



**КРАТКО ДОСТУПНО ПРОСТО**

**Людмила Зимина**

# 3D STUDIO MAX

**ВИРТУОЗНАЯ РАБОТА**

**3.0**

Обзор версий  
1.0, 2.0, 2.5 и 3.0

3D моделирование

Тонкости, хитрости  
и приемы работы

Вопросы и ответы

Расширения

**КОМПЬЮТЕРНАЯ  
ГРАФИКА  
без секретов**

и масса других  
полезностей...

**Людмила Зимина**

**Кратко, доступно, просто**

**Виртуозная работа в  
3D Studio MAX  
3.0**

ББК 32.973  
УДК 681.3

Главный редактор  
Александр Осипенко  
Корректор  
Ирина Царик  
Дизайн и верстка  
Борис Леонтьев, Ирина Царик

**Людмила Зимина**

380 Виртуозная работа в 3D Studio Max 3.0. /Кратко, доступно, просто/. — М.: Познавательная книга плюс, 1999. — 192 с.

**ISBN 5-8321-0033-6**

3D Studio Max 3.0 — самый известный и навороченный из всех 3D редакторов. Позволяет делать как статичные сцены так и трехмерную анимацию. Наследник знаменитых 3D Studio и Autocad. Кроме собственных средств моделирования, 3D Studio Max 3.0 имеет интерфейс для подключения модулей от сторонних фирм. И эти модули постоянно появляются на рынке. Так, что возможности 3D Studio Max легко расширяются.

Большинство новых функций 3D Studio Max 3.0 предлагают столь значительные возможности, что достижение аналогичных результатов традиционными методами в большинстве случаев невозможно, а в лучшем случае потребует несоизмеримо больших затрат сил и времени.

Copyright © Людмила Зимина 1999. Составление, перевод, комментарии.

Copyright © Издательство «Познавательная книга плюс», 1999.

Copyright © Студия дизайна и рекламы «МиК», 1999. Оформление этой книги.

Автор этого проекта благодарит сервера <http://prism.xproject.ru/> и <http://wagood.hypermart.net/notes1.htm> за оказанную информационную поддержку в процессе подготовки рукописи к изданию.

Авторские права на настоящую книгу, название серии «Кратко, доступно, просто™» и оформление самой серии принадлежат Борису Леонтьеву и издательству «Познавательная книга плюс». Полное или частичное копирование книги, а также публикация других рукописей в рамках этой серии без разрешения правообладателя является нарушением авторского права и может иметь юридические последствия в соответствии с действующим законодательством.

Идея создания серии «Кратко, доступно, просто» принадлежит А.И.Осипенко.

В информацию, включенную в данное издание, могут быть внесены изменения без специального уведомления. Программное обеспечение, описанное в этой книге, поставляется исключительно по лицензионному договору и может использоваться или копироваться только в соответствии с условиями этого договора. Копирование этого программного обеспечения на какой-либо носитель информации, если на это нет специального разрешения по лицензионному договору, является нарушением Закона Российской Федерации «О правовой охране программ для ЭВМ и баз данных» и норм международного права.

**Налоговая льгота «Общероссийский классификатор ОК 005-93-ТОМ2  
953000 — Книги и брошюры.»**

Издательство ООО «Познавательная книга плюс»

109316 г. Москва, Волгоградский пр-т, 14.

Лицензия ЛР № 066245 от 25.12.1998 г. Подписано в печать 31.05.99. Формат 60X84/16.

Гигиеническое заключение на серию «Кратко, доступно, просто»  
№ 11.99.6.927.1.99 от 29.01.1999 г.

Заказ 3774—99

Отпечатано с готовых диапозитивов в 12 ЦТ МО  
Тираж 3500. Объем 12 печ. л.

# **Если посмотреть на компьютерную графику со стороны**

Если посмотреть на компьютерную графику со стороны то:

- во-первых — это хорошее владение программами трехмерного моделирования. Владение настолько хорошее, что при создании проекта дизайнер не думает о том, как создать, а думает о том, что создать;
- во-вторых — это знание принципов построения и законов композиции, позволяющее добиваться решения конкретных задач по созданию интерьера, отвечающего не только эстетическим требованиям, но и удобствам проживания в нем;
- в третьих — знание одного из наиболее выразительных средств, применяемых в работе дизайнера-декоратора — цвета.

Учеными доказано, что оформление интерьера заметно влияет на психологические аспекты деятельности человека: активность, работоспособность, утомляемость, сосредоточенность, спокойствие, раздражительность и т.д.

Профессиональные 3D-пакеты сложны в освоении, в то время как простой софт не позволяет сделать многого, что хотелось бы реализовать, т.е. пакетов множество,

остается найти оптимум, удовлетворяющий следующим критериям:

- умеренные запросы в области железа, распространенность платформы, графический интерфейс, пригодный для интуитивного освоения;
- должное количество удобных и понятных инструментов для моделирования, приемлемое качество рендеринга;
- наличие online help;
- и последнее, но немаловажное: доступность персональной помощи в случае возникающих проблем.

Вполне возможно, что среди критиков этой книги найдутся поклонники Alias/Wavefront, Strata или Softimage. Кто-то, к примеру, прекрасно владеет Povray или Bruce, и действительно, возможности этих пакетов подчас позволяют решать маленькие проблемы с соразмерно небольшими затратами. Однако в качестве инструмента для дальнейшего изучения избран 3D Studio MAX.

Одним из главных преимуществ нашего избранника является возможность объектно-ориентированного моделирования, т.е. исходная геометрическая форма, равно как и последовательно применяемые к ней модификаторы, являются объектами, параметры которых могут в любой момент быть изменены. Это позволяет сделать процесс моделирования предельно структурированным, а заодно и избавляет от страха перед возможными ошибками.

Пристрастие к тому или иному типу графической оболочки – безусловно дело индивидуальное, но качество

## Если посмотреть на компьютерную графику

интерфейса 3D Studio MAX может быть относительно независимо оценено как удовлетворительное. Присутствует элемент эксплоративности, хотя «ткнуть и посмотреть, что получится» не всегда приводит к видимым результатам. Авторы этого пакета нашли разумный баланс между пиктографической и текстовой формой представления активных и информационных элементов.

Благодаря уже упомянутой объектной ориентированности, возможно практически безграничное модулярное расширение возможностей пакета, в результате чего, каждый способен сам составить себе набор любимых инструментов или попробовать что-то новое, просто подключив дополнительный plugin (надо заметить, что базовой подборки, приходящей со стандартной инсталляцией пакета, на первых порах более чем достаточно).

Работать с 3D Studio MAX вполне можно уже на P100 с 32MB RAM под Windows 95/98 и Windows NT4.0, в последнем случае можно рассчитывать на гораздо более стабильное функционирование. Особых, по разумным меркам, претензий к графическим возможностям не предъявляется, использование OpenGL-способной карты рекомендуется. Пакет – в зависимости от инсталляции – занимает от 60 до 200 MB на жестком диске, стоит учитывать возможность дальнейшего «разбухания», связанного с добавлением подключаемых модулей. Особых проблем при инсталляции MAXa до настоящего момента не встречалось, стоит упомянуть лишь о необходимости установки **Sentinel Hardlock Drivers** для работы программы.

# **3D Studio MAX**

Впервые программа под таким названием появилась в России в 1996 году, однако мало кто тогда ее воспринял в качестве чего-то принципиально нового. Представляем себе вытянутые лица будущих потомков, которые прочтут эту фразу. А оно так и было — долгие годы практически единственной полноценной программой для 3D на PC-платформе оставалась 3D Studio для DOS. Да и платформой для графики PC тогда обозвать было сложно — 386DX+8RAM.

Так вот и развивалась эта самая Студия вплоть до версии 3.0, может быть и дальше развивалась так же, но производитель Autodesk вовремя смекнул, что это не жизнь и, как результат, появилось в фирме еще одно подразделение — Kinetix. Сама фирма продолжила изыскания в разработке досовской версии, а ее подразделение было призвано создать на основе многолетних наработок нечто уникальное, мощное, рассчитанное на новое поколение компьютеров и художников.

Так же к разработке 3DStudio MAX приступила Yost Group в качестве независимой группы программистов еще до выхода 4-ой досовской версии в 1993 году. Общее время работы над проектом — 3 года и три месяца. Необходимо было не только разработать новый интерактивный Win32-интерфейс, но и максимально расширить возможности пакета, лишить его ограничений, которыми так страдали все без исключения графические программы того времени. Так они и развивались

параллельно — 3D Studio и 3D Studio MAX. После выхода 4-ой версии под DOS, Autodesk объявила о моратории на дальнейшее развитие 3D Studio и все силы бросила на это самое новое долгожданное нечто. Вот и воспринял ваяющий люд ее не как программу версии 1.0, а, скорее, как следующую версию после 4.0, критиковать начали — глючит, того нет, то есть, но не работает. Так оно и было — темпы развития рынка росли все стремительней, начали приходить на PC платформу «беженцы» с умирающей Амиги, среди которых были и уже известные Real3D, LigthWave 3D. Нельзя было больше ждать и на рынок выпустили еще не облупившегося монстра — 3D Studio MAX 1.0.

Затем посыпались версии 1.1, 1.2, программа становилась все стабильней и стабильней, а после выхода версии 2.0 производитель, наконец, признался, что это и есть то, что задумывалось. Досада какая — это произошло аж через полтора года после выхода первой версии! Но это произошло! Следующая версия 2.5, и среди даже самых грозных оппонентов не находится смельчаков пренебрегать ее мощностью. А открытая архитектура пакета породила на свет такое количество подключаемых модулей, что теперь уже и сложно придумать «непозубную» задачу для это счетной машинки.

## Интерфейс

Интерфейс программы во многом унаследовал решения старой DOS-овской версии.

Что можно было сказать практически сразу, еще с выходом самого первого 3D Studio MAXа, так это то, что разработчики нового интерфейса не просто справились



на 5 — они действительно создали новый интерфейс. Для 3D Studio MAX это не только пользовательская часть, но и внутренний API.

Главное, все операции выполняются в одном модуле, точнее в одном интерфейсе — и построение кривых, и их вытягивание, работа с геометрией, материалами, ключами, эффектами. Главное преимущество такого подхода — возможность анимировать сам процесс моделирования и редактирования, возможность применения динамических модификаторов на разных уровнях геометрии. Такое нововведение, как «Стэк модификаторов», отбросило конкурентов далеко назад.

На данный момент пакет работает со всеми тремя кнопками мыши и 102-мя кнопками клавиатуры, подавляющая часть функций доступна на 2-3 уровне меню, что тоже радует. Да и стэк упростил жизнь — всегда можно вернуться и изменить любую трансформацию, проведенную ранее. К слову, такое доступно еще только в Maya.

Настройка окон позволяет отображать в одном окошке ключевые кадры, а в другом — геометрию или отдельно **shape**, то есть редактировать их на одном уровне, результат любого действия отображается интерактивно. Жаль, что не пошли дальше, как в SOFTIMAGE 3D или Maya, — нет окна иерархии сцены и материалов. Правда материалам это не мешает — у них свое отдельное окошко, отображаемое поверх, тоже интерактивное — изменения цвета или любого другого параметра в редакторе материалов приводит к незамедлительному отображению результата в сцене.

Все менюшки организованы в виде сворачиваемых блоков — всегда легко убрать лишнее или просто подвинуть, «взяв» мышкой за свободное место. Для новичка, который знаком хотя бы с основными понятиями 3D, разобраться — дело двух-трех дней.

### **Многозадачность**

3D Studio Max эффективно использует все преимущества системы Windows NT: многозадачность, мультисвязность, симметричную многопроцессорность. Таким образом, дополнительному процессору может быть назначена работа со второй запущенной копией программы на том же PC, или он может использоваться совместно с первым в работе с единственной работающей копией. Использование второго процессора в целом повышает производительность системы более чем на 60%.

Несмотря на абсолютно новый интерфейс программы, время, необходимое на его освоение, потребует незначительное. Если вы знакомы с текстовым процессором типа Microsoft Word или работали с ранними версиями 3D Studio, то освоить новый интуитивный интерфейс программы для вас не составит труда.

### **Моделирование**

Это первый камень преткновения. Прапредок 3D Studio MAX всю жизнь претендовал на некоторую универсальность, но ее там никогда не было, а вот когда народ с ней действительно столкнулся — у многих опустили руки. В 3D Studio MAX можно моделировать что

угодно и как угодно, поддерживаются и реализованы все возможные технологии, однако мешать эти изыски между собой крайне не рекомендуется. Если работаешь с **patch**, то надо забывать про каркасный моделинг или NURBS. Этого как раз никто и не делал — мешали в кучу, получали неудовлетворительный результат, плевались, уходили. И зря!

Все объекты хранят свою «редактируемую историю». Это означает, что в любой момент может быть осуществлен доступ к любой команде по созданию и редактированию объекта, изменены ее параметры и эти изменения тут же отразятся на текущем состоянии объекта. Если в цилиндре не хватает сегментов, просто вернитесь к команде создания цилиндра и нажатием мыши измените число сегментов. Если заказчик в последнюю минуту потребовал посильнее изогнуть объект, просто обратитесь к стэку и измените угол на больший. Эта новая возможность, ранее доступная лишь пользователям high-end рабочих станций и получившая название «недеструктивное редактирование», вызывала восторженные реакции зрителей на демонстрациях пакета.

Помимо этого, у пользователей появилась возможность неограниченного использования команд **Undo/Redo**. Модели, создаваемые в 3D Studio Max, могут быть полигональными, Безье- и Би-сплайновыми. Теперь они не ограничены в размерах. Появился механизм объектной привязки: можно точно указать на вершины и ребра объектов в 3D, 2D и 2.5D. Объекты и подобъекты легко трансформируются относительно любой из множества возможных систем координат. Например, центр операции **Bend** может быть легко анимирован относительно контрольных точек или группы вершин,

которые собственно и поворачиваются. Можно создавать копии объектов, которые унаследуют топологию и анимацию исходного объекта. Можно помимо этого задать локальные, присущие только копии, возможности при сохранении общей зависимости от оригинала.

Модификаторы объектов позволяют деформировать их в собственных координатных системах. В пакет войдут модификаторы, выполняющие деформации, сходные с имеющимися на диске N3 из набора IPAS-процессов от Yost Group.

Очень интересны **Space Warps** — средства пространственной деформации.

Будучи связанными с конкретной геометрией сцены, они позволяют получить различные спецэффекты (волновые деформации, эффект «черных дыр», взрывов и т.д.) прямо в среде 3D Studio Max, без выполнения операции морфинга. Один объект может одновременно задействоваться в нескольких эффектах, тогда как в АХР-процессах это было невозможно.

Список доступных технологий моделирования — стандартный пакет: набор из 10 стандартных и 10 расширенных примитивов, выдавливание, кручение, выдавливание с фаской (**bevel**), по пути (**path extrude**), лофтинг, полный набор каркасного редактирования, патчи, NURBS с триммингом, metaNURBS, 6 вариантов «комбинирования» объектов, в том числе **boolean** с анимацией, — доступно через плагины, **meta clay** и интересная разновидность под названием **metaRey** (мышечный скелет с поддержкой анимации), очень специфичные объекты вроде воды или материи (говоришь сетке — «ты тряпка» — и она в ужасе падает на спинку

стула, образуемая рисунок складок), разновидности сплайновых поверхностей (daVinci3D).

Некоторые из технологий смело можно назвать уникальными. Практически не существует формы, которую было бы нельзя создать, но разобраться во всей этой куче без посторонней помощи сложновато — поэтому и любителей моделировать в 3D Studio MAX не так уж много. Просто перед тем, как что-то делать, нужно четко представлять, по какому пути идти и что в результате получится. Те же **MetaRey** очень хороши для органики, но на обычной машине среднего пользователя им не хватит ресурсов развернуться во всей красе в анимации, зато патчами можно сделать то же самое, в два-три раза медленней, но очень легко для последующего оживления. Большинство предпочитает путь быстрый и оказывается в тупике. Однако те, кто освоил моделинг в 3D Studio MAX, редко уже «прыгают» в другой пакет.

## **Материалы и визуализация**

Редактор материалов имеет плавающий характер (не вызывается как отдельный модуль). Удобные средства выбора цвета на панели. Каждая текстурная карта может иметь свою координатную привязку. В одном материале может быть задействовано несколько привязок.

Архитектура построения пакета делает его независимым от применяемого модуля просчета. Будут поддерживаться множество программ просчета. Разрабатывая 3D Studio Max, компания Autodesk не изменила своим принципам: программа имеет открытую архитектуру и оставляет разработчикам из третьих фирм прекрасную возможность для расширения

функциональных возможностей пакета. Эта особенность была восторженно воспринята теми фирмами, которые на сегодняшний день имеют уже немало наработок для 3D Studio. Ранее разработанные IPAS-процессы в чистом виде в Max'e использоваться не могут. Для этого потребуется некая предварительная работа по переносу программ. Дополнительно разрабатываемые программы получили название **Core Components**. Они полностью органично интегрируются в основной пакет и выглядят как его составная часть.

Материалы в 3D Studio MAX отличаются многофункциональностью. Поддерживаются процедурные и растровые текстуры практически для всех параметров, очень богатые возможности сочетания многослойных текстур — от альфа-канала до сглаживания различных материалов. Стоит и вспомнить, что поддерживается три текстурные координаты на предмет — мировые XYZ и пользовательские YUW1,2.

Это дает возможность накладывать текстуры, по крайней мере, двумя различными способами на один предмет. На первый взгляд, этого недостаточно, в LightWave 3D, к примеру, каждый материал может иметь свой уникальный способ наложения. Однако не будем забывать, что в 3D Studio MAX версии 2.5 основной упор сделан на NURBS моделирование, один объект может состоять из любого количества поверхностей, каждая из которых имеет свою уникальную сетку наложения текстуры и, если надо, материал. С этой точки зрения, двух вариантов координат хватает «за глаза».

Появились такие параметры для материалов, как «суперсэмплирование» (очень полезно при мелком

**bump-mapping**), **displacement mapping** (фича для NURBS — вроде бампа, только полностью трехмерные деформации, работающие гораздо быстрее, чем модификатор геометрии **displace**). Есть и «материалы-спецэффекты», вроде **Shadow Matte**, который очень сильно облегчает компоинг 3D и видео, или **Double Sided** — разным сторонам полигонов назначаются различные материалы.

Отдельного упоминания заслуживает тип материала **RayTrace**. Как видно из названия, этот материал считается по иному алгоритму, но это далеко не все — в нем реализованы очень симпатичные спецэффекты, вроде «театра теней» или игры света внутри полупрозрачных материалов.

Сам рэйтрэйсинг 3D Studio MAX отличается невообразимым количеством настроек способов его визуализации. Естественно, что разобраться в нем иногда бывает сложно. Кстати говоря, на этом типе материала виден очень интересный подход — можно создать любой внешний материал или текстуру. Именно этим обусловлено такое обилие модулей визуализации от третьих фирм.

Что касается модуля визуализации, то тут 3D Studio MAX, к сожалению, не блещет занебесным качеством. Есть и рэйтрэйсинг, и всевозможные атмосферные эффекты, огромное количество вариантов настройки и по скорости и по качеству, но, если надо действительно достичь потрясающего результата — лучше воспользоваться модулем третьих производителей, вроде RayMAX. Однако это не страшно — подмену можно будет обнаружить только только по изменившимся настройкам в окошке **RENDER** — все плагины работают внутри

основного интерфейса. Зато вам станут доступны даже такие «вкусности», как **rusted metal** — от него до настоящего компакта один шаг! Подобных модулей существует минимум пять, и есть, из чего выбрать. И выбирают — так для фильма «Армагеддон» после тестирования всего, чего только можно, остановились на модуле атмосферных эффектов **Afterburn** под 3D Studio MAX — в «аутсайдерах» оказались такие монстры как Alias Power Animator и SOFTIMAGE 3D.

### Анимация

Все характеристики объекта анимируются, вплоть до количества граней на его поверхности.

- Анимируются характеристики материала, подобъекты, их параметры, а также сам процесс моделирования.
- Удобные средства редактирования анимации.
- Встроенный метроном, импорт звуковых файлов и захват звука в реальном времени.
- Управление анимацией по кадрам или по временному коду.
- Мощные средства для организации циклов.
- Инверсная кинематика предполагает возможность зафиксировать объекты таким образом, чтобы они не двигались до тех пор, пока не будет определена вся кинематическая цепочка.

Анимируется все... и еще чуть-чуть. Помимо ставшего уже стандартом набора, есть отличная от всего



система костей со встроенной инверсной кинематикой (встроенной в кости — есть и просто ИК в самом пакете), возможность назначения любому анимируемому параметру одного из 13 контроллеров движения, в том числе **expression** и скрипты, динамика.

И теперь о главном — 3D Studio MAX не был бы 3D Studio MAX'ом без модуля анимации персонажей **Character Studio**. Именно благодаря этому модулю, пакет занял очень прочные позиции на рынке производителей игрушек, кое-где потеснив силиконовых монстров. Это так просто — сказал ему: «иди», и он пошел. Именно так, но еще и с возможностью **Motion Capture**. Кроме того, и динамика есть «довесная», когда резина гнется от удара, а металлический чайник может упасть, помяться и так и лежать помятым, как ни в чем не бывало (**HyperMatter**). Добавьте к этому частицы, многоканальный морфинг, анимацию материалов, процедурных текстур, постэффектов, и вы получите анимацию всего на свете.

## **Резюме**

«Модульность» 3D Studio MAX породила на свет уйму плагинов, только свободно распространяемых на данный момент насчитывается более сотни. Реализуются функции, которых изначально и не было в пакете — вроде волос или шейдеров для частиц. Вкупе со сравнительно невысокой ценой, средними ресурсными требованиями и гибкостью, этот пакет является наиболее предпочтительным выбором для видеографики, анимации персонажей и даже кинопроизводства. Но самое большое признание пакет получил среди разработчиков игр. Его ближайшие конкуренты «перетягивают» в цене на сам

софт в два раза минимум, а про железо и вспоминать не стоит.

Одно время значительную конкуренцию составлял LightWave 3D, однако на данный момент он потерял свои позиции — ядро пакета не менялось уже несколько лет и многие функции реализовать в нем корректно принципиально невозможно. А значит, в ближайшем будущем основную альтернативу 3D Studio MAX составят «силиконовые монстры» — у производителей будет хороший стимул к самосовершенствованию.

# 3D Studio MAX 2.0

При разработке 3D Studio MAX R2 учтены предложения пользователей пакета, накопленные в Autodesk Multimedia и Yost Group на протяжении последних семи лет. 3D Studio MAX является прекрасным воплощением того, что может быть создано в области трехмерной графики на основе самых распространенных технических средств. 3D Studio MAX R2 вышел в 1997 году и стал популярнейшей программой компании KINETIX. Пакет позволяет создавать фильмы, игры, видеопroduкцию и обеспечивает превосходное качество изображений, легкость создания анимаций и спецэффекты высокого класса.

Пакет 3D Studio MAX R2 включает более 1000 новых возможностей по сравнению с предыдущей версией MAX R1.2. При этом, цена MAX R2 невероятно низка при таких достоинствах программы. Моделирование 3D Studio MAX R2 позволяет создавать объекты различными методами. Некоторые из этих методов существовали в предыдущих версиях программы и были доработаны в новой версии (стандартные примитивы, лофт-объекты, объекты, полученные вращением или выдавливанием плоских форм, пэчковые поверхности).

Другие методы являются новыми: появились расширенные примитивы, NURBS-модели, существенно увеличен класс сложных (**Compound**) объектов.

Рассмотрим некоторые возможности моделирования и анимации NURBS-объектов, которые широко

используются для моделирования сложных криволинейных поверхностей. В одном известном голливудском фильме гигантская анаконда заглатывает человека. С появлением NURBS-моделирования, 3D Studio MAX позволяет легко получить такой эффект. Для этого можно создать NURBS-модель змеи, которую затем анимировать смещениями ее кожи, пасти и мускулов во время поглощения змеей жертвы. Анимация NURBS очень проста — достаточно включить кнопку **ANIMATE** и настроить контрольные вершины или их веса у NURBS-поверхности — анимация создается немедленно, причем анимированная поверхность получается сглаженной.

Для анимации NURBS кривых и поверхностей можно также использовать все обычные инструменты трансформации. Кроме того, пользователи могут анимировать NURBS (а также другие типы объектов) с использованием нового инструмента — языка **MAX Script**, или создавать анимацию в реальном времени, используя систему **Motion Capture**. В 3D Studio MAX Release 2 добавлены новые возможности редактирования на панели **Modify**. Появился новый класс модификаторов — **World Space**, работающих в мировом пространстве и позволяющих воздействовать на объекты в сцене. Расширился состав «обычных» модификаторов, работающих в объектном пространстве (**Object Space**). Это, прежде всего, модификаторы деформации свободных форм — **FFD** модификаторы, позволяющие изменять геометрические формы объекта путем трансформации контрольных точек модификатора. Контрольные точки представляют собой узлы пространственной решетки, окружающей объект.

Значительно расширены возможности стека модификаторов.

- Возможна конвертация Instance копий модификаторов в Reference.
- Можно копировать, перемещать и вставлять модификаторы внутри объекта.
- Можно копировать модификаторы из одного объекта в другой.
- **Undo/Redo** опции работают для всех операций на стеке.
- Можно конвертировать объекты в **Editable Mesh**, **Splines**, **NURBS** поверхности или кривые и редактировать их на новом уровне.
- Объекты типа **Editable Mesh** можно анимировать непосредственно на уровнях вершин, граней, ребер.

## **Специальные анимационные эффекты**

**Particle Systems** — это объекты, которые генерируют нередатируемые подобъекты, называемые частицами. **Particle Systems** используются для специальных эффектов. Многие сцены в кино дополнены дымами, взрывами, летающими обломками или искрами. **Particle Systems** могут быть использованы как средство для моделирования дождя, снега, пыли, пузырьков воздуха, а также для имитации объектов, поведение которых изменяется во времени: текущей жидкости, водопадов, языков пламени, потоков крови, ран на лице жертвы и т.д.

Многие компоненты **Particle Systems** работают с утилитой **Dynamics**. Эмиттером (источником) частиц можно назначить любой объект в сцене, а сами частицы могут быть любым объектом сцены.

### **Новые Space Warp деформаторы**

В версию 2 пакета 3D Studio MAX включены новые деформаторы объектов. К ним относятся **FFD Cyl**, **FFD Box**, **FFD Select** и **Conform Space Warps**, позволяющие деформировать и анимировать объекты в мировом пространстве путем трансформаций любого числа точек вспомогательной решетки.

Появилась целая группа **Space Warps**, работающая с частицами и динамикой для моделирования:

- гравитации, ветра, отражения частиц;
- силового поля, изменяющего геометрию (в том числе системы частиц), импульсного характера движения системы частиц или движения их по выбранному сплайновому пути;
- выталкивающей силы или закручивающего усилия для создания таких эффектов, как торнадо, штормы, стаи мотыльков, кружащихся вокруг горящей лампы и т.п.

