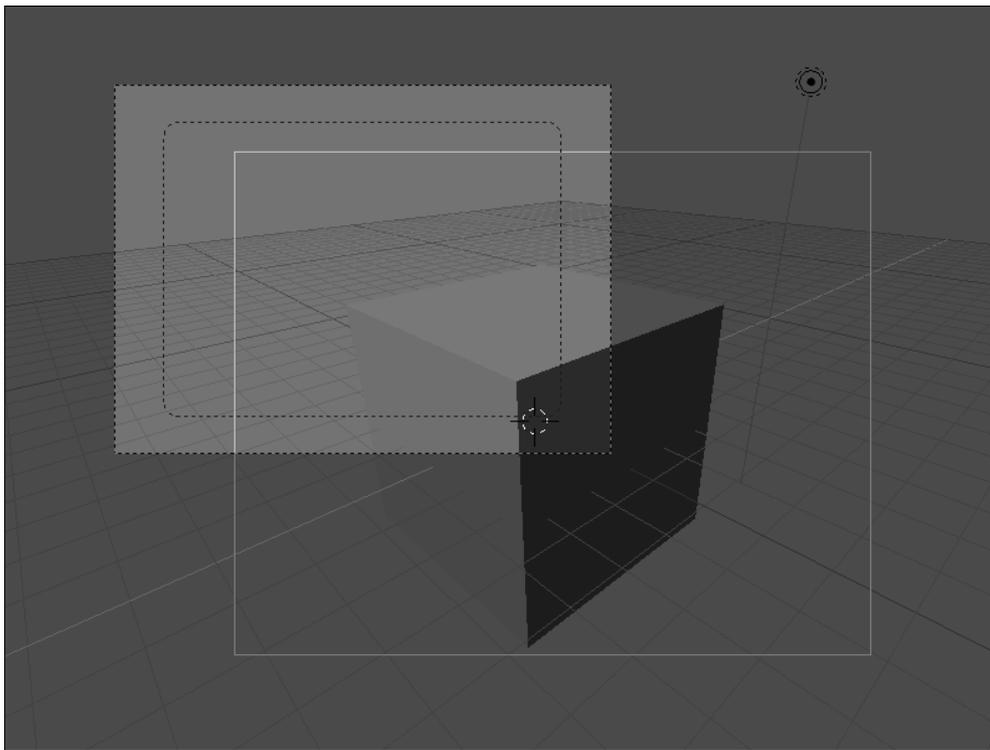


Рис. 7.17. Функция Shift в действии

Остальные функции предназначены для настройки отображения камеры в окне **3D View**:

- ◆ **Limits (Лимиты)**. Прямые, визуально указывающие фокусное расстояние для камеры (параметры *Clipping Start/End*);
- ◆ **Name (Имя)**. В окне просмотра камерой появляется ее название;
- ◆ **Title Safe (Титровальная рамка)**. Включение или отключение безопасной зоны для показа титров;
- ◆ **Mist (Туман)**. Прямые, указывающие границы действия тумана (см. разд. 7.3);
- ◆ **Passepartout (Перекрытие)**. Позволяет затемнить неактивную область экрана за пределами активной рамки. Уровень можно регулировать бегунком **Alpha** (Прозрачность).

При работе с анимацией часто бывает нужно заставить камеру не только следовать за определенным объектом, но и следить за ним вне зависимости от расположения самой камеры. Если в первом случае достаточно установить

иерархическую связь, то для слежения нужно использовать особый вид привязки. Для этого нужно выделить сначала камеру, потом объект слежения и нажать <Ctrl>+<T>. Чтобы удалить такую связь, используется сочетание клавиш <Alt>+<T>. Рассмотрим простой пример анимации камеры по пути с функцией слежения за объектом.

Откройте новый проект Blender и добавьте объект `Path`. Отредактируйте путь для желаемого перемещения камеры. Выделите камеру и добавьте ей привязку `Follow Path` (<F7>). В появившемся окне введите название самой кривой и нажмите кнопку **CurveFollow** (см. разд. 5.4).

Если теперь проиграть анимацию, то камера, как и положено, пробежится по проложенному пути.

Внимание!

При назначении привязки `Follow Path` для камеры, она удалится от пути на фокусное расстояние, которое указано в ее настройках. Измените параметр `Clipping End` или вручную передвиньте объект.

Теперь выделите камеру и затем, удерживая клавишу <Shift>, — объект слежения. Нажмите сочетание клавиш <Ctrl>+<T>. В появившемся меню нужно выбрать пункт **TrackTo Constrain**. Запустите проигрывание анимации.

Совет

В одной сцене можно использовать несколько камер. Для добавления новой просто выберите ее из меню создания объектов. Однако в сцене может быть только одна активная камера, которая учитывается при обработке. Для активации необходимо выделить нужную камеру и нажать <Ctrl>+<NumPad 0>.

7.3. Настройка окружения

Сцена имеет глобальные свойства, изменение которых влияет на конечный результат. Нажмите клавишу <F12>, чтобы обработать текущий кадр сцены. По умолчанию программа использует в качестве фона сплошной синий цвет. В действительности на его месте можно установить не только другой цвет, но и картинку, и даже процедурную текстуру имитации неба. Функции настройки *окружения* расположены в окне **Buttons Window** и доступны при нажатии кнопки **World buttons** (рис. 7.18).



Рис. 7.18. Нажатие этих кнопок откроет панели настроек окружения

Всего лишь несколькими щелчками мыши можно создать в сцене красивый градиентный фон, имитирующий небо. Настройки градиента и смешивания находятся на панелях **Preview** и **World** (рис. 7.19).

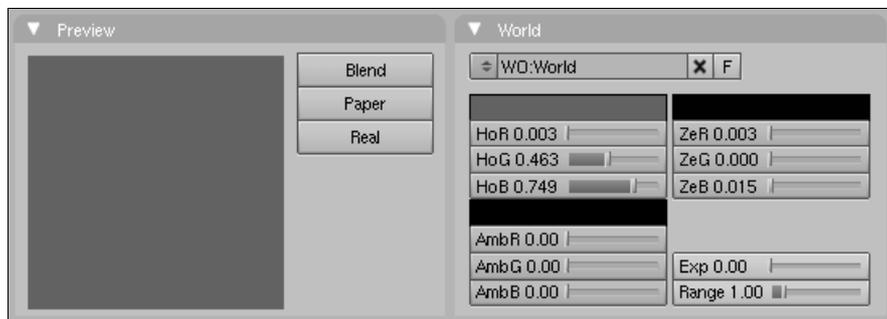


Рис. 7.19. Панели для настройки фона сцены

Цвет фона состоит из двух составляющих: горизонта и зенита. За это отвечают группы цветов **HoR (G, B)** и **ZeR (G, B)** соответственно. Сменить режим смешивания можно на панели **Preview** следующими кнопками:

- ◆ **Blend** (Смешивание). Основная кнопка, активирующая режим смешивания. Создает градиент от основания до верха картинке. При этом местоположение и ориентация камеры не учитываются;
- ◆ **Paper** (Бумага). Равноценно первому пункту, но с учетом поворота камеры. При наличии дополнительной текстуры учитываются ее координаты;
- ◆ **Real** (Реальность). Создается двойной градиентный переход по схеме *зенит + горизонт + зенит*. Учитывается расположение и ротация камеры.

