

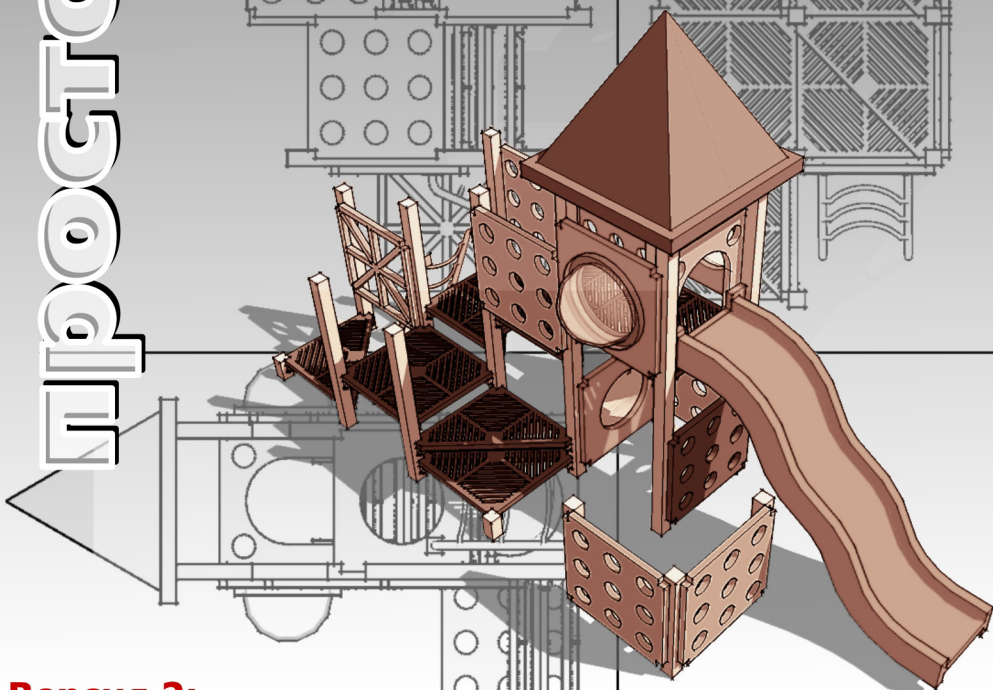
Александр Петелин

SketchUp

учебник-справочник

ПРОСТО 3D!

Книга 1. ПРАКТИК



Версия 2:
Google/Trimble SketchUp Pro 8

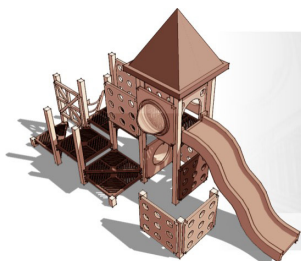
2011-2013



ОТ АВТОРА	5
Тема 1. НЕМНОГО ИСТОРИИ И ТЕОРИИ	
2000-2013	7
2D...	9
...и 3D	10
Камеры	11
Навигация, проекции	11
Тела, поверхности, кривые, полигоны...	13
Eng-Rus	15
Тема 2. ВСТРЕЧАЙТЕ – SKETCHUP!	
«Софт» и «железо»	17
Интерфейс	18
Тема 3. БАЗОВЫЕ ИНСТРУМЕНТЫ	
Principal (Основные инструменты)	23
Select (Выбор)	24
Component (Компонент)	25
Eraser (Ластик)	25
Paint Bucket (Палитра)	25
Drawing (Инструменты рисования)	25
Line (Линия)	25
Arc (Дуга)	29
Freehand (От руки)	30
Rectangle (Прямоугольник)	31
Circle (Окружность)	31
Многоугольник (Polygon)	32
Тема 4. НАВИГАЦИЯ В СЦЕНЕ	
Camera (Камера)	33
Orbit (Вращение)	34
Pan (Панорамирование)	34
Zoom (Лупа)	34
Zoom Window (Окно увеличения)	35
Zoom Extents (Показать все)	36
Previous (Предыдущий вид)	36
Next (Следующий вид)	36

Views (Виды)	36
Тема 5. ИНСТРУМЕНТЫ И ОПЦИИ МОДИФИКАЦИЙ	
<i>Push / Pull (Вдавить / Вытянуть)</i>	39
<i>Follow Me (Следуй за мной)</i>	41
<i>Offset (Контур)</i>	43
<i>Move (Перемещение)</i>	43
<i>Rotate (Вращение)</i>	46
<i>Scale (Масштабирование)</i>	46
<i>Faces / Surfaces (Плоские / Криволинейные поверхности)</i>	50
<i>Soften / Smooth Edges (Смягчение / сглаживание ребер)</i>	51
Тема 6. ПЕРВАЯ МОДЕЛЬ	
<i>Group (Группа)</i>	54
Выбор в быстрой последовательности	56
Выбор и создание группы через контекстное меню	56
Lock (Фиксация) группы	57
<i>Entity Info (Инфо по элементу)</i>	57
<i>Редактирование внутри группы</i>	62
Тема 7. ИЗМЕРЕНИЯ	
<i>Model Info / Units (Инфо по модели / Единицы измерения)</i>	65
Тема 8. СТРОИМ ТОЧНО	
<i>Управление инструментами рисования</i>	68
Line (Линия)	68
Arc (Дуга)	70
Rectangle (Прямоугольник)	72
Reverse Faces (Поменять стороны поверхности)	73
Circle (Окружность)	73
Polygon (Многоугольник)	75
<i>Управление FOV (фокусным расстоянием объектива)</i>	75
<i>Управление инструментами модификаций</i>	76
Push / Pull (Вдавить / Вытянуть)	76
Follow Me (Следуй за мной)	77
Offset (Контур)	77
Move (Перемещение)	77

Rotate (Вращение)	81
Scale (Масштабирование)	83
Конструкционные инструменты	82
Tape Measure (Рулетка)	83
Protractor (Транспортир)	85
Axes (Оси)	86
Строим модель в размерах	88
Тема 9. РАБОЧАЯ ВИЗУАЛИЗАЦИЯ	92
Настройки видеокарты	93
Hide / Unhide (скрыть / показать)	94
Стили отображения поверхностей и ребер	96
Стили поверхностей	96
Стили ребер	100
Shadows (Тени)	101
Тема 10. МАТЕРИАЛЫ	
Paint Bucket (Палитра)	107
Диалоговое окно Materials (Материалы)	111
Тема 11. ТЕКСТУРИРОВАНИЕ	120
Position (Позиция текстуры)	126
Make Unique Texture (Создать уникальную текстуру)	128
Combine Textures (Комбинировать текстуры)	129
Add Photo Texture (Назначить фототекстуру)	129
Тема 12. СЛОИ	130
Диалоговое окно layers (Слои)	131
Тема 13. СЦЕНЫ	
Диалоговое окно Scenes (Сцены)	138
Тема 14. СТИЛИ	
Диалоговое окно Styles (Стили)	143



SketchUp просто 3D!

Книга 1. ПРАКТИК

ОТ АВТОРА

Если читаете сейчас эти строки, значит настало ваше время проявить интерес к темам, обозначенным в названии книги! В мир 3D приходят по-разному – от желания просто как хобби, «для души» попробовать себя в новом жанре, новой программе компьютерной графики, – до профессиональной необходимости освоения нового рабочего инструмента. Сразу можно верить – и в том, и в другом случае вы на правильном пути, выбрав именно *SketchUp*!

О чем говорит название книги? Надеюсь, оно точно отражает задуманное содержание, назначение, организацию и стиль подачи материала. Итак, почему «*SketchUp*», почему «просто» и почему «3D»...

Во-первых – перед вами учебник-справочник по 3D-редактору *SketchUp*. Точнее, последней «под флагом» Google – *SketchUp Pro* 8-й версии... Подробнее почему именно так, а не по текущей версии от сегодняшнего владельца Trimble – *SketchUp 2013* в следующей главе... Далее мы будем подробно рассматривать все возможности и достоинства *SketchUp*, но сразу подчеркну главное, особенно ценное для вообще новичка в 3D или в этой новой для пользователя других редакторов программе – уникальную дружественную простоту освоения и не менее уникальные функциональные достоинства в работе.