### **Новое в анимации**

Анимация в 3D Studio MAX R2 основывается на анимационных контроллерах. Поэтому MAX имеет значительно больше возможностей для анимации объектов, чем другие программы. В 3D Studio MAX практически все можно анимировать на объектном или

подобъектном уровне, т.е. имеется уникальная возможность обращаться со всем в сцене, как с объектом. Это означает, что источники света и материалы — объекты, так же как и единичные вершины на NURBS кривой или опорная точка у вспомогательного Dummy объекта. Все эти объекты могут быть анимированы в MAX одним движением мышки.

Для задания более точных значений анимации можно вводить значения с клавиатуры в определенные числовые поля, такие, как в **Transform type-in** окне или в параметрах создания. Можно анимировать объекты, меняя параметры присвоенных им модификаторов как на объектном, так и на подобъектном уровне.

Другой путь анимации объектов в MAX R2 — использование новой утилиты **Motion Capture**, которая позволяет связать объект с внешним устройством, таким, например, как музыкальная клавиатура. В этом случае при нажатии на две разные ноты объект будет анимирован.

Инверсная кинематика в интерактивном режиме может быть просчитана для иерархии объектов с целью формирования движения в реальном времени. MAX может анимировать объекты одной и той же иерархии как в прямой, так и в инверсной кинематике. Можно анимировать передачу родственных связей ребенка от одного родителя к другому.

### **Анимационные инструменты**

В 3D Studio MAX появилась утилита **Dynamics**, которая позволяет создавать анимацию с учетом физики реального мира. Утилита может эффективно

использоваться при разработке игр, в которых быстрые и активные действия не могут быть анимированы вручную. С ее помощью можно моделировать движение реальных объектов при падении, скольжении, столкновении, отскоке и т.п., а также звездные системы, механизмы или ситуации на бильярдном столе. Результат анимации зависит от установки параметров, которые сами могут изменяться во времени (масса объектов, участвующих в динамической ситуации).

Например, при стандартной технологии анимации для задания движения мячу, прыгающему вниз по ступенькам, следует определить положение всех ключей и их свойства. Если же использовать утилиту **Dynamics**, можно задать физические свойства мяча и ступеней, добавить силу гравитации и дать задание системе на просчет этого взаимодействия.

Обычно утилиту **Dynamic** используют для первоначального просчета движения, а затем вручную редактируют полученные ключи анимации для получения лучшего результата.

## Продвинутый Rendering и камерные эффекты

Один из самых распространенных методов построения реалистичных изображений — метод слежения луча (**Raytracing**). С помощью этого метода можно создавать изумительные изображения трехмерного мира, той самой «компьютерной реальности», которая так широко используется в новом кино, рекламе и играх. Сам термин «слежение луча» отражает его сущность —



прослеживание луча света, отражаемого объектом в сцене: отражаясь, луч приобретает цвет и характеристики объекта, его отразившего, и позволяет построить на основе этого изображение всей сцены.

Селективный **Raytracing** работает как интегрированная часть построчного просчета (**scanline**) 3DS MAX 2. Выборочная трассировка применима для просчета отраженных или преломленных лучей, а также для отдельных поверхностей.

Новый **Raytracing** разработан с учетом адаптивного антиэлайзинга и поддерживает такие новые возможности, как затухание с расстоянием (дефокусировка) и внутриобъектные отражения. Просчеты могут быть улучшены при использовании преломления с рекурсией, атмосферных эффектов и адаптивного антиэлайзинга, размывания, назначения полупрозрачности и флюорисцентности.

Селективный **Raytracing** совместим с объемным освещением и все его параметры, которые связаны с изменением освещения поверхности в сцене, могут быть анимированы.

В **Video Post** добавлены оптические камерные эффекты. Они позволяют смоделировать явления, появляющиеся при попадании прямого света в камеру, свечения объектов, сверкания, расфокусировки камеры.

Линзовые эффекты могут существенно повысить реализм сцены. Все линзовые эффекты можно анимировать и присваивать как фильтры в **Video Post**. Параметры редактируются как события в **Video Post** и в **Track View** окне. Можно добавить множество слоев анимированных линзовых эффектов любой сцене. Все

эффекты имеют полноцветные окна предварительного просмотра, дающие возможность увидеть все изменения изображения события в **Video Post** без необходимости просчета всей сцены и уникальную возможность использования настраиваемого градиента цветов, позволяющего создавать свои собственные эффекты.

### Новое в материалах

В Редакторе материалов в 3D Studio MAX R2 появились большие изменения:

- включен режим **Drag and Drop** материалов, карт и сцен, ускоряющий создание сцен и назначение материалов объектам;
- улучшена работа с новым **Material/Map Browser**;
- расширены до 24 слотов количество образцов для редактирования и просмотра материалов;
- добавлены опции **Custom Objects, Lighting, Background** для образцов материалов;
- включены возможности поворота образца в слоте для просмотра материала;
- добавлено окно **Dynamic Propertie** для определения динамических свойств объекта с назначенным материалом;
- появился новый метод просчета материалов — **Blinn**, дающий более сглаженный результат по сравнению с **Phong**;
- возможна настройки цвета тени при использовании **Matte/Shadow** материалов;

- добавлены новые карты и опции настройки карт такие, как **Thin Wall Refraction**, создающие преломления тонкостенных поверхностях.
- **Refraction Dimming** — управление отражениями в тени. Используя новые каналы назначения карт, можно создавать специальные эффекты — воду, дым, планеты, волны и т.д.

## **Освещение и камеры**

Теперь все типы источников света обладают одинаковыми возможностями — все создают тени, могут затухать, проецировать изображение (включая **Omni**). Возможна глобальная подкраска и притемнение для всего освещения в сцене; появились новые параметры для настройки затухания света (**Custom**, **Linear** или **inverse-squared attenuation**, **Near** и **Far** границы затухания). Для получения реалистичного солнечного освещения и тени можно использовать систему **Sun** с точным размещением и анимацией солнца в соответствии с реальным географическим положением и временем года и суток.

Добавлен новый тип источника света — **Target Directional**, создающий точное параллельное освещение и тени. Появилась утилита **Camera Match**, настраивающая камеру для вписания сцены в назначенный фон — **Background**. В результате появилась возможность перемещать выбранную камеру так, чтобы ее вид соответствовал перемещению камеры в реальной сцене кино или видео. Эта возможность используется для создания специального эффекта, применяемого в кино, когда анимированные на компьютере модели при перемещении камеры по сцене оказываются на объектах,

которые существуют только на фоне. Камере можно назначать стандартные фокусные расстояния в сочетании с **Camera Aperture Width**.

В строке инструментов добавлены кнопки, позволяющие выравнивать положение камеры к нормали выбранной грани, а источники света — к положению блика.

# 3D Studio MAX 2.5

Со 2 июня 1998 года началась поставка в Европе последнего варианта 3D Studio MAX Release 2.5, в которой получили развитие возможности предыдущей версии, главные из которых перечислены ниже.

## Геометрия

- **Boolean** объекты заменены на **Boolean 2**, которые используют новый алгоритм построения объектов. По этому алгоритму получаются более приемлемые результаты при менее сложной геометрии. Появилась возможность выполнения булевских операций над удаленными в пространстве объектами (не обязательно частичное пересечение исходных операндов).
- Расширены возможности NURBS моделирования:
  1. Добавлены новые типы подобъектов **Curves**, позволяющих обрезать поверхности, используя кривую как границу обрезания, а также создавать **UV Loft** и **Sweep** поверхности. **Curves** (кривые) образуются как **Surface-Surface Intersection Curve**, **Normal Projected Curve**, **Vector Projected Curve**, **CV Curve on Surface**, **Point Curve on Surface**, **Surface Offset Curve**.
  2. Добавлен новый тип подобъектов **Points: Surface Curve Intersection Point**.

### 3. Добавлены новые типы подобъектов **Surface**:

- UV Loft Surface
- 1-Rail Sweep Surface
- 2-Rail Surface
- Multisided Blend Surface
- Multicurve Trim Surface

4. Появилась опция замены родителя зависимых подобъектов при редактировании зависимых подобъектов.

5. Добавлены новые опции **Rebuild** и **Reparameterize** для **CV curves** поверхностей, которые позволяют повысить точность и улучшить вид независимых подобъектов.

- С объектами типа **Editable Meshs** можно выполнять операции **Cut** и **Slice**. Кроме того, можно использовать режим **Drag & Drop** для назначения материалов выбранным группам граней.
- Включен новый модификатор **Slice**. Этот модификатор создает разрезающую плоскость для **Mesh** объекта, добавляя новые вершины и ребра на основе гизмо-модификатора (**Slice plane gizmo**). Вершины могут вставляться двумя способами: либо без изменения кривизны исходного объекта (**Refine**), либо путем среза части **Mesh** объекта выше или ниже плоскости **Slice**. Разрезающая плоскость может быть анимирована — перемещена или повернута во времени.

## Редактор материалов

В редакторе материалов режим **Drag & Drop** можно использовать при назначении материала группам граней для объектов типа **Editable Mesh** (раньше можно было это делать только через Edit Mesh модификатор). При этом автоматически создается **Multi-SubObject** материал.

## Утилиты

В состав утилит включена утилита **Camera Track**. Эта утилита анимирует перемещение камеры в сцене таким образом, чтобы показываемое ею изображение фона соответствовало изображению при движении реальной камеры, которая была использована при съемке фильма, назначенного в качестве фона в сцене. В результате изображение в сцене будет «вписываться» в фон и соответствовать ему.

Трекер создает **Gizmo**, который связывает камерные точки, созданные в сцене, с выбранными элементами в перемещающихся кадрах фона.

Кроме перечисленного, 3D Studio MAX Release 2.5 имеет адаптированный к новым возможностям язык **MaxScript** для создания анимаций и сценариев. Добавлены новые форматы файлов для хранения карт, фонов, изображений для проецирования источником света: Kodak Cineon, SGI (RGB), Quick Time.

# 3D Studio MAX 3.0

## Track Bar

Непосредственно снизу от **Time slider** появилось новое окно **Track Bar**, в котором легко можно управлять ключами выбранных объектов. Ключи отображаются так же как и в **Track View**. Здесь можно ключи двигать, копировать (с помощью **Shift**), удалять, менять их параметры. В общем, иногда можно полностью забыть о существовании **Track View**. Очень удобное дополнение.

## Toolbars

Все меню и toolbar'ы могут быть плавающими или прикрепляться сверху, снизу или сбоку. Иконки можно переставлять в любом порядке. Все, что мешает можно прятать, вызвав потом горячими клавишами. Настройки интерфейса (**Custom User Interface (UI)**) можно сохранять в файл, что позволяет чувствовать себя комфортно на любом компьютере (захватив с собой файл своих настроек). Плавающие toolbar'ы наконец-то позволяют полноценно использовать **Expert Mode**.

## Schematic View

**Schematic View** — графическое представление вашей сцены, дающее дополнительную возможность смотреть, организовывать и воздействовать на различные элементы сцены. Здесь видна вся иерархия, можно пользоваться



функциями **Copy** и **Paste** для изменения структуры объектов.

## Материалы

Помимо старых добрых **Constant**, **Phong**, **Blinn** и **Metal** добавлены новые типы закраски **Oren-Nayer-Blinn** (для тканей и других мягких материалов), **Ward Anisotropic** (для реалистичного металла и стекла), **ANSIO2** (тоже для металла и стекла с небольшими дополнениями) и **Strauss** (быстрое и легкое создание того же металла).

## Xrefs

**Xrefs** — ссылки на внешние файлы. Возможность повысить производительность группы людей, работающих над одним проектом: позволяет работать нескольким аниматорам с одной сценой, привлекать удаленных аниматоров. Предварительные модели включаются в финальную сцену в виде Xref-ссылок. При изменении в дальнейшем этих моделей, финальная сцена также обновится в автоматическом или ручном режиме.

## RAM Player

Этот плеер позволяет загружать в RAM некоторое количество кадров и воспроизводить их с нужной скоростью. Два канала дают возможность сравнивать два варианта.

### **Particle Systems**

Добавлена возможность столкновения частиц одной системы между собой (если ваш компьютер выдержит это). Мета-частицы теперь тоже размножаются.

### **Скрипты**

Расширено присутствие скриптов в MAX. Выполняют роль макросов, которые можно записать, повторить, вынести на панель в виде кнопки, и т.д. и т.п.

# Требования к оборудованию

- Минимальный процессор: Pentium с частотой не менее 150 МГц.
- Минимальная память RAM: 64 МБ.
- Файл подкачки: не менее 200 Мгб.
- Видео-карта: должна поддерживать минимальное разрешение 800x600 и 256 цветов.

Использование минимальной конфигурации не обеспечит нормальной работы художника, дизайнера. Для ускорения работы программы рекомендуется расширить оперативную память до 128 МБ, использовать двойной процессор Pentium Pro или Pentium II.

Значительный выигрыш можно получить за счет использования мощных видеокарт и 3D акселераторов. К рекомендуемым следует отнести следующие видеокарты:

- Matrox Millenium;
- ATI Graphics PRO Turbo;
- Diamond Stealth 64 Varm;
- Number 9 128 Imagine;
- Glint.

3D Studio MAX R2.5 поддерживает работу с акселераторами Open GL и Direct 3D, хотя рекомендуются 24-битные 3D акселераторы.

# Какое «железо» нужно для 3D программ

Вообще, программы трёхмерного моделирования являются наравне с пакетами обработки hi-end 2D графики и 2D видео одними из самых требовательных на рынке программного обеспечения для персональных компьютеров. Задавшись целью создать у себя на столе полноценную студию синтеза 3D графики, вы должны запастись 7-10 тысячами долларов. Если к тому же вы решите пользоваться только легальным ПО, то добавьте сюда ещё от тысячи до 50.000 долларов — в зависимости от «крутизны» и набора предполагаемого ПО. К счастью, компьютеры и комплектующие дешевеют прямо на глазах, что нельзя сказать о ПО. Однако, если вы просто хотите посмотреть, что может человек сделать с помощью компьютера и собственного воображения, то есть вариант использовать уже существующий компьютер, слегка расширив его возможности.

Обратите внимание, что если вы собираетесь серьёзно заниматься этим делом и имеете достаточные финансовые возможности, то лучше будет купить готовую графическую платформу у одного из известных производителей. Таким образом вы получите сбалансированную мощную платформу отличного качества и с гарантией работоспособности. Тем не менее, большинство профессионалов предпочитает самоличноковыряться во внутренностях компьютера, выбирая компоненты на собственный вкус.

Мы рассмотрим платформу Wintel — x86/WinNT. Использование в качестве операционной системы для производства графики на x86 чего-либо, отличного от Windows NT Workstation, вряд ли целесообразно.

### Монитор

Ну, вряд ли стоит говорить о том, как важен хороший монитор для производства графики. Как правило, работая с программой, вам необходимо 3-4 видовых окна, кроме того, программы размещают на рабочем столе немеренное количество всяких служебных, вспомогательных и прочих окошек. В итоге, работая на 15" мониторе (14" вообще в расчёт не берется — место им на помойке), вы имеете на экране свалку служебных панелек и окошек и крохотные видовые окна, на которых видны огрызки вашего проекта.

В общем, 17" — это минимум, если вам не надоели собственные глаза. В последнее время предпочтение отдается работе на 20" мониторах с разрешением 1280x1024. Кстати, совершенно не обязательно брать шикарный монитор с множеством настроек и цветокалибровкой. В процессе работы вам всё равно придётся сжимать рабочий материал в файлы AVI/MPEG и особых преимуществ при работе с монитором за 4000\$ вы не увидите. Другой вопрос — геометрия монитора и его сведение должны быть как можно лучше.

### Видеокарта

Тут дать определённый совет несколько сложнее. В последнее время на рынке появилось много видеокарт,

сочетающих в себе отличные возможности отображения двухмерной графики и ускорения трёхмерной. В принципе, для работы с 3D софтом подойдёт практически любая карта, начиная с простейших S3 с мегабайтом видеопамати. Однако вам мучительно будет наблюдать за медленной (как «Запорожец» на просёлочной дороге) перерисовкой окон.

Лучше брать одну из современных видеокарт, имеющую быстрый DAC и хороший процессор OpenGL ускорения. Обратите внимание, что частота DAC должна позволять вам работать в вашем любимом разрешении с частотой вертикальной развёртки, как минимум 80Hz. Иначе после пары часов работы вам неимоверно захочется выключить чёртов компьютер и пойти пить пиво в ближайшем баре. Одним из неплохих и недорогих решений является видеокарта Matrox Millenium G200.

Однако, есть ряд специализированных видеокарт, предназначенных для работы именно с такими пакетами. В процессе работы вам приходится часто передвигать объекты по экрану, изменять положение камеры, просматривать динамику анимации. В MAX'e, к примеру, есть возможность задать в видовом окне достаточно качественную картинку с приличным шейдингом и учётом источников света, да к тому же ещё и наложить текстуры на объекты. Это здорово помогает оценить конечный результат работы, но даже на 400-ом процессоре движение картинки в окне будет очень медленным, да, к тому же, весьма некачественным.

Решением являются видеокарты ускорения OpenGL-графики. Они позволяют переложить бремя обчёта графики с центрального процессора на

специализированный, а то и на несколько специализированных сразу. Естественно, качество и скорость просмотра возрастают многократно. Для того, чтобы выбрать подобную видеокарту, обратитесь к прайс-листам и рекомендациям специалистов фирм, занимающихся продажей подобного оборудования, так как желательно подыскивать видеоускоритель под конкретные нужды. Например, SoftImage 3D Extreme при установке на компьютер без такой платы сначала удивляется, что таковой не найдено, а потом честно заявляет, что вряд ли скорость работы с ним будет высокой.

### Процессор

Пакеты трехмерной графики являются наиболее требовательными к вычислительной мощи компьютера. Сам процесс рендеринга состоит в многократном повторении сложных операций с огромными массивами данных. Таким образом, производительность вашей работы будет напрямую зависеть от выбранного CPU. И если моделировать можно и на 100-133 MHz Pentium, то скорость рендеринга на подобном аппарате будет просто невыносимо медленной. Поэтому главная рекомендация будет очень простой — выбирайте самый производительный процессор из доступных вам.

Не рекомендуется использовать процессор, отличный от Intel Pentium. На Cyrix'ax, TI и других подобных клонах очень медленный математический сопроцессор, а AMD хотя и производит недорогие качественные процессоры, но вечно отстаёт от Intel по производительности. Хотя в последнее время, AMD представила технологию 3D Now!, которая уже поддерживается многими играми и ПО.

Возможно, эта технология вместе с возрастающей мощностью изделий AMD сделает его приемлемым решением для 3D-платформ.

Если вы хотите получить действительно хорошую производительность рендеринга, то купите двухпроцессорную системную плату и, соответственно, два PentiumII. Многие программы построены с учётом многопоточности современных ОС и использование 2-х процессоров даст 60, 80 а то и 90% прироста производительности. Правда, из всех ОС вам, в таком случае, остаётся использовать только Windows NT Workstation, так Windows 95, OS/2 и Windows 98 два процессора не поддерживают, а выбирать в качестве ОС Windows NT Server было бы... несколько нелогично. К сожалению, дополнительные модули для 3D-софта, как правило, не поддерживают двухпроцессорные компьютеры. Производительность двух PentiumII-400 будет уже приближаться к младшим моделям SGI.

### **Оперативная память**

Как правило, серьёзные пакеты декларируют минимальную потребность RAM 32Mb. И на этом количестве памяти они действительно запустятся и даже будут что-то делать. Но если вы хотите работать, а не созерцать негаснущую лампочку активности HDD, то лучше не стесняться и втыкать в системную плату 128Mb SDRAM. С этим количеством RAM вам уже будут не страшны комплексные сцены с большим количеством объектов.



### Винчестер

Сами пакеты занимают на диске относительно немного. Ядро отнимет у вас от 50 до 150Мб. С такой конфигурацией уже можно работать. Однако библиотеки качественных текстур способны занять несколько гигабайт.

Кроме того, 10 секунд 32-битного несжатого видео с альфа-каналом разрешения 640x480 (не годится в качестве материала для вещания, только для AVI/MPEG роликов) с частотой кадров 24 в секунду займёт почти 300Мб на диске. Двухминутный ролик 1280x1024 30 кадров/сек — около 18 Гб. Сжимать же рабочий материал в режиме реального времени не позволит никакой процессор. Таким образом, для того чтобы просто побаловаться, вам хватит 2-4 Гб, а серьёзные люди задумаются о покупке RAID гигабайтов этак на 48. Естественно, желательно подключение SCSI винчестера с качественным контроллером.

### Сетевая плата

Спросите, зачем сетевая плата? Во-первых, можно выделить файл-сервер для хранения временных файлов. Но главное — многие программы позволяют организовать распределенный рендеринг в сети.

Например, 3D Studio MAX распространяется с уже включённым в дистрибутив комплектом ПО для организации сетевой визуализации. В этом случае вам будет необходимо получить лицензию на право использовать MAX в сети и настроить свою сеть на поддержку протокола TCP/IP. Тогда одна из машин будет

выполнять роль сервера, раздавая задания клиентам и контролируя ход их выполнения. Производительность рендеринга возрастает пропорционально количеству машин в сети.

Многие студии по производству 3D анимации вынуждены создавать сети из десятков высокопроизводительных машин для того, чтобы в приемлемое время успевать выполнять качественную визуализацию длинных анимационных клипов. В принципе, возможна визуализация и через сеть Internet, но вряд ли это стоит делать из-за очень низкой скорости работы этой системы.

В любом случае, вам необходима высокопроизводительная сеть типа 100BaseT или ATM.

## **IntelliMouse и 3D Studio MAX**

IntelliMouse появилась на свет в 1997 году и пришлась по вкусу многим пользователям, как это не странно для продуктов Microsoft. Внешне эта мышка не сильно отличается от своих собратьев — дизайн с намеком на эргономичность, две кнопки, шнур. Одно нововведение — небольшое колесико, расположенное между кнопками.

И что же с этим нововведением делать?.. Главное достоинство в том, что вращая это колесо, можно двигать текст в окне вверх-вниз, не прибегая к помощи кнопок. Таким образом удобно просматривать многостраничные тексты и бродить по Internet.

Чем это колесико может помочь при работе с 3D Studio MAX? Здесь Microsoft остался Microsoft'ом: то, что великолепно работает в Word'e или Excel'e, напрочь

отказывается работать в других программах. Благо есть такой человек Jim Barry и его программа FreeWheel. Подключив ее в **StartUp**'е Windows, вы можете благополучно забыть о существовании двух кнопок MAX'a. Теперь можно масштабировать (вращая колесико) и панорамировать (нажав на колесико), не покидая рабочей области. Это настолько удобно, что поработав на IntelliMouse пару дней, вы станете испытывать дискомфорт на других мышках и постоянно протирать пальцем то место, где должно находиться колесико.

Программа FreeWheel бесплатная и взять ее можно с сайта [www.geocities.com/SiliconValley/2060/freewheel.html](http://www.geocities.com/SiliconValley/2060/freewheel.html).

Одно небольшое ограничение: эта программа работает, начиная со версии 2.0 3D Studio MAX, первая версия MAX'a не поддерживает работу со средней кнопкой, а, соответственно, и с колесом.

# Тонкости, хитрости и приемы работы

Трехмерные объекты (**mesh**), с которыми мы будем работать, на самом «нижнем» уровне являются ничем иным, как геометрическими поверхностями, состоящими из множества граней (**face**), которые, в свою очередь, образуются ребрами (**edge**) — соединениями между отдельными точками (**vertex**) объекта. Вполне возможно формировать объект, создавая точки, ребра и грани и воздействуя на них напрямую, однако гораздо чаще на помощь приходят иные средства моделирования.

Впоследствии поверхность объекта снабжается материалом, будь это просто цвет или набор текстур. Во втором случае требуется указать, каким образом двумерные текстуры накладываются на объект — определить «мэппинг» (**mapping**). В заключение остается осветить созданную сцену и выбрать точку наблюдения, поставив в ней камеру (часто ее функцию выполняют стандартные «обзорные» проекции).

Теперь за дело принимается «рендерер» (**renderer**), строя конечную картинку из «проволочной» геометрии, текстур и источников света. Технологии при этом применяются разные, 3DStudio MAX с некоторыми пор является членом (хоть и не совсем полноправным) клуба пакетов для трехмерной графики, поддерживающих **raytracing** — метод построения изображения, при котором

лучи света преследуются на их пути через виртуальный мир к виртуальному глазу.

### Сплайновое моделирование

Кубический сплайн — кусочно-гладкая параметрическая кривая, используется для аппроксимации формы, которая имеет более сложное задание. Существуют разнообразные способы представления кубических сплайнов, используемые в прикладных программах графики, каждый из них отличается различными свойствами и характеризуется своей областью использования.

#### Bezier-сплайны

Эти сплайны используются и в 2D прикладных программах для векторного рисунка, и в анимации по ключевым кадрам для задания траектории движения. Такие сплайны обычно управляются набором контрольных точек по кривой, с возможностью настройки и управления формой в окрестности этой точки. Управляя наклоном (производной) кривой, проходящей через данную точку, и гладкостью в ее окрестности, получают необходимое приближение. Обычно изменяется только наклон и с обеих сторон обеспечивается гладкость. В некоторых случаях, однако, симметрия может быть нарушена, наклон — тот же самый, но вторая производная меняется, или и первая производная и вторая производная в точке меняются и мы получаем излом.

Простота интерфейса, когда сплайны меняют форму, переходя от узла к узлу, обеспечивает большое удобство при моделировании и анимации, однако существуют

некоторые сложности, преодолеть которые при такой форме задания кривой не всегда возможно.

Мы можем увидеть иногда, как при движении внезапно изменяется ускорение, поскольку объект переходит на участок кривой с другой второй производной при смене ключевого кадра. Движение при этом выглядит рваным и неестественным.

### **В-сплайны**

Следующий класс кубических сплайнов, которые решают эту проблему. В-сплайны сохраняют тангенс угла наклона, и изменение кривизны является непрерывным в контрольных точках, что делает этот тип кривой оптимальным для анимации и создания 2D-форм. Однако цена этого — отсутствие удобного интерфейса для интерполяции — форма сплайна полностью задана контрольными точками, не лежащими на кривой.

### **NURBS (неоднородный рациональный В-сплайн)**

Термин, про который каждый знает, что это хорошо, но немногие могут объяснить — почему. NURBS — единица «контрольного списка», которую каждая трехмерная программа графики должна иметь по маркетинговым причинам. NURBS — специальный тип В-сплайна, разработанный для использования в трехмерной графике. NURBS — тип В-сплайна и имеет все свойства, которые имеют В-сплайны со всеми их преимуществами и недостатками. Он также неоднороден, что является полезным для установки ключевых кадров в произвольных точках времени.

Однако трансформации объектов, заданных в такой форме, более сложны и требуют большого количества

операций с плавающей точкой, которые не могут быть оптимизированы и требуют большего времени для вычислений. Поэтому большинство пакетов компьютерной графики, особенно на PC, справедливо считая скорость вычислений важнейшим фактором, избегают NURBS представления.