Кнопки **Paper** и **Real** могут использоваться одновременно. В этом случае получается двойной градиентный переход, как при одной нажатой кнопке **Real**, но при ротации камеры цвет горизонта всегда остается посередине (рис. 7.20).

Совет

Помимо градиента можно дополнительно использовать текстуры. Принцип работы с ними ничем не отличается от рассмотренного в главе 4 материала. Естественно, при этом должна быть включена кнопка **Blend**.

Полезной функцией при разработке космических сцен является возможность создания звезд. Для этого имеется специальный генератор, расположенный на панели **Mist/Stars/Physics** (рис. 7.21).

Генерируемые звезды не являются трехмерными объектами, но имеют зависимость от расположения камеры. Таким образом, при ротации камеры будет

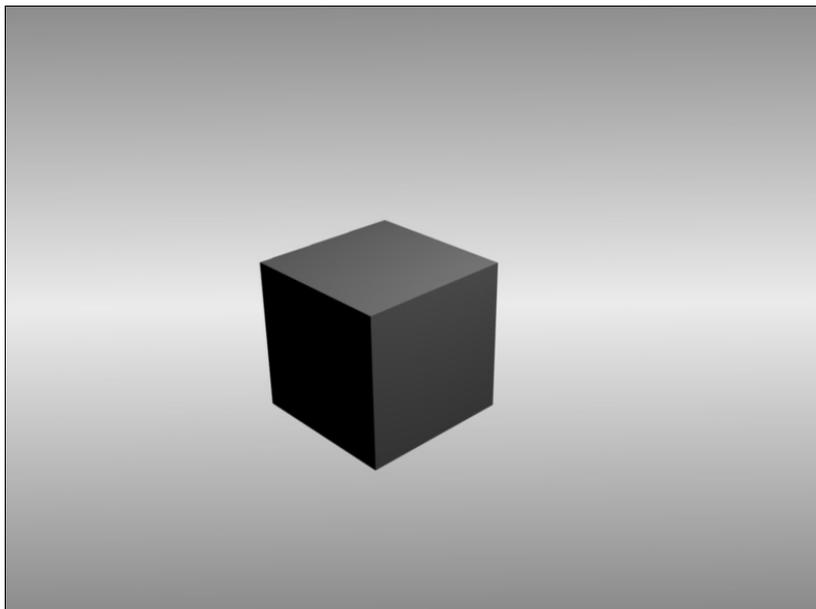


Рис. 7.20. Результат работы функции смешивания

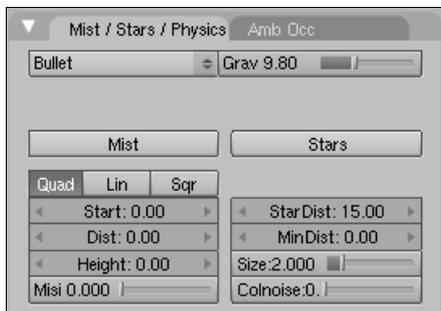


Рис. 7.21. Создание звезд и тумана

изменяться картина звездного неба. Для включения генерации достаточно нажать кнопку **Stars** (Звезды). Доступны следующие параметры:

- ◆ **StarDist** (Дистанция). Регулируется плотность звезд за счет изменения расстояния между ними;
- ◆ **MinDist** (Минимальная дистанция). Расстояние между звездами и камерой. В зависимости от этого параметра и фокуса самой камеры в ее поле зрения будет попадать определенное количество имеющихся звезд. Этим удобно пользоваться при создании сцены выхода в космос, где по мере удаления от планеты все большее количество звезд становится видимым;

- ◆ **Size (Размер)**. Среднее значение для размера звезды;
- ◆ **ColNoise (Цветовой шум)**. Позволяет добиться случайного применения цветов к звездной панораме.

Совместным использованием функций смешивания фона и генерации звезд можно с легкостью добиться красивой картинки звездного неба (рис. 7.22).

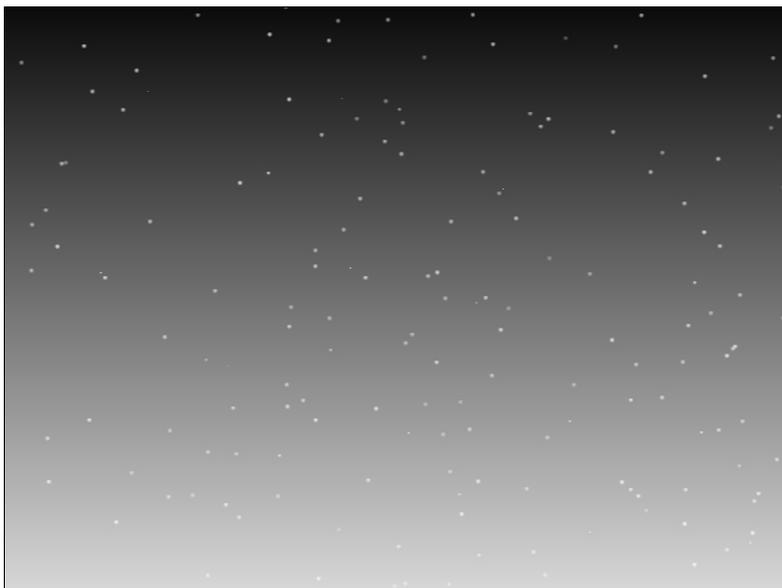


Рис. 7.22. Звездное небо и горизонт

Для придания реализма сцене можно использовать туман. Посмотрите на обработанную картинку, где кубы уходят в бесконечность (рис. 7.23). На самом деле такого ощущения не возникает. Дело в том, что в реальности далеко расположенные объекты из-за атмосферных явлений постепенно теряют четкость. Трехмерная картинка по умолчанию таким свойством не обладает, и все объекты, попадающие в поле зрения камеры, несмотря на удаление, будут все так же четко просматриваться. В случае выхода за фокусное расстояние камеры эти объекты будут грубо обрезаны. Функция тумана позволяет имитировать постепенное исчезновение объектов в зависимости от расстояния. Естественно, можно создать такой туман, что, как говорится, не будет видно собственного носа.