Во-вторых – решения по содержанию, структуре и стилю изложения учебной информации сложились из общения с читателями других моих учебников, в т.ч. предыдущих по *SketchUp* – в основном, начинающих пользователей. Стало ясно, что кроме подробного «сухого» руководства-справочника в манере *Help* с описанием пунктов меню, или пошаговых уроков решения отдельных задач (что предполагает уже наличие базовых знаний программы), часто требуется нечто третье... То самое «просто», последовательно, от простого к сложному, от общих понятий к нюансам, максимально подробно и наглядно иллюстрированное, и на практических примерах. Поэтому в итоге сложилось решение построить книгу, как тренинг с воображаемым «учеником», или как курс занятий с комментариями по ходу и ответами на типичные вопросы и затруднения. При этом за основу принят принцип постепенного «наращивания» багажа знаний – от самых необходимых и востребованных тем, до профессиональных тонкостей решения узкоспециальных задач, «секретов» и «хитростей» моделирования в *SketchUp*...

В-третьих – круг интересов и начальный уровень подготовленности пользователей к моменту знакомства со *SketchUp* бывает очень различен, как различна и персональная необходимость в глубине освоения программы. Для *SketchUp* это особенно актуально, поскольку многие начинают его освоение с совершенно «нулевого» уровня не только в 3D, но и вообще в компьютерной графике. Очевидно также, что даже опытные пользователи и даже в «своей» программе не задействуют всех 100% возможностей – только то, что действительно необходимо в рабочей практике или больше подходит для типично решаемых задач, а в

некоторые «уголки» программы могут так ни разу и не заглянуть. Отсюда – решение разделить материал на два уровня «глубины» освоения, условно обозначенные в названиях двух частей (книг) – «Практик» и «Эксперт». Первая книга проведет читателя от начала знакомства с миром трехмерной компьютерной графики, основными понятиями и принципами устройства 3D-программ до уровня уверенного владения основными, самыми востребованными и необходимыми инструментами *SketchUp*, готовности к решению типичных задач моделирования. Вторая книга углубит и завершит изучение всех возможностей, практических «тонкостей» и «хитростей» работы в *SketchUp*, знание которых и делает пользователя профессионалом, экспертом в программе.

Таким образом, каждый может выбрать и начать изучение со своего уровня подготовленности и по своим интересам, или обратиться к любой из еще не освоенных тем в нужный момент, в произвольном порядке. Хотя практика показывает, что иногда бывает очень полезно «вернуться к истокам» даже достаточно опытному пользователю – не исключено, что обнаружатся пробелы в уже, казалось бы, пройденном и освоенном...

Условные сокращения и обозначения в тексте:

Клик (кликнуть) – нажатие левой кнопки мышки (запуск опции)

СКМ – средняя кнопка (колесико) мышки

ПКМ – правая кнопка мышки

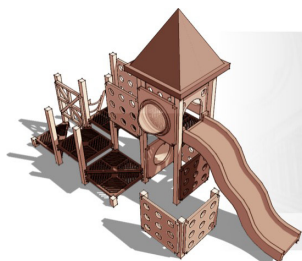
Shift + СКМ – сочетания клавиш с кнопками мышки

Ctrl+V – «горячие клавиши» (*keyboard shortcuts*) быстрого запуска команд

Конечно, приведенные в книге комментарии, замечания и оценки по тематике учебника выражают только личное мнение автора, исходя из личного понимания и опыта, и конечно, не претендуют на роль единственно однозначных и правильных. В том числе непосредственно по методике и способам решения задач моделирования – хотя бы по причине того, что один и тот же результат может быть достигнут самыми разными шагами и способами, а «универсал» *SketchUp* в этом плане предоставляет все возможности...

Буду благодарен за отзывы и пожелания по книге, готов ответить на возникающие дополнительные вопросы (пишите: prosketchup@narod.ru) и, конечно, – желаю творческих успехов!

Александр Петелин 2011 – 2013 гг.



SketchUp просто 3D!

Книга 1. ПРАКТИК

Тема 1. НЕМНОГО ИСТОРИИ И ТЕОРИИ

2000-2013

Как видим из приведенных дат, *SketchUp* имеет весьма солидный по меркам жизни компьютерных программ возраст – его «родитель», американская компания *@Last Software*, первую рабочую версию представила еще в 2000 г.

В 2006 г. *@Last Software* прекратила свое существование, поскольку компания вместе со *SketchUp* (уже 5-й версии) были приобретены всем известной корпорацией *Google*, «под флагом» которой за шесть последующих лет было выпущено еще три версии.

Весной 2012-го происходит еще раз неожиданная смена владельца – *Google* продает *SketchUp* компании *Trimble*, совершенно неизвестной до этого 3D-сообществу и вообще весьма далекой по профилю деятельности от этой тематики. И вот через год появляется наконец новая, текущая на сегодня версия – *Trimble SketchUp 2013*. Однако, не отвлекаясь здесь на подробности, можно сразу резюмировать – к сожалению, в ней не оказалось ни одного настолько реально значимого, обоснованного, полезного «плюса» – улучшения или нововведения, чтобы они хотя бы перевесили те «минусы», которые возникли в этой новой версии! Таким образом для практически работающего в программе очень сомнителен смысл перехода на 2013-й с предыдущего – отработанного, надежного, фактически последнего «гугловского» *SketchUp* 8-й версии (точнее – 8.0.16846), на основе которого и будем изучать программу.

Что же касается опасений пользователя (особенно начинающего) оказались на «перепутье» версий – на самом деле никаких «революций» в программе не произошло, практически все инструменты и опции остались все те же... А раз так, «косметические ремонты» новых версий (а большего вряд ли стоит ожидать и в будущем!) никак не мешают использовать для полноценного изучения программы учебные пособия по *SketchUp 8.0* (да и более ранним)...

Тем не менее для тех, кто все-таки захочет сам попробовать и оценить *Trimble SketchUp 2013*, по ходу изучения конкретных тем в учебнике приведены краткие комментарии по его явным отличиям и изменениям в интерфейсе и функциям.

SketchUp доступен в двух вариантах – профессиональном *SketchUp Pro* (на нем и базируются материалы книги), и бесплатном – просто *SketchUp* (в 2013-ой версии стал называться *SketchUp Make*). *SketchUp Pro* без покупки лицензии предоставляет после первой установки 8 часов пробного периода работы. Бесплатная версия отличается от *Pro* определенными функциональными ограничениями: недоступен или ограничен импорт и экспорт файлов ряда форматов, а также отсутствуют две «партнерские» программы-приложения: *LayOut* и *Style Builder*, входящие в комплектацию *SketchUp Pro*. Но сразу можно заметить, что эти ограничения для начинающего пользователя совершенно не критичны, так что можно не только сразу начинать

изучение *SketchUp*, но и полноценно работать над своими проектами.

Так что же такое *SketchUp*, и в чем его отличие от других, общеизвестных 3D-редакторов? Изначально «идеология» программы была сформулирована разработчиками @Last Software как «...оптимальное сочетание элегантности и свободы дизайна «от руки» со скоростью и гибкостью цифровых технологий...», и «3D – для всех!». И эти принципы были блестяще воплощены, прежде всего – в уникальном по простоте, удобству и функциональности интерфейсе, который обеспечивает как быстроту и легкость освоения, так и непревзойденную для 3D-редактора скорость работы. Благодаря этому *SketchUp* предоставляет максимальный простор именно для творческой части работы, дает возможность творить, фактически «не отрывая руку от карандаша» при минимальных затратах на технические действия. А отсюда – если для большинства пользователей освоение нового специфического мира трехмерного моделирования неизбежно весьма трудоемко и длительно, то со *SketchUp* это намного проще, легче, быстрее! Буквально в первый раз открыв программу, пользователь уже через полчаса начинает с увлечением скорее не работать, а играть в моделирование, даже если это его первая 3D-программа...

Редактор разрабатывался, в том числе, и для потенциального пользователя, у которого 3D – только один (или не основной) из профессиональных инструментов, или вообще просто хобби «для души». Понятно, что в этом случае у него нет необходимости, возможности, да и желания тратить долгие часы на освоение громоздких интерфейсов и огромного числа функций «тяжелых» универсальных 3D-редакторов «полного цикла». Остается только удивляться, насколько точно в свое время разработчики @Last Software «попали» в эту востребованную и перспективную «нишу» жанра...