Конечно, NURBS представление более точно отображает многие поверхности, такие, например, как круги и эллипсы. Это делает NURBS более интересным для систем автоматизированного проектирования и производства прикладных программ механического моделирования. Для аниматоров, однако, круги, которые могут быть сделаны B-сплайнами, или даже обычными Bezier-сплайнами достаточно близки к реальным и если вы больше заинтересованы в искусстве, чем в технологии, можете не вспоминать NURBS.

Однако с использованием аппаратного отображения поверхностей графической системой компьютера, может пригодиться другая особенность NURBS: возможность линейных преобразований самих контрольных точек и оптимизация вычислений на уровне их структуры, с последующим аппаратным отображением самой поверхности, и тогда этот способ представления получит большее распространение.

### **Просчет и фотореализм**

Термин фотореализм используется, чтобы описать компьютерное изображение, которое выглядит также реально, как на фотографии. Процесс компьютерной генерации изображения (просчет) или рендеринг, оценивался на заре своего появления исключительно с

точки зрения фотореалистичности. Однако, с развитием компьютерной графики этот критерий отходит на все более дальний план. И дело даже не только в том, что компьютерная графика все больше переходит в область искусства, а и в том, что основные принципы компьютерных построений, а следовательно, и все приемы и средства, которые позволяют добиваться впечатляющих эффектов, к реализму никакого отношения не имеют, и рождаются подчас исключительно в недрах самой компьютерной графики.

Рассмотрим основные методы визуализации компьютерной модели. Наиболее простым из таких методов является *метод Гуро*, который основывается на определении освещенности грани в ее вершинах с последующей интерполяцией получившихся величин на всю грань. Этот метод не обеспечивает гладкости изменения закраски и имеет много дефектов на изображаемой поверхности, однако является самым оптимальным в вычислительном отношении, поэтому часто используется в предварительных расчетах.

Как и метод Гуро, следующий метод — *Фонга*, при расчетах также опирается на интерполирование закраски, однако, в отличие от предыдущего метода, здесь интерполируется значение самого вектора нормали в данной точке, которое затем используется для вычисления цвета пиксела в соответствии с законами физики.

Вот тут-то и начинается... Произвольно меняя вектор «нормали» в точке, мы можем получать из острых ребер — гладкие, а из гладких поверхностей — шершавые. А также, здесь мы можем изменять цвет и свойства поверхностей,



используя специальную картинку — «текстуру» или «карту» того или иного свойства материала.

Существуют различные способы наложения текстур или непосредственной генерации в каждой точке по заданной формуле (такие текстуры называются процедурными, в отличие от простых, проективных). Несомненным достоинством моделей Гуро и Фонга является их сравнительная простота. Однако, вследствие значительных упрощений, получаемый результат не всегда оказывается удовлетворительным, поэтому приходится пользоваться более совершенными методами. Как, например, методом трассировки лучей (**ray-tracing**) или методом излучательности (**radiosity**).

При построении изображения методом трассировки, луч посылается в заданном направлении для оценки приходящей оттуда световой энергии. Эта энергия определяется освещенностью поверхности, встретившейся на пути луча, с учетом преломления, отражения и рассеяния. Поступая таким образом, можно определить освещенность каждой поверхности сцены и построить требуемое изображение.

Однако, полный просчет лучей от всех источников, называемый прямой трассировкой, из-за огромного количества вычислений не применяется. Кроме того, большая часть работы на прослеживание лучей, не попавших в глаз наблюдателю и определение освещенности невидимых поверхностей, окажется проведенной впустую. Поэтому на практике реализуется так называемая обратная трассировка, при которой отслеживаются только лучи, попавшие в глаз наблюдателя.

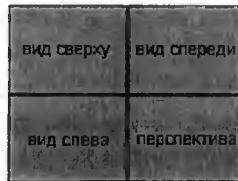
Основными недостатками метода трассировки лучей является неэффективность работы с диффузными поверхностями (которые, в основном, и существуют в природе) и то, что определение освещенности поверхностей проводится параллельно с построением изображения и зависит от положения наблюдателя.

Метод излучательности устраняет эти недостатки, обеспечивая высокую точность при работе с диффузными объектами и отдельное вычисление глобальной освещенности, независимо от положения наблюдателя. В основе метода лежит закон сохранения энергии в замкнутой системе. Все объекты разбиваются на фрагменты, и для этих фрагментов составляются уравнения баланса энергии. Грубо говоря, каждый элемент сцены выступает в роли источника света. Картина в этом случае наиболее достоверная, но вычислительная сложность непомерна даже для специализированных рабочих станций. Поэтому те немногие реализации, которые существуют сегодня, используются только при просчете одиночных изображений.

И, наконец, делая общие выводы о процессе 3D-визуализации, важно подчеркнуть необходимость качественного алгоритма просчета. И даже не в смысле математики, положенной в его основе, а в смысле его высокопрофессиональной реализации. **Renderman** фирмы Pixar не использует ни алгоритм трассировки лучей, ни, тем более, излучательность, однако считается одним из самых качественных и на нем-то, собственно, считались почти все широко известные работы для видео и кино.

### Что дальше?

Плавно переходим к работе в 3D Studio Max. Бегло ознакомимся с оболочкой и попробуем выявить элементы, представляющие для нас наибольший интерес, а заодно и просто попривыкнуть к обстановке, дабы на будущее знать примерно, что где лежит. Итак, смотрим и видим...



...4 проекции создаваемого нами мира. Их расположение, тип и количество можно менять. Активируется тот или иной вид кликом на его названии в левом верхнем углу.

Далее видим довольно типичное меню.



То, что нас непосредственно может интересовать, объяснений не требует. Более того, многие его команды продублированы на других панелях.

Далее видим пестрое сборище всевозможных инструментов.



Немного подробнее о наиболее важных.



Это классические **Undo** и **Redo**. Распространяются на множество шагов, но существуют и необратимые операции.



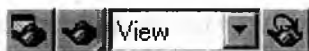
Курсор для выбора объектов, меню типов выбора (прямоугольник, эллипс, полигон), фильтр выбираемых объектов, вызов окна списка объектов для выбора по имени.



Эти кнопки представляют собой самые простые операции, которые можно совершить с объектом в пространстве — переместить, покрутить, пропорционально или непропорционально уменьшить или увеличить, опираясь при этом на различные системы координат, с возможностью ограничить изменения одной или двумя осями.



Кнопка вызова редактора материалов.



Основной доступ к рендереру — модулю, занимающемуся построением готовых картинок из

сделанной нами геометрии, присвоенных объектам поверхностей и расставленных источников света.



Панель контроля видов/проекций создаваемого мира для перемещения точки зрения наблюдателя, изменения направления и угла обзора, оптимизации вида и т.д. — не влияют на созданную геометрию (за исключением положения камеры).

Все, что мы до настоящего момента успели рассмотреть, больше напоминает своеобразный рабочий стол, на котором под рукой — в относительном (техническом) беспорядке — всегда самое нужное. Теперь на очереди строгая иерархическая система создания и модификации объектов — этакий шкаф с разложенными по полочкам и ящикам заготовками и инструментами. Вот как выглядит верхний уровень этой иерархии:



Работа по созданию модели, как правило, начинается с простейших двух- или трехмерных элементов, таких как прямая линия, сплайн, полигон, куб, шар и т.д. В результате взаимодействия примитивов по различным схемам и наложения на объекты или их части соответствующих модификаторов, возникают более сложные и приближающиеся к конечному идеалу модели.

Так, в меню **Create** мы, в первую очередь, сталкиваемся с возможностью выбора типа элемента, который мы хотим создать:



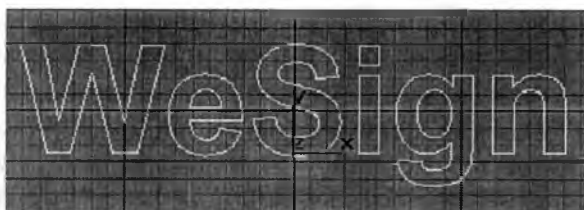
стандартный ли примитив, двухмерную фигуру, а может, источник света, камеру или что-то другое.

## Hello, world!

Итак, долгожданное начало! Попробуем сделать его достаточно простым и эффектным, и, не слишком размышляя над смыслом сего действия, придадим объем некоему тексту. Найти «создатель текста» не составляет большого труда — он именно там, где ему и положено быть, среди двумерных примитивов, в категории **Splines** на панели **Create**.



Появившаяся внизу диалоговая панель для ввода текста, особых объяснений не требует, теперь можно подстегнуть свое воображение в выборе шрифта и содержания. Касательно первого рекомендуется для начала ограничить выбор шрифтами «потолще» и без большого количества острых углов и узких переходов, хотя, в принципе, подойдет что угодно — в нашем случае это был простой **Arial Bold**. Когда выбор сделан, нажмем левой кнопкой мыши в центр поля **Front**. Вот и наш текст.



Самое время немного поиграть с режимом перемещения,



подвигав свежесозданный текст в разные стороны, а также с контроллерами видов,



разобравшись, что делает каждый из них — это совсем несложно.

Попробуем сделать наш текст трехмерным. Методов для этого существует множество, однако мы воспользуемся самым простым и удобным, который предлагает нам MAX — модификатором **Bevel**.



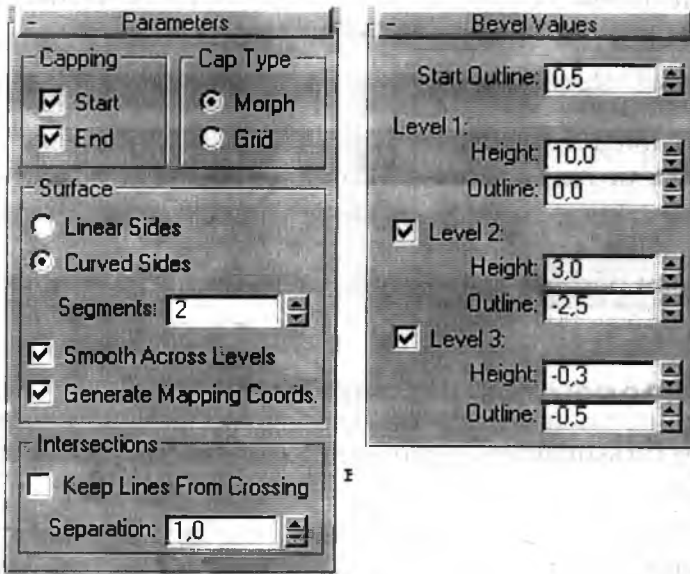
**Bevel** находится в стандартном наборе модификаторов и доступен через меню **more** на панели **Modify**. Модифицируемый текст, само собой, должен быть выбран.

Коротко о том, как работает этот модификатор: к контуру, взятому за основу, достраиваются на заданном расстоянии от него дополнительные «этажи» (**levels**) — максимум 3 штуки — из его копий, по необходимости расширенных или суженных (**outline**). Несколько сжимая контур текста при удалении от основы, можно получить «мягкую» выпуклую поверхность.

Можно посвятить некоторое время экспериментам с различными параметрами, однако для ориентации здесь приведена неплохо работающая комбинация (размер текста — 100). Пользоваться параметром **keep lines from crossing** рекомендуется только в крайнем случае, когда иным способом невозможно удержать линии контура от напользания друг на друга и образования артефактов при



изменении **outline** - связанный с ним алгоритм требует весьма интенсивных вычислений.



То, что у нас должно получиться, выглядит в поле перспективного обзора (**perspective**), в котором мы, ради исключительно эстетических целей, с помощью контроллеров видов несколько изменили положение точки наблюдения и масштаб, примерно так:



Сейчас вполне можно задаться вопросом, с какой, собственно, стати, объект в перспективном обзоре освещен, и чем? — ведь мы пока не создавали источников света. Дело в том, что для облегчения нам жизни на первых этапах работы со сценой, МАХ по умолчанию добавляет «параллельную» освещенность без теней. Организацией правильного освещения мы займемся несколько позже, а пока давайте посвятим некоторое время замене глупого шероховатого пластика (примерно так выглядит материал по умолчанию, цвет выбирается случайным образом), из которого сейчас сделан наш текст, на что-нибудь более впечатляющее.

### **Хромированный металл**

По сложившейся традиции обучения первым шагам в 3D, используется хромированный металл. Традиция неплохая, не будем ее нарушать.

Прежде, чем мы нажмем на кнопку вызова редактора поверхностей, позвольте немного предварить наше с ним знакомство коротким теоретическим отступлением.

Поверхности объектов в окружающем нас мире характеризуются множеством параметров — они по-разному поглощают, отражают и преломляют свет. Поскольку рендерер стремится добиться максимального реализма при построении изображений, можно быть готовым к тому, что материалы в виртуальной сцене также должны быть снабжены всевозможнейшими характеристиками, отражающими их оптические свойства.

Для начала разберемся с простейшей из них — цветом. Точнее, цветами, поскольку в редакторе материалов 3DStudio MAX базовых цветов — четыре:

- **Ambient** — цвет объекта с неосвещенной прямым светом стороны.
- **Diffuse** — цвет объекта с освещенной белым светом стороны.
- **Specular** — цвет ореола блика от яркого источника света.
- **Filter** — «окраска» света, проходящего через отчасти прозрачный материал. Этот пункт, строго говоря, на конечный цвет объекта прямо не влияет.

«Блестящность» объекта определяется двумя параметрами:

- **Shininess** — размер блика.
- **Shininess Strength** — яркость блика.

Кроме того, мы можем регулировать прозрачность (точнее, непрозрачность) объекта параметром **Opacity**, и сделать его светящимся, устанавливая **Self-Illumination**.

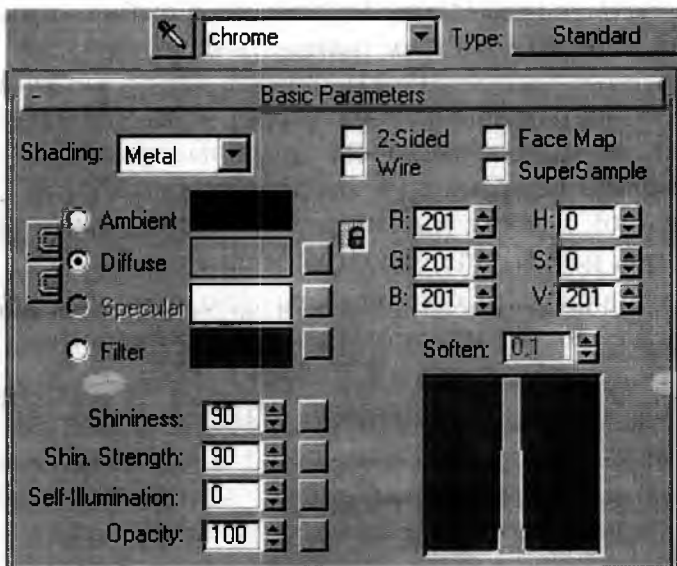
Изученного нами сейчас достаточно для моделирования поверхности объекта, в реальном мире сделанного из одноцветного материала равномерной матовости/блестящести. Однако поверхности реальных объектов достаточно редко одноцветны и равномерны. Гораздо чаще они текстурированы: на них бывают изображения, неровности, они обладают меняющейся в зависимости от точки объекта блестящостью и прозрачностью. А иногда в них к тому же что-то отражается. Как быть? Редактор материалов предлагает

## Тонкости, хитрости и приемы работы

---

наиболее логичное решение — нанести на объект причитающиеся ему текстуры в виде плоских картинок. Итак, вооруженные приобретенными знаниями, вперед.

В стандартном варианте редактор поверхностей предлагает 6 ячеек (slot) для одновременной работы с материалами и текстурами. В каждой из них — образец материала на освещенном шаре (вариант — кубе или цилиндре). Конечно, можно выбрать готовый материал из стандартной библиотеки, входящей в состав стандартной инсталляции пакета, однако мы учимся, а потому попробуем сделать свой собственный материал от начала и до конца.



За основу возьмем материал в любой ячейке и выставим **Shading**, цвета, **Shininess** и **Shininess Strength** в

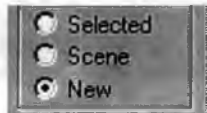
приведенные здесь значения. Заодно можно и назвать наш материал — **chrome**. Полученный образец материала хромированную сталь, однако, напоминает довольно слабо. Проблема очевидна — не хватает отражений.

На панели **Maps** найдем требуемый пункт — **Reflection** — и ткнем мышкой в пока еще пустую (в ней значится **None**) ячейку текстуры.



Небольшое замечание касательно терминологии: под текстурами (**map**) мы будем понимать нечто большее, чем проецируемые на поверхность объекта картинки (они являются лишь одним из параметров), в это понятие входят и так называемые процедуральные текстуры (изображения не хранятся в файле, а генерируются программно), а также многие другие данные, такие как искажения, размытость, фильтры и прочее.

В появившемся меню выбора типа текстуры, выберем **Raytrace** для достижения большей реалистичности в воспроизведении отражений. Теперь в объектах с поверхностью из этого материала отражались бы другие элементы сцены.

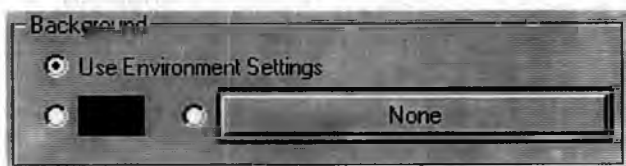


## Тонкости, хитрости и приемы работы

---

В нашем случае, однако, мы имеем в сцене выстроенные в ряд буквы, да и только. А потому отражений получилось бы до обидности мало. К счастью, текстура **Raytrace** предлагает возможность эмулировать окружение объекта другой текстурой — в нашем случае это будет просто некая картинка, имитирующая хромированную поверхность.

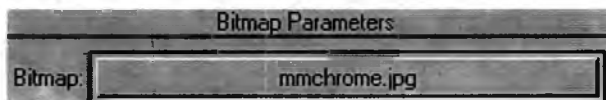
Обнаружив параметр **Background**, кликнем в опять же пока недействующую ячейку текстуры фона.

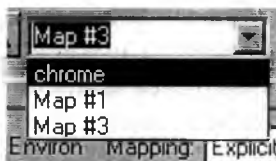


В появившемся — уже знакомом нам — меню выбора типа текстуры на этот раз выберем **Bitmap** — ведь на этот раз мы хотим использовать вполне конкретную картинку.



Наконец, установим параметр **Bitmap**, выбрав соответствующий файл.





В процессе создания материала мы незаметно спустились по лестнице иерархии на 2 уровня вниз (от самой поверхности к одной из её текстур и далее к её вспомогательному элементу). Вернемся обратно.

Наше творение должно теперь выглядеть примерно так:



Остался один важный шаг — присвоить материал выбранному объекту нажатием на кнопку



На этом работа с редактором материалов завершена, закроем его и вернемся к нашей сцене.

Наверное, уже сейчас ужасно не терпится посмотреть, что же мы там натворили. Что ж, активируем вид **Perspective** и обратимся к услугам рендерера

(**Render Scene**), выставим размер картинки поменьше (320x200 вполне подойдет) и ткнем в кнопку **Render** (в будущем, когда все параметры рендеринга выставлены, можно пользоваться **Quick Render**, расположенной непосредственно справа от **Render Scene**). Прилежно подождем, алгоритмы **Raytracing** требуют множества вычислений.

Обнаружить панель создания источников — не так сложно:



- **Spot** — конус света из определенной точки.
- **Free** — направленный поток параллельных лучей.
- **Omni** — точечный источник, излучающий равномерно во все стороны.

Для освещения сцены нам потребуются три лампы типа **Omni** потемнее и один **Spot** поярче. Начнем с **Omni**.

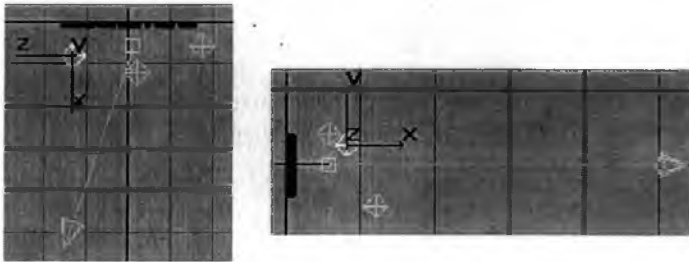
Выбрав **Omni** и слегка уменьшив яркость примерно до **90**, ткнем левой кнопкой мыши в окне **Top** где-нибудь



перед текстом — первый источник света создан! Еще две копии его можно сделать посредством клонирования — **Shift+drag-and-drop** в режиме перемещения.



Теперь очередь **Spot**. Установим его яркость примерно на **130**. Контроллером видов **Zoom** изменим масштаб сцены в окне **Top** так, чтобы можно было создать центр новой лампы на достаточном расстоянии от текста, ткнем левой кнопкой мыши и протянем линию — основное направление **Spot**. Теперь осталось разместить созданные лампочки, пользуясь видами **Top**, **Left** и **Front**:



Теперь волнующий момент — финальный рендеринг...



### Ландшафт

Иногда (и даже очень часто) бывает необходимо создать реалистичный природный ландшафт. Для 3DS была написана такая уйма plug-in'ов, позволяющих их моделировать (3DSurf, Miner, и т.д.), а в 3DSMAX есть несколько способов, позволяющих получить ландшафт, не прибегая к нестандартным plug-in'ам.

#### Способ №1

Создадим простой, но эффектный ландшафт. Для этого воспользуемся примитивом **Box**. При создании параллелепипеда укажите количество сегментов по двум измерениям побольше, как показано на рисунке.



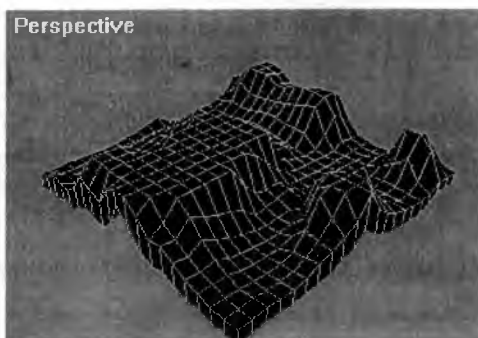
Теперь на получившийся сетчатый объект примените модификатор **Noise**. Проставьте флажок **Fractal, Roughness 0.0, Iterations 6.0, и Strength по Z равную 50** (или другую, в зависимости от ваших масштабов).

Ландшафт готов! Добавьте материал и небо по вкусу.



### Способ №2

Ландшафты, создаваемые предыдущим методом имеют один недостаток — некоторую монотонность. Реально в природе вы никогда не встретите горного массива, состоящего из часто натканных мелких холмиков. Так что предыдущий метод хорош для создания холмистой, но не гористой местности. А если нам нужен настоящий горный ландшафт, то можно воспользоваться следующим.



## Тонкости, хитрости и приемы работы

---

Начнем так же, как и в предыдущем способе — создадим **Box** с множеством вершин. Теперь мы воспользуемся модификатором **Displace**, который изменяет геометрию предмета по картинке или по **map**'у.

Допустим, картинки возвышений у вас нет (если есть, то вперед, она используется, как в **Bump**'е — чем светлее пиксел, тем выше). Остается использовать **map**.

Воспользуемся процедурным **map**'ом **Dent**. Это почти его прямое назначение — как **Bump**. Войдите в **Material Editor** и создайте новый **map** — **Dent**. Укажите следующие параметры:

Size=800

Strength=20

Iterations=2

X, Y, Z Tiling каждый по 0.2



Теперь примените модификатор **Displace** (**More** ⇒ **Displace**) на ваш **Box**, указав **Strength**, равную **35** (или другую — зависит от вашего масштаба), и **Dent-map**, который вы только что создали. Если вы хотите использовать не **map**, а просто картинку, то вместо **Map** нажмите **Bitmap**, и укажите на свою картинку.

Первоначальный **Box** должен искривиться. Те его вершины, которые при накладывании на него **map**'а должны бы были быть окрашены светлее, поднимутся, а те, которые темнее — опустятся. Теперь можно наложить на него текстуру, причем заметьте, что можно и не добавлять дополнительно модификатор **UVW Mapping**, т.к. сам **Displace** работает как **UVW**, если включить опцию **Apply Mapping**.



Вот вариант готового отрендеренного ландшафта — вид со дна каньона.

### Способ №3

Если все-таки вам не нравится ни один из способов выше, то вам ничего не остается, как только достать один из многочисленных **plug-in**'ов для **3DSMAX3** — генераторов фрактальных ландшафтов.

Пример использования одного из них — **Mountain** вы видите ниже. Без его помощи трудно создать, например, такие реалистичные крутые каменистые берега.



### Текстура ландшафта

Допустим, вы уже освоили технику создания ландшафтов, описанную в нашем совете по ландшафтам. Но есть одна маленькая проблема — эти способы отлично отвечают вашим запросам, когда вам нужны марсианские горы или лунные поверхности (т.е. поверхности однородного цвета — например, коричневого), а создать реалистичный ландшафт земного мира уже сложно. Этот совет рассказывает о том, как уже ранее созданный ландшафт раскрасить так, чтобы он был похож на кусок реалистичного земного рельефа.



Итак, приступим. Для начала создадим ландшафт любым из способов, описанных в предыдущей главе. В этой картинке использован **plug-in Mountain**.

Теперь задумаемся о том, что мы хотим увидеть в реальном земном ландшафте. (Кстати, 3D-художникам часто приходится думать о реальной природе явлений — привыкайте к этому). Итак, нам нужно следующее:

- В нижних точках рельефа — светло-зеленая трава.
- Выше — темно-зеленые кустарники и деревья.
- Еще выше становится холоднее и там нет никакой растительности — голая коричневая земля.
- На самом верху — снежные и ледяные шапки гор.



Теперь подумаем, как можно это сделать в 3DSMAX3. Порывшись на буржуйских сайтах, несложно откопать **plug-in Terrain Material**. Эта штука позволяет делать переход от одного материала к другому по высоте. Нам это не совсем подходит, т.к. у нас должно быть, как

минимум, четыре материала. Так что воспользуемся UVW Mapping'ом MAX'a, нанеся текстуру особым образом.

Закройте или прикройте 3DSMAX3 и запустите свой графический редактор (в данном случае предпочтительнее Fractal Design Painter, но подойдет и Adobe Photoshop). Мы особенно сложных текстур делать не хотим, так что у нас будет все просто. Создайте новый рисунок-полоску, например 100x300. Теперь нанесите на него градиентами отдельные цветные участки.

Теперь сделаем плавные переходы от одного градиента к другому. Для этого в большинстве редакторов есть функция **Smudge** (размазывание) или аналогичная (вот почему предпочтительнее FDP). Процесс размазывания очень творческий, но на самом деле требует минимального опыта и средней мышки.

Все практически готово. Осталось затайлить текстуру. Сделайте **Filter** ⇒ **Misc** ⇒ **Offset** и введите в X половину размера картинке по горизонтали. Картинка сместится и будет виден шов тайлов. Теперь размажьте и его, и получите окончательный вариант текстуры. Сохраните ее в один из каталогов **map**'ов 3DSMAX3 и закрывайте редактор.