Для включения тумана необходимо нажать кнопку **Mist** на панели **Mist/Stars/Physics** (см. рис. 7.21). Доступны следующие настройки:

- ◆ **Start (Начало)**. Расстояние от камеры до начальной точки тумана;
- ◆ **Dist (Дистанция)**. Глубина создаваемого тумана;

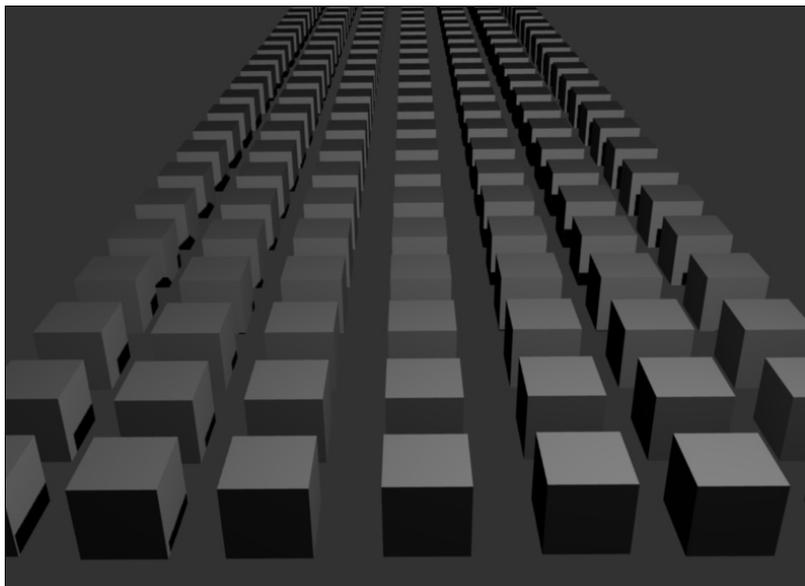


Рис. 7.23. Кубы, уходящие к горизонту

- ◆ **Height (Высота)**. Этот параметр позволяет регулировать высоту тумана. При нулевом значении плотность тумана одинаковая;
- ◆ **Mist (Интенсивность)**. Средняя плотность тумана.

Для удобства настройки параметров тумана следует включить кнопку **Mist** в опциях камеры (см. разд. 7.2). В этом случае программа создает прямую, визуально демонстрирующую настройки тумана (рис. 7.24).

Кнопки **Quadratic**, **Linear**, **Square** позволяют выбрать подходящий коэффициент рассеивания тумана в зависимости от удаления от камеры (рис. 7.25).

Настройки окружения позволяют использовать так называемый *окружающий свет*. Его особенность в том, что он не имеет отдельного источника света и равномерно освещает объекты со всех сторон с учетом наложения теней. Этот свет нужно использовать одновременно с другими источниками света для придания реалистичности. Для включения его необходимо нажать кнопку **Ambient Occlusion** на панели **Amb Occ** (рис. 7.26).

Рассмотрим некоторые элементы:

- ◆ **Use Falloff (Затухание)**. Включение этой кнопки обеспечит более качественный рендер теней. Дополнительно силу эффекта можно регулировать в окне **Strength (Сила)**;
- ◆ кнопки **Add (Добавить)**, **Sub (Вычесть)** и **Both (Оба)** позволяют подобрать оптимальный вариант яркости объекта;

