

Tallinna Ülikool
Informaatika instituut

3D modelleerimine programmi K-3D baasil
3D моделирование на базе программы K-3D

Seminaritöö

Autor: Juri Snežkov

Juhendaja: lektor Andrus Rinde

Tallinn 2009

| | |
|---|-------------------------------------|
| Введение | 3 |
| 1. Особенности K-3D..... | 4 |
| 1.1 Архитектура системы K-3D..... | 4 |
| 1.2 Поддержка практически всех платформ..... | 4 |
| 1.3 Поддерживаемые форматы | 4 |
| 1.3 Низкие системные требования | 7 |
| 1.4 Лицензия..... | 7 |
| 1.5 Иерархически упорядоченное, неограниченное количество отмен и повторов(undo/redo)..... | 8 |
| 1.6 Написание скриптов в системе K-3D..... | 8 |
| 1.7 Визуализация (Рендеринг) | 9 |
| 1.8 Анимация..... | 9 |
| 1.9 Моделирование | 10 |
| 2. Сравнение K-3D и Blender'a. | 11 |
| 2.1 Особенности..... | 11 |
| 3. Первые шаги, перед началом трехмерного моделирования в системе K-3D..... | 13 |
| 4. Работа с объектом и его составляющими..... | 15 |
| 5. Работа с сеткой..... | 27 |
| 6. Работа с визуализируемыми цветами. | 29 |
| 8. Пример трехмерного моделирования в K-3D. | 30 |
| Заключение..... | 34 |
| Ülevaade..... | Error! Bookmark not defined. |
| Список использованных в работе источников..... | 37 |

Введение

На данный момент имеется достаточно большое количество различных программ для работы с 3D графикой, как коммерческих, так и бесплатных. Некоторые из них, к примеру, Blender, довольно известны. Есть и менее известные, но почти ни в чем не отстающие, а иногда и превосходящие по возможностям программы. Автор данной работы открыл для себя программу, которая называется К-3D, и возможности которой, на его взгляд, достаточно впечатляющие.

Цель данной работы — познакомить читателей с особенностями системы К-3D, а также с возможностью создания в ней трехмерных моделей. После прочтения работы у читателя должно возникнуть четкое представление о том, что же представляет из себя система К-3D и как на базе данной системы можно создать собственные трехмерные модели.

Для достижения данной цели автор будет использовать в своей работе материалы с сайта разработчиков, а также свои практические наблюдения, описывающие интерфейс К-3D, его основные свойства и функции, и приведет примеры ключевых моментов в трехмерном моделировании посредством данной программы.

1. Особенности K-3D.

K-3D это полностью бесплатная система для создания трехмерных моделей, анимаций и изображений. Создатель системы — Тим Шид (*Tim Shead*) начал работу над ней еще в 1994 году [FAQ, вопросы “*What is K-3D?*”, “*Who wrote K-3D?*”, <http://www.k-3d.org/wiki/FAQ>]. Скачать K-3D можно с официального сайта <http://www.k-3d.org>. Далее автор проведет общий обзор K-3D, на основе мыслей разработчиков, отмеченных в разных разделах официального сайта.

1.1 Архитектура системы K-3D

Технологически высоко развитая архитектура системы K-3D предоставляет пользователям обширные возможности использования сторонних плагинов (*Plug-in*). Практически все элементы системы K-3D являются заменяемыми плагинами. Таким образом, пользователь может подобрать под себя наиболее удобные на его взгляд элементы, заменив стандартные. [FAQ, вопрос “*Which are the main features of K-3D?*”, <http://www.k-3d.org/wiki/FAQ>]

1.2 Поддержка практически всех платформ

Система K-3D поддерживается практически всеми платформами, включая следующие:

- GNU / Linux
- Win32
- MacOSX
- BSD
- Solaris

[Features, раздел *Platforms*, <http://www.k-3d.org/wiki/Features>]

1.3 Поддерживаемые форматы

Форматы геометрии: Wavefront OBJ и формат RAW полностью поддерживаются, другие

являются опциональными и зависят от инсталляции PLIB. Экспериментальные плагины включают в себя OpenFX, OFF, RIB и X форматы.

Форматы изображений:

- JPEG (все платформы).
- PNG (все платформы).
- TIFF (все платформы).
- OpenEXR (требуется опциональный плагин OpenEXR).
- BMP (требуется опциональный плагин ImageMagick).
- SUN (требуется опциональный плагин ImageMagick).

[Features, раздел *Supported Formats*, <http://www.k-3d.org/wiki/Features>]

Подробнее о форматах геометрии:

Благодаря плагин-ориентированной архитектуре, K-3D может экспортировать и импортировать файлы различных форматов. Некоторые из них требуют инсталляции проекта PLIB(*Portability Libraries*), целью которого является создание переносимых, системно независимых игр и интерактивных приложений. Перечислять абсолютно все форматы не имеет особого смысла, поэтому автор опишет только полностью поддерживаемые и экспериментальные форматы геометрии, не требующие инсталляции PLIB.

OBJ — это геометрические определения формата файла, впервые разработанные в Wavefront Technologies для их анимационного пакета Advanced Visualizer. Формат файла является открытым и был принят другими разработчиками приложений 3D графики и может быть экспортирован/импортирован в K-3D, e-Frontier's Poser, Autodesk's Maya, XSI, Blender, MeshLab, Misfit Model 3D, 3D Studio Max, Rhinoceros 3D, Hexagon, Newtek Lightwave, Art of Illusion и т. д. По большей части это общепринятый формат. Это простой формат данных, который содержит только 3D геометрию, а именно, позицию каждой вершины, связь координат текстуры с вершиной, нормаль для каждой вершины, а также параметры, которые создают полигоны. [Object Files, <http://local.wasp.uwa.edu.au/~pbourke/dataformats/obj/>]

RAW — формат данных, содержащий необработанные (или обработанные в минимальной степени) данные, что позволяет избежать потерь информации, и не имеющий четкой спецификации. В таких файлах содержится полная информация о хранимом сигнале, и она может быть несжатой, сжатой без потерь, или сжатой с потерями. [*Брюс Фрейзер* Реальный мир Camera RAW и Adobe Photoshop CS2 = World Camera RAW with Adobe Photoshop CS2, 2007. — С. 304.]

OpenFX – собственный формат одноименной программы, которая также как и K-3D, представляет из себя бесплатную студию моделирования и анимации с открытым кодом.

OFF(Object File Format) – используется для предоставления геометрии модели путем определения полигонов поверхности модели, при этом полигоны могут иметь любое количество вершин. [Object File Format, http://shape.cs.princeton.edu/benchmark/documentation/off_format.html]

RIB - Многие программы трехмерного моделирования не имеют высококлассных средств визуализации. Когда изображение подвергается визуализации, то ему придаются освещенность, текстура и цвет. Формат RIB (RenderMan Interface Bytestream protocol) содержится в самих программных средствах моделирования. Таким образом, пользователи могут экспортировать свои файлы в программу визуализации RenderMan компании Pixar. [Использование цифровых файловых форматов, <http://lesscan.info/5.php>]

X - формат файла для хранения 3D объектов, созданный компанией Microsoft. Данный формат файла является свободно распространяемым. Этот формат хранит информацию о геометрии 3D объекта (координаты вершин и координаты нормалей), текстурные координаты, описание материалов, пути и названия к текстурам, которые используются. Хранится иерархия объектов, хранится анимация, и хранятся привязки вершин к «костям» с описанием весов. В X файле может отсутствовать какая-либо информация об объекте (Например, в X файле могут содержаться только координаты вершин). [X

File Format Reference, <http://msdn.microsoft.com/en-us/library/bb173014.aspx>]

1.3 Низкие системные требования

K-3D будет работать на любом компьютере, но для полной реализации возможностей K-3D необходима поддержка графического аппаратного ускорителя OpenGL
Рекомендуемые системные требования:

- **CPU:** 400Mhz
- **RAM:** 256 MB
- **Video:** Хорошая OpenGL карта
- **Hard Disk :** Должно хватить 150 MB. (но если вы собираетесь пересобрать ядро системы K-3D то, в зависимости от вариантов настроек, понадобится около 0.5 — 2 GB свободного места)
-

Также стоит учесть, что большинство документаций связанных с системой K-3D находятся в Интернет сети. Поэтому к системным требованиям стоит добавить и наличие интернет-соединения.

[FAQ, вопрос “*What are the hardware requirements for running K-3D?*” <http://www.k-3d.org/wiki/FAQ>]

Для своей работы я использовал домашний компьютер со следующими параметрами:
CPU: 2.0 Ghz Dual Core, RAM: 1024 MB, Video: Radeon X1600 256 MB. Естественно, с такими параметрами компьютера, работа в K-3D проходила без каких-либо проблем. Среднее время рендеринга(*Rendering*) было около трех секунд, хотя оно конечно больше зависит от того, какое количество данных необходимо обработать. Большинство моих моделей были простыми, более сложные будут обрабатываться дольше по времени.

1.4 Лицензия

В 1999 K-3D получила лицензию «free-as-in-freedom GNU General Public License »
[FAQ, вопрос “*Who wrote K-3D?*”, <http://www.k-3d.org/wiki/FAQ>]. Эта лицензия

позволяет бесплатно копировать, распространять дистрибутив системы K-3D и изменять ее программный код(*Open source*).

Таким образом программа является полностью бесплатной и имеет ряд других свобод. (с текстом лицензии на английском языке можно ознакомиться здесь <http://www.k-3d.org/wiki/GPL>)

1.5 Иерархически упорядоченное, неограниченное количество отмен и повторов(*undo/redo*)

Каждое внесённое изменение в вашем 3D проекте в K-3D запоминается программой и вносится в *Node History*. Это позволяет вносить изменения, перескакивая на несколько шагов назад и наблюдать каким образом они изменят весь проект в целом.