Снова откройте MAX и загрузите свою сцену с ландшафтом. Идите в **Material Editor** и возьмите новый материал. Присвойте ему в **Diffuse Map Bitmap**'ом нашу текстуру рельефа (для контроля лучше установите кубик, а не шарик). Подержайте за остальные настройки материал, чтобы он выглядел получше (уберите блеск и т.п.). Присвойте материал своему объекту — ландшафту.





Держитесь, осталось немного... Теперь надо присвоить хитрый **UVW Mapping** объекту. Добавьте модификатор **UVW Mapping**. Убедитесь, что он имеет тип **Planar** и нажмите магическую кнопку **Sub-Object**. Разверните прямоугольник так, чтобы он был перпендикулярен плоскости ландшафта, как показано на рисунке. Отожмите **Sub-Object** и кликните на **Fit**. Все! Можно рендерить!

Ну в общем неплохо, да? Снизу у нас все зеленое, выше — коричневое, а шапки гор — белые. Но как-то все нереалистично выглядит. У нас плоская и гладкая текстура, а в природе ничего такого нет. Надо бы добавить немного **Noise'a**.



Переключимся опять в **Material Editor** и изменим тип **Diffuse Map**'а с **Bitmap** на **Mix**. На вопрос о том, хотим ли мы оставить прошлый **Bitmap** как **Sub-Map**, ответьте "да". Теперь во второй ячейке создайте **Noise Map** с черно-белым **Noise**'ом помельче (размер где-нибудь 3-5). Теперь поставьте **Mix Amount 0.15** и посмотрите на результат. Задайте **Bump Map** как черно-белый **Noise** помельче (1-2) и отрегулируйте силу **Bump**'а, чтобы он хорошо смотрелся.

Все! Можно рендерить!



### Как создать море

Будем ваять море. Это то самое, которое плещется и сверкает в лучах нежного южного солнышка в бабье лето. Погода — спокойная, вид с берега, волнение — 0.5-1 балл, тип погоды — наиболее благоприятный для 3D. Будем считать, что поблизости нет речек с грязными водами, давно не было ни дождя, ни шторма. В общем, круг задач поставлен и описан — можно приступать.

Начнем с понимания моря в такую погоду. Во-первых, динамика — волны небольшие, ветра почти нет, а значит и нет барашков на волнах. Лёгкая рябь, да и только. Потом цвет — кроме того, что есть понятие "цвета морской волны", надо бы напомнить, что вода в море окрашена по-разному — ближе к берегу она зеленее, дальше от берега стремится к ультрамарину. Да и взвешенные в воде частицы дают о себе знать — солнечные лучи, преломленные поверхностью, придают воде легкий разброс по температуре цвета и светлости. Вот, в принципе, и всё. Приступим.

Во **viewport'e TOP** создайте **CV Surface** размера **1000x1500** (лучше работать именно в таких больших масштабах, чтобы не было мучений с длинами волн вроде 0.00001). Это будет поверхность моря. Теперь скопируйте эту поверхность, переместив её с **Shift** во **viewport'e FRONT** по оси **Y** единиц на **100** — это будущие облака. (Лучше сразу обзывать все соответствующими именами — море пусть будет "море", облака — "небо").

Теперь нужно создать "окружение" — полусферу с центром координат в точке **0,0,0** радиуса **1000**. Далее эту сферу надо переместить к нижнему краю "моря" во **viewport'e TOP** так, чтобы центр был на нижнем крае, а край сферы касался дальнего конца объекта "море". Теперь смасштабируйте сферу во вьюпорте **Front** через **Non-Uniform scale** по оси **Y** процентов на **50**. После этого выберите вместе "небо" и "окружение", разверните им нормали, присвоив модификатор **Normal**.

Осталось создать направленную камеру во **viewport'e TOP** (ближе к нижнему краю с прицелом наверх). Угол камеры лучше взять увеличенный, **35** мм в

самый раз. Можно немного "увеличить" будущее пространство сцены, повернув "небо" так, чтобы оно почти касалось дальнего края "морья". С геометрией покончено!

Можно приступать к материалам. Приступим к разукрашиванию — здесь, коротко о главном. Начнем с "окружения". Это фактически купол неба, а значит и цвет его — градиентная заливка. Создадим соответствующий материал с текстурой **Gradient** на **Diffuse** — от небесно-синего до голубого (или желтоватого, если рядом город). Для корректности наложения текстуры примените модификатор **UVW Mapping** во **viewport'e FRONT** так, чтобы карта была параллельно **viewport'у** и корректно "сидела" на объекте. Этот материал должен светиться (**Self Illumination = 90-100**). Добавим облака — это белая поверхность с маской прозрачности **Noise (Fractal)** и самосветимостью 50 процентов. Поигравшись параметрами **High** и **Low**, можно регулировать плотность облаков.

А теперь — к главному. Море... Для начала цвет — при удалении от камеры он должен густеть. А значит это градиент. Кроме того, внутренние блики воды — это **Noise (Turbulence и Inverted)**. А значит карта цвета будет **Mix Градиента** и шума на 0.05-0.1. Теперь волны. Не бывает моря без волн, но и волны на море не бывают одного типа. Есть побольше — основные, вызванные ветром, землетрясением и ещё бог знает весть чем, и рябь. Так вот, для бампа мы тоже используем микс карт — воды и шума. Вода будет создавать крупные волны, а шум — рябь. Кроме того, можно при помощи ряби показать неравномерность ветра и морских течений.

Для этого нужно создать **MASK** течений и использовать его для ряби. В нашем случае сойдет и просто обычный шум большого размера, а значит задача упрощается — просто вместо белого цвета шума-маски будем использовать мелкую карту шума ряби. Можно усложнить задачу, задав обычный шум для белого и **turbulent**'ный для черного — это будет ближе всего к истине. Ну и, конечно же, отражения — какая же это вода без отражений? В качестве карты отражений в данной сцене можно смело использовать **Raytrace**, так как она не содержит в себе никаких глюков. Но главное — запомните, что коэффициент отражения воды не так велик, как кажется. Не переборщите.

Теперь осталось осветить всё это дело. Вы никогда не задумывались, откуда на море светит солнце? Если вопрос показался вам глупым — значит не задумывались. Где вы видели море? По телеку или у нас на Кавказе! А по телеку — опять-таки у нас на Кавказе. А значит чаще вам приходится смотреть на море против солнца! Именно такое положение светила покажется наиболее реальным.



Для света используем 1 основной **Direct Light** с **Raytrace**'овыми тенями, исключив из его освещения "окружение". Кроме того, один точечный источник покажет нам положение солнца.

Осталось добавить **Flare**'иков и тумана по вкусу и море готово. Заметим, что в данном случае можно всё анимировать.

### Как сделать реалистичный взрыв и осколки от объекта

Часто используемым эффектом (особенно в космических сценах, анимациях и т.п.) является взрыв. Для взрыва в **MAX**'е предусмотрен стандартный способ — **Combustion** и модификатор **Bomb**. Но это не самый хороший способ, и часто результат не устраивает — дело в том, что **Bomb** разделяет объект, в лучшем случае, на правильные формы групп **face**'ов (в худшем — на отдельные **face**'ы). Если же нужен более сложный эффект — например, разделение объекта на несколько неровных кусков — используйте следующий способ. Демонстрация будет происходить на традиционном объекте для всяких издевательств — стандартном **MAX**'овском чайнике.



Создаем **Create** ⇨ **Particle Systems** ⇨ **Particle Array**. Располагаем этот **Particle Array** где-нибудь в сцене (не важно где). Выбираем объект, который будем взрывать (**Pick Object**). Ставим следующие установки:

- Viewport Display — Mesh
- Particle Type — Object Fragments

Все. Основной взрыв уже готов — запустите анимацию, и увидите, как ваш объект взорвется и разлетится на отдельные **face**'ы (как и **Bomb**).



Если вы хотите что-то более сложное — поэкспериментируйте со следующими установками:

**Particle Type: Object Fragment Controls: Thickness** — толщина осколков. Т.к. чайник у нас имеет достаточно толстые стенки, надо увеличить этот параметр.

**Particle Type: Object Fragment Controls: Number of Chunks** — очень важный параметр, регулирует количество кусков-осколков, на которые разлетится чайник. Если надо, чтобы объект не разлетался на **face**'ы, то поставьте нужный параметр.

**Particle Rotation: Spin Speed Controls: Spin Time** — еще один важный параметр для увеличения реалистичности — контролирует вращение осколков. Если этот параметр не **0**, то осколок будет делать полный оборот за это количество кадров.

Для придания окончательного вида сцене, можно отрегулировать скорость и другие общие параметры — ведь **Particle Array** — это стандартный **Particle System**. В большинстве случаев также надо спрятать основной объект в момент взрыва (в отличие от **Bomb**, **PArray** этого не делает) в **Track View** ⇒ **Add Visibility Tracks** ⇒ **Visibility**. Время начала и конца взрыва также можно регулировать в основных настройках **PArray**.



Заметьте на этой картинке основные отличия **Bomb**'а от использования **PArray**:

- Осколки имеют произвольную форму — не одинарный **face**, и не простая группа **face**'ов.
- Осколки имеют толщину, т.е. считается, что чайник был полым и толстостенным.



- Осколки вращаются в полете.
- Осколки представляют собой стандартные **particle**'ы, и к ним можно применять любые **space-warp**'ы (такие, как **gravity**, **wind** и т.д.).

Можно разделить объект на куски по принципу сглаживания, т.е. объект разделится по линиям около острых углов и прочих хрупких мест.

Можно использовать функции **spawn**'а для обработки ситуаций удара осколков о предметы — подумайте, как здорово будет выглядеть, когда чайник распадается на два осколка, потом каждый из них еще на два, каждый еще и так далее.

Рендерить сцену надо через **VideoPost**. С ней показывается чайник, падающий со стола на пол и разбивающийся на осколки.

## Как сделать параллельные лучи света

Иногда можно создать хороший эффект, показав, например, момент выстрела лазерного орудия, стреляющего пучком параллельных лучей. Этого эффекта можно достигнуть несколькими способами.

### Способ №1

Этот способ заключается в использовании **Volume Light** на **Directional** источнике света. Посмотрим его действие на примере: создадим простую сцену — что-то типа лучевого пулемета, взяв массив из 6 трубок, скрепив их кольцом. Теперь попробуем сделать так, чтобы из верхнего дула вырывался луч света.

## Тонкости, хитрости и приемы работы

---

Создадим стандартный источник света **Target Direct**, так чтобы сам источник находится в дуле (неглубоко, но и не совсем на поверхности), а цель источника далеко на предполагаемой линии лазера. Теперь проставьте параметры **Hotspot** и **Falloff** так, чтобы они соответствовали размерам вашего дула (например, дуло имеет радиус **30**, **Falloff** — **15** и **Hotspot** — **10**).



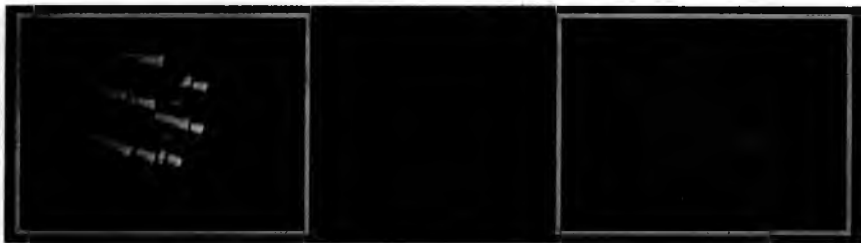
Если у вас изначально не было поставлено источников света, которые бы освещали сами стволы, поставьте их сейчас. Они больше не будут освещаться автоматически, т.к. вы добавили источник света.



Теперь самое важное: чтобы свет было видно, добавьте **Volume Light**. Войдите в меню **Rendering** ⇒ **Environment** и в **Atmosphere** добавьте **Volume Light**, затем нажмите **Pick Light** и выберите ваш direction-источник света.

Самое интересное — это настройки **Volume Light**:

- **Fog Color** — цвет самого света.
- **Attenuation Color** — цвет света, в который свет будет переходить при затухании.
- **Density** — "плотность" света в процентах. 100% — непрозрачный свет.
- **Max Light** — максимальный уровень света. "Плотность" реально не может быть больше его.
- **Min Light** — минимальный уровень света. Уровень света вне луча.
- **Atten.Mult** — коэффициент затухания.



Все, луч готов!

### Создание фонтана

Перед тем как браться за изготовление самого фонтана (с технической точки зрения это не так уж сложно), необходимо ответить на несколько базовых вопросов: как оно движется и почему образует именно эту форму.

На вырывающийся из сопла поток воды действуют две основные силы — земное притяжение с одной стороны и центробежная сила с другой (последняя возникает при выходе воды под напором из круглой трубки). Форма является результатом взаимодействия капель воды — разлетаясь из сплошной струи, они рассыпаются на множество отдельных брызг. В 3D Studio MAXR есть все необходимые инструменты для имитации вышеперечисленных эффектов — **SpaceWarps: Gravity, Motor, PS: Blizzard** — система частиц с поддержкой клэй-технологии, построенной на "слипаниии" частей, подобно каплям жидкости в невесомости.

Итак, приступим! Создав в проекции **TOP Particle Blizzard**, перейдите во **FRONT** и поверните частицы на **180** градусов. Опять же в проекции **TOP** создайте **gravity** (в любом месте) и **MOTOR** (с центром посередине системы частиц), активируйте их (**Bind to SpaceWarp**). Просмотрите получившуюся анимацию.

Наполнив содержанием, переходим к форме — выбрав в настройках **Blizzard Particle type**, измените его на **MetaParticle**. Далее необходимо настроить параметры сетки, по которой из частиц будет выстраиваться результирующий объект. Наиболее оптимален такой вариант — для **ViewPort** половина размера частицы и четверть для визуализации. Далее для отображения в окнах

можно выбрать **MESH (default ticks)**. Но наберитесь терпения или, что еще лучше, измените **Emission** с **Rate** на **Total**, далее цифра зависит от мощности вашего компьютера.

Мы создали основную струю воды и крупные капли, таким же образом создаются мелкие брызги, с той лишь разницей, что используются обычные сферические частицы вместо **MetaParticles** для экономии ресурсов. Ниже приведены две картинки — одна использует обычные частицы с **Motion Blur** (16 минут на кадр), другая — моментальный снимок "настоящей" струи из метачастиц (6.5 часов на кадр при максимальной детализации).



### Больше зеркал, хороших и разных!

Зеркальные поверхности — один из особых предметов гордости Autodesk и Kinetix — ведь именно их 3DS'ка смогла быть не только трехмерным моделлером, но еще и крутой рендеринговой системой и рендерить такие вещи, как **bump**, отражения, прозрачность и т.п. До этого, конечно были системы типа POVR, но они, в основном, — рендереры, т.е. моделировать в них особенно нельзя.

Мы рассмотрим четыре основных способа создания зеркал, по возрастающей сложности.

#### Способ №1 — Reflect/Refract

Создадим сферу и сплющим ее так, чтобы она была похожа по форме на линзу. Теперь идем в **Material Editor** и создаем новый материал. Можете оставить все основные установки так, как они есть — вполне подойдет почти любой шейдинг, блеск и т.п. Наша цель — раздел **Maps** и пункт **Reflect** в нем. Нажмите на него, и из появившегося списка **New** выберите **Reflect** ➔ **Refract Map**. Можете не менять никаких установок, вернуться в основной материал и присвоить этот материал вашему объекту — сплющенной сфере. Все — рендерите.

Быстрый и простой метод. Но это единственное его достоинство. Дальше идут одни недостатки. Во-первых — это очень плохое качество изображения. Даже на той сжатой из 640x480 до 200x150 картинке вы можете

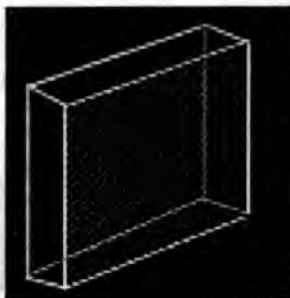


заметить ребристость границы — поверьте, она выглядит действительно отвратительно. Во-вторых, это качество изображения (количество пикселей) уменьшается с отхождением объекта от сферической формы. Для идеальной сферы этот способ очень привлекателен, но его ни в коем случае нельзя использовать на плоских объектах, и не рекомендуется на несферических объектах.

### Способ №2 — Flat Mirror

Этот способ применим только для идеально ровных поверхностей. Если у вас хотя бы на одну миллионную координата сдвинута — уже не сработает. К тому же, способ не так прост, как №1.

Итак, приступим. Создайте **Box** так, как показано на картинке.



Материал нам понадобится не простой, а многокомпонентный, а точнее двухкомпонентный. Создайте новый материал **Multi** ⇨ **Sub-Object**. Сделайте **Set Number** и укажите 2 компонента. Нам надо создать два простых материала: один — собственно зеркало, а другой — то, что будет как бы обрамлять его. Зеркало будет вторым материалом, а обрамление — первым — это важно.

## **Тонкости, хитрости и приемы работы**

---

Тыкаем мышкой во второй материал, если нужно — регулируем основные параметры, но самое главное — создаем **Flat Mirror Map** в разделе **Reflect Map** (так же, как и **Reflect ⇔ Refract**). Теперь самое интересное. Берете свой **Box**, и добавляете модификатор **Edit Mesh** (можно также сконвертировать **Box** в **Editable Mesh**). Теперь переключаете **Sub-Object** на **Face** и выделяете ту поверхность, что показана штриховкой на вышеприведенном рисунке.

Теперь ищите **Material ID** и вводите туда 2. Все — мы готовы к рендерингу.



Хорошее качество изображения — пожалуй, почти идеальное. Но только для идеально плоских поверхностей. Метод необходим там, где нужны точные, качественные и самое главное быстро читающиеся зеркала. По сравнению с рейтрейсингом, этот метод дает очень большой выигрыш в быстродействии.

### **Способ №3 — Raytrace**

**Raytrace** — это встроенный рейтрейсер 3DSMAX, доступный еще во второй версии. Рейтрейсинг или трассировка лучей — это способ построения изображения, при котором каждая точка поверхности, выбираемая с каким-либо шагом дискретизации испускает "луч",



который просчитывается по обычным физическим законам (преломления, отражения) и, таким образом, получается идеальная картинка. Минусы такого подхода тоже очевидны — очень низкая скорость по сравнению со всем остальным. Продемонстрируем его на не самом традиционном цилиндрическом зеркале. Сделайте цилиндр и расположите его примерно так, как показано на картинке.

Создайте новый материал и в **Reflection Map** задаем новый **map** — **Raytrace** (процесс аналогичен заданию **Reflect/Refract** из №1). Присвойте материал объекту и рендерите.

Идеальное качество изображений. Очень низкая скорость просчета — самая низкая из всех методов. Проблема в том, что встроенный рендерер МАХ'а очень и очень медленный по сравнению со своими аналогами.

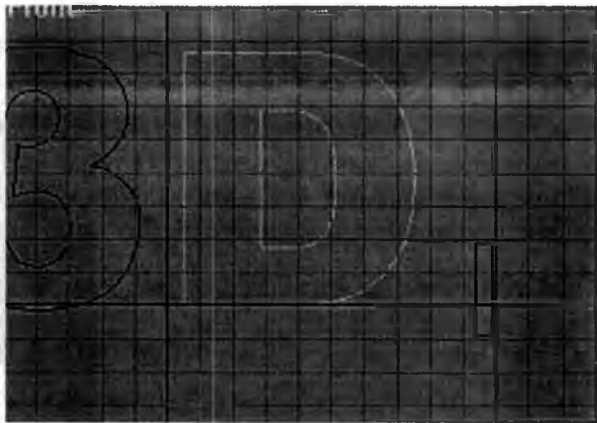


### Растущий текст

Это очень простой эффект, его знают практически все бывалые МАХ'еры. Построен он на том, что в 3DSМАХ'е можно анимировать все — даже настройки деформаций лофта. Попробуем сейчас анимацию такого растущего из ничего лофта простого текста.

Итак, первый шаг — создание сплайна текста. Здесь первая тонкость — сплайн должен быть замкнутым многоугольником (плавным или угловатым), а это значит,

что надо создавать каждую букву текста по отдельности (**Create** ⇒ **Spline** ⇒ **Text**). Для простоты создадим надпись из всего двух букв (вернее цифры и буквы) — "3D".



Но и тут уже кроется опасность! Буква "D" на самом деле состоит из двух сплайнов — внутреннего и внешнего! Для устранения этого недостатка буквы D, нам придется разделить ее на два разных сплайна. Выберите ее и во вкладке **Modify** примените модификатор **Edit Spline** (Редактирование сплайна). Теперь в **Sub-Object** выберите пункт **Spline** и выделите один из сплайнов буквы "D", который бы вы хотели отделить. Прокрутим вниз правую панель с кнопками и внизу увидим кнопку **Detach**. Смелее нажимайте и пишите название для нового объекта — отделенного сплайна!

Теперь надо создать сплайн-сечение, которое мы будем двигать по пути-тексту. Пойдя по пути наименьшего сопротивления, создадим простой прямоугольник (**Create ⇒ Spline ⇒ Rectangle**).

Все готово для лофтинга текста. Выберите один из сплайнов текста и сделайте **Create ⇒ Lofting ⇒ Loft**. Теперь нажмите **Get Shape** и выберите свое сечение — прямоугольник (если **Get Shape** не нажимается, то это означает, что вы где-то создали не одинарный сплайн, а двойной — такую букву, как например "D" или "O"). Одна буква есть! Прodelайте ту же операцию для каждого сплайна-пути и получите полный текст.



Осталось его проанимировать — это, в общем, не так сложно. Мы будем анимировать деформацию масштабирования. Изначально (на первом кадре анимации) у нас не должно быть видно ничего — т.е. по всей длине пути лофт должен иметь масштабирование до 0%. В конечном кадре должно быть все на 100%. А так как нам нужно, чтобы буквы как бы разматывались из

## Тонкости, хитрости и приемы работы

---

ленты, то будем двигать пределы этого перехода 100% в 0% от начала к концу пути по мере анимации.

Возьмите один из получившихся лофтов и перейдите к его вкладке **Modify**. В самом низу найдите **Deformations/Scale** и нажмите его. Теперь нажмите большую красную кнопку **Animate** на главном экране 3DSMAX и перемотайте движок кадров на начало вашей анимации. Добавьте на кривую деформации две точки с помощью команды **Insert Control Point** и расположите их, как показано на рисунке:



Теперь зафиксируйте второе положение точек в позиции конечного кадра анимации, как показано на этом рисунке:



Все! Анимация готова! Можете отрендерить и всем показывать! Вот кадр номер 50 из финальной анимации. Как обычно, можно скачать финальный вариант сцены.

## Anti-aliasing. Что это такое?

Сравните эти две картинки:

**ANTI-ALIAS|ANTI-ALIAS**

Чувствуете разницу? На картинке слева (без anti-alias'a) изображение выглядит ступенчатым, а на картинке справа (с anti-alias'ом) закругления букв обработаны серыми точками, что создает впечатление округлой формы, хотя на обеих картинках то же самое разрешение.

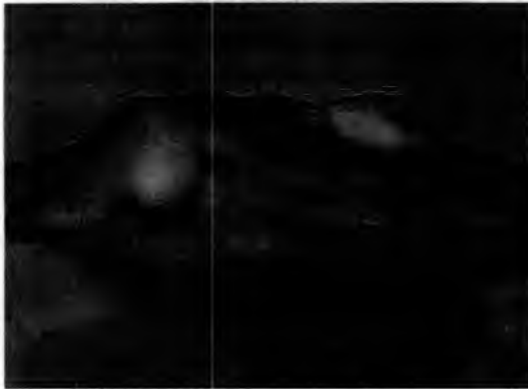
Вывод? Используйте **anti-alias** всегда для создания более плавных изображений — в 3D и 2D. Неплохо было бы иметь и anti-alias крупных шрифтов при просмотре web-страничек. Для этого вы можете использовать MS Plus! в котором есть функция **Smooth edges of screen fonts** или Windows 98, в которой эта функция встроена.

## Mountain Texturing

Для создания ландшафтов есть три простых способа:

### Способ №1

Нарисовать вид сверху в любом графическом редакторе, а потом с помощью **displacement'a** выдавить полученную текстуру на **box'e** с достаточной плотностью сетки. В этом случае яркость пикселей **grayscale**-карты определяет глубину, на которую та или иная часть **box'a** будет выдавлена. Если провести достаточно времени, то можно создать очень хороший ландшафт и, самое главное, он будет выглядеть именно так, как вы и хотели.



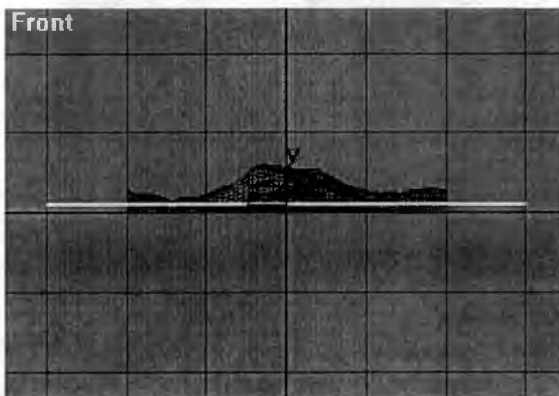
### Способ №2

Можно не рисовать ничего, а просто создать этот самый **box** и хорошенько потаскать его за вершины, не забывая **Affect Rigion** держать включенным или вообще взять **patch** или **NURBS** и разгуляться всерьез.

### Способ №3

Если рисовать не получается или времени нет, в таком случае надо поискать плагин, который все делает сам, благо для **MAX**'а он есть и может быть даже не один. Плагин **Mountain** как раз и создает подходящую поверхность. Но вот, горы есть и сразу возникает вторая проблема, а как их собственно оттекстурировать? Для первого способа всегда остается созданная карта, которую можно ограниченно использовать как маску или **blend**-карту, чтобы получить более естественный вид гор. А если мы любим баловаться плагинами и таскать **box**'ы и **patch**'и за уши, это что же, круглый сирота получается?

Если гора не нравится, то всегда можно поиграться с **Seed**'ом (генератором случайных чисел в плагине), и заделать себе такую, чтобы по душе была.



А теперь самое время создать немного дополнительной геометрии... В проекции **Top** создаем **box**, покрывающий нашу гору, как бы овцу.

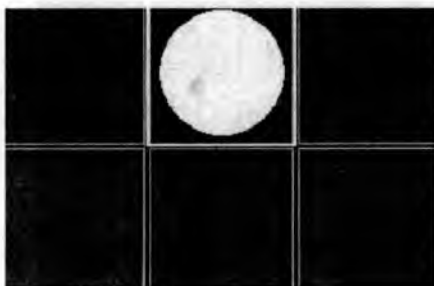


## Тонкости, хитрости и приемы работы

---

Сразу возникает вопрос, а зачем мы это делаем? А для того чтобы создать эти самые карты для масок, и **blend'ов**. Но перед основным шоу позвольте сделать еще несколько загадочных пасов руками. Создадим еще пару материалов...

То есть один материал совершенно черный и самосветящийся на 100%, второй материал настолько же бел, насколько черен первый и светится с не меньшей силой.