Сейчас уже, пожалуй, можно уверенно сказать, что *SketchUp* стал родоначальником идеологии целого направления максимально дружественных к пользователю программ, ориентированных именно на простоту, легкость освоения, скорость и удобство работы, но в то же время обладающий мощным функционалом. Характерно, что аналогичный подход все чаще прослеживается и в новых 3D-редакторах разного назначения. Более того – в некоторых из них можно заметить явные «скетчаповские» черты в принципах решения интерфейсов и вообще в системе управления.

Надо отметить также, что стало фактом широкое признание *SketchUp* сторонними разработчиками «партнерских» программ, которые сейчас как правило включают в свои продукты или непосредственную поддержку моделей (файлов) *SketchUp*, или обмен данными с ним через специальные плагины.

SketchUp, что отражено и в самом его названии (*Sketch* – эскиз, набросок, *Up* – вверх) предназначен для эскизного, поискового 3D-моделирования – изначально и прежде всего – с архитектурной специализацией. Однако на самом деле, благодаря гибкости заложенных в нем механизмов моделирования, возможности и области применения оказались намного шире – *SketchUp* с успехом используют для разработки разнообразных проектов во всех жанрах дизайна, рекламы, инженерном проектировании, кино- и игропроизводстве... трудно перечислить все.

Интересно проследить историю роста популярности *SketchUp* у «3D-общественности». Долгое время понятие 3D однозначно было связано с «тяжелыми» редакторами, прежде всего – с «патриархом» жанра – *Autodesk 3ds Max*. Да и сейчас это достаточно характерное заблуждение, особенно у тех, кто «не в теме» – например, у работодателей дизайнеров и заказчиков. Видимо поэтому довольно долго со *SketchUp* был знаком только узкий круг специали-

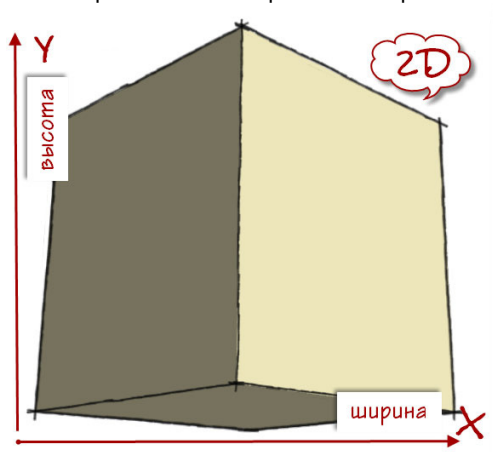
стов, в основном – архитекторов, а у нас в стране он оставался практически неизвестным. К тому же дружелюбность интерфейса программы, как ни парадоксально, иногда работает «на негатив» – создает при первом поверхностном знакомстве впечатление несерьезного, даже неполноценного редактора. На самом же деле за обманчивой простотой, даже «игрушечностью» интерфейса (в т.ч. в названиях некоторых инструментов) скрываются замечательные возможности для решения практически всех задач низкополигонного 3D-моделирования, высокая точность и удобство построений уровня профессиональных CAD (Систем компьютерного черчения) редакторов.

И конечно, явный скачок роста популярности *SketchUp* произошел с момента «привязки» программы к Интернет-проектам *Google 3D-модели (3D Warehouse)* и *Google планета Земля (Google Earth)*, для чего в *SketchUp* были встроены специальные опции и инструменты. Очевидно кстати, что в свое время причиной выбора и приобретения *SketchUp* корпорацией *Google* очевидно стали именно его простота и доступность – альтернативы на роль необходимого для таких проектов «народного» 3D-редактора просто и не было... да пожалуй, нет и до сих пор.

2D...

Начнем с обзорного знакомства с базовыми основами компьютерной графики вообще и 3D-моделирования в частности – понятиями, терминологией и «правилами игры». Этому имеет смысл уделить внимание, поскольку незнание элементарной «азбуки» изучаемого жанра в дальнейшем, при освоении конкретных инструментов и опций моделирования, порождает множество затруднений.

Итак, каждый может взять в руки карандаш и изобразить на листе бумаги, например, простой кубик. На определенном этапе компьютерного прогресса появились технологии компьютерной графики – возможность получать аналогичный результат в специальных программах – графических редакторах, например – самых распространенных у нас *Adobe Photoshop* или *CorelDraw*. Конечно, чисто технически с этого момента началась настоящая революция в части инструментов, методов и производительности рисования, но суть осталась та же – плоское изображение на некой плоской основе: там – на бумаге, здесь – в рабочем пространстве редактора, а для пользователя – фактически на экране монитора.



Теперь представим, что нам надо получить изображение этого же кубика, но в дру-

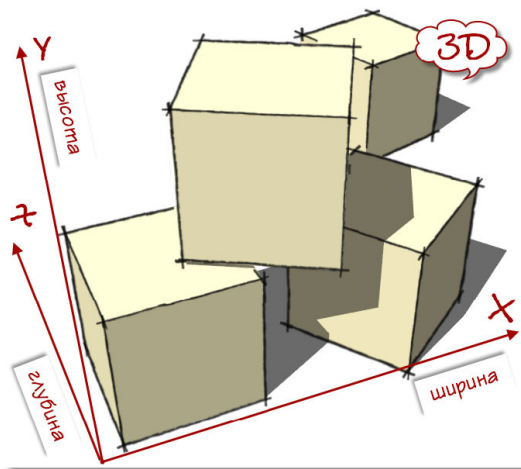
гом ракурсе (т.е. требуется показать его с другой стороны), изменить размер и положение на листе. В случае «ручного» рисования любой новый ракурс означает только одно – все опять «с нуля», заново... В компьютерном графическом редакторе можем без проблем быстро изменить размеры и положение уже существующего рисунка, удалить, приблизить картинку на экране, но вот другой ракурс... и здесь означает рисование еще одной новой картинки.

Суть в том, что рисуя на листе бумаги, мы неизбежно имеем дело с т.н. двухмерным рабочим полем, с двумя *Dimension* (размерами, измерениями), откуда и пришло сокращенное обозначение – 2D. Или, по-другому – при этом используем только два размерных понятия и измерения – «ширина (длина)» и «высота».

В этих же параметрах работают и механизмы рисования 2D графических редакторов – в них положение любой точки в числовых значениях определяется координатами по двум осям: горизонтальной и вертикальной (обозначаемых обычно X и Y). При этом заметим, что даже если откроем в редакторе фотографию, на которой зрительно вполне явно «читается» глубина пространства (например, перспектива улицы), все равно картинка, как объект редактирования, остается двумерной. Конечно, у 2D-графики имеется весомое достоинство – относительная простота в понимании сути рабочего пространства и управления инструментами рисования. Ведь 2D-редактирование фактически аналогично привычному «ручному» рисованию – например, интуитивно очевидно, что произойдет, если провести линию от точки А к точке Б, переместить нарисованную фигуру по горизонтали, вертикали, и как это сделать...

...и 3D

В отличие от 2D, трехмерная, или 3D-графика, базируется на принципиально другом подходе, прежде всего – к рабочему пространству. Здесь два привычных понятия и измерения – «ширина, высота», дополняются третьим – «глубина», и таким образом положение элементов любого объекта однозначно определяются тремя пространственными координатами с числовыми значениями по каждой из осей (традиционно – X, Y, Z). А само трехмерное пространство – это фактически виртуальный мир, который во многих 3D-редакторах так и называется – *World (Мир)*, а его оси – *World Axes (Мировые оси)*.



Также для такого пространства, «мира», используется аналогичное понятие – «Сцена». Работу в 3D-редакторе правильной уже называть не «рисованием», а «построением», по-

сколько изображаем не картинку, не плоскую фигуру, а строим именно полноценную (пусть и виртуальную) объемную форму. Такая форма (или «модель»), как и в реальности, имеет три измерения в физически реальных величинах. И именно это дает возможность свободно манипулировать объектом в трехмерной пространстве проекта-сцены.

Что же принципиально иного нам дает 3D-моделирование по сравнению с 2D-графикой? В самом широком смысле – мы получаем уникальную и удивительную возможность свободно, как угодно, и сколько угодно раз манипулировать однажды построенной моделью, изменяя и дополняя множество ее параметров в трехмерном, т.е. максимально приближенном к реальному восприятию, пространстве.