1.6 Написание скриптов в системе K-3D

Любые простые скрипты(*Script*), созданные пользователями, могут быть использованы для различных целей. К примеру, скрипты могут быть запущены из командной строки для запуска частичной обработки(*batch processing*), или при помощи пользовательского интерфейса для автоматизации повторяющихся задач. Благодаря системе плагинов в K-3D, скрипты могут быть интегрированы в программу таким образом, что они не будут отличаться от скомпилированных в пользовательском интерфейсе C++ плагинов. Также в K-3D скрипты могут быть автоматически запущены, когда включаешь или выключаешь программу, создаешь или закрываешь документ. Скрипты в K-3D можно использовать и для записи интерактивных туториалов(*Tutorial*) [Scripting, раздел *Overview*, <http://www.k-3d.org/wiki/Scripting>]. Далее автор перечислит движки для написания скриптов, которые поддерживает система K-3D.

- Движок Python (рекомендуемый движок для создания новых скриптов).
- Движок K3DScript (движок для написания незначительных скриптов для туториалов и макросов).
- Поддержка альтернативных движков для написания скриптов, которые могут

быть установлены в систему K-3D в виде плагинов.

[Scripting, раздел *Scripting Languages*, <http://www.k-3d.org/wiki/Scripting>]

1.7 Визуализация (Рендеринг)

Команда разработчиков настоятельно рекомендует использовать для визуализации изображений интерфейс Aqsis как наиболее приспособленный к системе K-3D. Кроме Aqsis, система поддерживает также следующие интерфейсы:

- RenderMan(высокоуровневый интерфейс программирования приложений для описания трехмерных сцен для движков рендеринга): Aqsis, Pixie, VMRT, PRman, 3Delight, Render Dot C.
- Альтернативные движки и модели рендеринга. К примеру, Yafray, созданный первоначально для Blender'a.

[Features, раздел *Rendering*, <http://www.k-3d.org/wiki/Features>]

- Так как все указанные авторами движки рендеринга работают автономно, это дает пользователям преимущество — они могут продолжать работать с программой, во время выполнения процесса рендеринга или добавить в очередь несколько задач рендеринга.

[FAQ, вопрос “*Which rendering engines does K-3D support?*”, <http://www.k-3d.org/wiki/FAQ>]

1.8 Анимация

- Возможность профессиональной анимации кинематографического качества
- Часть любого объекта может быть соединена с любой совместимой частью другого объекта
- Анимация любых элементов.
- Неограниченное число каналов анимации.
- Каналы кривых Безье.
- Анимация операций моделирования.

[Features, раздел *Animation*, <http://www.k-3d.org/wiki/Features>]

1.9 Моделирование

- Мощный график сцен процедурного моделирования с полной историей моделирования.
- Выбор: Объекты, сетки, грани, края, заплатки, кривые, группы точек, точки.
- Типы геометрии: Полигон, сплайн, подразделение, *blobby*.
- 3D Примитивы: Конус, окружность, подушка, цилиндр, диск, сетка, параболоид, многогранник, сфера, тор.
- Текст: поддержка FreeType2.

Инстанцирование: создание дублей без добавления геометрии в сцену.

[Features, раздел *Modeling*, <http://www.k-3d.org/wiki/Features>]

2. Сравнение K-3D и Blender'a.

Для многих программ по трехмерному моделированию Blender является главным конкурентом. От пользователей поступало много вопросов по этому поводу, поэтому разработчики K-3D решили сравнить свой проект с Blender'ом и дать по этому поводу свою трезвую оценку, выложив свои мысли на официальном сайте K-3D в разделе часто задаваемых вопросов - <http://www.k-3d.org/wiki/FAQ>. На данный момент Blender имеет более полный набор функций, чем K-3D, в то же время K-3D имеет больше амбиций за счет своей архитектуры. Многие, включая кодеров Blender'a признают, что всем элементам Blender'a необходима полная смена дизайна и переписывание программного кода. С другой стороны элементы K-3D часто хвалят за их дизайн и за то что все элементы в этой системе являются плагинами. Хотя в K-3D отсутствуют некоторые важные особенности, основа этой системы, по мнению разработчиков, является непревзойденной и многообещающей. Таким образом у K-3D большой потенциал за счет уникальной архитектуры и прочного фундамента на котором можно построить все недостающие функции и элементы, имеющиеся у других программ и изобрести новые.

2.1 Особенности

- K-3D интерфейс нового поколения использует GTK2, чтобы предоставить графический пользовательский интерфейс, который может быть интегрирован на любую платформу. Так как все элементы в K-3D являются плагинами, включая пользовательский интерфейс, то он может быть заменен при желании на другие специализированные интерфейсы.
- Поддержка повторов/отмен(*undo/redo*) в Blender'e является недавним дополнением, а в K-3D эта возможность была реализована с самого начала. K-3D поддерживает не только неограниченное количество повторов/отмен для любых действий пользователя, включая добавление скриптов, но и иерархические повторы/отмены, которые позволяют пользователю перескакивать назад и вперед через множества изменений, связанных в ветви.
- У Blender'a имеется встроенный движок для рендернга и скудная поддержка интеграции других движков, таких как Aqsis. В K-3D же все части являются

плагинами, включая движки рендеринга, освещения и материалы, поэтому практически любой другой движок рендеринга, как и другие элементы, может быть интегрирован в K-3D.

- K-3D — процедурный модельщик, а это значит, что он хранит всю историю пользовательских изменений произведенных при создании модели. Пользователь может войти в историю своей модели, внести изменения и посмотреть, как эти изменения повлияют на последующие шаги в истории.
- K-3D с самого начала была написана так, чтобы поддерживать стандарт RenderMan, который используют профессионалы для создания сложных компьютерных анимаций.

[FAQ, вопрос "*How does K-3D compare to Blender?*", <http://www.k-3d.org/wiki/FAQ>]

3. Первые шаги, перед началом трехмерного моделирования в системе K-3D.

Перед тем как начать создание собственных серьезных моделей, рекомендуется для начала ознакомиться с основными функциями и возможностями системы K-3D путем простейших экспериментов. Интерфейс программы интуитивно понятен — и это я считаю большим плюсом. Даже не знакомый с моделированием человек сможет довольно быстро разобраться в K-3D. Управление камерой также является довольно простым. Все возможные комбинации поворотов камеры производятся при помощи клавиши Shift, правой кнопки мыши и колесика для скроллинга. На что же стоит обратить внимание в первую очередь?

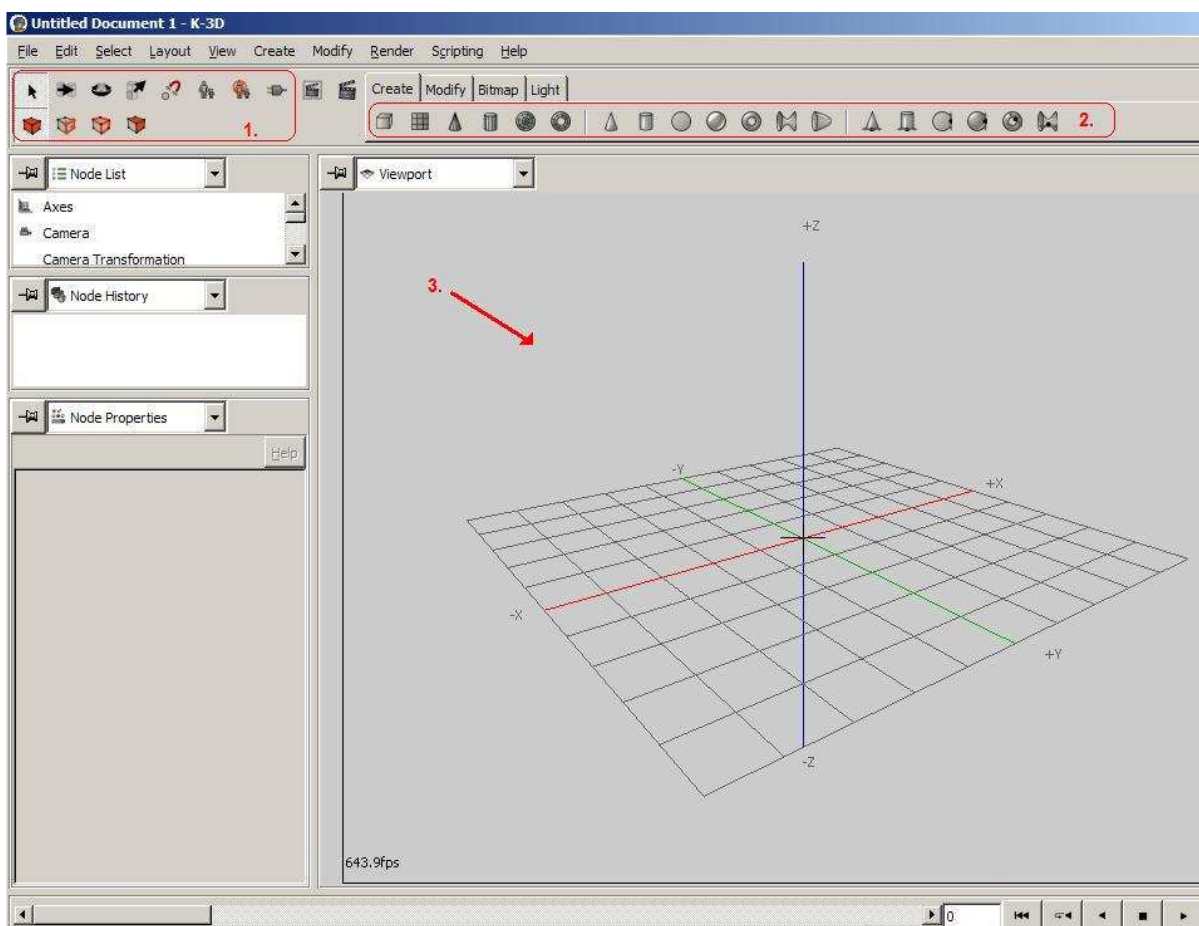




Рисунок 1. Интерфейс K-3D

На *Рисунке 1* автор данной работы обозначил цифрами то, с чем стоит разобраться с самого начала. Цифрой 1. - обозначены инструменты для работы с объектом. Сюда входит изменение положения, размера, поворот и притягивание объекта, работа с

гранями, ребрами и точками. Цифрой 2. обозначены стандартные заготовки объектов, такие как куб, цилиндр, сфера, конус и другие, более сложные объекты. Цифрой 3. обозначена плоскость, на которой и будет происходить создание трехмерной модели.