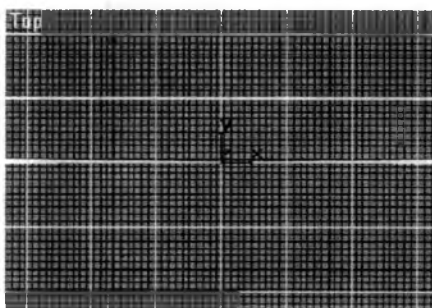


После этого присваиваем белый материал **box'у**, а черный — нашим скалистым горам.





Переходим в вид **Тор**, добиваемся с помощью **zoom'a** чтобы гора занимала весь вид по ширине. Зачем? Поскольку полученной картинкой в дальнейшем мы собираемся воспользоваться для наложения текстур на горы, то желательно чтобы стороны картинке были пропорциональны сторонам гор и если не отрегулировать вид, то потом предстоит немного поработать в Photoshop'e, урезая картинку до нужного размера. Рендерим в разрешении 300 на 300.



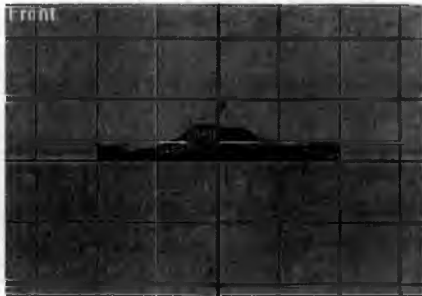
Спокойно! Это не шкура освежеванной коровы, а наша первая карта. Почему первая? Ну сами представьте, горы высокие, внизу травка растет, чуть повыше каменные стены до небес и на самом верху снежка напорошено.



## Тонкости, хитрости и приемы работы

Так вот, с полученной картой мы реализуем только травку и камни, а снежок-то и не выпадет, так что продолжаем...

Перемещаем **box** повыше, не забывая, что часть горы, которая возвышается над **box**'ом будет в снегу.



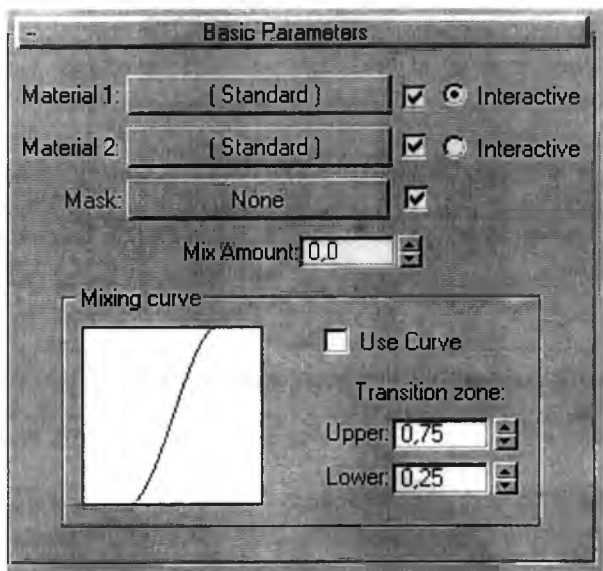
Когда подходящий уровень найден снова напрягаем CPU рендером...



Теперь у нас есть все необходимое, чтобы сделать красивую текстуру. Идем в **Material Editor** и начинаем химичить. Сначала изменим тип материала со **Standart**'а на **Blend**.



Первую маску засовываем в слот **Mask** как **Bitmap**, **Material 1** делаем красным, **Material 2** синим. Позже мы, конечно, заменим это красно-синее безобразия на настоящие текстуры, но для наглядности так лучше. Накладываем материал, рендерим...



Сложно предположить, что MAX скажет, после того как вы нажмете на кнопочку **Render**. Во-первых, он может хмуро заявить, что на вашем **Box'e**, то есть — горе, нет **mapping'a**, тогда бегите скорее в панель **Modify** и наложите **mapping** с помощью **UVW-Mapping** модификатора. Во-вторых, текстура может неправильно отображаться, тогда следует пойти в тот же модификатор **UVW-Mapping** и подкрутить **Gizmo**, на **Sub-Level'e**.

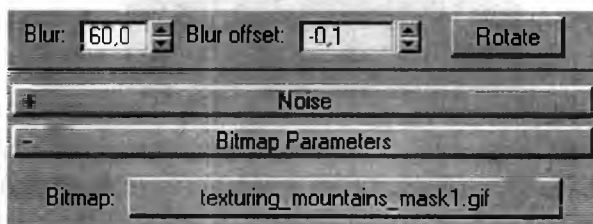
## Тонкости, хитрости и приемы работы

---

Резкая граница смотрится весьма неестественно, неплохо бы ее размыть.



Не стоит делать этого в Photoshop'е. В MAX'e получится не хуже. Заходим в слот **Mask** и в параметрах **Bitmap**'а изменяем значение **Blur**.



При значении 60 картинка заметно похорошела.



Помнится, еще был обещан снежок. Так вот, чтобы он выпал, надо поработать. Заходим в **Material 1** и меняем его тип со **Standart**'а на **Blend**. MAX предлагает сохранить нынешний красный материал как суб-материал — соглашайтесь. А дальше та же история, слоту **Mask** назначаем вторую маску, разблуриваем ее известным способом, в слоте **Material 2** создаем материал под цвет крокодила Гены. Рендерим.



А теперь осталось только заменить все эти красно-сине-зеленые несуразности на нормальные материалы. Вместо синенького — травку, вместо зеленого — камень, а красное на снежок заменить.



Метод можно расширять и усложнять.

Вместо трех материалов создать четыре, пять или даже больше, если надо отобразить различные скальные породы или просто разнообразить путь к вершине.

Вох'ы можно не только постепенно поднимать, но и наклонять на небольшой градус, чтобы граница не выглядела слишком горизонтальной.

Вспомогательные материалы (у нас в примере это белый и черный) можно сделать светло-серым и темно-серым соответственно, или добавить какую-нибудь текстуру в слот **diffuse** и должным образом уменьшить интенсивность **self-illumination**, тогда материалы будут просвечивать сквозь друг друга в самых неожиданных местах.

## Использование Photoshop для создания текстур

Работая с любой 3D-программой, часто встает вопрос о текстурировании объектов — в простейшем случае создании плоских изображений, которое бы проектировалось на нужный объект. Опишем несколько приемов, используемых для получения разных текстур в Photoshop.

### Помятость (bump)

Откроем новый файл и наделаем в нем черно-белых облаков. Для этого **reset**-ьте цвета (кнопка "D") и сделайте **Filter** ⇒ **Render** ⇒ **Clouds**.



Теперь на полученное применим фильтр **Distort** ⇒ **Ocean Ripple** со средним размером ряби и высокой силой. Увеличьте контраст изображения на 40-50 единиц (**Image** ⇒ **Adjust** ⇒ **Brightness-Contrast**) — и перед вами хорошая **bump**-карта для накладывания, например, для изображений помятости ткани или сморщенности и т.п.



### Расщелины (**bump, diffuse**)

Откройте новый файл, скажем 150 на 150. Возьмите два первоначальных цвета (черный и белый — кнопка "D") и примените **Filter** ⇒ **Noise** ⇒ **Add Noise** с параметром **Monochromatic** и уровнем около 75.



Теперь сделайте **Image** ⇒ **Adjust** ⇒ **Brightness-Contrast** и увеличьте контраст где-то на +50 или +55. Этот параметр подбирается на глаз, так, чтобы общий суммарный цвет всех точек примерно был на 50% серым. После этого нужно еще немножко разблюрить (**Filter** ⇒ **Blur** ⇒ **Blur**) картинку.



Все это теперь надо превратить в расщелины. Применим фильтр **Crystalize** со средним уровнем "кристаллизации" (например, 25).

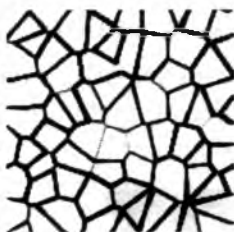




Ну, что, начинают уже проглядываться расщелины? Осталось немного — еще один фильтр. На этот раз — **Filter ⇒ Stylize ⇒ Find Edges**.



Ну вот практически и все. Осталось лишь сделать все расщелины более-менее равномерными. Сделайте **Image ⇒ Adjust ⇒ Levels**. Параметры такие: **Input Levels** — 194 1.00 255, а все остальное — оставьте, как есть. Все — можно использовать как **bump** или раскрасить текстурой, как **diffuse**. Одно “но” — эту текстуру очень сложно затайлить стандартными методами, так что советуем делать сразу столько, сколько нужно (пусть одна будет большая, типа 2000 на 2000, но чтобы не тайлилась).



### Пузырьки (bump, diffuse)

Ну, это совсем просто. Возьмите текстуру из предыдущего способа и сделайте **Filter ⇒ Blur ⇒**

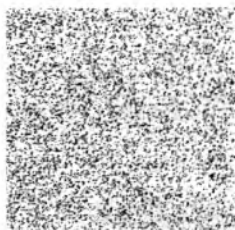
## Тонкости, хитрости и приемы работы

**Gaussian Blur** с параметром 4. Для bump'a сойдет и так, для диффуза раскрасьте как надо через **Image** ⇒ **Adjust** ⇒ **Hue-Saturation** с **Colorize**.

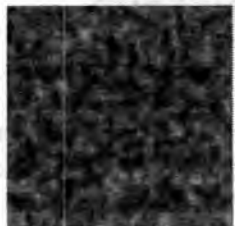


### **Каменистая поверхность (bump, diffuse)**

Создайте новый файл. Начнем с шума — традиционно, **Filter** ⇒ **Noise** ⇒ **Add Noise, Monochromatic**.



Теперь размытие — **Filter** ⇒ **Blur** ⇒ **Gaussian Blur**, коэффициент — 1-2.



И, наконец, выпуклости — **Filter** ⇒ **Stylize** ⇒ **Emboss**.  
Ориентировочные параметры:

**Height**=3,

**Amount**=100%.

Ну и для диффуза — раскрасить.



### **Дерево (diffuse)**

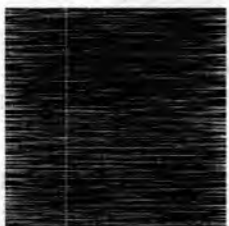
Создаем новую картинку, и, как всегда, добавляем шум — **Filter** ⇒ **Noise** ⇒ **Add Noise**. Лучше не монохроматичный.



Теперь делаем **Filter** ⇒ **Blur** ⇒ **Motion Blur** с расстоянием в 30-40 пикселей. Получились полосы, похожие на текстуру дерева, только очень размытые.



Делаем **Filter** ⇒ **Sharpen** ⇒ **Sharpen** для выделения текстуры дерева. Все — теперь осталось только раскрасить ее.

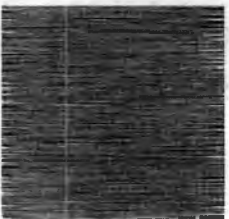


Раскрашиваем: **Image** ⇒ **Adjust** ⇒ **Hue-Saturation**, со включенной опцией **Colorize**, ориентируемые параметры:

**Hue** — 35,

**Saturation** — 60,

**Lightness** — +20.

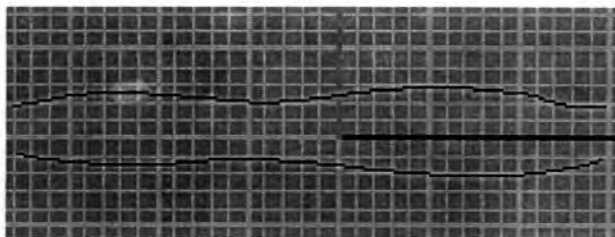


Получается что-то похожее на полированный паркет. Параметры дерева легко меняются, как параметры окраски. Осталось только отрезать края (там тоже получилась интересная текстура — текстура "расщепленного дерева" — если нужна — сохраните ее в отдельный файл), и затайлить все.

### Моделирование руки по 4 огибающим

**Rhino3D** — один из самых лучших NURBS-моделлеров. Этот совет посвящен моделированию в нем органической поверхности — руки. Он направлен на то, чтобы показать один из самых базовых методов построения поверхностей — построение по четырем огибающим.

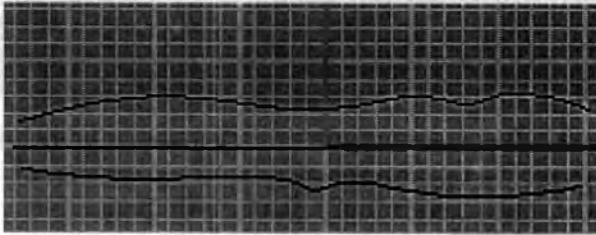
Приступим. Запустите **Rhino3D** и начните новый проект. Для начала нам нужны собственно 4 огибающие кривые для руки. Для их построения воспользуемся инструментом **Curve**, рисующем интерполированные кривые.



Выберите вид **Тор** и постройте две кривые, так, как показано на рисунке — они будут означать граничные значения будущей поверхности нашей руки. Сначала она

немного утолщается, потом утоньшается к локтю, и затем идет достаточно большой изгиб (только не переборщите) — это мускулы — бицепсы.

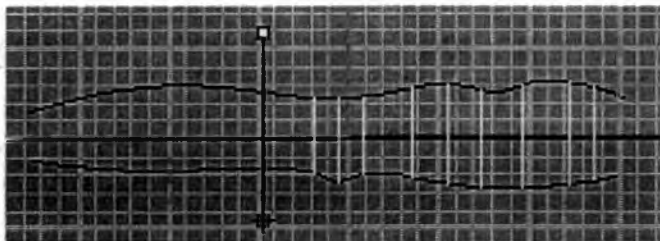
Теперь надо построить еще две кривые — ограничение по виду спереди. Перейдите в вид **Front** и постройте еще две кривые. Можете придать им некоторую бугристость — обычно мускулы в вертикальном положении напрягаются больше, чем в горизонтальном. И не забудьте про локоть!



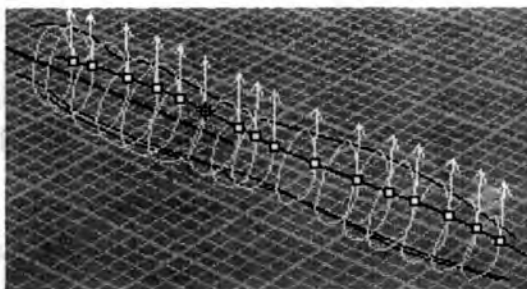
Все — творческая работа почти окончена. Теперь остальное — дело техники. Теперь нам надо построить много сечений нашей будущей поверхности и пролофтить их. **Rhino3D** выполняет построение сечений поверхности, заданной четырьмя огибающими без проблем — используйте команду **CSec Profile** из **Curve Tools**.

Теперь очень аккуратно выберите свои четыре кривые в следующей последовательности: передняя, верхняя, задняя, нижняя (в общем, в любой последовательности по, или против часовой стрелки). Для этого удобно использовать вид **Perspective**. Теперь нам надо указать, где проводить сечения — т.к. у нас форма достаточно сложная, лучше сделать это вручную. Сечения

проводятся, как линии в любом виде, хотя лучше всего использовать сейчас **Front**. Проведите их около десяти-пятнадцати, обязательно выделяя основные выпуклости и вогнутости поверхностей. Для проведения прямых линий, удобно зажимать **Shift** — тогда линии будут проводиться только параллельно сетке — как в Photoshop'e.



Закончив делать сечения (не слишком увлекайтесь!), нажмите **Enter**. Сечения уже выделены — осталось только сделать лофт. Он находится в **Surface** ⇒ **Loft**.



Теперь еще одна важная часть — надо чтобы векторы направления и точки приложения векторов были одинаковы для всех сечений. Если что-то не так — поправьте, подвинув вектор или изменив направление командой **Flip**. Есть еще команда **Auto**, которая делает все

## Тонкости, хитрости и приемы работы

---

автоматом, но не всегда правильно. Все — жмите **Enter** и в диалоге говорите **ОК**. (Проверьте, чтобы в диалоге **Closed Loft** был выключен!) Чтобы при будущих построениях эта куча сечений вам не мешалась — пока она выбрана, нажмите **Hide**.

Конечный результат — рендеринг.



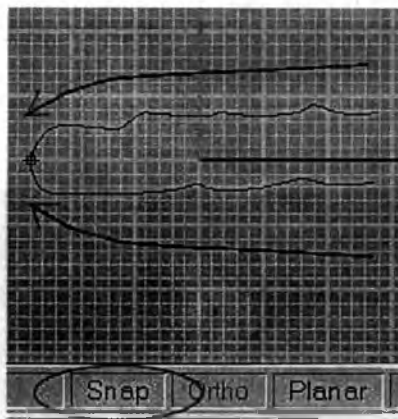
## Моделирование одного пальца руки

Здесь будет описан еще один распространенный метод — лофтинг огибающих.

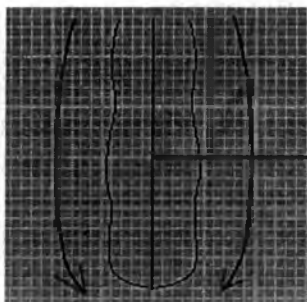
Итак, поехали. Нарисуйте инструментом **Curve** две кривые в виде **Right**. Главное — это чтобы их концы совпадали. Для этого проще всего воспользоваться **Snap**'ом — курсор будет ходить по клеточкам сетки. Следите также, чтобы вы вели обе кривые в одном направлении. Т.е. если вы заканчиваете первую кривую общей точкой со **Snap**'ом, то вторую ведите в том же



направлении (они показаны на рисунке красными стрелками).



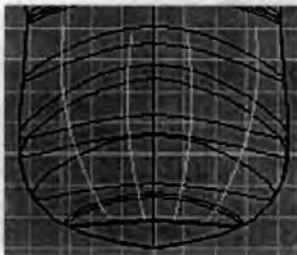
Теперь сделайте еще две кривые в виде **Top**. Они также должны смыкаться в одной и той же точке — опять используйте **Snap**. Так как сверху палец симметричен, вы можете воспользоваться инструментом **Mirror** для того, чтобы нарисовав лишь одну кривую, отзеркалить ее и получить вторую. Не забывайте также про направления! Ведите кривую в направлении, показанном стрелкой.



Теперь снимите все выделения и выполните команду **Loft**. Теперь по очереди (по/против часовой стрелки) в виде **Perspective** выберите все кривые. После выбора всех четырех, нажмите **Enter**. В появившемся окошке обязательно включите **Closed Loft**.

Уже почти все! Для ленивых — все. Палец готов, можете делать с ним все, что хочется. А для тех, кто не прочь сделать еще кое-что для реалистичности, почему бы не сделать этого. Не для кого не секрет, что на пальцах иногда растут такие роговые пластинки, которые люди в шутку называют ногтями. Попробуем их сделать.

На конце пальца (где вы, конечно, заранее предусмотрели место для ногтя), прочертите четыре кривые — ограничителя ногтя. Две из них можно даже отзеркалить. Выделите все четыре кривые. Теперь инструментом **Project** спроектируйте эти кривые на поверхность пальца.



Получилось 3 серии кривых: одна — изначальная — посередине пальца, вторая — лежащая сверху пальца и третья — снизу. Выберите и удалите первую и третью. То, что осталось, выберите по очереди (важно!) и выполните всю ту же самую банальную команду **Loft**. В окошке

опций уберите **Closed Loft** и, чтобы не перегружать сцену ненужными деталями, установите **Rebuild with 5 points**. Все!



Покрасьте получившуюся пластинку в какой-нибудь другой цвет, можете немножко поднять ее для видимости и отрендерите!



### **Моделирование полной руки — ладонь, пальцы, плечо, предплечье**

Итак, у нас уже есть палец и рука. Пора сделать полную руку — пальцы, ладонь, предплечье и плечо.

## Тонкости, хитрости и приемы работы

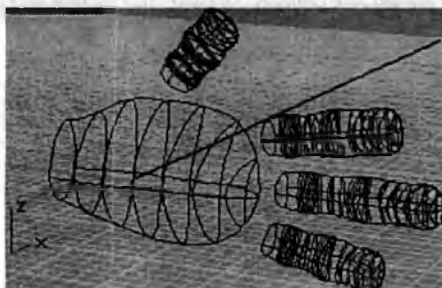
---

Задача поставлена — сделать ладонь, собрать ладонь и пальцы в кисть руки и присоединить кисть к предплечью.

Сначала — ладонь. Она выполняется тем же методом, что и палец (лотфинг огибающих). Единственное что — следите за направлением кривых, не забудьте про **Closed Loft** и для облегчения последующего редактирования сделайте **Rebuild with 8 points**.

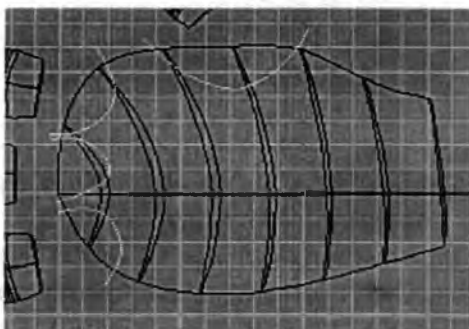


Теперь начинается самое интересное! Подгрузите ваш палец (**File ⇒ Merge**). Поверните его, если нужно, передвиньте и смасштабируйте так, чтобы он подходил к ладони. Теперь с помощью команд **Copy** и **Rotate** (с опцией "C" — **Copy**) размножьте свой палец. Каждый палец надо немножко изменить — повернуть, растянуть (**Scale-1D**) и т.п. Если у вас пальцы с ногтями — следите, чтобы вы двигали и трансформировали ногти вместе с пальцами. У вас должно получиться примерно вот это.



Пусть будет всего 4 пальца (ну не человек это будет, если кто еще не понял). Посмотрим на то, что у нас получилось. У нас есть 4 поверхности с "дырками" — пальцы и одна поверхность с одной дыркой для предплечья — ладонь. Чтобы присоединить пальцы к ладони, можно воспользоваться командой **Blend Surface**, но для нее нам нужны "дырки" с обеих сторон — и в пальцах, и в ладони. Ну, не беда — надо вырезать отверстия в ладони!

Проведите 4 кривые так, как показано на рисунке.

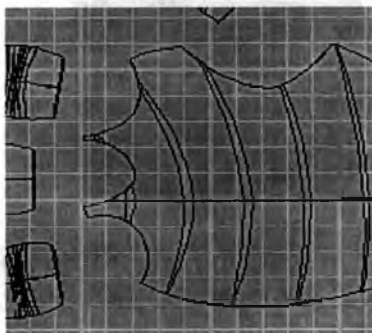


Этими кривыми мы как бы отрубим часть ладони. Они должны быть как можно более аккуратными и ни в коем случае не пересекаться, хотя и располагаться достаточно близко друг к другу.

Просто так отрезать ими мы не можем — нам надо, чтобы кривые лежали на поверхности объекта ладони. Спроектируем их на нее — команда **Project**. После проектирования ваши первоначально нарисованные кривые уже не нужны — можете удалить их.

Доктор сказал резать — значит резать! Настало время для команды **Split**, в ее основном применении. Нажмите эту кнопку, выберите ладонь как объект и все три кривые как **Cutting Objects** (режущие объекты). Жмите **Enter** и ждите. Эта операция — одна из самых долгих по времени, так что для сложных объектов порой приходится ждать несколько минут.

Все готово — нужные нам куски отрезаны. Выделите их и удалите. Режущие кривые нам теперь тоже не нужны — можете их спрятать или стереть. Дырки есть — можно использовать **Blend Surface**. Выполните эту команду (она находится в **Surface Tools**).



Теперь, после таких мучений, приставить предплечье к кисти руки вам не составит никакого труда — тем более, что дырки уже есть — осталось только сделать **Blend**.



## Как сделать правильную гайку с помощью лофта Rhino

Этот tutorial совсем не сложен. Он скорее направлен даже не на получение результата, а на получение некоего убеждения в мощности инструмента Rhino3D, имя которому **Loft**. Итак, мы будем рисовать гайку. Все и так умеют, да? Случайно не экструднутый сплайн, так как на рисунке?



Теперь идите, отыщите где-нибудь гайку и посмотрите на нее. Что, похоже? Не очень, правда? Ладно, приступим к рисованию правильной гайки.

Настоящая гайка — это лофт двух вещей — окружностей и описанных вокруг них шестиугольников. Начнем с окружности. С помощью простейшего инструмента **Circle** нарисуем окружность. Теперь выберем инструмент **Polygon** и введем следующие коррекции с клавиатуры:

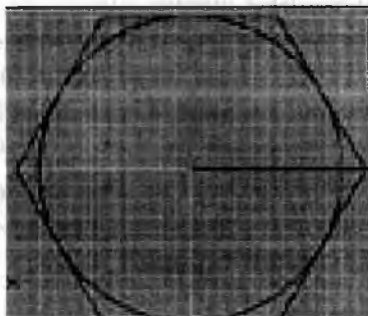
- **Num** — количество сторон;
- **6** — у нас шестиугольник;

## Тонкости, хитрости и приемы работы

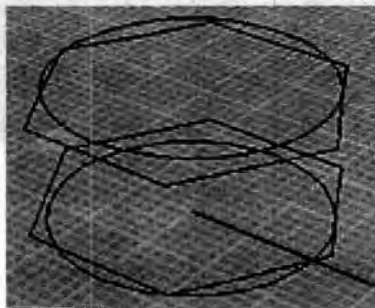
---

- **Circ** — наш шестиугольник.

Теперь постройте многоугольник, указав тот же центр, что и у окружности, и тот же радиус. У вас должно получиться то, что показано на рисунке.



Теперь надо откопировать (**Copy**) оба объекта и разместить их друг над другом, так, как показано на рисунке.



Теперь все готово к лофту.



Если хочется, то можно вставить еще несколько окружностей для внутренней части гайки и фасок, но сейчас для нас это несущественно, поэтому не будем на этом останавливаться. Выбираем **Loft** и выделяем окружность, шестиугольник, шестиугольник и окружность в порядке снизу вверх или сверху вниз. Для правильной подгонки линий связи проще всего использовать автоматику (**auto**). Теперь нам надо не плавную поверхность, а строгую, четкую и техничную, так что выбираем **Style: Straight Sections**. Все, лофтим.

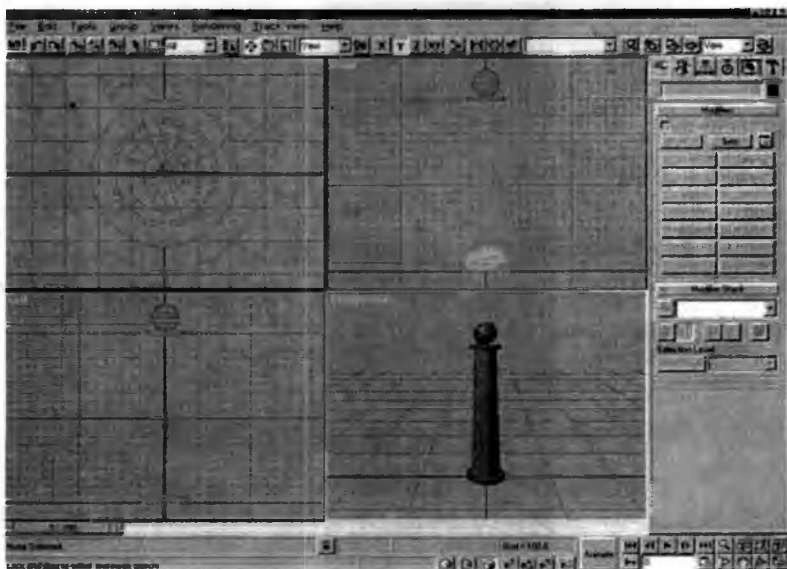
Теперь, если хочется, то **Cap Planar Holes** или переделайте все с окружностями и **Closed Loft**'ом. Вот и результат. Ну как? Больше похоже?