КАМЕРЫ

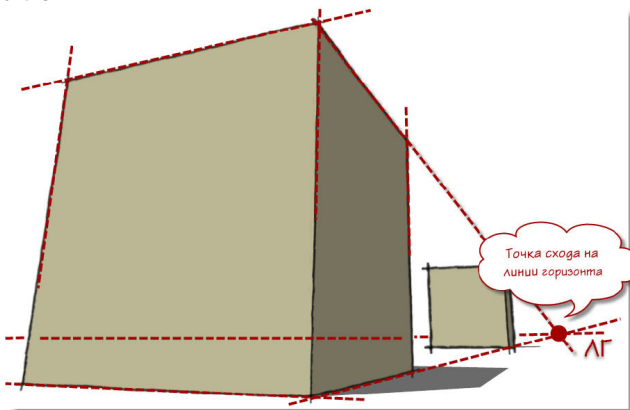
Итак, в трехмерной сцене произвольно меняем положение модели в пространстве – в нашем примере перемещаем и поворачиваем трехмерный кубик так, чтобы увидеть любую из его сторон. Однако рассмотреть тот же кубик, или любой другой трехмерный объект можно и по-другому, используя еще один базовый механизм 3D-моделирования – т.н. **Camera (Камера)**. Что под этим понимается? В принципе, это то, что в каждый момент времени видим на экране монитора, это как бы окно в наш виртуальный мир, созданный в данном проекте. Или, точнее: представьте, что смотрите на сцену через объектив фотокамеры. Почему именно объектив, а не просто «окно»? Дело в том, что механизм камеры дает дополнительные возможности управления показом сцены за счет изменения параметров ее «объектива». Т.е. можно физически достоверно имитировать диафрагму, глубину резкости, фокусное расстояние, ширину поля взгляда и др. настройки реальной оптики фотокамеры. Кроме того, доступны опции зуммирования, т.е. увеличения, уменьшения (или приближения, удаления) изображения нужной части сцены на экране. Пока на этом останавливаться не будем, но тому, кто достаточно основательно занимался фотографией, эти термины скажут многое...

НАВИГАЦИЯ, ПРОЕКЦИИ

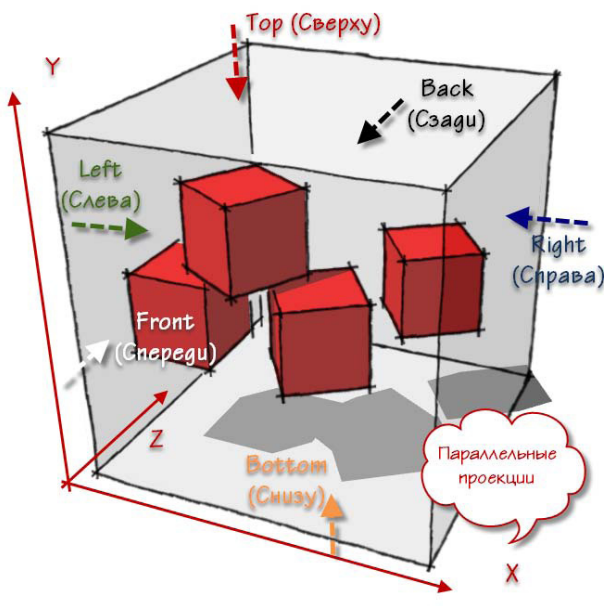
С понятием *Камеры* в 3D связано еще одно – т.н. «Навигация в сцене». Проще говоря, это опции управления камерой, а значит, взглядом в сцену. Возвращаясь к нашему примеру – мы можем рассмотреть обратную сторону кубика не трогая его, а просто «объехав» камерой вокруг объекта и направив объектив (т.е. взгляд) в нужном направлении. Таким образом, имеем возможность рассматривать сцену, как бы разгуливая по нашему трехмерному миру с камерой в руках. Нетрудно заметить, что часть аналогичных опций обзора рабочего пространства есть и в 2D-редакторах, однако в 3D их значимость и возможности несоизмеримо выше. Во-первых, манипуляции камерой занимают не меньшую часть времени, чем собственно редактирование геометрии объектов, а во-вторых, без свободного владения опциями навигации можно просто «потеряться» в трехмерном пространстве...

Однако свободная «ручная» камера иногда может не устраивать при работе с трехмерными объектами – при таком варианте отображения зачастую бывает неудобно, да и просто невозможно выполнять точные построения, перемещения, трансформации геометрии. Ведь такая камера имитирует «человеческий» взгляд на мир, в котором всегда присутствуют искажения геометрии по законам линейной перспективы – т.е. одинаковые по размерам, но более удаленные объекты будут казаться меньше, чем находящиеся ближе к наблюдателю. Это, собственно, и обеспечивает визуальную достоверность визуализации трехмерной сцены, но и означает также, что неизбежны зрительные искажения и линейных раз-

меров, и направлений.



Для решения этой проблемы в 3D-редакторах используются специальные, т.н. «стандартные» камеры – **Параллельные (ортогональные) проекции (или Виды)**. В принципе, эти термины наверняка знакомы каждому еще из школьных уроков черчения, однако в 3D они имеют свою специфику... Представим, что наша сцена (т.е. все ее объекты) находится внутри воображаемого куба с прозрачными стенками – т.е. сцена ограничена 6-ю смежными, взаимно перпендикулярными плоскими поверхностями. При этом линии пересечения (границы) этих поверхностей (ребра куба) направлены строго вдоль соответствующих осей сцены X,Y,Z, а потому сами поверхности ориентированы строго параллельно воображаемым плоскостям, образуемым каждой парой осей (их еще называют «планами осей»).



Теперь установим нашу камеру, например, на верхнюю плоскость так, чтобы ее «объ-

ектив» был направлен внутрь куба. Условимся также, что в этом варианте камера будет всегда «прижата» к поверхности куба, т.е. может перемещаться, «скользить» только по ней (или по параллельной ей плоскости) – получаем требуемый вид *Top* (*Сверху*). Или, что аналогично – в плоскости, плане осей X-Z. Нетрудно заметить, что в этом случае фактически получаем аналог двумерного рабочего пространства (нет «глубины» перемещения камеры), а значит, нет и перспективных искажений – что, собственно, и требовалось. По аналогичному принципу работают и остальные параллельные виды, а всего их шесть: *Top* (*Сверху*), *Left* (*Слева*), *Right* (*Справа*), *Front* (*Спереди*), *Back* (*Сзади*), *Bottom* (*Снизу*).

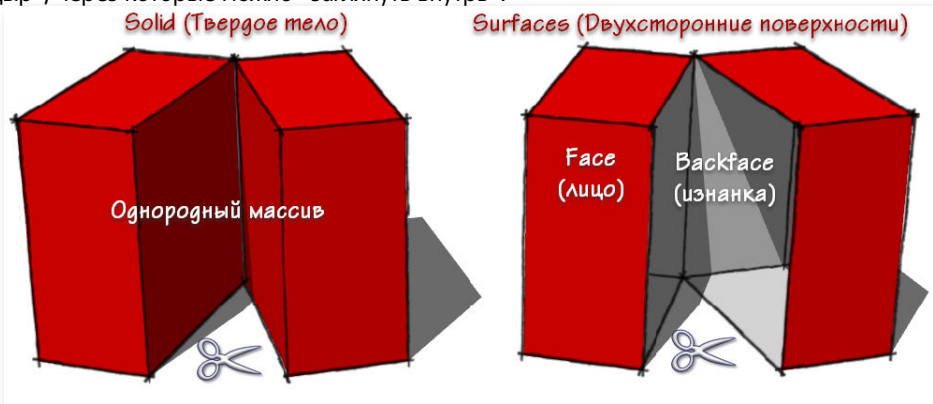
Но здесь все-таки надо уточнить, что это упрощенная модель – на самом деле на появление перспективных искажений влияет и один из «технических» параметров виртуального «объектива» – т.н. *FOV* (*Ширина поля зрения*)... но пока на этом задерживаться не будем.

Итак, имеем общепринятую во всех 3D-редакторах основную систему ориентации в пространстве за счет камер (видов) – «свободной» (одной или нескольких) и 6-ти стандартных (фиксированных проекций), на которые можно мгновенно переключаться в любой нужный момент.

ТЕЛА, ПОВЕРХНОСТИ, КРИВЫЕ, ПОЛИГОНЫ...

Алгоритмы 3D-моделирования (и использующие их редакторы) по наиболее распространенным математическим моделям можно условно разделить на две большие группы – т.н. *solids* (*твердотельные или сплошные*) и *surfaces* (*поверхностные*).