4. Работа с объектом и его составляющими.

Рассмотрим простейшие инструменты для работы с объектом. Выбираем режим *Select Objects*, обозначенный данным рисунком - . Это режим для работы с выбранным ранее цельным объектом. Для примера автор выбрал своим объектом — куб, который и буду использовать далее в своих примерах. Сначала рассмотрим возможности передвижения куба. Для этого выбираем функцию *Move*, обозначенную данным рисунком - .

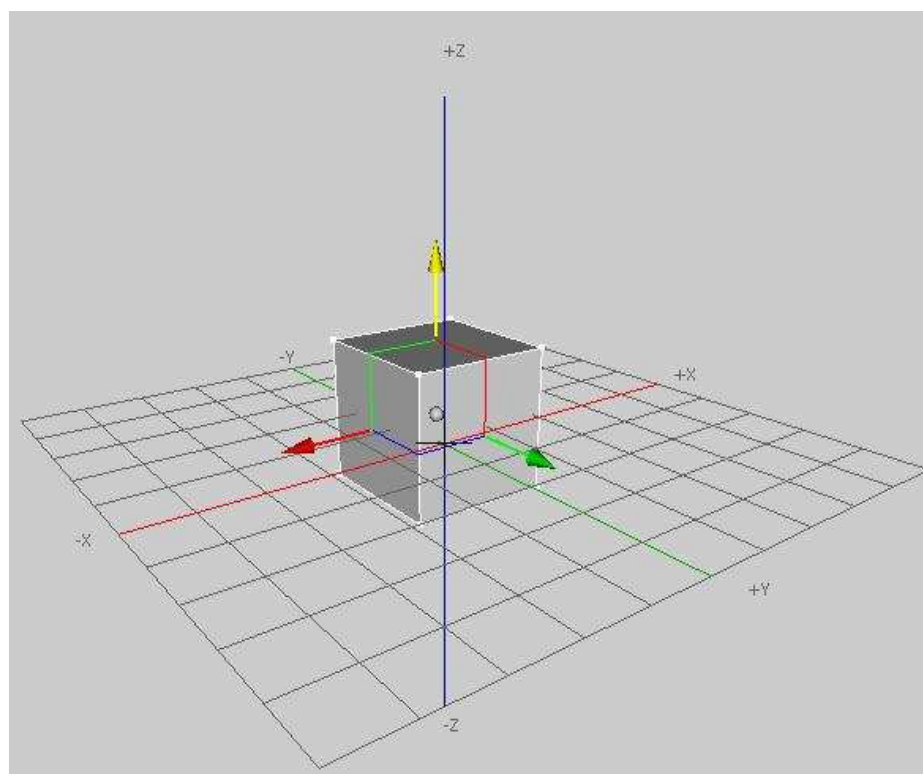



Рисунок 2. Режим *Select Objects*, функция *Move*

На *Рисунке 2* мы видим три разноцветных стрелки и шарик в центре объекта. Данные стрелки предназначены для передвижения объекта по осям X(красная), Y(зеленая), Z(желтая). Шарик же предназначен для произвольного передвижения объекта на плоскости.

Далее рассмотрим возможности вращения куба. Для этого выберем функцию *Rotate*, которой соответствует рисунок - .

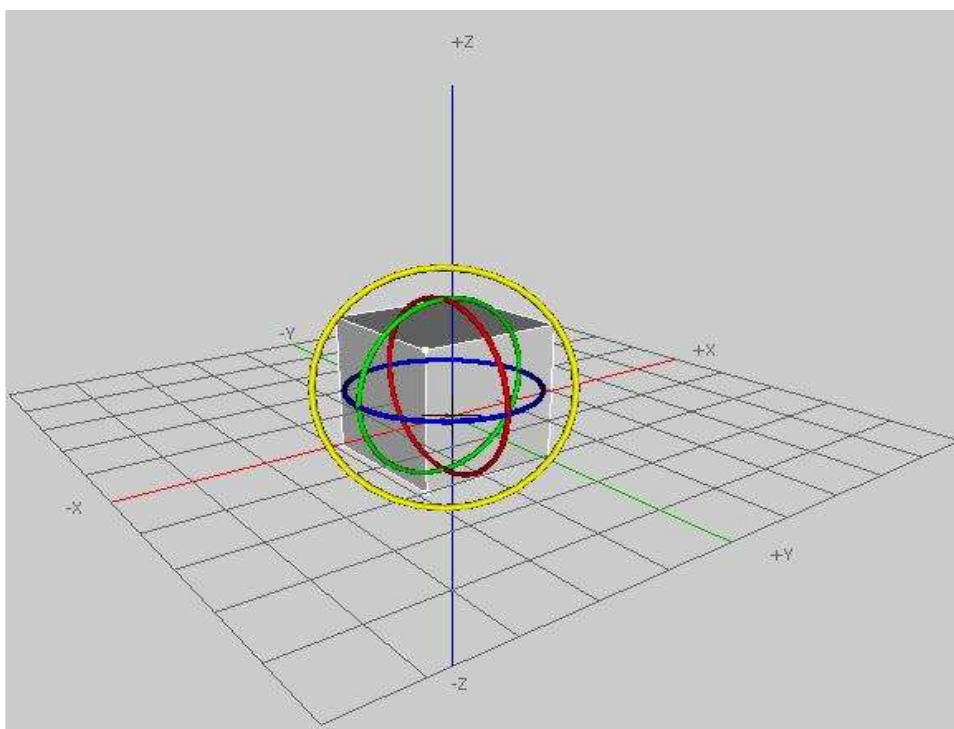



Рисунок 3. Режим Select Objects, функция Rotate

На *Рисунке 3* мы видим разные кольца — потянув за любое из них, можно вращать куб по выбранному кольцу. Таким образом, кольца являются как бы возможными орбитами вращения объекта.

Теперь перейдем к рассмотрению возможности изменения размера куба. Для этого выбираем функцию *Scale*, которой соответствует рисунок - .

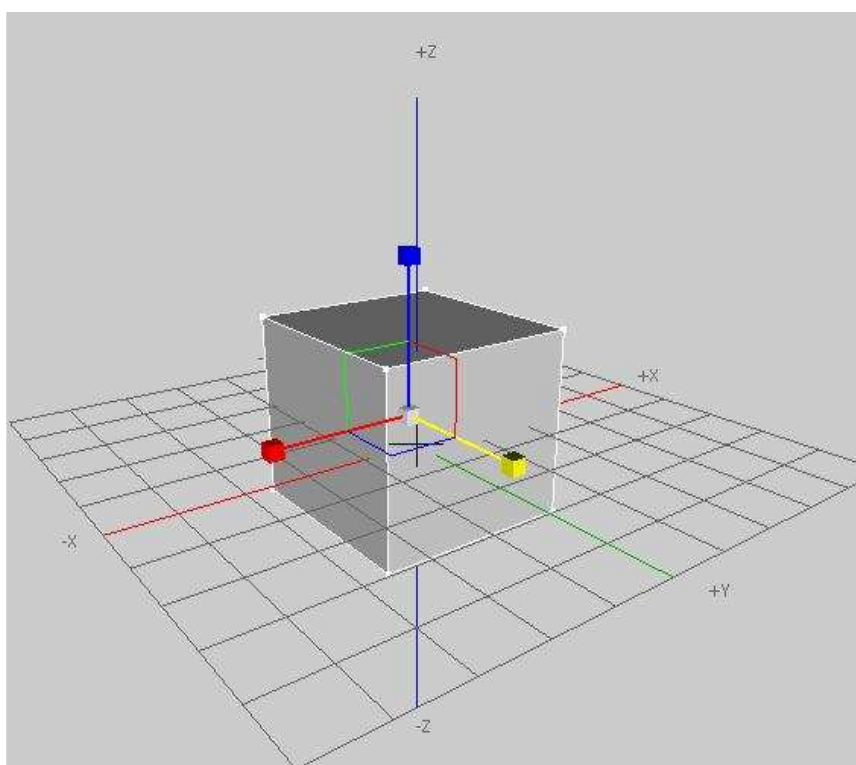



Рисунок 4. Режим *Select Objects*, функция *Scale*

Как мы видим из *Рисунка 4*, изменение размеров определено тремя стрелками с кубиками на конце, которые по аналогии с цветными стрелками, рассмотренными при перемещении объекта, растягивают объект по различным осям. Роль кубика в центре объекта при этом сильно отличается от роли шарика при перемещении. В данном случае при помощи кубика можно увеличить или уменьшить весь объект в целом.

Автор данной работы рассмотрел наиболее простые из возможностей работы с цельным объектом, но в К-3D те же самые функции применимы и к отдельным точкам на объекте, ребрам и граням. Такие возможности делают трехмерное моделирование в К-3D очень гибким, удобным и довольно простым. Рассмотрим теперь и их тоже. Начнем с работы с точками на объекте, для этого перейдем в другой режим работы - *Select Points*, который обозначен на панели рисунком -  .

Перемещение точек — первое, что я рассмотрю. Для этого выберем уже ранее рассмотренную функцию передвижения - *Move*.