### 3D моделирование

3D моделирование — одно из самых интересных приложений в области компьютерных технологий. С помощью современных программ человек может фактически создать свой собственный мир трёхмерных образов и «дорабатывать» его по своему усмотрению. Ни один современный боевик, комедия или просто интересный кинофильм уже не представляется нам без серии потрясающих спецэффектов, которые заставляют многих не верить, что происходящее на экране нарисовано на компьютере, а не происходило на съёмочной площадке «взаправду».

Пусть нам хочется нарисовать простенькую колонну с шаром на вершине. Колонна у нас будет усечённым конусом с цоколем и опорой.



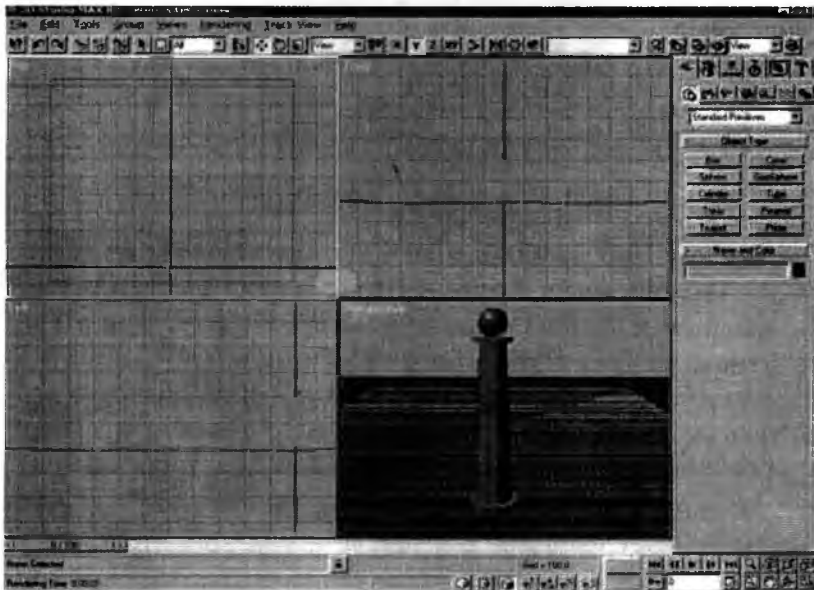
Как вы видите, 3D Studio MAX по умолчанию показывает 3 проекции — сверху, слева и спереди — и перспективу.

Однако в окне перспективы показывается очень некачественное, черновое изображение, предназначенное для максимально быстрой перерисовки. Многие возможности 3D Studio MAX и спецэффекты становятся видимы только после визуализации или рендеринга. Суть его в том, что из простейших примитивов, созданных вами в процессе моделирования, генерируется ряд поверхностей по весьма сложным алгоритмам — с учётом источников света, текстур, отражения, атмосферных эффектов строится фотореалистичное изображение. Попросим 3D Studio MAX визуализировать нам эту сцену с параметрами, принимаемыми по умолчанию.



Не правда ли, не слишком «фотореалистичное» изображение? Ну что же... Это только начало. На этом изображении цвета объектов оставлены по умолчанию и поэтому на границе колонны и основания видна неприятная граница, несмотря на то, что она слегка смягчена фильтром антиэйлиэсинга (**antialiasing**) 3D Studio MAX. Этот фильтр предупреждает резкие, «рублёные» кромки на границе двух объектов путём их взаимного размытия.

Логично предположить, что коколь, колонна и основание — единое целое. Для более качественного рендеринга с учётом этого, объединим до этого разные объекты путём создания **Compound** — составного объекта **Boolean**. Кроме того, добавим большой прямоугольник под колонну, олицетворяющий «почву».



Как мы видим, за счёт того, что в сцену был добавлен несоразмерно большой объект — прямоугольник, «основание» — видовые окна, значительно уменьшили свой масштаб из-за применения операции **Fit**, т.е. «подогнать» масштаб под размеры сцены. Перспективу же мы специально увеличили и сдвинули, чтобы последующая визуализация получилась более наглядной.



Ну вот, это изображение уже представляет из себя что-то. Использовался стандартный источник света, так как мы ещё не определили своих источников. При визуализации был сделан более качественный шейдинг, т.е. мягкий переход окраски граней объекта с учётом его освещённости. Однако, что и говорить, объект выглядит скучновато и не эффектно. Как можно это побороть?

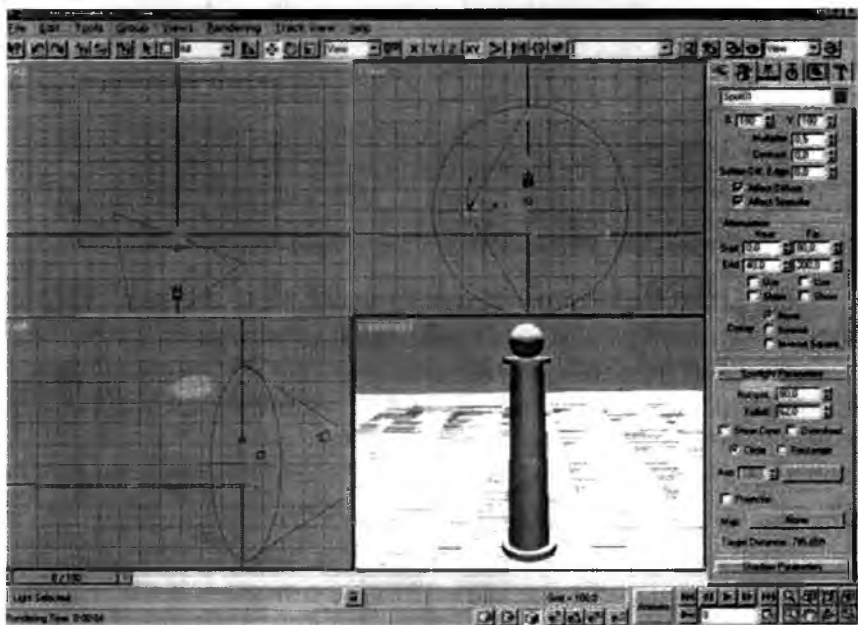
После того, как мы создали простую геометрическую модель, не грех выбрать для неё «сцену». Для этого мы сначала создадим камеру. Применяемый по умолчанию в

## Тонкости, хитрости и приемы работы

---

нижнем правом окне вид **Перспектива**, в принципе, неплох, но он не позволяет нам легко и быстро менять точку обзора. Если же мы создадим камеру и выберем её в окне перспективы, то меняя её расположение и ориентацию в других видах, сможем в реальном времени отслеживать изменения «пейзажа» в окне камеры.

Кроме того, создадим подсветку для сцены. Пусть колонна будет освещена мягким боковым светом, а её задний вид теряется в полутени. Это важно, так как мы не успели создать обстановки на заднем фоне и вид уходящего в никуда основания сцены будет выглядеть абсолютно непривлекательно.



Не обращайте внимания на «засвеченность» правого нижнего вида — дело в том, что MAX некорректно отображает освещённость сцены в окне предварительного просмотра при наличии сложных источников света.



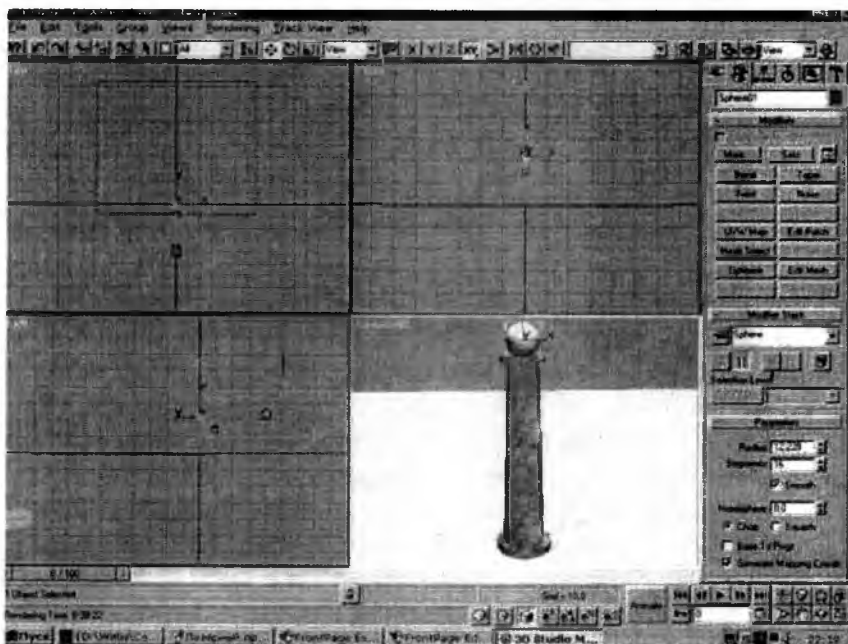
Ну вот, уже что-то. К сожалению, приходится записывать изображение с большими потерями в качестве, поэтому обратите внимание, что полосам на «полу» сцены мы обязаны не визуализатору MAX, а погрешностям формата JPEG.

Но где мы видели ярко-зелёную колонну с фиолетовым шаром на ней? Разве что на абстрактных полотнах. Все поверхности в реальной жизни представляют из себя сложный и слабо поддающийся математическому описанию набор цветов. Поэтому, хотя в библиотеке MAX есть «процедурные» материалы, во

## Тонкости, хитрости и приемы работы

---

многих случаях дизайнер просто сканирует реальную поверхность с хорошим качеством, а потом «натягивает» полученную текстуру на поверхность. Параметры этого наложения можно варьировать, в результате чего получаем поверхности, очень близкие к реальным. Выберем для пола текстуру травы с небольшими неровностями, а для колонны — текстуру мрамора с мягкими тенями. Шар пускай будет отсвечивать серебристым цветом.



Обратите внимание, что MAX может показывать текстуры и в окне предварительного просмотра. Надо заметить, что делает он это не очень корректно, поэтому этот уровень визуализации годится только для



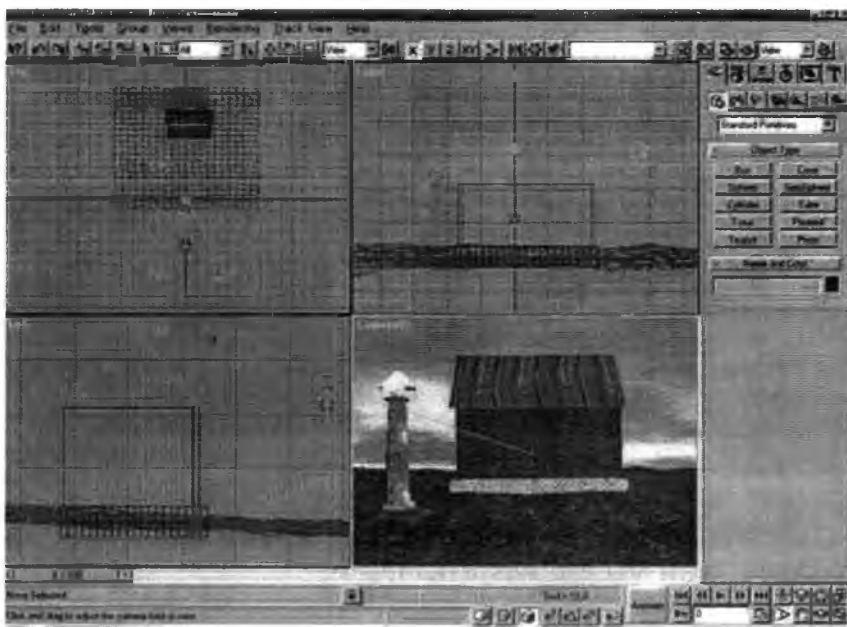
приблизительной прикидки. Вы можете исправить положение, установив на свой компьютер профессиональную видеокарту, ускоряющую отображение OpenGL-сцен. К тому же подобная карта будет визуализировать предварительный вид сцены намного быстрее, так что вы сможете даже на полном экране просмотреть будущий видеоклип с довольно неплохим качеством. А вот на скорость окончательной визуализации специальная видеокарта не влияет никак.



Ну вот, это сцена уже годится для того, чтобы распечатать её на струйном принтере, полюбоваться секунд 20 и выкинуть в корзину. Больше на ней рассматривать нечего. Любая сцена нуждается в заднем плане. У нас же на заднем плане — зияющая пустота. Как бы сделать так, чтобы взгляд зрителя имел какой-то фон и

колонна не была такой одинокой? Немного пораскинув мозгами, идём дальше!

Кроме того, придумаем фон, немного трансформируем нашу колонну. Из колонны она станет уличным фонарём, освещающим траву перед домиком. Сам домик пока только приблизительно набросаем, не создавая подробных деталей. Кроме того, слегка изомнём «траву», создавая иллюзию неровной полянки.



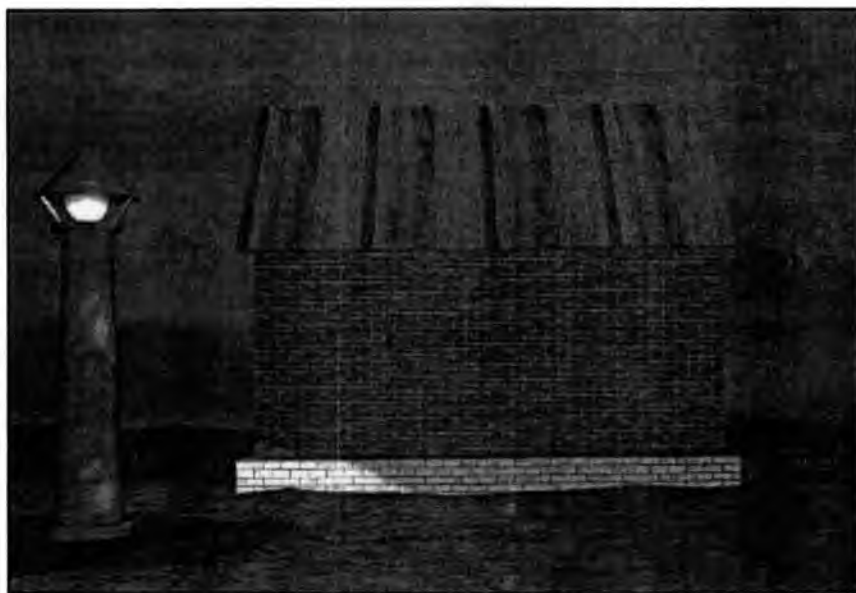
Как видим, 3D Studio MAX умеет создавать достаточно наглядное изображение в окне камеры (да и в любом другом). Оно вполне годится для предварительного

просмотра сцены. Правда, из-за того, что в окне просмотра неправильно учитываются источники свет (точнее, они обсчитываются слишком примитивно), сцена выглядит излишне светлой. Кроме того, никаких теней и иных световых эффектов вы в окне не увидите... Зато на достаточно мощном процессоре или с помощью специализированной видеоплаты, вы сможете просматривать анимацию в большом окне чуть ли не в режиме Real-time.

Для того чтобы подобрать лучшее соотношение между качеством предварительного изображения и эффективностью работы, можно выбирать из различных режимов качества отображения.



Естественно, изображение будет значительно приятнее, если мы его визуализируем.

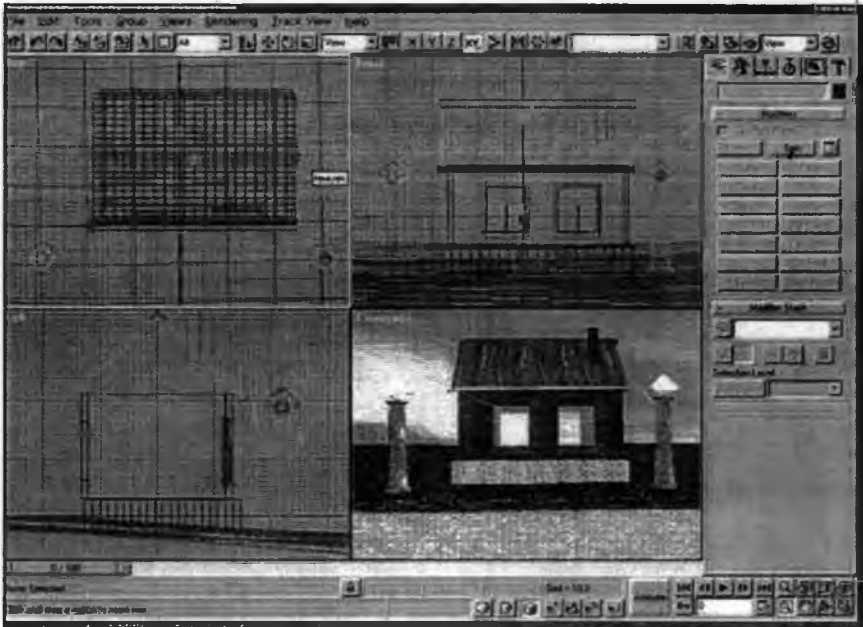


Не правда ли, не слишком похоже на картинку в окне просмотра. Добавились световые эффекты от фонаря, сама картинка стала более тёмной. Ну и, конечно, текстуры визуализировались с гораздо лучшим качеством. Конечно, домик пока напоминает хибарку дядюшки Тома, но ничего — мы его достроим и разукрасим. Но это всё.

В составе следующей сцены применены в основном материалы типа **Bitmap**. То есть, с помощью сканированного растра мы пытаемся изобразить материалы реального мира.

Но вещи обычно имеют, кроме цвета и фактуры поверхности, ещё большое количество свойств — таких, как прозрачность, отражение, преломление, пропускание света и так далее.

Попробуем повысить реалистичность нашей сцены, используя в ней материалы с этими свойствами.



Во первых, более подробно детализируем сцену. Домику стоит прерисовать конёк на крыше, окна с подоконниками, трубу, и добавить вторую колонну, она же — фонарь для симметрии. Кроме того, пусть домик у нас будет стоять на берегу озера.

Первый из материалов с расширенными свойствами — это материал занавесок и оконного стекла. Занавески сделаны наполовину прозрачными, а стекло — прозрачным практически полностью.

В результате мы можем видеть свет, льющийся из окна на берег и траву.

Воду сделаем чуть прозрачной, дабы немного видны были элементы дна. Кроме того, применим очень интересный, но крайне ресурсоёмкий приём — создание отражающего свет материала.

Как известно, все материалы отражают свет, кроме абсолютно чёрных тел, но такими в реальном мире являются, вроде, только звёзды — «карлики». Кроме того, они этот свет также поглощают и пропускают. Все эти параметры различны для света с разной длиной волн. Для того, чтобы синтезировать такие эффекты в пакете трёхмерной графики, используется методика трассировки лучей — **ray tracing**. Её суть состоит в том, что мы рассматриваем отдельно каждый луч света, испускаемый источником. При этом для каждого встретившегося предмета, исходя из заданных пользователем типов материалов, вычисляем коэффициент пропускания, отражения и поглощения (вообще-то, есть ещё ряд других, но менее заметных свойств). После этого продолжаем отслеживать путь модифицированного луча, до тех пор, пока дальнейшее отслеживание не станет ненужным. Количество шагов трассировки лучей может быть задано пользователем, как параметр подсистемы визуализации. К сожалению, этот метод, необходимый для создания фотореалистичных световых эффектов, очень ресурсоёмкий и, зачастую, увеличивает время визуализации во много раз.

Спускаясь с высот на землю, заметим, что в 3D Studio MAX есть специальная карта материала — **Flat Mirror**, предназначенная для создания плоских зеркальных

поверхностей. Пусть у нас вода будет спокойной — вечер всё-таки... Поставим коэффициент отражения в 25%. Кстати, чтобы ощутить эффект от всех этих прозрачностей и отражений, следует заставить все основные источники света отбрасывать тень и установить режим обсчёта теней в **Raytraced**.

Также для использования отражения следует в параметрах визуализации установить флажок **Auto Reflect/Refract Maps**. По завершении всех этих действий и выполнения визуализации, получаем следующий результат:



Не правда ли, отражение домика и фонарей в воде придают этой сцене реалистичный и интересный вид?

К сожалению, время визуализации увеличилось ни много ни мало — в 10 раз. Красота требует жертв.

В следующем шаге мы применим и другие «продвинутые» материалы 3D Studio MAX для придания нашей сцене ещё большей реалистичности и законченности.

В составе 3D Studio MAX имеются инструменты, придающие дополнительную реалистичность изображению. Это так называемые атмосферные эффекты. Они, в отличие от геометрических примитивов, представляют из себя скорее растровый эффект, хотя их параметры и задаются аналитически.

Первый такой эффект — это туман. Подробнее с ним можно ознакомиться здесь. Есть и его разновидность — трёхмерный туман, имеющий больше параметров настройки.

В состав 3D Studio MAX 3.0 включен интересный эффект под общим названием **Combustion**. Он не имеет какой-то определённой характеристики, так как при желании, с его помощью можно смоделировать весьма широкий класс объектов — от пламени пожара до взрыва бомбы или дыма от костра.





Это один из примеров применения эффекта. На самом деле, **Combustion** имеет большое количество параметров, действие которых редко можно точно предугадать. Чаще приходится подбирать значения разных коэффициентов, добиваясь желаемого эффекта.

И, наконец, один из самых интересных эффектов — объёмный свет — **Volume Light**. Представьте себе, что сквозь узкое отверстие в тёмной пыльной комнате светит лучик солнца и вы поймёте механизм действия этого эффекта.

Вот пример применения такого света в нашей сцене:



Если вы заметили, фонари теперь не только отбрасывают тени, но и виден их свет, отражаемый пылью в воздухе.

Используя возможность 3D Studio MAX синтезировать цветной объёмный свет, можно создавать интересные динамические ролики, где **Volume Light** будет исходить от вывесок, источников света и прочих светящихся объектов.

## Как сделать красивый 3D текст

Сначала нам следует создать плоский, «обычный» текст — таким вы видите его в своём любимом текстовом редакторе. На рисунке показано расположение примитива «текст» на панели 3DSMAX.



Нажмите кнопку **Text**. В раскрывшемся свитке выберите параметры создаваемого объекта, в том числе шрифт, размер шрифта, выравнивание, стиль и кернинг.

В результате вы получите плоский текст, отображаемый в видовом окне белыми линиями. Но это не то, что мы хотели, правда? Для создания настоящего текста следует применить операцию **Extrude**. Видели, как кондитер выдавливает на торт из специального шприца крем? Суть модификатора **Extrude** весьма близка этому. Укажите толщину создаваемого объекта. Не стоит делать его слишком «толстым» — это плохо скажется на его виде после визуализации.



Вот теперь это полноценный трёхмерный объект, к тому же достаточно сложный. Не стоит перегружать сцену текстом, так как требования к памяти при этом возрастут весьма заметно.



Конечно, такое изображение выглядит скучновато. Добавим источник света и наложим на текст текстуру:



Ну вот. Теперь раскрасьте это изображение, как вам нравится. Соль и сахар по вкусу!

### Как сделать лампу

#### Черновое построение.

На проекции **Front** сплайнами (**Create** ⇒ **Shapes** ⇒ **Splines** ⇒ **Line**) строим заготовки для объектов: ножка лампы, патрон, лампочка, абажур, часть держателя абажура. Для простоты построения делаем все сегменты прямолинейными.

#### Уточнение формы сплайнов.

Уточняем форму каждого сплайна. Для этого выбираем сплайн, и на закладке **Modify** выставляем

## Тонкости, хитрости и приемы работы

---

параметры группы **Interpolation** для удаления угловатостей сегментов сплайна. При нажатой кнопке **Sub-Object** в режиме **Vertex** редактируем форму сплайна.

**Совет:** В первую очередь выделить все вертексы, щелкнуть на любом выбранном вертексе правой кнопкой мыши и установить тип вертекса **Smooth**. А затем **Bezier** или **Bezier-Corner**. Это обычно облегчает дальнейшее редактирование формы сплайна.

### Создание из сплайнов объемных объектов.

Для каждого объекта, кроме части держателя абажура, применяем модификатор **Lathe**. Сразу после применения модификатора необходимо установить нужное выравнивание оси вращения (и/или направление оси).



В данном случае для всех объектов хватает нажать кнопку **Align:Min**. Необходимо установить чекбокс **Weld Core** для слияния вертексов у полюсов объекта. **Flip Normals** — при необходимости. **Segments** можно увеличить для уменьшения угловатости объектов (особенно для крупных деталей).

Для создания части держателя абажура будем использовать **Loft** (**Create** ⇨ **Geometry** ⇨ **Loft Object** ⇨ **Loft**).

Построенный сплайн будем использовать как путь для лофт-объекта. Необходимо еще построить сплайн-сечение объекта: **Create** ⇒ **Shapes** ⇒ **Splines** ⇒ **Circle** с небольшим радиусом.

Выбираем сплайн, который будет путем, нажимаем кнопку **Loft**, нажимаем кнопку **Get Shape** и указываем курсором на объект-сечение, т.е. окружность. Для более мягкой формы в параметрах лофт-объекта устанавливаем **Skin Parameters** ⇒ **Options** ⇒ **Path Steps 10**. Для того, чтобы видеть на всех видах лофт-объект полностью отмечаем чекбокс **Display** ⇒ **Skin**.

### Создание держателя абажура.

Для клонирования объекта поворотом, необходимо правильно установить **pivot-point** этого объекта.

Отмечаем построенную часть держателя абажура. На закладке **Hierarchy** (кнопка **Pivot** нажата) нажимаем кнопку **Affect Pivot Only** и совмещаем ставший доступным **pivot** в виде **Top** с осью вращения лампы. Отжимаем кнопку **Affect Pivot Only**.

Клонируем объект. При активном виде **Top** в **toolbar** нажимаем кнопку **Array** и в появившемся диалоге устанавливаем: вращение по оси **Z** на **120** градусов, **Array Dimensoins 1D Count=3**, **Type of Object — Instance**.

Теперь достраиваем два куска трубы: крепление к ножке лампы и кольцо, на котором лежит абажур.



### Пэтчевое моделирование

Перед тем как приступить непосредственно к моделированию, надо понять, что требуется для построения модели.

Из инструментов нам понадобятся **Line (Spline)**, **Quad Patch** и модификатор **Edit Patch**.

#### **Анализ**

Для быстрого и правильного построения модели необходимо совершенно четко себе представить, что вы хотите получить в итоге. Т.е. закрыть глаза и представить в уме моделируемый объект. Посмотреть на него с разных сторон. Постараться выяснить закономерности его формы.