По теме твердотельного моделирования можно коротко сказать, что этот метод типично используется в специализированных редакторах для инженерно-конструкторских задач. Т.е. там, где важно получить модель (например, деталь машины), несущую ее точные физические характеристики – вес, объем, центр тяжести и т.п. Кроме того, благодаря этому можно комбинировать тела, объединяя и вычитая их, и создавать таким образом новые, более сложные модели. Принципиальная разница между *solids* и *surfaces* становится очевидной, если разрезать объект – *solids* всегда представляют собой абсолютно замкнутые, сплошные, «глухие» объемы. Упрощенно, главный признак таких объемов – они никогда не имеют «дыр», через которые можно «заглянуть внутрь».



Поверхностный же метод построения трехмерных форм направлен только на визуализацию внешней оболочки модели, его главная задача – показать «как это выглядит», без уче-

та того, что находится внутри объекта. Сравнить, что лучше – алгоритмы *solids* или *surfaces* не имеет смысла, поскольку они предназначены для разных целей и задач...

SketchUp – типичный поверхностный моделиер, поскольку здесь объекты всегда формируются из **Faces (Плоских поверхностей)** и **Surfaces (Криволинейных поверхностей)**, т.е. представляют собой полые оболочки. Толщина этой оболочки «нулевая» (условная), или по-другому – не принимается во внимание. А вот еще одно свойство поверхностей в *SketchUp* очень важно – каждая и всегда имеет две стороны – лицевую и изнаночную, точнее, в терминологии *SketchUp*, внешнюю – **Face** и внутреннюю – **Backface**. Стороны имеют определенные различия в свойствах, что в ряде опций редактирования принципиально – далее мы подробно будем рассматривать эту особенность.

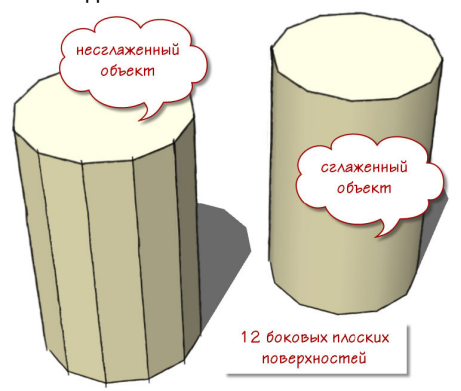
Возникает вопрос – а зачем тогда мы вообще говорим о твердотельном моделировании? Дело в том, что на самом деле не существует такого абсолютно «жесткого» разделения 3D-редакторов по этому принципу – как правило, в них в отдельных опциях применяются и тот, и другой алгоритмы. Так, в *SketchUp* всегда присутствовали (пусть в неявном виде) несколько типичных твердотельных опций, а с 8-й версии понятие и алгоритмы *solids* уже используется в виде полноценного набора специализированных инструментов.

Существует еще одно условное деление 3D-редакторов по используемым математическим моделям – на **curves-based** (на основе кривых, или *NURBS* (сплайновые) и **polygonal** (полигональные) – с поверхностями на основе плоских многоугольников. На их сравнении задерживаться не будем, можно привести только очень условную аналогию (понятную для знакомых с 2D-графикой), иллюстрирующую принципиальную разницу полигонального и сплайнового механизмов: сплайны – это как бы те же управляемые кривые Безье в векторных редакторах, а сетки полигонов можно сравнить с мозаикой пикселей растровых имиджей.

SketchUp по этим признакам – типичный **полигональный моделиер**. В таких редакторах основой всей геометрии является отрезок прямой линии – **Line**, а поверхности (грани) всегда состоят только из плоских **Polygons (Многоугольников)** со сторонами из этих прямых отрезков, которые в этом качестве являются **Edges (Ребрами)** объектов. Т.е. «настоящих» кривых линий и образованных из них плавно искривленных поверхностей в полигональном моделировании просто не существует! Выбор и устройство такой математической модели обусловлен стремлением обеспечить максимальную простоту и скорость редактирования, а также возможность назначения поверхностям объектов «отделочных» материалов, визуализации. Очевидным преимуществом полигональных моделей является и то, что они проще и «легче» в части вычислительной нагрузки на компьютер, а потому такие редакторы работают практически в реальном времени, с мгновенной реакцией на действия пользователя. Ну и кроме того, именно этот механизм предоставляет максимальные возможности для высококачественной визуализации трехмерных форм средствами специализированных редакторов фотореалистичного рендеринга и других решений задач подобного типа...

А как же быть с плавными криволинейными поверхностями? Поскольку в нашем распоряжении есть только плоские полигоны, предусматривается специальный механизм визуального сглаживания. Фактически при этом работает скрытие, зрительное размывание «жестких» границ между смежными поверхностями (т.е. этих границ мы просто не видим), хотя при этом никакого реального изменения геометрии не происходит. Таким образом создается иллюзия плавной криволинейности, что дает возможность даже при простой геометрии получать достаточно сложные формы. Или по принятой терминологии – строить т.н. **Low-Poly (Низкополигональные)** модели, т.е. с минимальным количеством формообразующих поверхностей. Все это обеспечивает высокую скорость редактирования за счет снижения нагрузки на «пересчет»

формы, а также «легкие» файлы моделей.



Запомним эту принципиальную основу полигонального моделирования, характерную и для *SketchUp* – ее непонимание является часто источником типичных затруднений многих начинающих.

Однако достоинства принятых механизмов моделирования оборачиваются в *SketchUp* и проблемной стороной – неизбежной повышенной нагрузкой на визуализацию, т.е. «перерисовку» изображения на экране при любом действии пользователя в реальном времени. Хотя на самом деле и эта проблема решается умелыми методами работы, о чем мы будем подробно говорить далее.

В завершении, для полноты картины, можно еще отметить характерные для *SketchUp* ограниченные возможности в части работы с т.н. «органикой» – т.е. с «нерегулярными» поверхностями сложной кривизны и высокой (*Hi-Poly*) детализации (природные объекты, персонажи и т.п.). Однако это уже совершенно другая «ниша» 3D-редакторов соответствующей специализации, обладающих оптимизированными под эти задачи механизмами...

ENG-RUS

Хотя эта тема напрямую не относится к содержанию учебника, хочу сразу, «на старте» изучения *SketchUp* сказать по этому поводу несколько слов, чтобы внести ясность и обозначить свою позицию по отношению к «русификациям» и «локализациям». Итак, типичная тема в обсуждениях какой-либо новой программы, новой версии и т.п.: – «А русификатор к ней есть?.. где найти?..». Сразу внесем ясность – термин «русификация» – это «народный» аналог понятия «локализация». В чем разница? Локализация – это выпускаемая разработчиком версия программы на определенном языке, которая, как правило, включает в себя переводы всех текстовых элементов интерфейса (меню и др.), а в идеале – и переведенные *Help* (Справку) или *Guide* (Руководство) – лучшие источники информации для изучения программы. А «русификация» – это самодельные любительские переводы текстовых элементов интерфейса, сделанные энтузиастами в меру своих способностей и возможностей, а потому чаще всего, как минимум – неполные, не говоря о прочих проблемах с качеством...

Итак, объективно ситуация с этой темой выглядит следующим образом:

1. Бытует мнение (скорее – заблуждение, почти миф), что русифицированную программу легче начинающему осваивать с «нуля»... Однако на самом деле это – не более чем иллюзия «знакомости», ведь пункты меню, кнопки – не более чем некие условные картинки, слова, ярлычки для запуска определенных опций! И от того, что вместо «*Follow Me*» видим пункт меню «*Ведение*»,

знаний о том, что это и как работает, никак не прибавится, поскольку за каждым таким условным обозначением может стоять несколько страниц инструкции по применению... А после того, как эти инструкции прочитаны, инструмент практически опробован, понят и освоен, проблема запоминания значения одного слова – пункта меню, просто снимается. Кстати, по этой же причине часто звучащий аргумент в пользу русифицированного интерфейса: – «Потому что английский не знаю», совершенно неубедителен – речь ведь идет не об изучении языка, а только об очень ограниченном наборе слов в интерфейсе программы! Ну и в конце концов, если пользователь не в состоянии запомнить эти понятия, может вообще стоит оценить смысл продолжения освоения такого сложного жанра, как 3D графика...