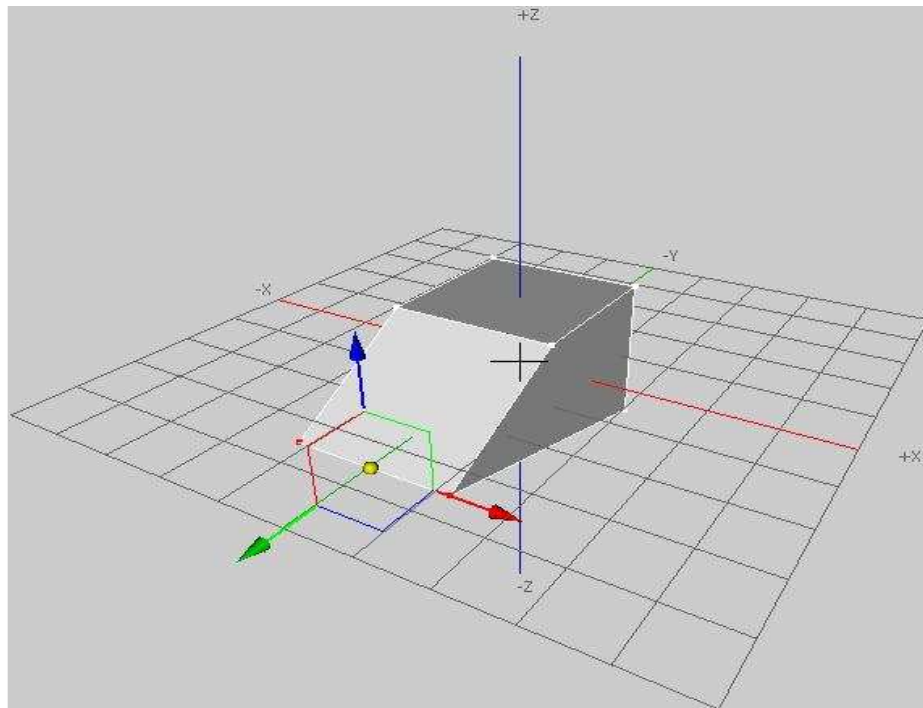


Рисунок 5. Режим *Select Points*, функция *Move*

Как видно из *Рисунка 5* автор выбрал для примера на кубе 2 точки и переместил их произвольно. Количество точек, которые можно выбрать — не ограничено, в выборе направления также — полная свобода. Одну точку можно выбрать, кликнув на нее мышью, а для выбора большего количества точек необходимо удерживать клавишу *Shift*.

Теперь развернем несколько точек на кубе, используя функцию *Rotate*.

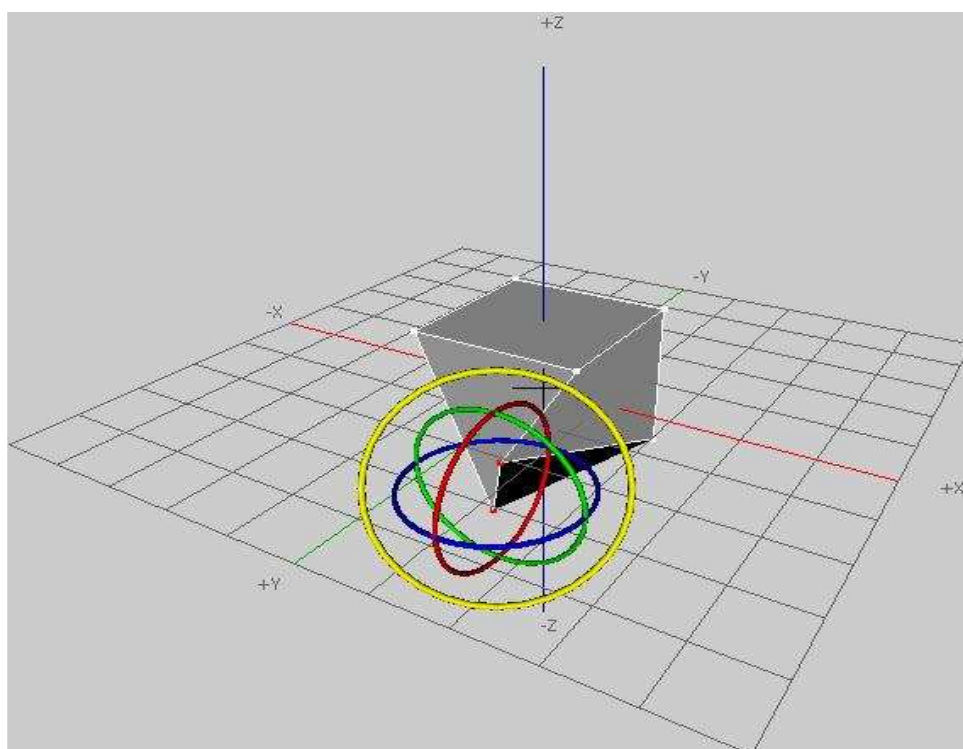


Рисунок 6. Режим *Select Points*, функция *Rotate*

В данном случае, который описан на *Рисунке 6*, автор взял 2 точки с нижнего ребра куба и повернул их. Получилась довольно интересная новая фигура. Экспериментируя подобным образом можно создать и более сложные объекты, которые пригодятся для будущих серьезных работ. В данном случае 1 точку повернуть нельзя, необходимо выбрать минимум 2 точки.

При изменении расстояния между точками, используя функцию *Scale* также необходимо выбрать как минимум 2 точки. Результат некоторых изменений расстояния между точками в кубе представлен на *Рисунке 7*.

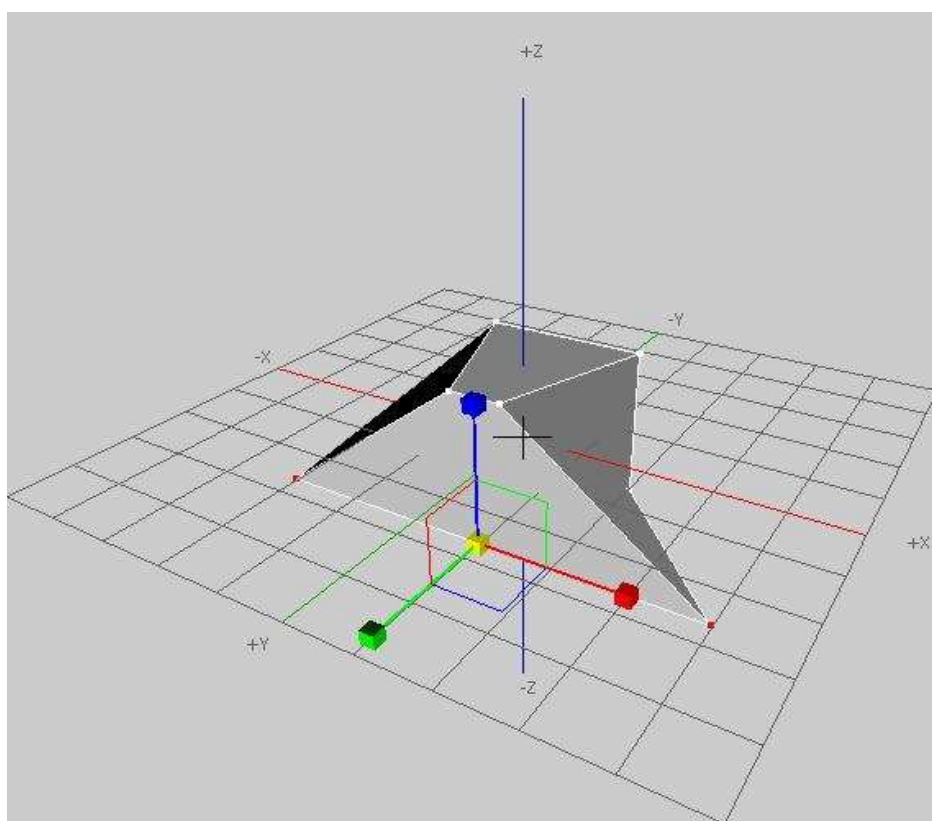



Рисунок 7. Режим *Select Points*, функция *Scale*

Перейдем к рассмотрению нового режима — *Select Lines*, предназначенного для работы с линиями, или ребрами многогранников. Этому режиму соответствует рисунок - 

Сначала, как и в предыдущих случаях, рассмотрим возможность передвижения линий при помощи функции *Move*.

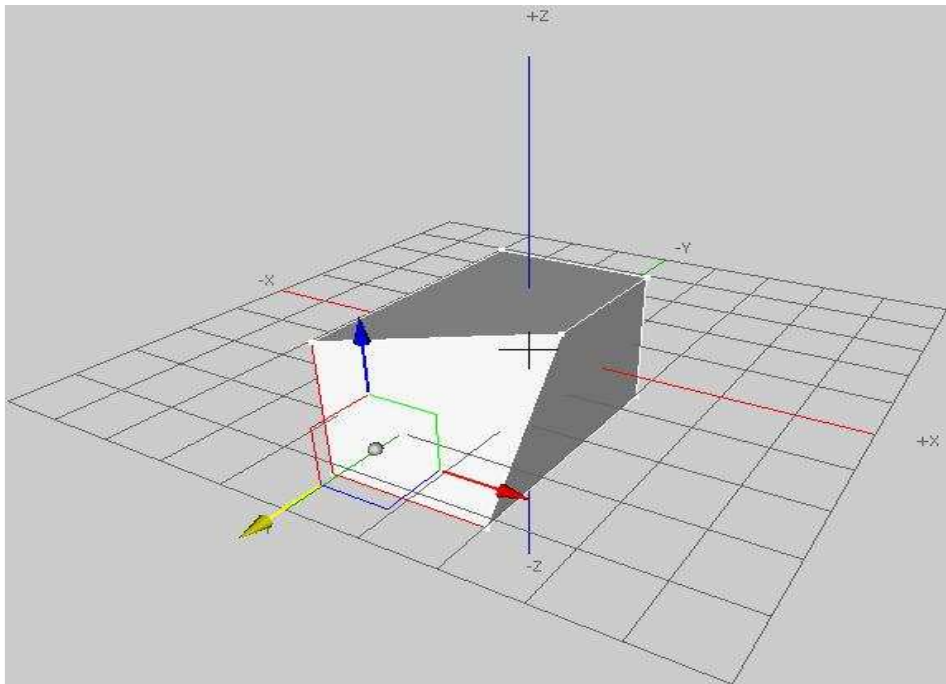


Рисунок 8. Режим *Select Lines*, функция *Move*

На *Рисунке 8* видны две выделенные автором линии, которые он переместил по оси *Y*. Количество одновременно выделяемых линий для перемещения — не ограничено. То же самое можно было бы сделать и в режиме выбора точек, но в сложной фигуре, где точек очень много, проще использовать именно режим работы с линиями.