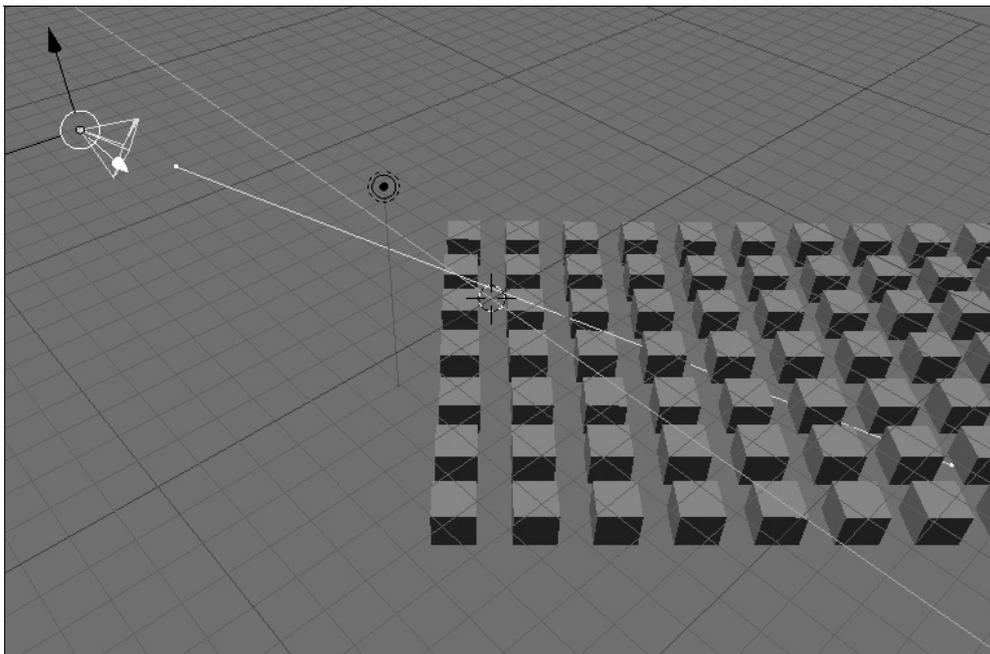


Рис. 7.24. Визуальная демонстрация области тумана

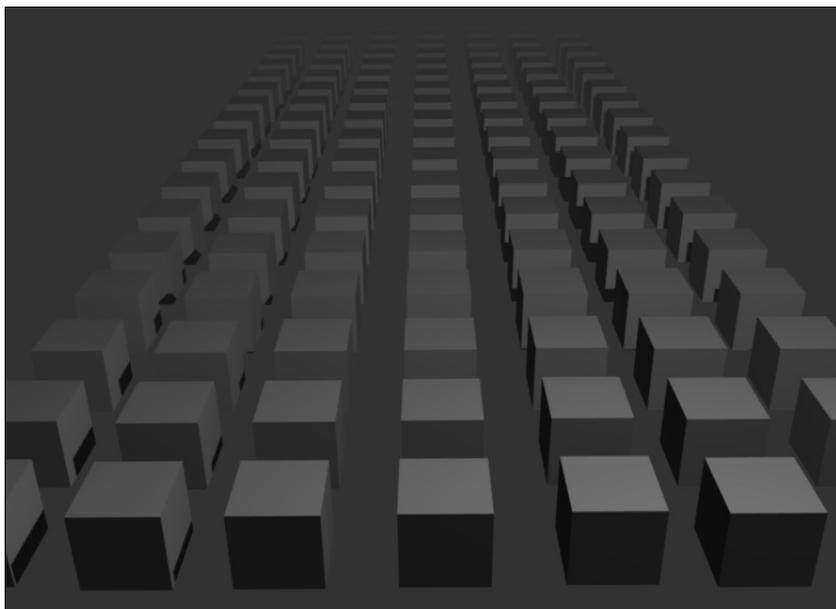


Рис. 7.25. Обработанная картинка с использованием тумана

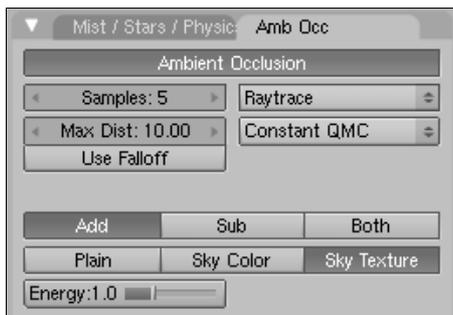


Рис. 7.26. Настройки окружающего света

- ◆ кнопки **Plain** (Простой), **Sky Color** (Цвет неба), **Sky Texture** (Текстура неба) отвечают за источник цвета, который будет использовать окружающий свет. В случае с **Plain** подходящий цвет можно настроить в панели **World** в опциях **AmbR (G, B)**. Два остальных варианта берут за основу цвета градиента или текстуру окружения;
- ◆ **Energy** (Энергия). Сила освещения.

Внимание!

Использование окружающего света позволяет достичь замечательного результата, но требует немалых вычислительных мощностей.

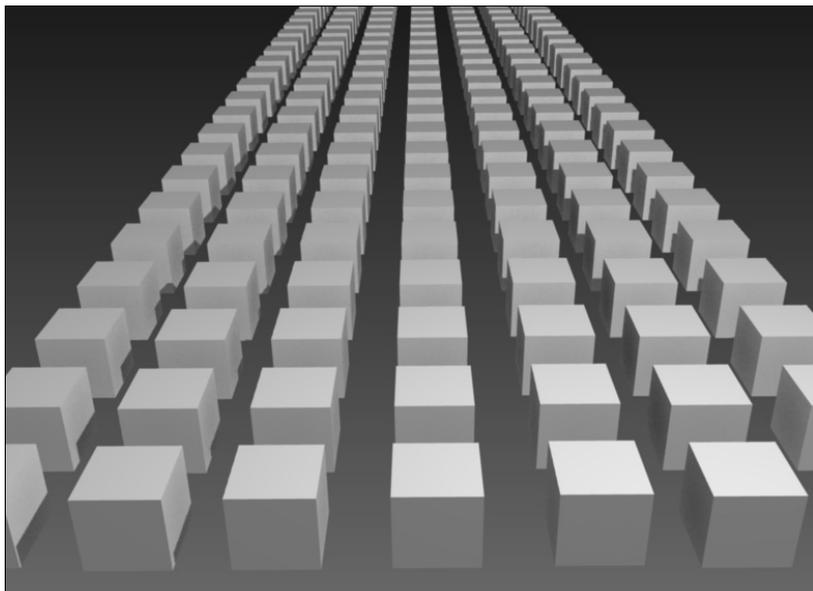


Рис. 7.27. С окружающим светом картинка выглядит эффектнее

ГЛАВА 8



Система рендеринга Blender

Наверное, самый радостный момент в любой работе — это получение осязаемого результата, в нашем случае — картинки или видеофайла. Но для этого нужно немного поднапрячься, ведь от правильной настройки системы обработки (или, как ее еще называют, *рендера*) зависит не только качество, но и время выполнения. Не секрет, что обработка занимает немало времени, а его подчас совсем не хватает. Эта глава поможет вам разобраться с рендером программы, настроить его под свои задачи и добиться оптимального равновесия между качеством и затратами времени.

8.1. Основы обработки

Заключительным этапом в разработке сцены является ее окончательная обработка в виде отдельной графической картинки или видеофайла. Система рендеринга программы Blender позволяет обрабатывать как отдельные картинки, так и анимационные последовательности. Помимо своей основной функции рендеринга сцены, программа умеет создавать карты нормалей, высот, окружения. Все это также располагается на панелях обработки и имеет специфическое название — выпечка. С этим вы уже сталкивались, когда разговор шел о работе с материалами. Настройки рендера программы располагаются на панелях, которые вызываются клавишей <F10> или соответствующей кнопкой на заголовке **Buttons Window** (рис. 8.1).



Рис. 8.1. Кнопки для вывода панелей рендера

Экспериментируя с программой по мере чтения книги, вы не раз использовали клавишу <F12> для вывода в отдельном окне результата обработки теку-

шего кадра. Это окно, несмотря на свой простой вид, имеет несколько полезных для работы функций:

- ◆ Масштабирование. Используйте для этого колесико мыши;
- ◆ Панорамирование. Нажмите центральную кнопку мыши и перемещайте в нужную сторону;
- ◆ Пипетка. Если нажать и удерживать левую кнопку мыши, то при перемещении курсора в нижней области окна будет выводиться информация о пикселе под курсором: координаты, цветность, глубина по Z.

Для сохранения результата обработки на диск достаточно нажать клавишу <F3> или выбрать пункт меню **File | Save Rendered Image** (Файл | Сохранить картинку). По умолчанию файл сохраняется в формате JPEG с разрешением 800×600 пикселей. Основные настройки вывода располагаются на панели **Format** (рис. 8.2).

Программа поддерживает большое количество графических и видеформатов. Подобрать подходящий можно в меню, расположенном на этой панели. В зависимости от выбора Blender может предложить уточнить настройки формата в отдельном окне (рис. 8.3).