2. Изначальное освоение программы в русифицированном виде и работа в ней означает, что про исходные, «родные» понятия, названия и т.п. надо просто забыть, иначе придется периодически выяснять – «а как это называется по-английски?» (и наоборот). При этом ни о каком полноценном («на одном языке» в прямом и переносном смысле) общении с большинством коллег-пользователей не может быть и речи! Это относится и к освоению новой информации (в т.ч. учебной) и, естественно, добавляются проблемы перехода на новые версии, патчи, плагины и т.п. Пример из жизни – с общепринятым в 3D графике понятием *Edge* (линия – граница между двумя смежными поверхностями): как будто чтобы специально запутать и исказить смысл, его в общении называют по-русски и «краем», и «ребром», и «гранью», каждый раз долго выясняя при этом, что каждый из собеседников имел в виду... Причем, что самое забавное, рано или поздно все равно звучит этот термин на английском – только так приходит однозначное понимание у собеседников, о чем идет речь...

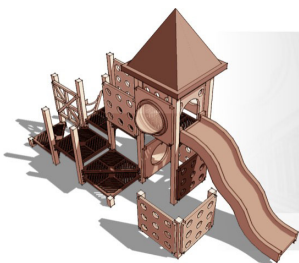
Иногда звучит аргумент сторонников русификаций, что это полезно на начальном этапе освоения программы, а потом можно перейти и на английскую версию. Но на самом деле это нереально и бессмысленно, а точнее означает только одно – последующее полное переучивание на совершенно другой интерфейс с ломкой уже наработанных навыков и автоматизма...

3. Возможности и качество собственно перевода – только в редких случаях и только для некоторых понятий можно подобрать полноценный точный аналог этой тематики на русском. В результате зачастую видим буквальный перевод, который совершенно не отражает суть в тех же названиях инструментов, т.е. основной аргумент сторонников русификации о «понятности», а значит собственно идея перевода вообще теряет смысл! Пример: даже в официальной русской локализации *SketchUp 7* одним и тем же понятием «масштаб, масштабирование» были переведены две принципиально разные опции – *Zoom (Лупа)* и *Scale (Масштабирование)*.

4. Даже в официальных локализациях не исключена вероятность появления дополнительных непредсказуемых программных ошибок и сбоев, а в любительских, понятны, тем более.

Можно продолжать и далее, но резюмируя только перечисленные проблемы, можно сказать, что нет ни одного рационального аргумента «за» русифицированный интерфейс. А если пользователь уже имеет опыт работы в англоязычных программах той же специфики (а очень многие термины и понятия в них аналогичны и повторяются), переход на русифицированные ничего, кроме появления дополнительных проблем, не дает.

Однако вы конечно заметили, что в уже прочитанном тексте все специальные термины и названия приведены в двойном (англо-русском) варианте – так будет и далее, поскольку изучать мы будем именно «английскую» версию программы с английским интерфейсом. Можете спросить – а зачем же тогда в этой книге автором все-таки приведены русские переводы элементов интерфейса? Да, на этапе начального освоения программы это имеет смысл – не более чем как «рабочий» прием для быстрого запоминания их функциональной сути с обязательным подробным истолкованием – что это, и как работает.



SketchUp просто 3D!

Книга 1. ПРАКТИК

Тема 2. ВСТРЕЧАЙТЕ – SKETCHUP!

В этой теме рассмотрим инсталляцию SketchUp, впервые откроем программу, и обзорно познакомимся с устройством ее интерфейса.

«СОФТ» И «ЖЕЛЕЗО»

Один из первых логичных вопросов, возникающих у будущих пользователей *SketchUp*: – «А будет ли программа работать на моем компьютере?»

Во-первых, *software* – доступны версии под операционные системы: *Windows (XP, Vista, 7, 8)* и *Mac OS X*. Во-вторых, *hardware* – с этим ситуация довольно своеобразная и требует некоторых комментариев. Поскольку программное ядро *SketchUp* разрабатывалось весьма давно (понятно, под «железо» той поры) и фактически осталось прежним, обладатели даже весьма «пожилых» компьютеров могут быть почти на 100% уверены, что проблем не будет. Другой вопрос – сложность, «тяжесть» конкретной модели, проекта, с которыми приходится работать – в этом случае весьма ощутимы возможности вашего компьютера, которые будут проявляться в комфортности и скорости работы.

Программа предъявляет весьма скромные требования к производительности процессора и оперативной памяти. Несколько сложнее ситуация с видеокартой – визуализация (рендеринг) в *SketchUp* происходит в режиме *realtime*, т.е. одновременно с процессом моделирования в реальном времени, с использованием стандартного графического 3D-движка *OpenGL*, который оптимально работает под управлением видеокарты. И хотя программа будет работать и на примитивном, интегрированном в «материнку» видеоадаптере (типично для компьютеров офисного класса, особенно - дешевых ноутбуков), на быструю и комфортную работу рассчитывать не приходится. Бывают случаи и критичной непригодности видеокарты (или ее драйвера) для работы в *SketchUp*. Ну а вообще серьезно заниматься графикой (а тем более – 3D) без подходящей видеокарты просто нереально... Какие видеокарты можно рекомендовать для работы со *SketchUp*? Такой однозначный список привести сложно, но по отзывам пользователей известно, что чаще проблемы возникают с картами *ATI* и наоборот, реже всего – с картами *NVIDIA*.

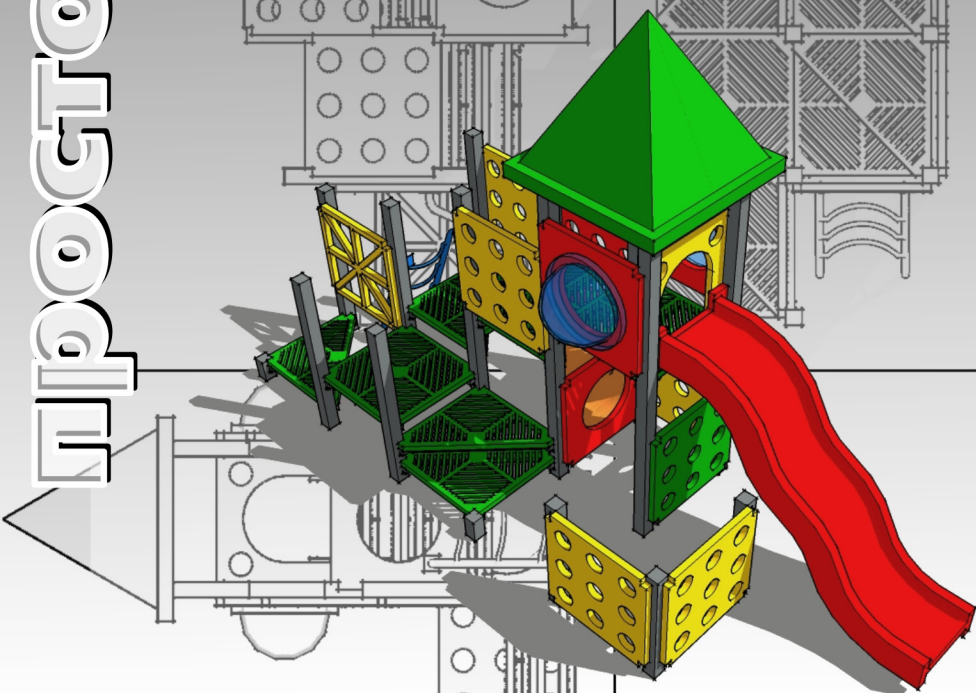
Конечно, на скорость работы влияет и общая производительность системы – понятно, что чем мощней, тем лучше! Однако для счастливых обладателей мощных «сегодняшних» компьютеров есть огорчительное обстоятельство – *SketchUp* «не умеет» в полной мере использовать, например, все возможности *multi-core* процессоров и будет работать только на одном «ядре». Таким образом, поменяв старенький процессор на несоизмеримо более мощный последних поколений, на явно ощутимый эффект «в разы» рассчитывать не приходится. Кроме того, хотя *SketchUp* будет без проблем запускаться на 64-bit версиях *Windows*, но работать будет как 32-bit приложение...

SketchUp

учебник-справочник

ПРОСТО 3D!