Переходим к функции вращения — *Rotate*. Линия представляет из себя соединение двух точек, а это значит, что для вращения достаточно выбрать одну линию. Рассмотрим это на примере куба.

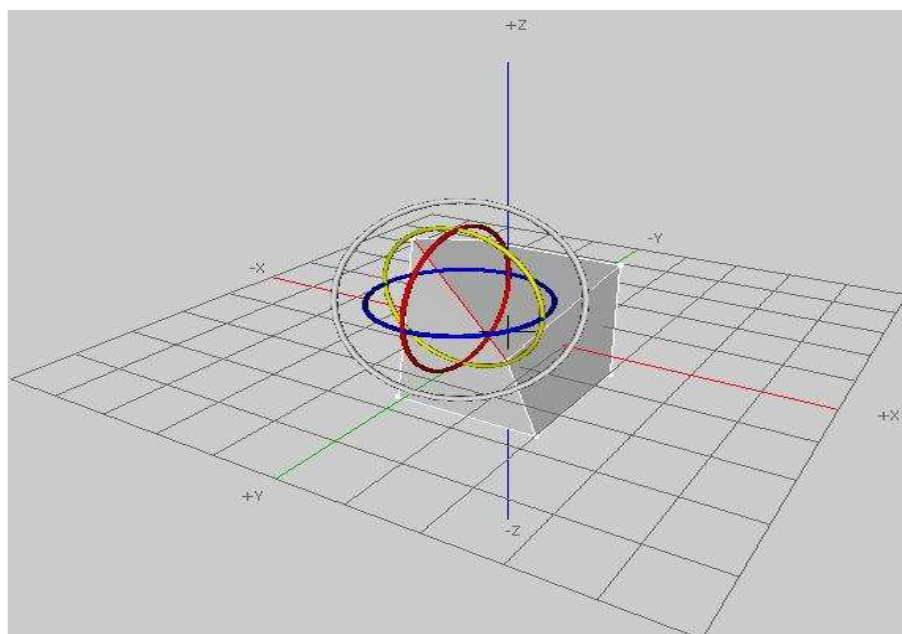


Рисунок 9. Режим *Select Lines*, функция *Rotate*

На *Рисунке 9* красным выделена та линия, которую автор развернул для примера. Количество одновременно вращаемых линий также не ограничено, и также как и в предыдущем примере, данное действие можно было бы воспроизвести и в режиме работы с точками, но это не во всех случаях является удобным. Режим работы с точками — для более детальной обработки модели.

Следующая функция — *Scale*, которая как мы уже знаем, предназначена для изменения размера или расстояния. При работе с линиями данная функция сужает или расширяет одну или несколько выбранных линий, изменяя объект в целом пропорционально изменению линий. Как это выглядит на кубе, можно увидеть ниже, на *Рисунке 10*, где размер двух выбранных линий был уменьшен.

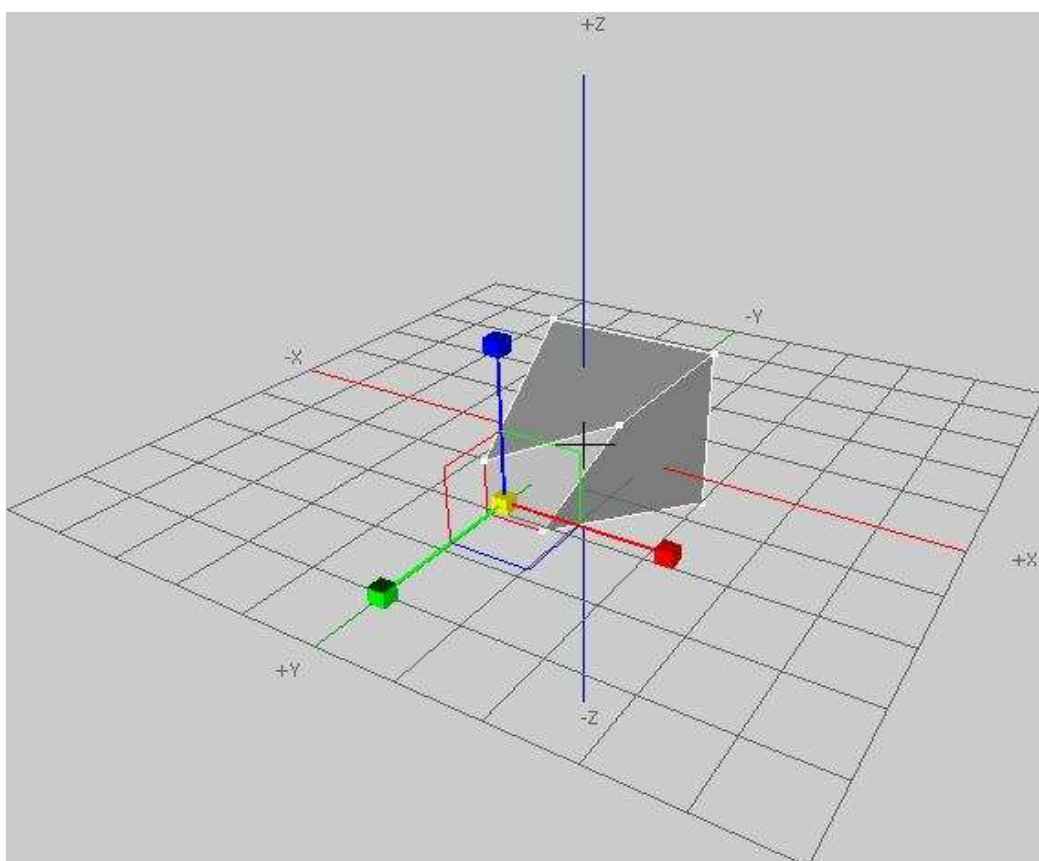



Рисунок 10. Режим *Select Lines*, функция *Scale*

Последний режим работы с объектом, который автор бы хотел рассмотреть в своей работе, называется *Select Faces*. Он предназначен для работы с определенными участками на объекте или гранями на геометрической фигуре. Одновременно можно выделять сколько угодно участков на объекте и работать с ними всеми вместе. Этому режиму соответствует следующий рисунок на панели инструментов -  .

Для начала рассмотрим передвижение двух граней на кубе при помощи функции *Move*. Не обязательно использовать цельные грани объекта, можно разбить объект сеткой и выделять и работать с участками, к примеру, в центре одной из граней куба. Пример можно увидеть на *Рисунке 11*, который находится ниже. Здесь мы видим, что у куба выделено 2 грани, окрашенных красным цветом, которые перемещены по оси X.

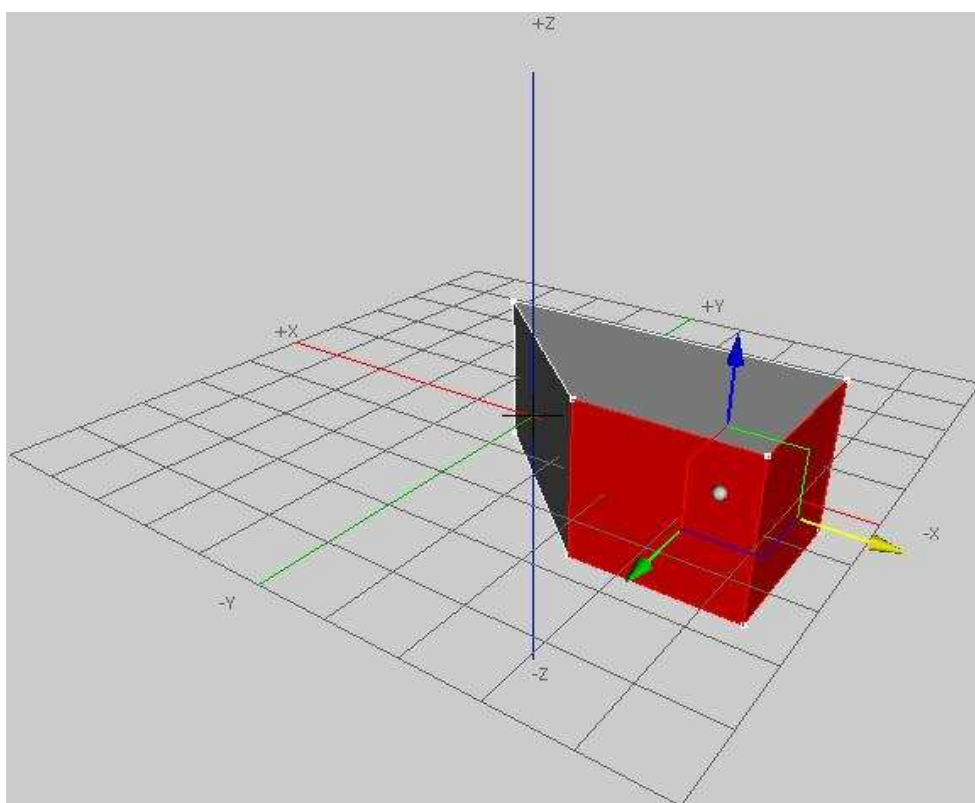


Рисунок 11. Режим Select Faces, функция Move

Разворот одной из граней куба, при помощи функции *Rotate*, мы рассмотрим при помощи *Рисунка 12*. Стоит учесть, что можно выделить сколько угодно граней и то, что грани объекта можно разбить на более мелкие участки, при помощи сетки, и далее работать уже с ними.