Возможно расчленить объект на более простые составные части, которые без ущерба для окончательного результата, могут быть созданы как отдельные объекты.

Является ли объект симметричным (осе-, зеркально- и пр.), содержит ли подобные части? Если да, то вполне вероятно, что можно вначале построить только часть объекта, а остальные части получить с помощью того или иного вида клонирования, возможно, с последующей доводкой откопированной части. Например, при моделировании кисти руки можно непосредственно моделировать только два пальца: «большой» и любой другой. Дальнейшие рассуждения относятся к объекту целиком (если он не содержит повторяющихся частей) или к части объекта.

Далее необходимо в уме проанализировать форму объекта. Провести по его поверхности характерные линии. Эти линии должны служить как бы каркасом объекта, помещенного в координатное пространство (центр объекта совпадает с началом координат). Какие линии будут характерными?

- Линии сечения объекта координатными плоскостями.
- Линии, ограничивающие отверстия.

Часто характерными линиями могут являться линии, проведенные по выпуклостям или вогнутостям объекта. На этапе освоения рекомендовано использовать их. Однако эти элементы поверхности могут быть смоделированы и другими способами.

### Синтез

Линии МАХ'а (сплайны) строятся по опорным точкам (вертексам). И вертексы занимают одну из ключевых позиций в данной методике. Поэтому после того, как каркас из характерных линий построен, надо, хотя бы примерно, представить, в каких местах этих линий будут опорные точки и сколько их должно быть. Полезно также попытаться соединить вертексы соседних характерных линий дополнительными линиями, чтобы получился более детализированный каркас. Каждая ячейка этого каркаса должна представлять собой криволинейный трех- или четырехугольник. Это важно. Если это требование не выполняется, то необходимо достроить вспомогательные линии. Если получившийся каркас имеет много треугольных ячеек, то он построен неверно. Треугольные ячейки должны использоваться только в случае необходимости, т.к. осложняют построение и окончательную доводку модели. Попробуйте построить каркас немного по другому и/или добавить вертексы в характерные линии. Желательно, чтобы ячейка не имела резких (больше чем на 90 градусов по одной из осей) перегибов.

Почему ячейки каркаса должны иметь три или четыре угла? Потому что в МАХ'е используются два вида patch'ей: треугольный и четырехугольный. Каждая ячейка каркаса должна быть закрыта одним patch'ем.

Чем линий каркаса будет больше, тем точнее будет модель, и тем больше времени потребуется на моделирование (временные затраты возрастают нелинейно).

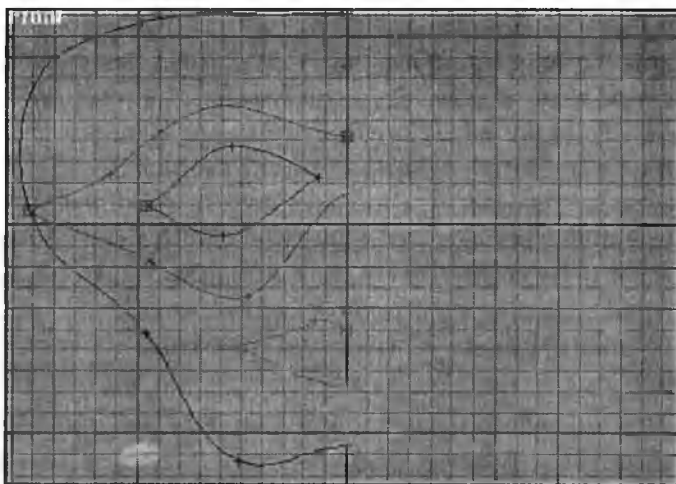
Теперь можно приступать непосредственно к моделированию.

Для примера построим модель некоей маски.

### Шаг 1. Выявление закономерностей

Маска является симметричной относительно плоскости  $YZ$ . Следовательно, можно строить только половину (допустим, левую).

### Шаг 2. Построение характерных линий

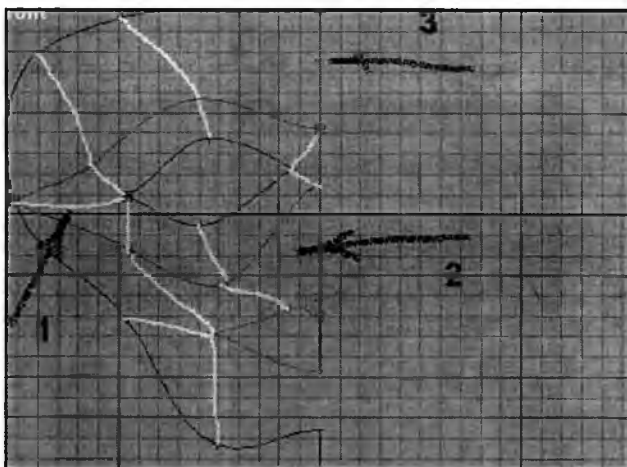


На виде спереди и слева строим сплайнами характерные линии. При регулировке сплайнов обращаем внимание на то, чтобы «правые» (на виде спереди) вертексы имели координату  $X$ , равную 0 (для облегчения в дальнейшем сопряжения половинок).

## Тонкости, хитрости и приемы работы

---

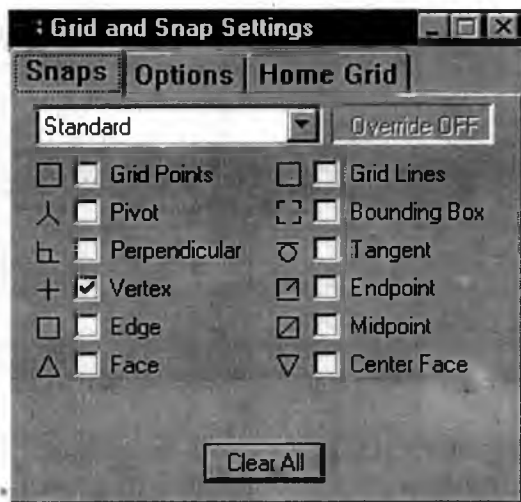
Мы выбрали, в качестве характерных линий, линии сечения координатными плоскостями, линии, ограничивающие рот (губы) и глаз, а также линии переносица-бровь-висок и переносица-скула-висок.



Далее окончательно регулируем сплайны. При построении линий и при дальнейшем построении модели удобно пользоваться «прилипалкой» (Snap).



Режим «прилипания» переключается с клавиатуры клавишей **S**. Перед использованием необходимо указать, к каким объектам будет прилипать курсор. Для этого на кнопке **Snap Toggle** необходимо щелкнуть правой клавишей мыши. Появится диалог настройки. Для наших целей необходимо установить прилипание к вертексам.



Для каждого сплайна (можно групповой операцией) устанавливаем флажок **Vertex Ticks** в разделе **Display Properties**, закладки **Display**.

### Шаг 3. Создание Patch-поверхности

Создаем простой прямоугольный patch (**Quad Patch**) **1x1**, размером примерно равным ячейке каркаса над глазом (вид спереди). Сдвигаем его (на виде слева) примерно на уровень той ячейки, которую он должен закрывать.

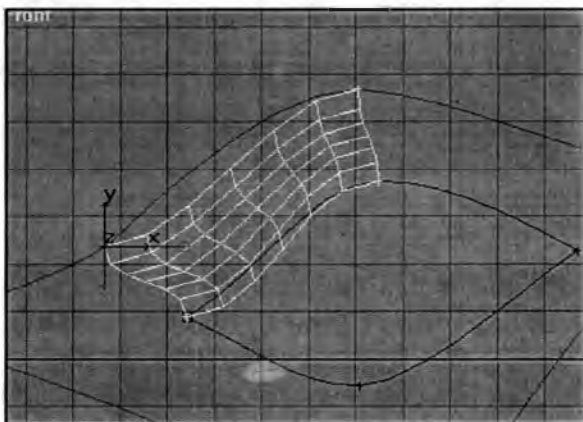
Если «прилипалка» еще не настроена, то это необходимо сделать сейчас.

Работаем в режиме **3D Snap** (прилипание к вертексам).

Накладываем модификатор **Edit Patch**.

### А

**Sub-Object — Vertex.** Дополнительно запретим показ каркаса patch'a: сбрасываем флаг **Lattice** в группе **Display** раздела **Edit Vertex**. Включаем «прилипалку» (клавиша **S**). Беремся по очереди за каждый вертекс patch'a и «прилипаем» его к соответствующему вертексу характерных линий.

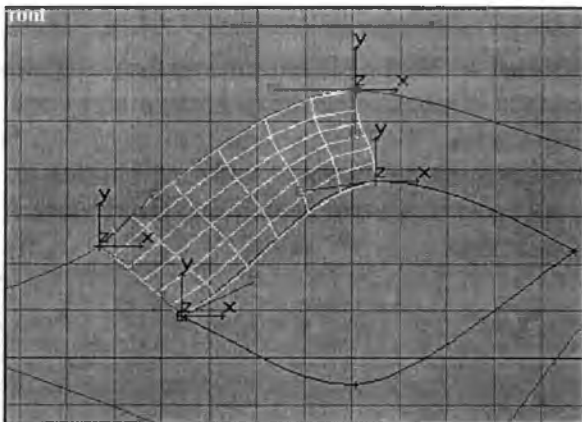


### В

Запрещаем прилипание (**S**) и «в черновую» регулируем форму patch'a тангентами. Общее правило: тангенты соседних вертексов (по общей стороне) не должны перекрываться в одной плоскости.

Также желательно, чтобы для соседних вертексов тангенты доходили до  $1/3$  стороны.

Окончательная регулировка формы поверхности тангентами осуществляется после всех «пристыковок» к данному и соседним с ним вертексам.



### В

Устанавливаем **Sub-Object Edge** (ребро, сторона), выбираем одну из сторон (при отключенном **Display Lattice** просто перемещается триплет координат, а при включенном, соответствующий элемент каркаса (**Lattice**) подсвечивается красным) и наращиваем еще один patch кнопкой **Add Quad** (или **Add Tri** для треугольного).

### Г

Переходим к пункту А. При стыковке patch'ей, наращенных на различных ребрах, необходимо объединять (**Weld**) вертексы. Для этого, при **Sub-Object Vertex** надо областью выделить оба вертекса и нажать кнопку **Weld**.

Обычно увеличивается параметр **Weld Treshold** для гарантированного срабатывания. При объединении вертексов объединяются и соответствующие ребра, что исключает щели между соседними patch'ами.

Если на выделенном вертексе шелкнуть правой кнопкой мыши, то появится горячее меню. В нем будут два пункта, отвечающие за тип вертекса: **Coplanar** и **Corner**. Эти типы вертексов patch'a соответствуют типам вертексов сплайна **Bezier** и **Bezier-Corner**.

### **Шаг 4. Окончательная модель**

Для этого установим **Pivot Point** в начало координат и отзеркалим созданный patch-объект как **Instance**.

Теперь видны допущенные ошибки. На крае половинки маски (по центру) не точно выставлены тангенты: заметен шов. От внешнего уголка глаза идут «морщины» — надо дополнительно регулировать вертексы. Недостаточна проработка на уровне характерных линий.





Для окончательного варианта необходимо довести половинку маски до нормального состояния, приаттачить вторую половинку к первой и объединить «пограничные» вертексы. При этой операции, кстати, уберется шов между половинками.

Вот, в основном, и все. Подробности использования инструментов можно найти в штатном хелпе.

### **Дополнительные замечания**

Характерные линии и каркас из характерных линий можно строить исходя из различных соображений. Можно, например, брать набор сечений объекта плоскостью или двумя, или тремя ортогональными плоскостями. Можно дотраивать вспомогательные линии сплайнами, как и характерные. Иногда это помогает, иногда начинают мешать лишние линии и вертексы при регулировке patch'ей.

Теоретически, можно вообще обойтись без каркаса из сплайнов.

## **Метод раздельного освещения и затенения сцены**

Крупные части изображения, не обязательно 3D, их внешний вид, композиция действуют на сознание зрителя, а мелкие — на подсознание. Картинка «смотрится» тогда, когда мелкие ее детали не вызывают у человека подсознательного конфликта с тем, что он привык видеть. То есть, человек может даже не понимать, что конкретно его смущает. Задача художника — учитывать все мелочи при построении изображения. При моделировании (особенно если нужно добиться «фотореализма») огромное

значение на качество имиджа играет правильное освещение сцены. В реальной жизни любой предмет освещается далеко не одним источником света. Первоначально их может быть один, два... (скажем, солнце, лампа), но свет от них, попадая на предметы, окружающие моделируемый, отражается (рассеивается) и, тем самым, мы получаем еще множество источников. И поэтому добиться требуемого результата, манипулируя источниками света, которые одновременно и освещают, и создают тени — нелегко, а в некоторых случаях — невозможно. Гораздо удобнее сначала осветить предмет так, чтобы он выглядел «как живой», а затем, независимо от освещения (!), создать нужные тени. 3D Studio MAX позволяет это сделать достаточно просто.

А теперь — суть метода:

- Освещаем предмет (сцену) традиционно. (Только у источников света не включаем **Cast Shadows**).
- Создаем источник света. Например — Spot01.
- Копируем его так, чтобы копия осталась в том же месте, что и оригинал (!).
- Далее, в Spot01 включаем **Cast Shadows**.
- В Spot02 **multiplier** устанавливаем отрицательным, но равным по модулю **multiplier** в Spot01.

Теперь все параметры Spot01 компенсируются Spot02, кроме теней. Мы получили сложный источник света (Spot01+Spot02), который создает тени и не влияет на освещенность сцены! Для удобства можно «подлинковать» оба источника к Dummy-объекту. А

используя **Exclude/Include** можно добиться влияния источников только на данный объект(ы).

Тут есть большое поле для экспериментов. Например, можно и в Spot02 включить **Cast Shadows**, но в Spot01 поставить **Sample Range=5** (к примеру), а в Spot02 — равным 0. Тогда мы получим только контур тени объекта. И так далее...

Кто-то может спросить: «К чему все это — ведь в **Exclude/Include** есть опция **Exclude/Include cast shadows**». Действительно — есть, но при этом источник света продолжает освещать предмет (сцену), т.е. чтобы не менять освещенности — одним источником все-равно не обойтись.

Кроме того, все написанное выше — не догма, а просто совет. Не обязательно им всегда пользоваться, к каждой конкретной ситуации нужно подходить по-своему. 3D Studio MAX хорош тем, что один и тот же результат можно получить 1000 способами. Был предложен еще один... Надеемся, что это поможет кому-нибудь в работе.

# Расширения 3D MAX

Вся система 3DSMAX3 (и MAX1) состоит из plug-in'ов. Что такое plug-in? **Plug-in** — это программный модуль 3DSMAX3, который позволяет добавить какую-то новую функцию в основную программу. Основная программа 3DSMAX3 (так называемая **core**) писалась с расчетом на то, что все действия будут делать plug-in'ы.

Взглянем поближе на директорию **3DSMAX3** (будем считать, что она установлена у вас в \3DSMAX3). Вы увидите три поддиректории, относящиеся к plug-in'ам:

- **STDPLUGS** — в этой директории хранятся так называемые стандартные plug-in'ы. Они служат, например, для рисования примитивов, импорта файлов для текстур и т.п.
- **PLUGINS** — здесь хранятся нестандартные plug-in'ы, например, написанные пользователями. Они позволяют расширить диапазон функций 3DSMAX3.
- **PLUGCFG** — здесь обычно лежат файлы конфигурации plug-in'ов. В основном туда их кладут стандартные plug-in'ы.

Как выглядят plug-in'ы? Plug-in поставляется в виде одного (реже — нескольких) файлов в расширениями **.DL?** и некоторыми другими. Это переименованный обычный DLL Windows. По расширению, данному файлу можно определить тип plug-in'a:

- **BMF** — (BitMap Filter) Фильтры файлов графики.

- **BMI** — (BitMap Import) Поддержка форматов файлов графики.
- **DLC** — (Controller) Анимационный контроллер.
- **DLE** — (Export) Экспорт сцен в другие форматы.
- **DLF** — (Font) Возможность использовать другие типы шрифтов.
- **DLI** — (Import) Импорт сцен из других форматов.
- **DLM** — (Modifier) Новый модификатор.
- **DLO** — (Object) Создает объекты.
- **DLR** — (Render) Рендеринг и объекты рендеринга.
- **DLS** — (Shape) Все, связанное со сплайнами.
- **DLT** — (Texture) Процедурные текстуры и material editor.
- **DLU** — (Utility) MAX-утилиты.
- **FLT** — (FiLTer) VideoPost-фильтр.

И в конце — маленькая неприятность. Дело в том, что почти все plug-in'ы для MAX1 или MAX2 не подходят к MAX3 (хотя есть исключения) и наоборот.

### Plug-in Shag: Fur

Основной отличительной чертой этого «генератора волос» является его скорость. Дело в том, что он не генерирует вообще никакой геометрии. Без него волосы обычно делаются либо созданием **вручную/полуавтоматически (Compound Scatter)** геометрии волос, либо хитрыми системами партиклов.

Первое — достаточно сложный и трудоемкий процесс, создающий полумиллионифейсные сцены, которые, к сожалению, пока не каждый может себе позволить. А хитрая система систем партиклов плоха тем, что практически невозможно выполнить анимацию полученной модели — партиклы-то движутся, и остановить их без дополнительных plug-in'ов не так-то просто. Да и сами партиклы тоже достаточно прожорливы в смысле железа.

**Shag: Fur** же делает свои волосы не за счет геометрии, а как «атмосферный эффект», примерно так же, как всем известный **combustion**. Используя специальные **Hair-Enabled Lights** (свои собственные источники освещения) это plug-in достигает даже эффекта тени от волос на волосатом объекте.



С помощью использования специальных векторов, возможна настройка таких сложных опций, как подгибание волос — ведь не у всех же волосы стоят торчком. Этими же векторами можно уложить волосы так, чтобы у шерсти было направление — т.е. «по шерсти» и «против шерсти». И все это, естественно, можно анимировать.

**Sub-Material**'ы в качестве параметров также существенно облегчают работу и улучшают результаты — например, можно назначить меппинг **Noise** и на объекте частично проступят «проплешины». Материал волос, длину, плотность распыления — все можно регулировать.



У **Shag: Fur** куча применений. Не считая тривиальных — волосы и шерсть — это трава, листва деревьев, вообще всевозможная растительность, оригинальные эффекты (вместо обычного **bump** можно сделать что-то действительно объемное), при небольшом желании с его помощью можно даже моделировать какие-то объекты (например, когти у злобного пришельца получатся очень интересными, если их сделать не монолитной геометрией, а из жесткой лоснящейся щетины).

К сожалению, эти эффекты не так легко даются. Основные концепции (особенно векторы) даются относительно сложно для понимания, а если учесть, что хотя **Shag: Fur** и очень быстрый, но простая сцена рендерится все равно долго (где-то минуты 2-3), то на обучение уходит где-то с пол-дня. Тutorials, идущие с

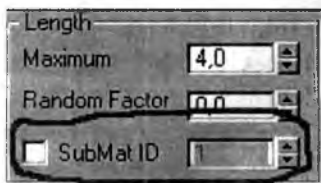
plug-in'ом, достаточно примитивны и не показывают самой сути.

Можно сказать, что в целом plug-in получился очень хорошим, и был бы просто отличным, если бы не некоторая сложность в обучении и использовании — ну, а это на самом деле объяснимо тем, что никто до этого ни с чем подобным не сталкивался и, соответственно, общаешься с «новыми технологиями». Plug-in'у можно обещать долгую жизнь, хорошую продаваемость и множество применений.

### Волосатая дубинка

Предположим, у нас есть дубинка, хорошая дубинка, но хотелось бы, чтоб она была волосатой! Никак не иначе, а именно чтобы самый «дубинистый» конец был просто лохматым и темным, а «недубинистый» — светлым и почти бритым! Да еще и чтоб на волосы было похоже! Представили? Значит пора браться за дело!

Загрузим plug-in **Shag: Fur** и назначим в качестве **furry object** нашу неудавшуюся палицу. Естественно — будем «шерстить» целиком. Но сначала остановитесь и внимательно взгляните на изображение. Особенно стоит задуматься над сочетанием, показанным ниже.





Как вы думаете, что это? Правильно! Именно здесь незадачливый программист заложил возможность присвоения текстуры параметру, просто объекту надо назначить мультиматериал, **diffuse** которых будет использоваться в качестве текстуры. Оттуда же можно вытащить и цвет волос. Приступим!

Перейдите в **Material Editor** и измените тип материала дубинки на **Multi-SubObject**, сохранив сам материал дубинки. Проследите, чтобы объект имел на всей поверхности материал с **ID1**. Теперь прикинем — на **ID2** положим длину волос, на **ID3** — плотность, на **ID4** — материал. Все, что осталось сделать — это положить **gradient** на **diffuse** в **ID12** и настроить **ID3**. Обратите внимание, что волосам более присущ металлический блеск.

Осталось лишь подогнуть волосы. Для этого создайте **Lean Vectors (Create ⇒ Helper ⇒ ShagFur ⇒ Vector)** так, чтобы максимальная зона влияния покрывала треть дубинки, а минимальная с запасом покрывала все целиком. Назначьте этот вектор в качестве **Lean Direction** и добавьте немного **Bend**. Отрегулировав параметры, стоит все же сконвертировать свет в **Hair Enabled** (это делается в модуле **Shag:Render/Tools**) и посмотреть результат.



## Вопросы и ответы

### Как обойти проблему несовместимости форматов при импорте файлов Adobe Illustrator в 3DS MAX?

Загвоздка в формате текста, а не самого AI. Многие программы сбрасывают AI в юниковом формате — т.е. в конце строки стоит только код 0x0A, а MAX желает в досовском — т.е. в конце строки должно стоять 0x0D, 0x0A. Проблему можно решать различными способами.

1. Открываем в **MultiEdit for Windows**, загружаем файл AI с указанием формата Unix, сохраняем как (**Save As**) файл в формате DOS.

2. Открываем AI-файл в редакторе ДООС Навигатора. Сохраняем с `_другим_именем`.

3. Пишем сами на любом языке программирования или ищем простенькую утилиту, которая производит вышеописанное преобразование.

### Как сделать зеркало?

В MAX'е зеркальная поверхность делается 3-мя способами: один для плоской поверхности, другой для криволинейной, третий общий (но тормозной).

1. Для криволинейной поверхности делаем стандартный материал с картой отражения (**Reflect/Refract**) на **Reflect** (100%).

2. Для плоской поверхности в общем случае делается **Multi/Sub Object** материал с 2-мя подматериалами. Один подматериал — любой стандартный (не зеркальный), другой — стандартный с картой **Flat Mirror** на **Diffuse**.

Далее работаешь как обычно с **Multi/Sub Object** материалом. Только не советуем на смежные некопланарные фэйсы ставить зеркальный подматериал — могут вылезти глюки на картинке.

3. Рейтрейсовый материал (MAX 2+) — самый тормозной, но и дающий самые реалистичные результаты. В редакторе материалов выбирается тип материала **Raytrace**. В простейшем случае требуют настройки 2 параметра: **Diffuse** и **Reflect** (степень отражения).

### Пакетный рендер. Как поставить несколько проектов в очередь для просчета?

Необходимо настроить сетевой рендер. После этого будет возможно ставить проекты в очередь на обсчет (все как при сетевом рендере), но доступная рабочая станция будет в сети только одна.

### Как настроить сетевой рендер?

Сетевой рендер возможен при работе под Windows NT, Windows 95, а также в смешанной сети NT-95. Также возможен обсчет очереди проектов на одной станции.

Распараллеливание просчета одного проекта по сети может быть выполненном только при генерации последовательности кадров. Каждый отдельный кадр будет

целиком считаться на одной станции. AVI, FLI и прочие анимационные файлы будут просчитываться целиком на одной станции.

Для настройки сетевого рендера необходимо выполнить следующие шаги:

1. Установить и настроить TCP/IP. Необходимо прописать IP-адреса для всех рендер-станций в локальной сети. Рекомендуются серии 192.100.100.\*, 192.168.100.\*, где «\*» — число от 1 до 255. IP-адреса станций в сети должны быть уникальными. Маска подсети 255.255.255.0.

Для Windows 95. В файле **WINDOWS\hosts** (можно откопировать **hosts.sam**) прописать соответствие IP-адресов именам станций (по строке для каждой станции), например:

**192.168.100.1 StationA**

**192.168.100.2 StationB**

Для Windows NT 4.0 соответствующий файл:

**WINDOWS\system32\drivers\etc\Hosts.**

2. Удалить все файлы и подкаталоги из каталога **MAX\NetWork** или создать такой каталог в случае его отсутствия.

3. При сетевом рендере необходимо иметь одну управляющую станцию-менеджер и одну или более станций-серверов (на которых будет производиться обсчет). Станция-менеджер может совмещать функции рендер-сервера (возможность очереди проектов на одной станции!).

На станции-менеджере необходимо запустить программу **MAX\Manager.exe**.

При первом запуске возможна только установка параметров (оставляем все по умолчанию). При повторном запуске будет выдано окно с логом и кнопки настройки. (Для MAX 1.x **MAX\manager.exe -d**).

На станции-сервере необходимо запустить программу **MAX\Server.exe**. При первом запуске возможна только установка параметров. В поле **Manager name or IP-address** вписываем имя станции-менеджера, остальное оставляем по умолчанию. При повторном запуске будет выдано окно с логом и кнопки настройки. (Для MAX 1.x **MAX\server.exe -d**).

Следует помнить, что все сетевые установки MAX'a на всех станциях в сети должны совпадать.

4. Проект, предназначенный для сетевого рендера, не должен быть привязанным к той станции, на которой он был создан. Каждая станция-сервер получает копию проекта. Поэтому, по возможности, пути должны быть в UNC, т.е. **\\Station\Volume\Directory\File.Ext**. Исключения составляют те пути, которые совпадают для всех станций-серверов (например путь к стандартной библиотеке карт, ссылка на которую разрешается через диалог **Preferences** в MAX).

Проект обязательно должен быть сохранен перед сетевым рендером.

5. Для сетевого рендера на всех станциях-серверах необходимо запустить **MAX\Server.exe**. На станции-менеджере необходимо запустить (в добавок к **server.exe**) **MAX\Manager.exe**.

Запускаем MAX, подготавливаем проект. В диалоге **Render Scene** отмечаем чекбокс **NetRender** и нажимаем кнопку **Render**. В следующем диалоге (**Network Job**

**Assignment**) в редактируемом выпадающем списке **Network Manager** вписываем имя станции-менеджера и нажимаем кнопку **Connect**. После этого в списке, расположенном ниже, появится список имен станций-серверов. В этом списке необходимо отметить те станции, которые должны считать данный проект. Чекбокс **Inactive** отложит выполнение просчета до активизации задачи в **Queue Manager**'е (**MAX\QueueMan.exe**).

Подробное описание настройки сетевого рендера содержится в **Help**'е.

### Как настроить Glow в MAX?

Эффект в видеопосте. Для его работы необходимо установить канал материала (в редакторе материалов кнопка с цифрой под сэмплами и/или канала объекта. Плагин (стандартный) **Lens Effects (apollo.flr)**.