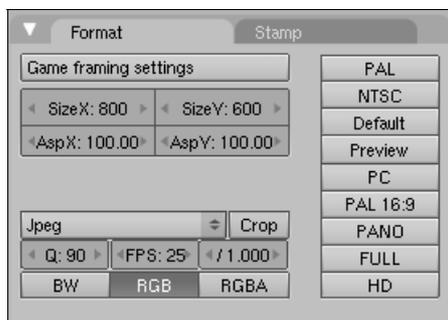


Рис. 8.2. Настройки формата рендера

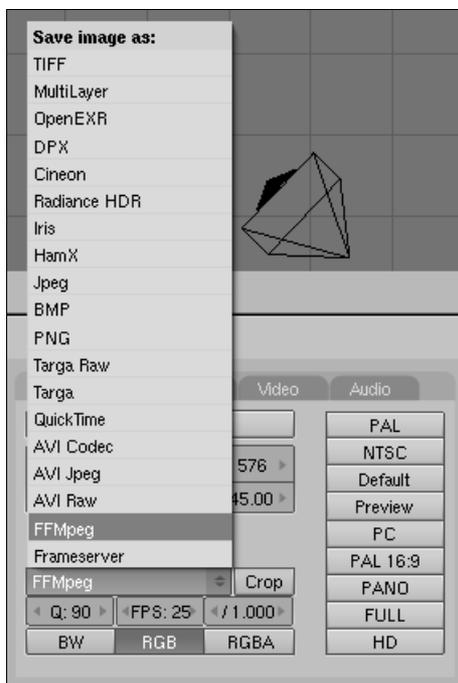


Рис. 8.3. Выбор формата для обработки

Следующий шаг — определиться с разрешением обрабатываемой картинки. Сделать это можно вручную, заполнив поля **SizeX** (Размер по горизонтали) и **SizeY** (Размер по вертикали), или воспользоваться уже имеющимися заготовками. Также следует учитывать значения **Aspect Ratio (AspX и AspY)** — соотношение размеров сторон пиксела.

Совет

В России, равно как и в Европе, используется телевизионный стандарт PAL (720×576, 25 кадров в секунду). Поэтому для вывода анимации предпочтительно использовать заготовку *Pal*. Для предварительного просмотра результата очень хорошо подходит заготовка *Preview*.

После выбора подходящего формата можно смело нажимать кнопку **Render** для обработки текущего кадра. Однако перед рендером анимации необходимо подкорректировать некоторые параметры в панелях **Anim** (Анимация) и **Output** (Выход). Панель **Anim**, предназначенная для обработки анимации, содержит поля для указания начального и конечного кадра (**Start** и **End**). В панели **Output** нужно указать путь для сохранения результата. Вот теперь можно нажимать кнопку **Anim** для рендера анимации.

8.2. Дополнительные возможности

Существует немало параметров, влияющих на качество и содержимое обрабатываемой сцены. Рассмотрим наиболее полезные.

При просмотре обработанной анимации можно заметить, что движение объектов выглядит неестественно. Причина этого в особенностях нашего зрения видеть быстро двигающиеся объекты размыто, в то время как программа с максимальной четкостью штампует кадр за кадром. Для реализации размытости можно воспользоваться эффектом *Motion Blur* (Размытие движения). Необходимые функции расположены на панели **Render** (рис. 8.4).

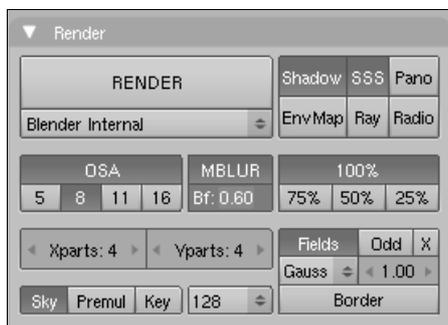


Рис. 8.4. Основные настройки рендера и обработка активного кадра

Нажмите кнопку **MBLUR** и отрегулируйте силу эффекта в поле **Bf** (Blur factor, качество размытия). Использование этой функции дает реалистичный результат, но очень сильно замедляет обработку (рис. 8.5).

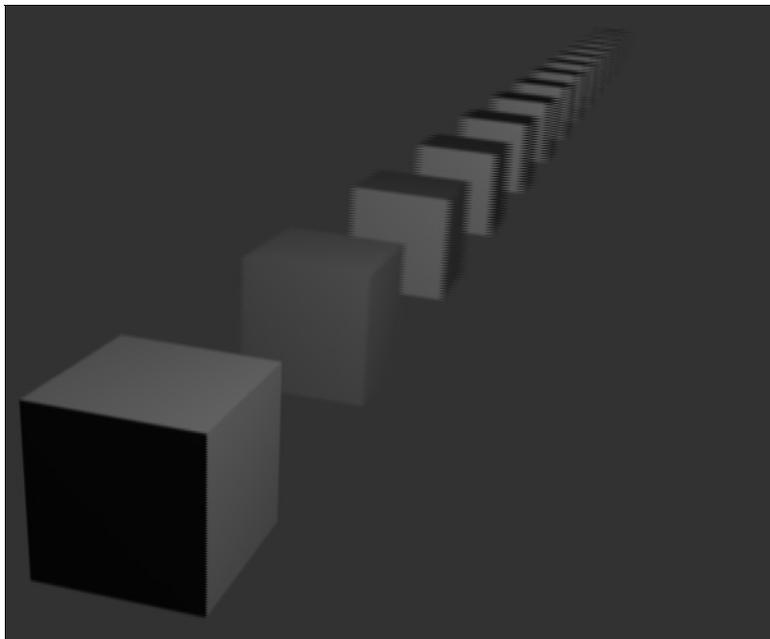


Рис. 8.5. Эффект Motion Blur

Программа на этапе моделирования позволяет применять различные способы сглаживания объектов. Однако при растеризации изображения тонкие линии и края моделей могут содержать артефакты в виде зубчиков. Для удаления ненужных искажений рендер программы Blender использует метод *сглаживания* (anti-aliasing), суть которого заключается в разбиении пиксела на несколько образцов и вычислении цвета по среднему значению соседних пикселей. На языке Blender этот процесс называется *оверсэмплингом* (OSA). Настройки сглаживания можно найти на панели **Render**. Нажмите кнопку **Osa** и выберите желаемое качество в диапазоне от 5 до 16 (см. рис. 8.4).