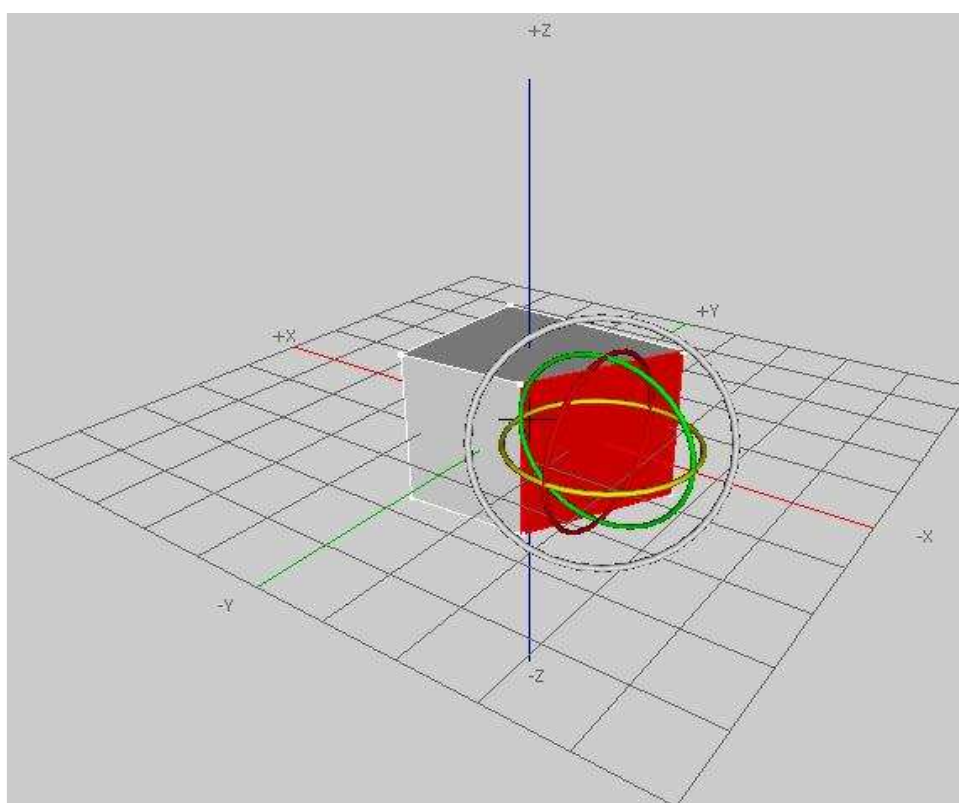


Рисунок 12. Режим Select Faces, функция Rotate

Изменение размера одной из граней куба, при помощи функции *Scale* автор данной работы покажет на следующем рисунке. Как автор уже отмечал ранее, если разбить куб сеткой на более мелкие участки, то и с ними можно будет работать, пример работы с сеткой я приведу позже. На *Рисунке 13* видно, что автор уменьшил одну из граней куба.

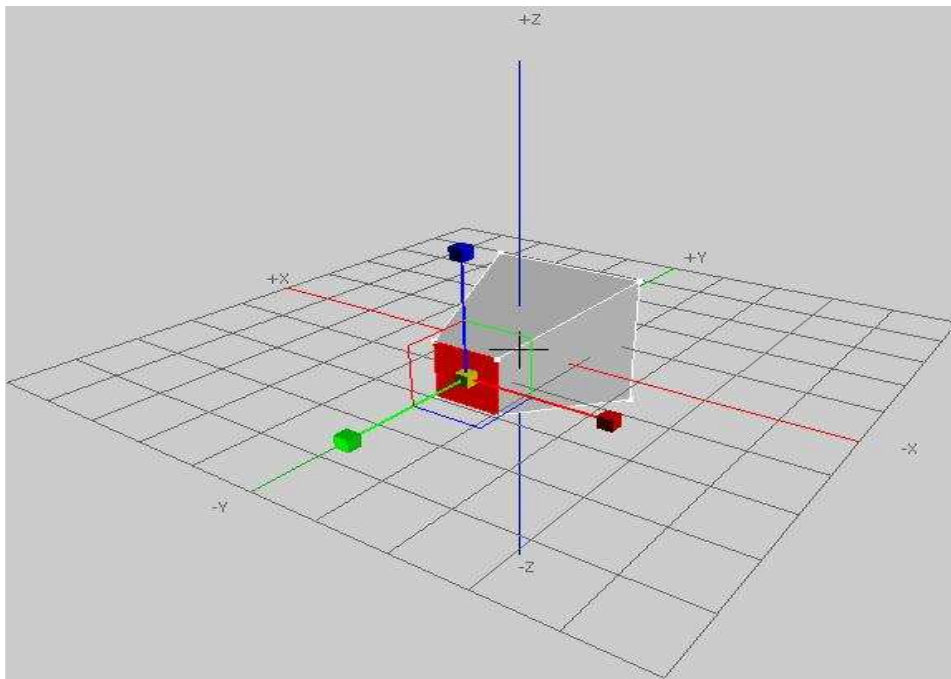


Рисунок 13. Режим Select Faces, функция Scale

5. Работа с сеткой.

В любой программе для трехмерного моделирования объект покрыт вспомогательной сеткой, для удобства работы с ним. В K-3D сетка (*Mesh*) также играет большую роль при создании собственных моделей. На *Рисунке 14* автор покажет основы работы с сеткой. Для примера он, как и в предыдущих случаях, взял обычный куб. Цифрой 1. на рисунке автор отметил три поля, в которых задается плотность сетки. При помощи правого клика мышью на объекте открывается меню, одним из разделов которого является модификатор сетки - *Mesh Modifier*, который отмечен цифрой 2. на рисунке. Модификатор сетки предоставляет огромный перечень функций, с помощью которых можно изменять сетку объекта так, как вам необходимо. Для примера автор выбрал один из модификаторов, который сглаживает углы - *MakeSDS*. На рисунке он отмечен цифрой 3. Результат изменения куба данным модификатором также можно увидеть на *Рисунке 14*.

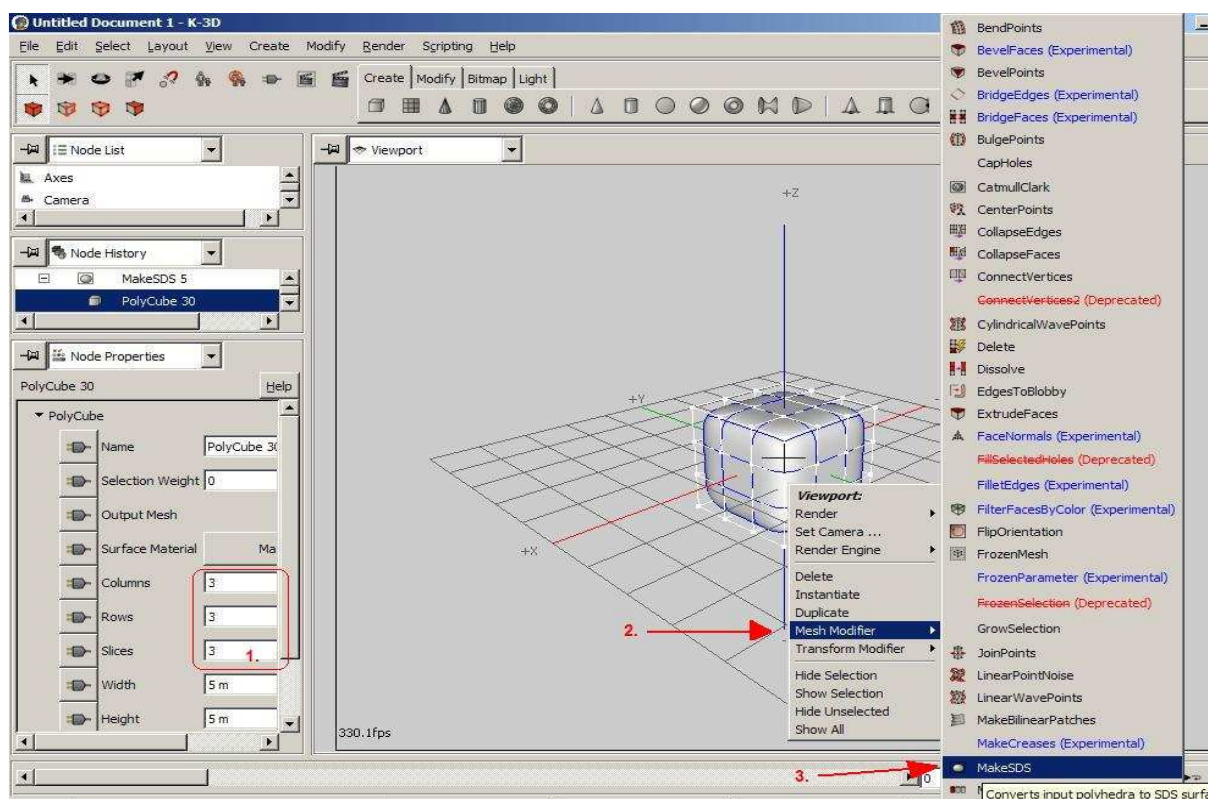


Рисунок 14. Работа с сеткой

На данный момент в K-3D имеется более 30 рабочих модификаторов сетки, среди которых есть несколько экспериментальных, находящихся в стадии тестирования и

отмеченных синим цветом. Также есть не рекомендуемые для использования модификаторы, которые отмечены красным цветом. Стоит отметить такие наиболее часто используемые модификаторы как *BendPoints* — изгиб объекта по точкам, *TaperPoints* — деформация объекта по точкам конически, *TwistPoints* — скручивание объекта по точкам, *ExtrudeFaces* — выталкивание выделенных частей объекта или граней фигуры. Результаты работы этих модификаторов автор покажет на следующем рисунке.