Книга 2. ЭКСПЕРТ

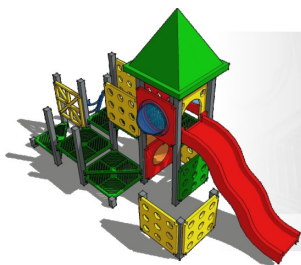


Версия 2:
Google/Trimble SketchUp Pro 8

2011-2013

ВЕДЕНИЕ	4
Тема 15. КОМПОНЕНТЫ	
Make Component (Создать компонент)	5
Редактирование компонентов	9
Components (Браузер компонентов)	13
Моделируем компонентами (учебный проект)	19
Outliner (Структура групп и компонентов)	24
Тема 16. РАЗМЕРЫ, ТЕКСТЫ	
Dimension (Размеры)	27
Text (Текст)	32
3D Text (3D-текст)	35
Тема 17. РАЗРЕЗЫ	
Section Plane (Секущий план, плоскость)	37
Display Section Planes (Показать секущую плоскость)	40
Display Section Cuts (Показать отсечения)	40
Тема 18. РЕЛЬЕФЫ	
From Contours (Из контуров)	43
From Scratch (Из линий)	45
Smooove (Присоска)	45
Stamp (Штамп)	47
Drape (Драпировка)	49
Add Detail (Добавить детали)	50
Flip Edge (Отразить ребро)	51
Тема 19. МОДЕЛИРОВАНИЕ ОТСЕЧЕНИЯМИ	
Intersect Faces (Отсечение поверхностей)	52
Solid tools (Инструменты сплошных тел)	55
Тема 20. ИМПОРТ/ЭКСПОРТ 2D-ГРАФИКИ	
Импорт растровых файлов	61
Экспорт растровых файлов	64
Распечатка на принтере	66
Импорт векторных файлов	67
Экспорт векторных файлов	70

Тема 21. СОВМЕЩЕНИЕ С ФОТО	
Выбор, подготовка фото	77
Загрузка фото, настройка камеры	78
Построение 3D-модели по фото	80
Проецирование фото (текстуры) на модель	81
Размещение модели на «фотосцене»	82
Тема 22. ИМПОРТ/ЭКСПОРТ 3D-МОДЕЛЕЙ	
Импорт 3D-моделей	84
Экспорт 3D-моделей	86
Тема 23. GOOGLE. ГЕОЛОКАЦИЯ	
Инструменты Google	93
Геолокация	97
Авторские права	100
Тема 24. ДИНАМИЧЕСКИЕ КОМПОНЕНТЫ	102
Управление DC в сцене	103
Разработка DC	105
Тема 25. УПРАВЛЕНИЕ ПРЕЗЕНТАЦИЯМИ	107
Анимация – настройки параметров	108
Walkthrough (инструменты Прогулки)	110
Fog (Туман)	113
Анимация – запись в фильм	114
Серия покадровых снимков	116
Google SketchUp Viewer	116
Тема 26. УПРАВЛЕНИЕ ПРОЕКТОМ	
Главные меню	117
Preferences (Основные настройки)	126
Тема 27. ЗАДАЧИ И РЕШЕНИЯ. ПЛАГИНЫ. ПРОГРАММЫ-ПАРТНЕРЫ	133



SketchUp просто 3D!

Книга 2. ЭКСПЕРТ

ВВЕДЕНИЕ

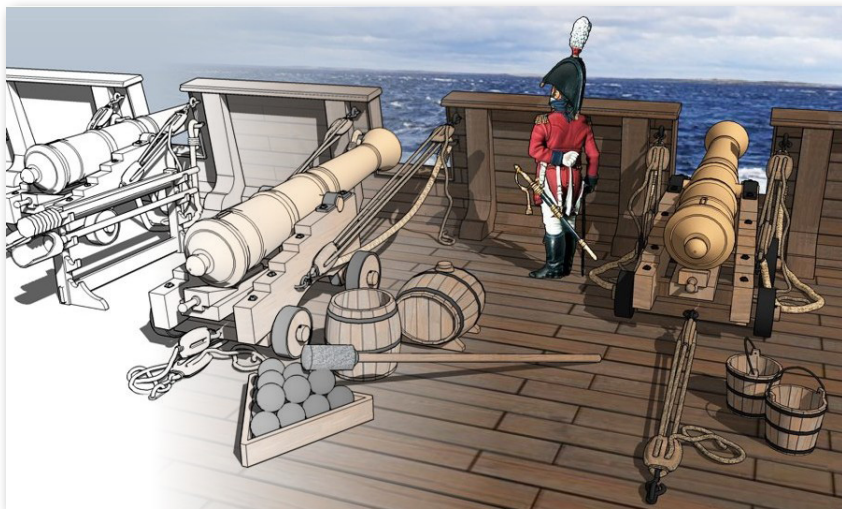
Освоив инструменты и опции, рассмотренные в первой части (книге) нашего курса, вы уже достаточно свободно можете работать в *SketchUp*, успешно решая подавляющее число типичных практических задач. Однако «за кадром» у нас пока еще остался целый ряд возможностей программы, не менее интересных и нужных чем уже знакомые, – в том числе некоторые из них предназначены для решения таких специфических задач, которые просто не могут быть реализованы без их применения.

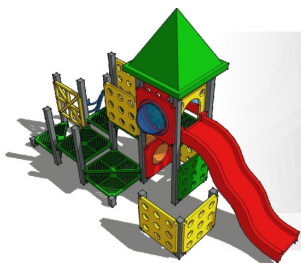
Кроме того, пройдем по ряду частично рассмотренных ранее инструментов и опций, полностью завершив их изучение.

Особое внимание уделим также финальной стадии работы над проектом, его наглядной и эффектной презентации, «подаче» – ведь это не менее важная составляющая работы профессионала, чем собственно моделирование.

Не менее интересна тема, мимо которой никак не пройдет продвинутый «скетчапер», – многочисленные скрипты-плагины, расширяющие возможности программы настолько, что можно говорить о совершенно другом *SketchUp*...

Ну и в завершение курса будут представлены некоторые подсказки и «хитрости» из опыта «экспертов» в программе, в том числе решающие отдельные «врожденные» проблемы и ограничения *SketchUp*...





SketchUp просто 3D!

Книга 2. ЭКСПЕРТ

Тема 15. КОМПОНЕНТЫ

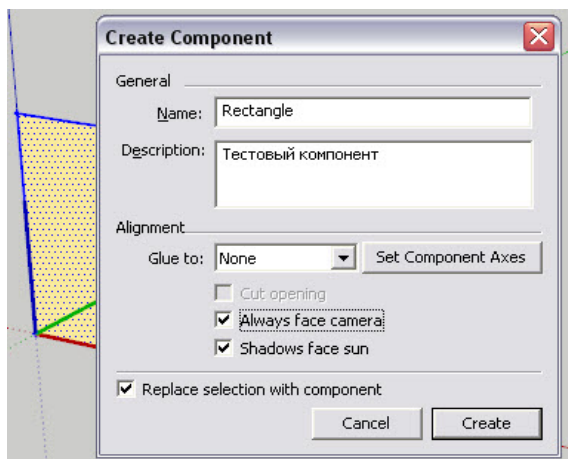
Начиная изучение инструментов в первой части курса, в группе Principal (Основные) мы уже упоминали опцию (и кнопку) *Make Component* (Создать компонент). Тогда мы отложили эту очень важную для эффективной работы тему – сейчас разберемся с ней подробно.



MAKE COMPONENT (СОЗДАТЬ КОМПОНЕНТ)

Выбираем абсолютно любой элемент построения (даже отдельную линию) и видим, что тут же кнопка опции показывает готовность к применению – выходит из «затененного» состояния и выделяется голубым контуром. Включение *Make Component* доступно также и из меню *Edit*, и из контекстного меню выбранного – используем то, что привычней и удобней. А обратная опция – возвращение составляющих компонент элементов в исходное «разрозненное» состояние – та же, что и для группы – *Explode* (Разорвать).

Посмотрим, как это работает, на простейшем тестовом примере – строим прямоугольник, выделяем его весь, включаем опцию *Make Component* – открывается диалоговое окно *Create Component* (Создать компонент), в котором назначаются его параметры (свойства).



Сразу отметим главное: по сути, *Компонент* – это «продвинутая» группа (то есть обладает всеми ее возможностями), имеющая одно основное дополнительное уникальное свой-

ство – способность существовать одновременно в двух состояниях – **Definition (Оригинала)** и **Instance (Вставки)** в сцену, между которыми поддерживается постоянная интерактивная связь. Это означает, что каждая вставка постоянно «считывает» все свойства оригинала и, соответственно, редактирование самого оригинала тут же одновременно воздействует на все его вставки в сцене. И наоборот – «внутреннее» редактирование любой из вставок одновременно редактирует тем же образом и все другие вставки в сцене и оригинал компонента в проекте.