Рассмотрим еще один очень интересный эффект создания "мультиязышной" окантовки, когда вокруг каждого объекта в сцене прорисовывается линия определенного цвета. Этот эффект разумно использовать совместно с шейдером **Toon** (см. разд. 4.3) для окончательного достижения рисованного кадра. Для активации эффекта нажмите кнопку **Edge** на панели **Output**, а в окне **Edge Setting** (Настройки окантовки) можно выбрать цвет и толщину линии (рис. 8.6).

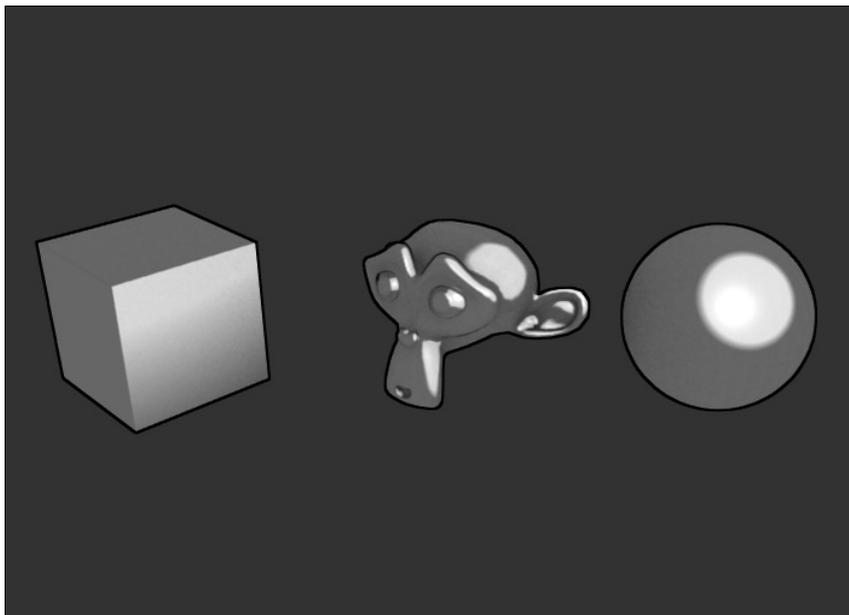


Рис. 8.6. Эффект Edge совместно с шейдером Toon

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ 1

Список основных горячих клавиш

Файловые операции

Клавиши	Описание
<Ctrl>+<X>	Создание нового проекта. Программа выведет предупреждение об удалении предыдущей сцены
<F1>	Открытие сохраненного ранее проекта
<Shift>+<F1>	Загрузка дополнительных ресурсов без удаления активного проекта
<F2>	Сохранение текущего проекта. В появившемся окне можно использовать клавиши <+> и <-> для инкремента номера файла
<Ctrl>+<W>	Быстрое сохранение
<F3>	Сохранение заранее отрендеренной картинки в графический файл. По умолчанию используется JPEG. Для изменения формата картинки необходимо настроить рендер программы
<Ctrl>+<Q>	Выход из программы. Обязательно перед этим сохраните свой проект. Перед закрытием Blender не выводит сообщение о несохраненных данных

Выбор режима работы окна *Buttons Window*

Клавиши	Описание
<F4>	Настройка игровой логики
<F5>	Панель материалов, шейдеров, ray tracing

(окончание)

Клавиши	Описание
<F6>	Работа с текстурами
<F7>	Основные параметры текущего объекта, управление привязками
<F8>	Глобальные настройки сцены, окружения
<F9>	Основные инструменты редактирования объекта (доступны в режиме редактирования)
<F10>	Окно настроек рендера программы
<F11>	Показ последней отрендеренной картинки
<F12>	Запуск рендера сцены с параметрами по умолчанию

Управление просмотром сцены

Клавиши	Описание
<NumPad 0>	Камера
<NumPad 1>	Передний план
<NumPad 3>	Вид сбоку
<NumPad 5>	Переключение проекций с ортографической на перспективную
<NumPad 7>	Вид сверху
<NumPad точка>	Центрирование выделенного объекта
<NumPad />	Переключение с глобального на локальный вид
<Home>	Просмотр всех объектов в сцене
<NumPad 2>, <NumPad 4>, <NumPad 6>, <NumPad 8>	Ротация сцены в окне 3D View
<NumPad +>, <NumPad ->	Масштабирование сцены
<Shift>+<F>	Режим "плавающей", или, как ее еще называют, "игровой" камеры. Управление мышью и клавишами <WASD>. Для выхода используется клавиша <Esc>

Создание и редактирование объектов

Клавиши	Описание
<Пробел>	Вывод меню создания объектов
<S>	Масштабирование
<R>	Вращение
<G>	Перемещение
<N>	Панель трансформации объекта
<M>	Управление расположением объекта в системе слоев
<H>	Скрытие выделенных объектов. Нажатие <Alt>+<H> возвращает все на место
<Tab>	Переход в режим редактирования
<O>	Включение и отключение режима пропорционального редактирования (доступно только в режиме редактирования)
<W>	Контекстное меню возможных операций для объекта
<A>	Выделение всех объектов в сцене или структуры объекта в режиме редактирования. Повторное нажатие снимает выделение
	Прямоугольная рамка выделения. Доступна в объектном и редактируемом режимах программы
<X>	Удаление всего объекта целиком или выделенных его частей

ПРИЛОЖЕНИЕ 2

Инсталляция

2.1. Установка программы в Windows

Установка Blender для операционной системы Windows 98/XP/Vista является стандартной и происходит точно так же, как и любых других программных пакетов. Откройте на сопроводительном диске папку install и запустите файл blender-2.48a-windows.exe (рис. 2.1).



Рис. 2.1. Начальный этап установки Blender

Нажмите сначала кнопку **Next** (Продолжить), а в появившемся окне с лицензией кнопку **I Agree** (Согласен). Следующее окно позволяет выбрать компоненты для установки (рис. 2.2).