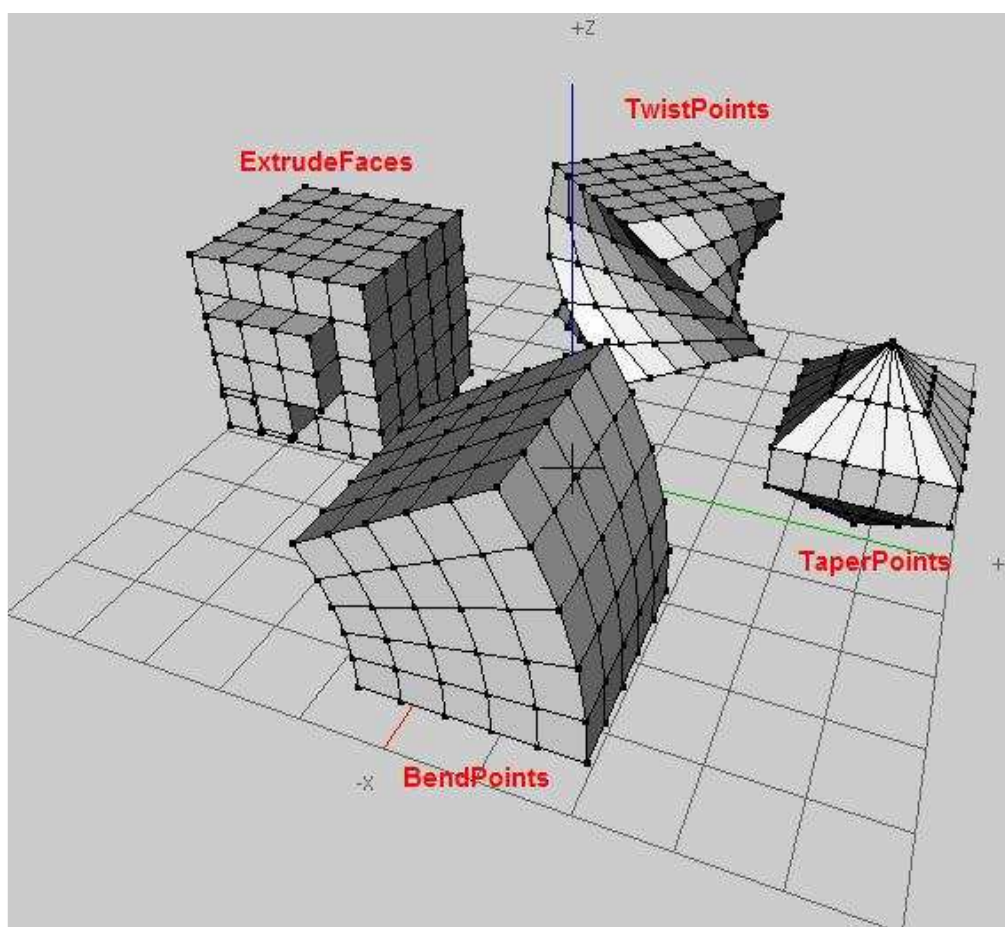


Рисунок 15. Пример использования модификаторов сетки

Все эти модификаторы были применены автором на кубах, где было необходимо — заданы произвольные значения углов и множителей, в случае *ExtrudeFaces* также была случайным образом выбрана область на грани куба, так чтобы работа модификатора была видна наглядно.

6. Работа с визуализируемыми цветами.

K-3D предусматривает возможность нанесения цвета на любой из объектов, так, чтобы после выполнения рендеринга, мы видели цветную картинку. Для изменения цвета необходимо выбрать *Material* с панели *Node List*, которая находится в верхнем левом углу, под режимами работы с объектом, которые мы рассмотрели ранее. Теперь достаточно изменить свойство цвета — *Color*, так как вам необходимо. После рендеринга мы видим, что наш объект изменил цвет на выбранный. При добавлении следующего объекта он по умолчанию получает тот же самый цвет, который имеет первый объект. Для того чтобы изменить цвет второго объекта необходимо создать свой дополнительный материал. Нужно зайти в панель *Node Properties*, в нижнем левом углу, предварительно выбрав на панели *Node List* тот объект, цвет которого вы хотите изменить. Теперь, в *Node Properties*, необходимо найти *Surface Material* и, кликнув на него мышью, выбрать опцию *New RenderMan Material*, которая позволяет вам создать свой собственный материал для нанесения цвета на объект. [Surface Color Tutorial, http://www.k-3d.org/wiki/Surface_Color_Tutorial] Стоит учесть, что при создании собственного материала можно использовать функцию непрозрачности - *Opacity*, которая позволяет выбрать второй цвет который будет не прозрачным, а первый цвет, выбранный при помощи функции *Color*, станет полупрозрачным.

Экспериментируя с этими функциями можно создать довольно таки красивые комбинации цветов. К примеру, можно имитировать полупрозрачность воды или стекла.

Для того чтобы создание собственных материалов для нанесения цветов на объекты было более понятным, я добавил 3 рисунка, которые приведены ниже. На *Рисунке 16* изображено создание нового материала, а на *Рисунке 17* показаны возможные настройки этого материала, включая выбор цвета и непрозрачности. Настройка самого цвета также является достаточно детальной, это можно увидеть на *Рисунке 18*.

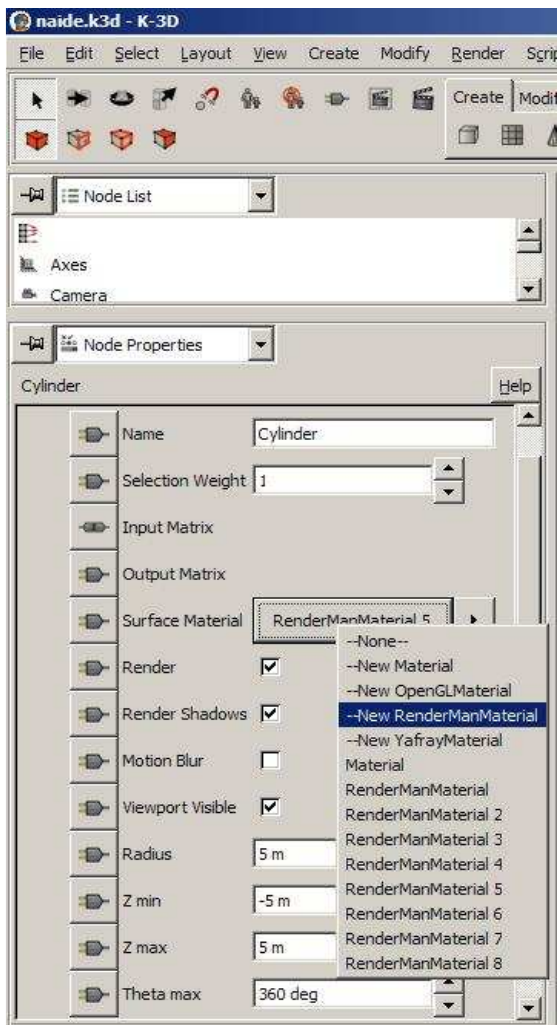


Рисунок 16. Создание нового материала

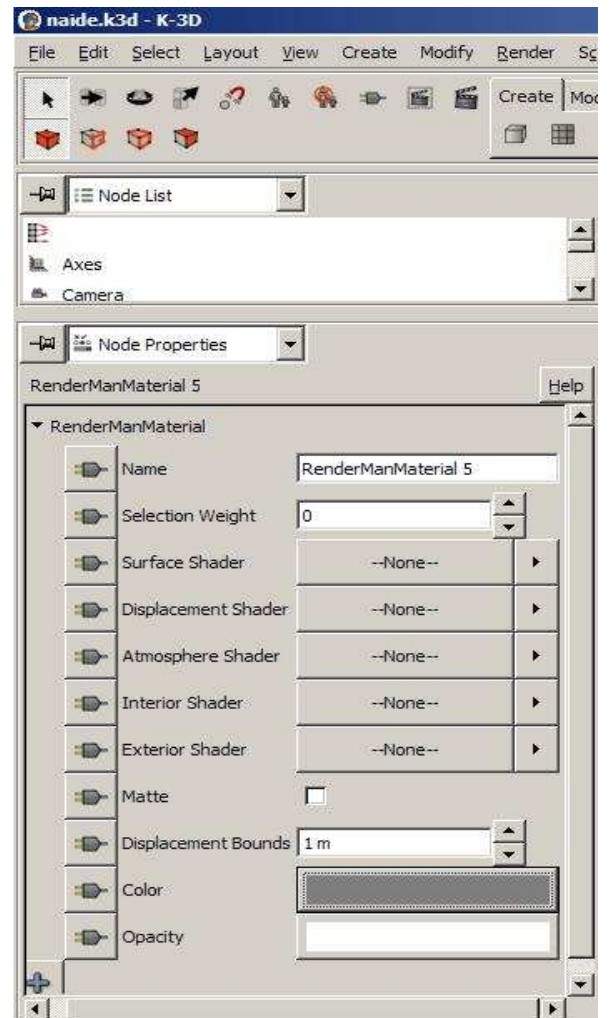


Рисунок 17. Настройка материала

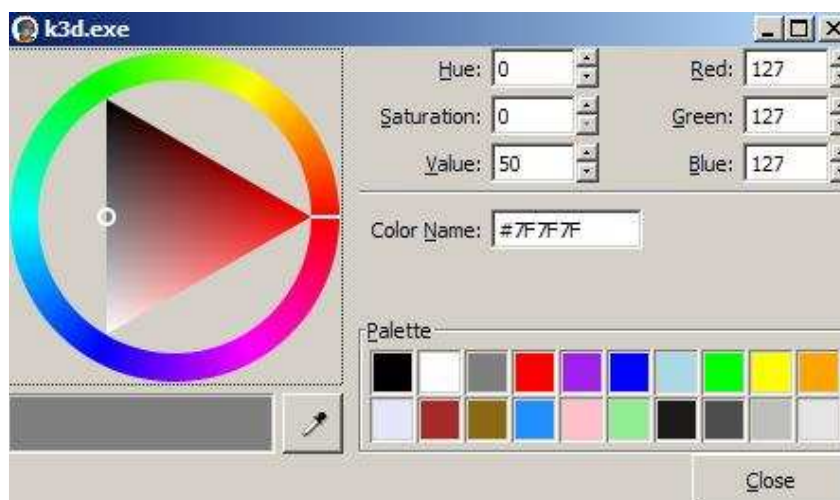


Рисунок 18. Настройка цвета

8. Пример трехмерного моделирования в K-3D.

Для примера автор решил создать пейзаж, на котором изображено средневековое здание, соединенное с башней, луна, колодец и ведро рядом с ним. В схематичном виде это выглядит так, как показано на *Рисунке 18*, расположенном ниже.