General (Общее):

Name – назначенное программой автоматически индивидуальное имя оригинала компонента с порядковым номером (может быть и так) в виде **Component #**, которое, естественно, тут же можно изменить на свое (еще раз напомним – лучше сразу на латинице).

Description (Описание) – здесь при желании можно вписать любую дополнительную (справочную) информацию, которая, конечно, на собственно моделирование никак не влияет.

Aligment (Выравнивание):

Здесь устанавливаются параметры ориентации и воздействия на другие элементы построений, с которыми вставки компонента будут появляться в сцене:

Glue to (Приклеить к) – этот выпадающий список используется для назначения принудительной ориентации «основания» вставки, которым она будет «садиться» на поверхности в сцене: **None (Никакая)**, **Any (Любая)**, **Horisontal (Горизонтальная)**, **Vertical (Вертикальная)**, **Sloped (Наклонная)**. У такой вставки при создании компонента появляется серый прямоугольник, обозначающий план (основание) приклеивания на поверхности (кроме варианта **None**). Например, если назначили вариант **Horisontal**, то уже не сможем поместить вставку основанием на вертикальную поверхность. Не вдаваясь в дальнейшие подробности, можно сразу сказать, что опция отличается насколько путаной логикой, что польза от нее весьма сомнительна, потому на практике рациональней при создании любого компонента обойтись назначением ориентации **Any**, а лучше **None** (что и предлагается по умолчанию). Намного проще осевую ориентацию установки вставки поменять в любой момент по конкретным обстоятельствам опциями редактирования осей оригинала компонента (см. далее). Впрочем, ничто не мешает при желании разобраться с этим механизмом самостоятельно по подробней и использовать в работе, если так покажется удобным...

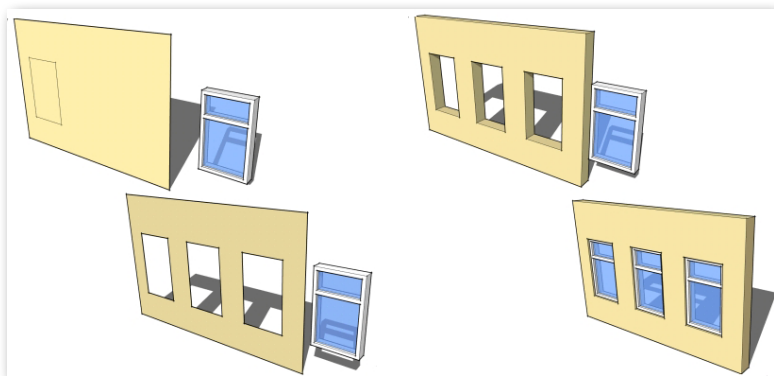
Set Component Axes (Установить оси компонента) – опция позволяет назначить оригиналу компонента «индивидуальные» оси, независимые от общих осей сцены (по умолчанию они назначаются при создании автоматически по направлениям основных осей сцены). Надо подчеркнуть, что этим чрезвычайно полезным свойством обладают только **Компоненты** из всех элементов построений **SketchUp**, что позволяет редактировать их геометрию, ориентированную любым образом в пространстве сцены, а также то, каким образом они появляются и работают в сцене.

Порядок действий здесь аналогичен тому, что мы уже изучали в инструменте управления осями сцены **Axes** (см. тему 8 «Строим точно»), – после выбора опции курсор изменится на символ с осями, который нужно поместить и кликнуть в точке начала («нуля») координат компонента. Выбор «нуля» зависит от характера объекта-компонента и планов по его дальнейшему использованию. Например, для компонентов, которые должны «стоять на земле», имеет смысл назначить его на **endpoint** какого-либо ребра поверхности низа основания (на-

пример, ножки стула) – тогда при помещении в сцену он будет автоматически «ставиться на землю» нужным образом. Далее вращаем символ осей до нужной ориентации красной оси (фиксируем вторым кликом) и завершаем опцию третьим кликом, показав направление зеленой оси (синяя ось устанавливается автоматически по положению красной и зеленой). Отметим еще раз, что положение «нуля» и ориентацию осей оригинала компонента можно быстро поменять и впоследствии, на уже готовом объекте его контекстной опцией **Change Axes** (**Сменить оси**), которая работает аналогично.

Cut Opening (Поместить с вырезанием) – опция становится доступна при выборе любого варианта *Glue to* (*Приклеить к*), отличного от *None*. И хотя рекомендовал этой опцией вообще не пользоваться, все же стоит знать, что *Cut Opening* используется для назначения вставке компонента свойства «прорезания под себя» сквозного проема в поверхности, на которую она «приклеивается» своим основанием в соответствии с ориентацией, назначенной в *Glue to*.

Здесь стоит задержаться и ответить на постоянно повторяющийся вопрос начинающих пользователей по типичной ситуации: – как размещать компоненты-окна на стене и сразу прорезать под них проемы с помощью этой опции? Короткий ответ – фактически никак... Кроме уже указанной проблемы сложности настройки этих параметров, есть и еще одна, весьма весомая – *Cut Opening* действует только на одну, единичную поверхность, а поскольку стены в построениях обычно имеют некую толщину, будет прорезана только ее внешняя поверхность, а внутренняя – нет. Есть разные приемы решения этой задачи, но в любом случае потребуются «многошаговые» манипуляции с дополнительными построениями, использованием механизмов «отсечения» объемов (о них далее), удалением лишнего и так далее. Отсюда еще раз напрашивается вывод: – практическая польза опции сомнительна, особенно в сложных случаях, поэтому рациональней решать задачи такого рода другими способами. Например, для той же вставки окна в стену быстрее и удобнее просто заранее разметить и «выдавить» проемы в габариты окна, а затем «размножить» по ним окна-компоненты перемещением с копированием, как показано на рисунке ниже.



Always face camera (Всегда лицом к камере) – используется для назначения свойства вставке компонента всегда быть «повернутой лицом» к камере. Что это значит, для чего используется и как правильно установить эту функцию? Очень просто: – «лицо» – это такое положение компонента в момент создания, при котором оно («лицо») находится в плоскости включенного вида *Front* (*Спереди*). Или по-другому – в красно-синем плане осей. То есть,

например, сначала включаем вид *Front*, затем выставляем в нем объект – будущий компонент «лицом» к камере, и потом при создании компонента в диалоговом окне включаем (ставим галочку) на пункте *Always face camera*, который и фиксирует эту ориентацию как «лицо» компонента. Заметим, что вращаться за камерой вставка компонента будет относительно своей вертикальной (синей) оси и точки «нуля» осей. И конечно, в любой момент можем скорректировать «лицевую ориентацию» на вставке компонента опцией *Change Axes* (удобней – через контекстное меню), о которой было сказано выше.

Always face camera – очень нужная и полезная функция для использования в подходящих задачах 2D («плоских») объектов-компонентов в качестве альтернативы несоизмеримо более «тяжелым» аналогичным 3D-объектам. Этот достаточно типичный механизм (и прием) для многих редакторов и программ-рендеров с успехом используется для наполнения сцены «массовкой» средних и дальних планов – деревьями, фигурами людей и т. п. В сочетании с грамотной постановкой камеры, эти объекты (*общепринятый термин – billboard*) в сцене практически визуально неотличимы от «настоящих» 3D-объектов. Подробнее создание *billboard* рассмотрено в конце нашего курса.



Shadows face sun (Тень от лица) – опция работает только при включенной предыдущей (*Always face camera*) для отображения падающей тени от контурных очертаний 2D-вставок компонентов таким образом, как будто «солнце» светит им точно «в лицо» при любом повороте камеры (и самой вставки компонента).

Как это работает, показано на примере ниже, где левый прямоугольник – обычный объект, средний – вставка компонента с включенной опцией ***Shadows face sun***, правый – с выключенной. Как видим, не всегда при включение этой опции вставка компонента однозначно «правильней» выглядит – это зависит и от характера очертаний ее контура, повторяемого тенью, и от места «нуля» его осей, и от текущего ракурса обзора сцены – пробуем, выбираем оптимальный, наиболее «естественный» в данном случае (компоненте) вариант...