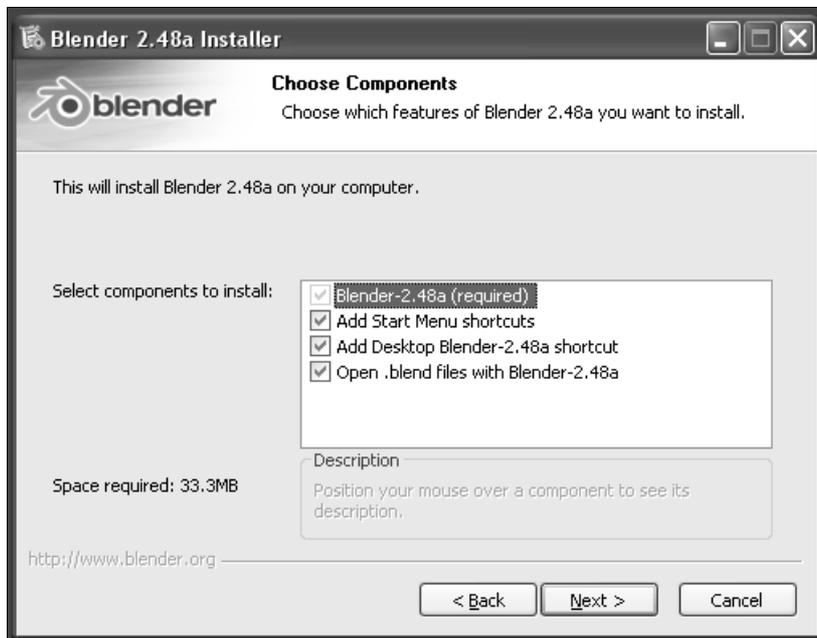


Рис. 2.2. Выбор компонентов

Предлагаемый список состоит из четырех опций:

- ◆ **Blender-2.48a**;
- ◆ **Add Start Menu shortcuts** (Добавить ярлык в меню Пуск);
- ◆ **Add Desktop Blender-2.48a shortcut** (Добавить ярлык на Рабочий стол);
- ◆ **Open .blend files with Blender-2.48a** (Открывать файлы с расширением blend в Blender-2.48a).

Настройте установщик так, как вам необходимо, и нажмите кнопку **Next**. Следующее окно позволяет выбрать место для расположения выполняемых файлов программы. Оставьте все как есть и переходите к очередному этапу (рис. 2.3).

Все скрипты и настройки программы обычно хранятся отдельно от выполняемого кода, и это окно позволяет выбрать место для хранения:

- ◆ **Use the Application Data directory** (Использовать папку Application Data). Стандартный способ Windows размещения настроек программ персонально для каждого пользователя.

Пример:

- C:\Documents and Settings*имя пользователя*\Application Data\Blender Foundation\;
- ◆ **Use the installation directory** (Использовать установочную директорию). В этом случае все файлы программы находятся в папке, указанной в предыдущем окне установщика;
- ◆ **I have defined a %HOME% variable** (Использовать заданную переменную HOME).

После выполнения указанных действий начнется установка программы.

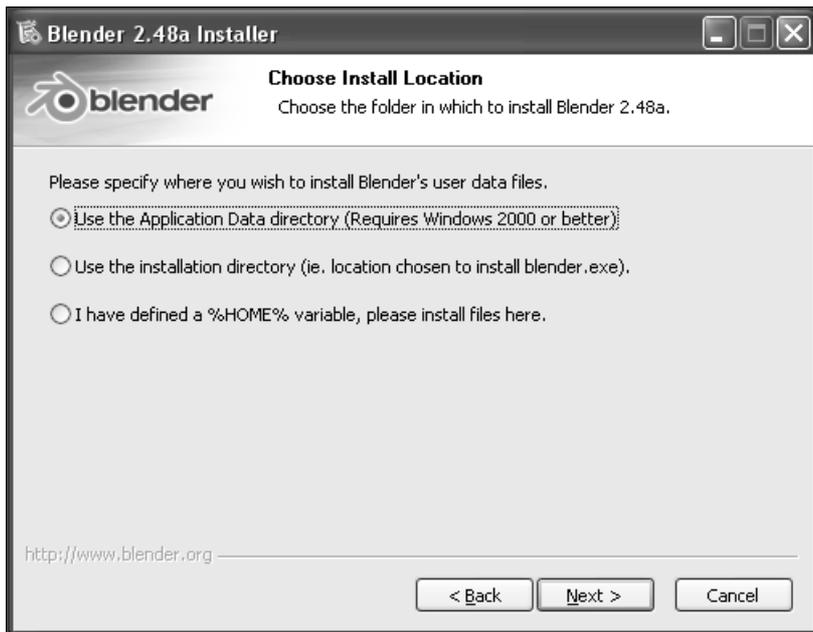


Рис. 2.3. Выбор места для файлов настроек программы

Внимание!

Blender в стандартной комплектации поставляется с уже компилированными библиотеками Питона (Python), что необходимо для нормальной работы программы, но в некоторых случаях нужно дополнительно устанавливать интерпретатор языка. Как правило, это необходимо для использования сторонних скриптов. При отсутствии Питона в системе программа на заключительном этапе выдаст соответствующее предупреждение. Установочный файл интерпретатора можно найти в папке install сопроводительного диска.

2.2. Установка программы в Linux

На сопроводительном диске в папке `install` находятся бинарные установочные файлы для 32- и 64-битной систем. Так как Linux славится большим количеством дистрибутивов с различными менеджерами окон, то проще всего будет объяснять процесс установки, опираясь на командную строку.

Сначала необходимо скопировать подходящую для вашей системы сборку в папку пользователя. Для этого откройте терминал и выполните следующие команды:

- ◆ `$ cd /cdrom/install` — переход в каталог диска с файлом;
- ◆ `$ cp blender-2.48a-linux-x86_64.tar.bz2 ~` — копирование файла в папку пользователя.

Следующий шаг — распаковка архива. Так как файлы хранятся в двойном архиве, то придется использовать две команды:

- ◆ `$ bunzip2 blender-2.48a-linux-x86_64.tar.bz2` — распаковка архива `bzip`;
- ◆ `$ tar xvf blender-2.48a-linux-x86_64.tar`.

После обработки последней команды в папке пользователя будет находиться каталог с одноименным названием. Теперь самое время решить, куда установить программу. Для этого чаще всего используется каталог `/opt`. Выполните перемещение, но уже с правами суперпользователя:

```
# mv blender-2.48a-linux-x86_64 /opt
```

В отличие от Windows папка `.blender`, содержащая скрипты и настройки, должна находиться обязательно в каталоге пользователя. Как вы знаете, по правилам Linux все файлы и каталоги, начинающиеся с точки, являются скрытыми. Поэтому для проверки результата копирования можно выполнить команду `ls` с параметром `-a`:

```
$ cp -r /opt/blender-2.48a-linux-x86_64 /.blender/ ~ | ls -a
```

Для работы Blender обязательно нужно установить Python версии 2.5. Выполните это в пакетном менеджере вашего дистрибутива. Пример для дистрибутива, основанного на Debian:

- ◆ `# aptitude search python` — ищем пакет с подходящим именем;
- ◆ `# aptitude install python2.5` — установка.