1. Установить канал объекта отличным от 0 (0 — по умолчанию).

2. В видеопосте кнопки **Add Scene Event** (добавить просчет сцены); **Add Image Filter Event** (добавить фильтр для изображения) в диалоге, в выпадающем списке выбрать **Lens Effects Glow**, кнопка **Setup** — настройка эффекта; **Add Image Output Event** — добавить запись картинки в файл.

3. Настройка эффекта. Там все достаточно прозрачно. Кнопка **Preview** разрешает показ. Кнопка **VP Queue** заставляет показывать в **preview** реальную сцену из очереди видеопоста (в противном случае тестовая картинка **MAX**).

Все настройки анимируются. Ниже — некоторые основные установки.

### Страница **Properties**

- **Object ID** — тот, который был установлен в п.1;
- **Material ID** — канал эффектов материала (устанавливается в редакторе материалов кнопкой **Material Effects Channel**).

### Страница **Preferences**

- **Size** — размер эффекта свечения;
- **Color: Gradient** — свечение будет генерироваться в соответствии с установками, сделанными на странице **Gradients**; **Pixel** — свечение будет рассчитываться, исходя из цвета пиксела объекта; **User** — свечение будет рассчитываться, исходя из указанного цвета;
- **Intensity** — плотность эффекта.

Есть еще плагины, реализующие эффект **Glow**:

- **Glow+**
- **RealLensFlare**

## **Bomb** и его параметры. Как «взорвать» объект?

Для встраивания **Bomb** в объект необходимо нажать кнопку **Bind to Space Wrap** (в toolbar), отселектировать **Bomb** и, не отпуская левой клавиши мыши, дотянуть пунктирную линию до нужного объекта. При удачном встраивании в стеке объекта, появится строка **MeshBomb Binding**.

**Bomb** разрывает объект по фейсам. Для лучшего эффекта необходимо назначить объекту 2-х сторонний материал и желательно наличие достаточно большого количества фейсов.

«Сила» **Bomb** действует от иконки бомбы во все стороны, с учетом значения параметра **Gravity** (сила тяжести, направленная вниз по мировой оси **Z** при положительных значениях).

Параметры:

- **Strength** — сила «взрыва» — скорость разлета осколков.
- **Spin** — вращение осколков.
- **Falloff (Falloff On)** — сферическая зона «захвата» фейсов. Фейсы объекта, не попадающие в эту зону «взрываться» не будут.
- **Fragment Size: Min, Max** — минимальное и максимальное количество фейсов в одном «осколке».
- **Gravity** — сила тяжести.
  - **0** — невесомость;
  - **>0** — осколки будут падать вниз по мировой оси **Z**;
  - **<0** — вверх.
- **Chaos** — степень хаотичности разлета осколков.
- **Detonation** — время (по таймлайну) начала взрыва. До этого момента бомба не работает.
- **See** — начальное значение для датчика случайных чисел.



## Где устанавливается пивот?

Закладка **Hierarchy** командной панели (справа), кнопка режима — **Pivot**.

Кнопка **Affect Pivot Only** позволяет перемещать и вращать **pivot**, не трогая самого объекта. Кнопка **Affect Object Only** позволяет перемещать, вращать, масштабировать объект, не трогая **pivot**. Кнопка **Reset Pivot** возвращает **pivot** к первоначальному виду.

При работе с **pivot** работает функция **Align** (кнопка в toolbar).

## Как разным частям одного объекта назначить различные материалы?

Для 3DS MAX всех версий:

1. В редакторе материалов необходимо сделать материал типа **Multi/Sub-Object**. В настройках материала указать необходимое количество подматериалов (там по умолчанию 6 или 10 в зависимости от версии). После этого надо настроить каждый подматериал.

Порядковый номер подматериала является ID'ом.

2. Назначить объекту этот материал.

3. Применить к объекту модификатор **Edit Mesh** ⇒ **Faces**. Для каждого подматериала, выбрать все грани, которые должны иметь одинаковый материал, и в разделе **Material** в строке ввода ID указать порядковый номер подматериала.

### Как в доске прорезать надпись?

1. Рисуем замкнутый сплайн (например, квадрат).
2. В той же плоскости делаем текст.
3. Выбираем сплайн, **EditSpline**, отжимаем кнопку **Sub-Object**, ждем **Attach** и указываем на текст. Теперь у нас получился один объект из двух сплайнов.
4. Нажимаем **Sub-Object** и выбираем **Spline** в выпадающем списке.
5. Селектируем первый сплайн (должен стать красным) нажимаем кнопку соответствующей булевой операции, ждем кнопку **Boolean** и выбираем второй сплайн.

### Как отрендерить объект с отбрасываемой тенью на невидимый объект?

То есть нужно получить картинку с альфа каналом, с объектом и тенью от него (по альфа каналу) на допустим якобы имеющуюся стенку за объектом, при этом на картинке этой стенки не должно быть.

В качестве «стенки» используем бокс или патч с материалом **Matte/Shadow**. В редакторе материалов параметры:

- **Matte: Opaque Alpha** — непрозрачность по альфе — отключить.
  - **Shadow: Receive Shadows** — принимать тени — включить.
  - **Affect Alpha** — тени по альфе — включить.
- Остальные параметры — по необходимости.

## Чем LightWave лучше 3D MAX?

Тем, что LightWave бедноват интерфейсом...

В отличие от глобальных пакетов типа 3DMax и Softimage LightWave изначально был ориентирован на применение в видео производстве и использование людьми различного технического уровня.

Внешняя простота LightWave маскирует исключительно мощный и надежный инструмент. Среди основных достоинств пакета следует особо отметить качество просчета, нетребовательность к ресурсам и, самое главное, эффективный интерфейс. Стоило бы заметить, что при сравнении пакетов люди зачастую обращают основное внимание на специфические функции и возможности, забывая, что в основе своей работа с трехмерной графикой требует применения методов и приемов, практически одинаковых во всех системах. И в отличие от многих, LightWave предоставляет возможность исключительно быстро и удобно манипулировать инструментами для создания и модификации объектов.

Большинство функций вызываются с клавиатуры, и позволяют как интерактивное применение, так и задание числовых параметров, панель которых также вызывается одной кнопкой...

Также LightWave содержит несколько уникальных функций, с помощью которых создание сложных форм становится исключительно простой задачей. В частности MetaForm, инструмент для создания сглаженных моделей, по мощности использования несравним со стандартным сплайновым моделированием.

Для исключительного качества просчета LightWave использует достаточно простой метод **Antialias**. В точках, где цветовая разница превышает порог, задаваемый **Adaptive Sampling, LightWave** просто просчитывает участок вокруг точки с каждой стороны, затем интерполируя цветовое значение. При уровне **Antialias Low** обсчет происходит по одному разу с каждой стороны, соответственно два раза при **Medium**, и четыре раза при **High** уровне. Количество проходов при этом соответственно: 5 (1+4) при **Low**, 9 (1+8) при **Medium**, 17 (1+16) при **High**.

Соответственно время счета одного кадра при использовании **Antialias** зависит от количества точек, требующих сглаживания, и может быть в четыре (low antialias)/восемь (medium) /шестнадцать (high) раз больше чем при отсутствии antialias.

### **3D MAX работает только под 32М?**

Смотря что называть работой... В **readme** написано, что он медленно работает, если стоит небольшой размер **swap**. Рекомендуется поставить его минимально на **100 Мб**. Идеально было бы иметь своп в 200...300 Мб... Неплохо иметь еще и 3D видеоакселератор, так как сам интерфейс перегружен элементами, а их еще надо прорисовывать.

### **Как делать анимацию?**

Жмешь кнопку **Render** вот и все... можно анимировать, если непонятно, то, к примеру, можно объект сдвинуть...

## Как бомбу использовать?

Подложить (**create bomb**) под объект (**link to**), зажечь шнурок (**modifi bomb # frame nj explode**) и наслаждаться разрушением. Гравитацию можно добавить по вкусу.

## Как спрятать объект?

В **Track View** выбираешь объект, кликаешь в иконку с глазиком (**Add Visibility Track** называется) и добавляете ключи. Переходите в **Function Curves** и редактируете: меньше **0** — объект невидим, больше — видим. Все очень просто!

# Приложения

## Ссылки на ресурсы Internet

**<http://www.cray.onego.ru/3d>**

Русский 3D Центр. Все о 3D: вебринг, статьи, советы, галерея, тесты железа.

**<http://www.sim.ru/lightwave3d>**

Русскоязычный сайт по LightWave.

**<http://www.newtechniques.com/TekTicker>**

Ежедневные новости от журнала NewTechniques.

**<http://www.animationartist.com>**

Он-лайнный журнал Animation Artist. Обновляется ежедневно: 2d/3d новости, туторы, интервью, книги...

**<http://www.digitalproducer.com>**

Он-лайнный журнал Digital Producer.

**<http://www.3dup.com>**

Мощный специализированный на 3d поисковик. Сюда попал из-за своего раздела Spotlights, обзорающего последние 3d новости.

**<http://www.VFXPro.com>**

VFXPro — ежедневные новости в области визуальных эффектов (FX): новый софт, хард, фильмы, новости компаний, предложения работы.

**<http://www.3dartist.com>**

Он-лайнновый вариант журнала «3D Artists». Основное направление — технологии создания трехмерных сцен, обсуждение новых пакетов и инструментов.

**<http://www.cgw.com>**

Он-лайнновый вариант журнала «Computer Graphics World», отдающий предпочтение таким темам, как CAD, анимация, визуализация и мультимедиа.

**<http://www.serious3d.com>**

Журнал профессиональных туторов «Serious 3D». Все очень интересно, но практически все только за деньги.

**<http://www.visualmagic.awn.com>**

Журнал визуальных эффектов «Visual Magic». Периодически отсюда можно скачать (в PDF формате) полную версию одноименного бумажного журнала.

**<http://www.625-net.com>**

Журнал «625». Не совсем о 3d, но зато по-русски.

**<http://www.3dcafe.com>**

Один из самых популярных 3d сайтов, где можно найти очень много полезных вещей. Все, что можно взять бесплатно (туторы, текстуры, модели, шрифты, звуки...) в разделе FREE STUFF.

**<http://www.3dark.com>**

Еще один ведущий 3d сайт. Масса разнообразной информации по 3d.

**<http://www.members.xoom.com/surender99>**

Сайт поддержки турнира SU.RENDER 99 между подписчиками эхоконференции FIDO SU.RENDER.

**<http://www.raph.com/3dartists>**

Галерея «3d Artists», самая большая 3d галерея.

**<http://www.avalon1.viewpoint.com>**

Одна из самых больших библиотек моделей, текстур и утилит.

**<http://www.3dcafe.com>**

Еще одна большая библиотека моделей.

**<http://www.meshmart.org>**

И еще одна библиотека.

**<http://www.boardhost.com/boards/sherlock>**

Место, где можно обменяться своими моделями. Что-то похожее на Fido: пишете свою заявку (Что-то вроде — Очень нужна модель пятиногого динозавра) и может быть вам кто-нибудь и поможет.



## Список использованной литературы

**Ландшафт**

Михаил Якшин

**Все о 3D**

<http://cray.onego.ru/3d/>

**3D Studio MAX R3**

Сергей Гашников

**IntelliMouse и 3D Studio MAX**

Сергей Гашников

**Mountain Texturing**

Юрий Стоцкий

**Текстура ландшафта**

Михаил Якшин

**Море**

Виталий Башкатов

**Взрыв**

Михаил Шатилов

**Лучи лазера**

Михаил Якшин

**Создание фонтана**

Виталий Башкатов

**Зеркальные поверхности**

Михаил Якшин

### **Растущий текст**

Михаил Якшин

### **3D Studio MAX**

Виталий Башкатов

### **Anti-aliasing**

Михаил Якшин

### **Plug-in'ы 3DSMAX2**

Михаил Якшин

### **Волосатая дубинка**

Виталий Башкатов

### **Shag: Fur**

Михаил Якшин

### **Создание текстур**

Михаил Якшин

### **Моделирование руки по 4 огибающим**

Михаил Якшин

### **NURBS-Палец**

Михаил Якшин

### **NURBS-Полная рука**

Михаил Якшин

### **Сплайновое моделирование**

Журнал «625-net news» ([reclama@625-net.com](mailto:reclama@625-net.com))

### **Пэтчевое моделирование**

Алекс Бухтенко

**Как сделать лампу**

Алекс Бухтенко

**Синтез 3D графики шаг за шагом**

Максим Ральников (<http://wing.da.ru>)

**Новые возможности 3D Studio MAX**

Steepler Graphic Center

**3D Studio MAX**

Олег Татарников

**Метод раздельного освещения и затенения  
сцены**

Валерий Пуцал

### Всероссийский Клуб Вебмастеров

Всероссийский Клуб Вебмастеров, именуемый в дальнейшем Клуб, создан по решению физических лиц — граждан Российской Федерации в форме общественной организации. Клуб является общественным объединением и осуществляет свою деятельность в соответствии с Федеральным законом «Об общественных объединениях» и Гражданским Кодексом Российской Федерации. Клуб является основанной на членстве общественной организацией, созданной на основе совместной деятельности для защиты общих интересов и достижения уставных целей объединившихся граждан.

Официальное полное название на русском языке: Общественная организация «Всероссийский Клуб Вебмастеров». Официальное краткое название на русском языке: «Вебклуб». Официальное полное название на английском языке: «All-Russia Webmasters Club». Официальное краткое название на английском языке: «WebClub».

В соответствии с действующим законодательством Клуб имеет право образовывать представительства, филиалы, отделения и другие обособленные подразделения, входить в ассоциации, союзы и другие объединения.

Территорией действия Клуба является Российская Федерация.

Клуб ведёт свою деятельность, проводит собрания и размещает информацию, касающуюся деятельности Клуба, на веб-узле в сети Internet по адресу <http://www.webclub.ru>

## **Издательство “Познавательная книга плюс”**

ЗАО “Познавательная книга плюс” является эксклюзивным дилером по продаже книг издательств:

- “Бином”
- “Познавательная книга плюс”
- “Лаборатория базовых знаний”

издающих в год более 100 наименований книг по различным тематикам.

### **Взаимовыгодное сотрудничество в реализации:**

- книг по программному обеспечению и компьютерам.

Если вас заинтересовало наше предложение, то пошлите на наш адрес [roznkn@orc.ru](mailto:roznkn@orc.ru) реквизиты и профиль деятельности вашей организации, название региона, в котором вы работаете и пожелания по сотрудничеству. В ответ вы получите наши условия работы и прайс-лист нашей продукции.

### **Реквизиты издательства “Познавательная книга плюс”**

Мы ждем ваши предложения и надеемся на дальнейшее сотрудничество. Обращайтесь за более подробной информацией:

*адрес:* 109202, г. Москва,  
Перовское шоссе, 10/1

*тел.:* (095) 171 1954, 174 7616, (095) 373 0420

*факс:* (095) 170-6674

*web-сайт:* <http://www.orc.ru/~poznkn/>

*e-mail:* [poznkn@orc.ru](mailto:poznkn@orc.ru)

## Студия дизайна и рекламы МиК

### Масса услуг

Спектр предоставляемых услуг МиК весьма обширен. МиК обеспечивает полный диапазон услуг по дизайну, рекламе, маркетингу, предпечатной подготовке, разработке web-серверов, созданию персональных web-страничек и баннеров.

### Стандарт де-факто

Мы используем самое последнее программное обеспечение как на базе Macintosh, так и Windows NT.

### Наша концепция

Мы разграничиваем рекламу на престижную (формирование долговременного образа), на рекламу продукции, на имиджевую рекламу (публикация статей, информация о пресс-конференциях и т. д.), на рекламу распродаж и на рекламу "public relations" (разъяснения, извинения, возмущения).

Дизайн-студия МиК утверждает, что какая бы ни была профессиональная реклама и солидная Ваша компания, для ее большей известности необходима тщательно спланированная рекламная кампания.

### Обращайтесь!

Создание Вашего проекта начнется сразу после встречи с Вами или Вашим представителем!

Пишите! Мы сможем найти общий язык:  
**mik\_moscow@hotmail.com**. Звоните! По этим телефонам Вы получите любую интересующую Вас информацию:  
**(095) 324-3781, (8902) 681 0085** (бесплатно: мобильный МТС).  
Посетите наш web-сервер: <http://members.rotfl.com/bobleon>.

## **Книги серии «Кратко, доступно, просто» издательства «Познавательная книга плюс»**

Борис Леонтьев. Тонкости, хитрости и секреты Microsoft Windows 98.

Борис Леонтьев. Как установить и настроить Microsoft Windows 98.

Борис Леонтьев. Тонкости, хитрости и секреты Internet.

Борис Леонтьев. Хакеры, взломщики и другие информационные убийцы.

Борис Леонтьев. Web-дизайн: тонкости, хитрости и секреты.

Борис Леонтьев. Ваш незаменимый деловой партнер Lotus Organizer для Windows 98.

Борис Леонтьев. QuarkXPress для Windows 98: Дизайн и верстка в самой популярной настольной издательской системе.

Антон Высоткин. Все, что вы хотите знать о портативных компьютерах.

Антон Высоткин. Модемы: тонкости, хитрости и секреты.

Ашот Халафян. Как прописаться в Москве: Консультации юриста. (серия книг «Библиотека «Законодательство»)

Эдуард Фелистов. Введение в архитектурно-пространственное моделирование проектных решений в программе ArchiCAD 6.0.



Эдуард Фелистов. **Архитектурно-пространственное моделирование проектных решений в программе ArchiCAD: Справочное пособие.**

Иван Фролов. **Как собрать компьютер: тонкости, хитрости и секреты.**

Владислав Пузырев. **Microsoft Outlook Express для Windows 98: Электронная почта Internet и конференции пользователей Usenet.** (серия книг «Windows 98 без секретов»).

Сергей Ивановский. **Операционная система Linux для начинающих и не только.** (серия книг «FIDONet для начинающих и не только»).

Борис Леонтьев. **Все лучшие русскоязычные ресурсы Internet: справочное пособие.** (серия книг «Internet без секретов»).

Владислав Пузырев. **Глобальная некоммерческая информационная сеть FIDONet для начинающих и не только.** (серия книг «FIDONet без секретов»).

Владимир Иконников, Борис Леонтьев. **Adobe Photoshop 5.0 для начинающих и не только.** (серия книг «Компьютерная графика без секретов»).

Иван Фролов. **Джойстики: Все, что вы хотите знать.** (серия книг «Компьютер без секретов»).

Алексей Петровский, Борис Леонтьев. **Эффективный хакинг.** (серия книг «Хакерство без секретов»).

Людмила Зимина. **Виртуозная работа в 3D Studio Max 3.0.** (серия книг «Компьютерная графика без секретов»).

Антон Высоткин. **Все, что вы хотите знать о Мультимедиа.** (серия книг «Компьютер без секретов»).

Антон Высоткин. **Microsoft Plus! 98 для начинающих и не только.** (серия книг «Windows 98 без секретов»).

Георгий Мысин. **Как решить проблему 2000 года.** (серия книг «Компьютер без секретов»).

Владимир Иконников. **Adobe Illustrator 8.0 для начинающих и не только.** (серия книг «Компьютерная графика без секретов»).

Петр Карабин. **Эффективный фрикинг, секреты телефонной линии или тайны «телефонных» хакеров.** (серия книг «Хакерство без секретов»).

Антон Высоткин. **Мультимедийные приложения для Windows 98.** (серия книг «Windows 98 без секретов»).

Владислав Пузырев. **Как общаться с другими людьми через Internet с помощью ICQ.** (серия книг «Internet без секретов»).

Алексей Фридман. **Программирование на Turbo Pascal для начинающих и не только.** (серия книг «Программирование без секретов»).

Алексей Фридман. **Программирование на C++ для начинающих и не только.** (серия книг «Программирование без секретов»).

Алексей Фридман. **Программирование на Assembler для начинающих и не только.** (серия книг «Программирование без секретов»).

Алексей Фридман. **Microsoft Visual Basic 6.0: тонкости, хитрости и секреты.** (серия книг «Программирование без секретов»).

Эдуард Фелистов. **MathCAD для начинающих и не только.** (серия книг «САПР без секретов»).

Эдуард Фелистов. **PCAD для начинающих и не только.** (серия книг «САПР без секретов»).

Алексей Петровский. **Баннерная реклама в Internet.** (серия книг «Internet без секретов»)

Михаил Зотов. **Офисные приложения для Windows 98: текстовый редактор Microsoft WordPad и графический редактор Microsoft Paint.** (серия книг «Windows 98 без секретов»).

Андрей Шестов. **Служебные приложения для Windows 98.** (серия книг «Windows 98 без секретов»).

Петр Карабин. **Internet-приложения для Windows 98.** (серия книг «Windows 98 без секретов»).

Алексей Петровский. **Маркетинг и Internet или все, что вы хотите знать о web-маркетинге.** (серия книг «Web-маркетинг без секретов»).

Петр Карабин. **Все, что вы хотите знать о компьютерных вирусах.** (серия книг «Компьютер без секретов»).

Иван Фролов. **Все, что вы хотите знать о персональных компьютерах.** (серия книг «Компьютер без секретов»).

Дмитрий Царик, Борис Леонтьев. **Тонкости, хитрости и секреты компьютерных игр: коды, читы и солюшены.**

Андрей Быстров. **CorelDRAW 9.0 для начинающих и не только.** (серия книг «Компьютерная графика без секретов»).

Владимир Иконников. **Дизайн и верстка в Adobe PageMaker 7.0 для начинающих и не только.** (серия книг «Настольное издательство без секретов»).

**Борис Леонтьев. Дизайн, верстка и компьютерная графика для начинающих и не только.**

*(серия книг «Настольное издательство без секретов»).*

**Борис Леонтьев. Дизайн книги для начинающих и не только или как профессионально подготовить книгу к печати.**

*(серия книг «Настольное издательство без секретов»).*

**Борис Леонтьев. Как профессионально подготовить рекламу для периодических изданий.**

**Борис Леонтьев. Как профессионально подготовить рекламу для Internet.**

*(серия книг «Internet без секретов»).*

**Алексей Фридман. Microsoft Word 2000 для начинающих и не только.**

*(серия книг «Windows 2000 без секретов»).*

**Алексей Фридман. Microsoft Excel 2000 для начинающих и не только.**

*(серия книг «Windows 2000 без секретов»).*

**Борис Леонтьев. Операционная система Microsoft Windows 2000 для начинающих и не только.**

*(серия книг «Windows 2000 без секретов»).*

**Борис Леонтьев. Тонкости, хитрости и секреты Microsoft Windows 2000.**

*(серия книг «Windows 2000 без секретов»).*

**Борис Леонтьев. Как установить и настроить Microsoft Windows 2000.**

*(серия книг «Windows 2000 без секретов»).*

# Содержание

*Если посмотреть на компьютерную графику со стороны . . . . . 3*

## **3D Studio MAX**

Интерфейс . . . . .	7
Многозадачность . . . . .	9
Моделирование . . . . .	9
Материалы и визуализация . . . . .	12
Анимация . . . . .	15
Резюме . . . . .	16

## **3D Studio MAX 2.0**

Специальные анимационные эффекты . . . . .	20
Новое в анимации . . . . .	21
Анимационные инструменты . . . . .	22
Продвинутый Rendering и камерные эффекты . . . . .	23
Новое в материалах . . . . .	25
Освещение и камеры . . . . .	26

## **3D Studio MAX 2.5**

Геометрия . . . . .	28
Редактор материалов . . . . .	30
Утилиты . . . . .	30

### 3D Studio MAX 3.0

Track Bar .....	31
Toolbars .....	31
Schematic View .....	31
Материалы .....	32
Xrefs .....	32
RAM Player .....	32
Particle Systems .....	33
Скрипты .....	33

<b>Требования к оборудованию .....</b>	<b>34</b>
--	-----------

### Какое «железо» нужно для 3D программ

Монитор .....	36
Видеокарта .....	36
Процессор .....	38
Оперативная память .....	39
Винчестер .....	40
Сетевая плата .....	40
IntelliMouse и 3D Studio MAX .....	41

### Тонкости, хитрости и приемы работы

Слайновое моделирование .....	44
Просчет и фотореализм .....	46

Что дальше? . . . . .	50
Hello, world! . . . . .	53
Хромированный металл . . . . .	57
Ландшафт . . . . .	65
Текстура ландшафта . . . . .	69
Как создать море . . . . .	73
Как сделать реалистичный взрыв и осколки от объекта . . . . .	77
Как сделать параллельные лучи света . . . . .	80
Создание фонтана . . . . .	83
Больше зеркал, хороших и разных! . . . . .	85
Растущий текст . . . . .	88
Anti-aliasing. Что это такое? . . . . .	92
Mountain Texturing . . . . .	92
Использование Photoshop для создания текстур . . . . .	101
Моделирование руки по 4 огибающим . . . . .	108
Моделирование одного пальца руки . . . . .	111
Моделирование полной руки — ладонь, пальцы, плечо, предплечье . . . . .	114
Как сделать правильную гайку с помощью лофта Rhino . . . . .	118
3D моделирование . . . . .	121
Как сделать красивый 3D текст . . . . .	138
Как сделать лампу . . . . .	140
Пэтчевое моделирование . . . . .	143
Метод раздельного освещения и затенения сцены . . . . .	152

### Расширения 3D MAX

Plug-in Shag: Fur . . . . .	156
Волосатая дубинка . . . . .	159

### Вопросы и ответы

Как обойти проблему несовместимости форматов при импорте файлов Adobe Illustrator в 3DS MAX? . . .	161
Как сделать зеркало? . . . . .	161
Пакетный рендер. Как поставить несколько проектов в очередь для просчета? . . . . .	162
Как настроить сетевой рендер? . . . . .	162
Как настроить Glow в MAX? . . . . .	165
Bomb и его параметры. Как «взорвать» объект? . . . .	166
Где устанавливается пивот? . . . . .	168
Как разным частям одного объекта назначить различные материалы? . . . . .	168
Как в доске прорезать надпись? . . . . .	169
Как отрендерить объект с отбрасываемой тенью на невидимый объект? . . . . .	169
Чем LightWave лучше 3D MAX? . . . . .	170
3D MAX работает только под 32М? . . . . .	171
Как делать анимацию? . . . . .	171
Как бомбу использовать? . . . . .	172
Как спрятать объект? . . . . .	172



## Приложения

Ссылки на ресурсы Internet . . . . .	173
Список использованной литературы . . . . .	176
Всероссийский Клуб Вебмастеров . . . . .	179
Издательство "Познавательная книга плюс" . . . . .	180
Реквизиты издательства "Познавательная книга плюс" . . . . .	181
Студия дизайна и рекламы МиК . . . . .	182
Книги серии «Кратко, доступно, просто» издательства «Познавательная книга плюс» . . . . .	183



**Издательство  
«Познавательная книга плюс»**



**Книги серии  
"Кратко, доступно, просто"**

**Телефоны:**

**(095) 171 1954**

**(095) 174 7616**

**(095) 373 0420**