Вот и все. Для запуска Blender используйте команду:

```
$ /opt/blender-248a-linux-x86_64/blender
```

ПРИЛОЖЕНИЕ 3

Описание компакт-диска

На прилагаемом компакт-диске находятся проектные и инсталляционные файлы программы Blender, текстуры, а также мультфильм Big Buck Bunny, сделанный в Blender (см. табл.).

Все рабочие файлы примеров располагаются в папке `examples` и в подпапках с нумерацией, совпадающей с нумерацией глав книги. Каждый файл имеет в своем названии номер главы и сокращение от темы проекта. Пример:

◆ `.\examples\2\2_retopo.blend` — пример работы с инструментом `Retopo` для главы 2.

Если для проектного файла требуется дополнительный ресурсный файл, то название последнего совпадает с именем проекта. Пример:

◆ `.\examples\4\normal.tga` — карта нормали для файла `4_normal.blend`.

Для просмотра фильма в системе Windows следует предварительно установить кодек `ffdshow` (в той же папке). В системе Linux возможен просмотр с помощью программы `MPlayer`.

Таблица. Описание компакт-диска

Папки	Описание	Главы
<code>\examples</code>	Примеры и файлы рабочих проектов	1, 2, 3, 4, 5, 6, 7
<code>\install</code>	Установочные файлы Blender для Windows 98/XP/Vista и Linux (32/64)	
<code>\movie</code>	Короткометражный мультипликационный фильм, смонтированный в Blender организацией Blender Foudation, и кодек <code>ffdshow</code> для просмотра его в Windows	
<code>\textures</code>	Набор свободных текстур, распространяемых по лицензии Creative Commons	

Предметный указатель

3

3D cursor 16

A

Action editor 191, 192, 194

Alpha 101, 120

Array 63

Auto IK 181

B

Background image 71

Blinn 107

Bones 173

Bump map 129

C

Circle 24

Collision 207, 209

Colorband 120

Cone 25

Constraints 163

CookTorr 106

Cube 24

Curve Taper 93

Cylinder 25

D

Damp 212

Diffuse Color 99

Displacement map 136

Domain 220

Draw normals 131

E

Edit mode 26

Empty 117

Erase 35

Extrude 31

F

Follow path 163

Fresnel 105

Friction 215

G

Gloss 111

GLSL 103

Gravitate 215

H

Halo 227, 234

◇ flare 119

◇ lines 118

◇ rings 118

◇ star 118

◇ определение 116

I

IcoSphere 24
IOR 112
IPO 154, 158, 160, 192
IPO curve editor 154

L

Lambert 103, 107
Lattice 54, 172

M

Map input 124
Map to 126
Mass 216
Merge 34
Mesh 23
Minnaert 105
Mirror Color 99
Monkey 25
Motion blur 247
Multires 60

N

NLA editor 196
Normal map 131, 133

O

Object mode 26
Orco 125
Oren-Nayar 104, 107
OSA (oversampling) 102, 248
Outliner 19

P

Path 163
Phong 106
Plane 24
Pose mode 178
Preview 102, 109, 123, 226
Pull 211
Push 212

R

Ramps 120
Ray mirror 109
Ray transp 109
Reflection 104
Render 99, 113, 245, 247
Retopo 51
RGB 101
RMS 107

S

Sample 100
Sculpt mode 26
Shadow buffer 232
Skinning 186
Smooth 42
Specular Color 99
Speed 216
Stick 125
Subdivide 35

T

Texture paint 26, 145
Timeline 150
Toon 106, 107, 248

U

Unwrap 138
UV calculation 137, 138
UV/Image Editor 137, 142
UVsphere 24

V

Vertex groups 114, 187
Vertex paint 26

W

Weight paint 25, 189

X

X-ray 186

А

Абсолютные ключи 168
Арматура 172, 175

Б

Базовый ключ 168
Бампинг 129
Блеск 111
Булевы операции
◇ difference 58
◇ intersect 58
◇ union 58
◇ определение 58

В

Выдавливание 31
Выпечка 133, 220

Г

Гравитация 215
Группы
◇ вершин 114
◇ объектов 16

З

Звено арматуры 172

И

Иерархия 18
Инверсная кинематика 181
Индекс преломления 112

К

Камера 234
Каналы IPO 157
Каркас 23
Карта
◇ нормалей 131
◇ смещения 136
Кинематика 179

Ключевой кадр 148, 153
Конус 25
Кости 173
Кривая
◇ NURBS 70
◇ Безье 70
◇ определение 69
◇ редактирование 71
Круг 24
Куб 24

М

Маркер 153
Масса 216
Массив 63
Масштаб объекта 14
Материал
◇ понятие 97
◇ прозрачность 101
◇ цвет блика 99
◇ цвет диффузии 99
◇ цвет отражения 99
Модификатор
◇ lattice 54
◇ mirror 41
◇ simpledeform 91
◇ subsurf 44
◇ определение 40
Мэппинг 123

Н

Нормали 131

О

Объединить 34
Объектный режим 26
Ограничения 163
Относительные ключи 172

П

Перемещение объекта 13
Плоскость 24
Поверхности 79

Предварительный просмотр 102
Привязки 163
Проецирование текстуры 123
Пропорциональное редактирование 28
Процедурные текстуры 123

Р

Разбивка 35
Развертка 138
Размытие движения 247
Редактор
◇ IPO 154
◇ действий 191, 192, 194
Режим
◇ позы 178
◇ редактирования 26
Рельефные карты 129
Решетка 54
Ротация объекта 14

С

Свет
◇ area 226
◇ hemi 226
◇ lamp 225
◇ spot 226, 227, 232
◇ sun 226
◇ окружающий 241
Сгладить объект 42
Скелет 172
Скорость 216
Слои 20
Сплайн 69
Сфера 24
Сцена 21

Т

Текст 83, 89
Текстура 122

Текстурные каналы 122
Теневой буфер 232
Траектория движения 162
Трение 215
Туман 236, 240

У

Удаление 35

Ф

Физика
◇ cloth 207
◇ fields 216
◇ fluid 220
◇ soft body 211
Фоновое изображение 71

Ц

Цилиндр 25

Ч

Частицы 199, 204, 208

Ш

Шейдер
◇ бликовый 106
◇ диффузный 103
◇ определение 103
◇ рамповые 120
Шрифты 84

Э

Эмиттер 199