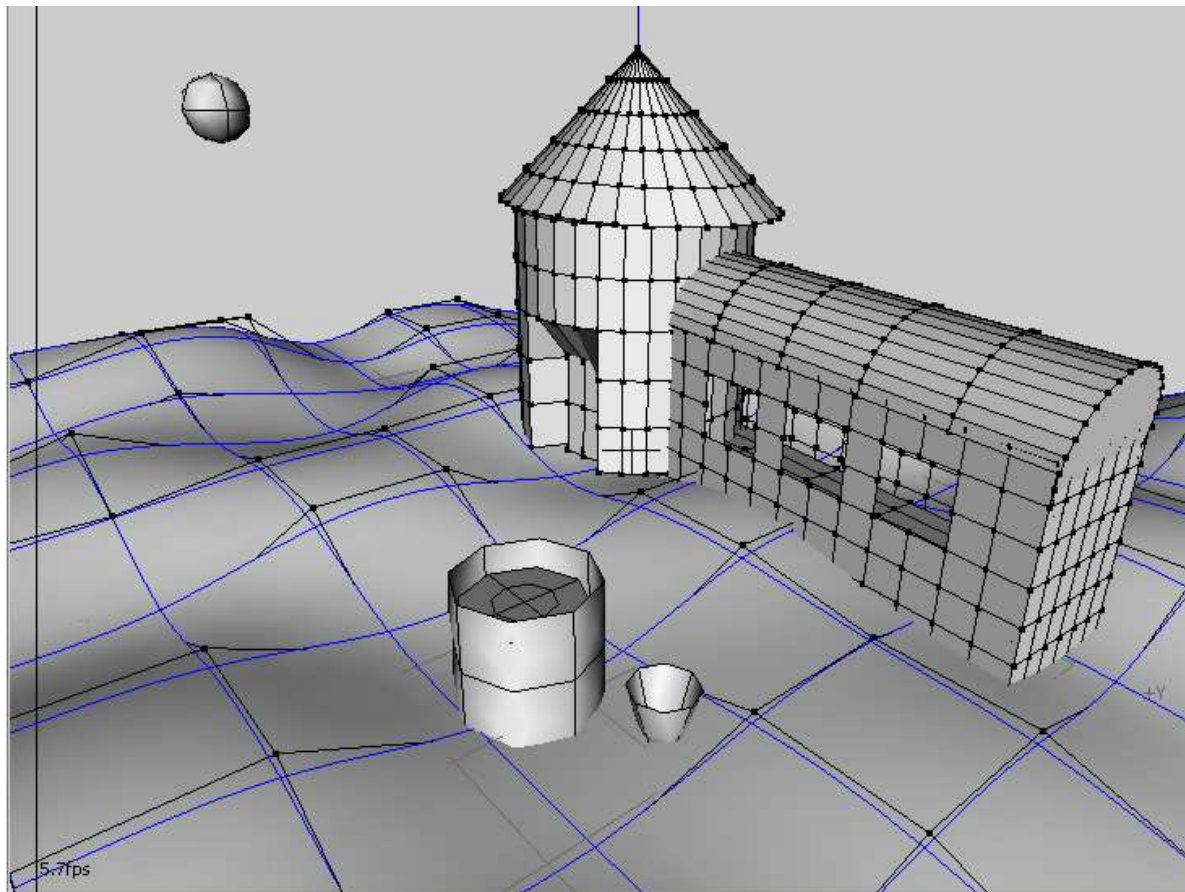


Рисунок 19. Пример работы в K-3D в схематичном виде.

Сначала автор данной работы создал поверхность при помощи стандартного объекта — *Polygonal Grid*. Используя режим *Select Faces* и функцию *Move*, он приподнял некоторые участки на почве, чтобы она получилась неровной, затем он применил модификатор сетки - *MakeSDS*, чтобы почва получилась гладкой, без углов. Затем автор добавил один из стандартных видов цилиндра — *Polygonal Cylinder*. Изменил его положение и размер соответственно при помощи режима выбора объектов — *Select Objects* и функций *Move* и *Scale*. Выбрав определенный участок на цилиндре в режиме *Select Faces*, автор вдавил его внутрь, используя функцию *Move*, чтобы образовалась арка под дверь. Саму дверь автор создал еще из одного *Polygonal Grid'a* и, используя

Move и *Scale* в режиме *Select Objects*, подстроил ее под арку в цилиндре. Крышу башни автор создал также из стандартного объекта — *Polygonal Cone*, соответственно изменив ее размер, используя функцию *Scale*, и положение, используя функцию *Move*. Затем из стандартного объекта куба — *Polygonal Cube*, автор создал пристройку к башне, вырезав в ней окна при помощи одного из модификаторов сетки, который называется *Delete*. В режиме *Select Faces* автор выбрал на сетке куба участки под окна и применил к ним модификатор *Delete*. Через окна стало видно неровную почву, которую он сделал с самого начала. Для того чтобы накрыть ее, автор выбрал стандартный объект *Polygonal Grid*, изменил его размер и разместил чуть выше уровня почвы, так, чтобы это было похоже на пол в помещении. И пристройку, и пол он также разместил в нужных местах при помощи функции *Move* и изменил их размеры, чтобы они соответствовали друг другу при помощи функции *Scale*. Для создания крыши пристройки автор использовал один из стандартных фигур-цилиндров — *Polygonal Cylinder*, и обрезал его при помощи модификатора сетки *Delete*, в режиме *Select Faces*, для того чтобы в окнах здания не было видно половины цилиндра. Для создания колодца автор использовал другой стандартный вид цилиндра — *Cylinder primitive*, который представляет из себя полый цилиндр. При помощи функций *Move* и *Scale* автор поставил его на нужное мне место и изменил его размер. Для того чтобы изобразить воду в колодце он использовал стандартный объект — *Disk Primitive*, который подогнал под размер колодца и расположил немного ниже краев колодца, при помощи функций *Move* и *Scale*. Для создания ведра автор использовал стандартный объект параболоид - *Paraboloid Primitive*. Сам по себе параболоид уже напоминает по форме ведро, поэтому он всего лишь развернул его при помощи функции *Rotate*, изменил размер и положение при помощи уже известных нам функций *Move* и *Scale*. Последнее что автор создал в этой работе — луна. Для ее создания он взял объект *Sphere Primitive* и расположил его соответствующим образом при помощи функции *Move*, размер сферы изменил при помощи функции *Scale*. После создания материалов для нанесения цвета на объекты (об этом автор более подробно рассказал в предыдущем разделе Работа с визуализируемыми цветами, под номером 6), автор придал каждой детали свой цвет. Результат рендеринга можно увидеть на *Рисунке 20*.



Рисунок 20. Результат рендеринга в K-3D

При создании цветов автор применил элементы непрозрачности — *Opacity* для воды в колодце. В результате стенку колодца видно через воду. Более детально это можно рассмотреть на *Рисунке 21*.

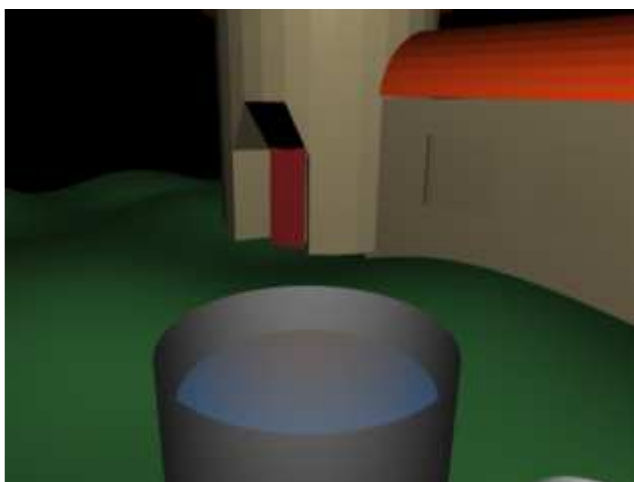


Рисунок 22. Использование Opacity

Результат использования модификатора сетки *Delete* также можно увидеть поближе. На *Рисунке 22*, видно, что окна в здании сквозные.

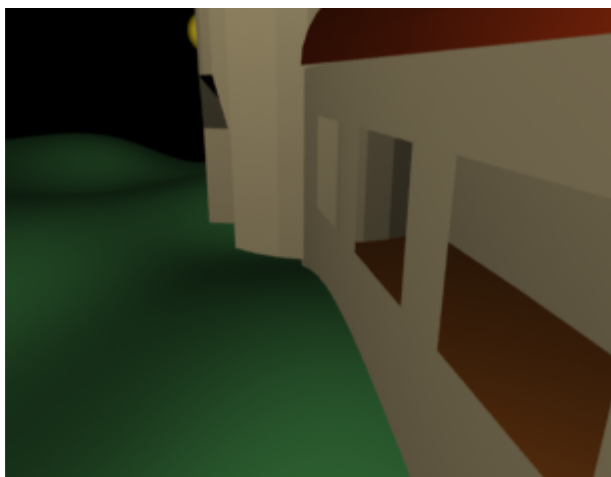


Рисунок 21. Результат использования модификатора сетки - Delete

Заключение.

В данной работе изложены далеко не все возможности K-3D в области трехмерного моделирования, но, используя знания, полученные после прочтения этого материала, можно создавать сложные модели. В своей работе автор отметил наиболее важные основы создания трехмерных моделей, а также показал на своем примере их практическое применение.

В наше время имеются и другие программы 3-D моделирования, которые превосходят набором возможностей программу K-3D, но даже на фоне других коммерческих и некоммерческих проектов потенциал K-3D достаточно большой. Благодаря своим уникальным особенностям, K-3D является очень удобной программой и предоставляет полную свободу во всем - как в выборе инструментов, каждый из которых является плагином и может быть заменен при желании (плагин-ориентированная архитектура), так и в возможности распространения программы и изменения ее программного кода(*Open Source*).

Неоспоримым плюсом данной программы является также ее адаптированность ко всем наиболее известным платформам, таким как GNU/Linux, Win32, MacOSX, BSD, Solaris. Также стоит упомянуть, что система K-3D имеет достаточно низкие системные требования и может быть запущена практически на любом компьютере.

Особую гибкость K-3D придает поддержка форматов других программ трехмерного моделирования. K-3D может открывать и изменять работы, созданные в других программах, и также сохранять измененные работы в форматах, соответствующих другим программам. Данная возможность реализуется также при помощи плагинов, которые требуют отдельной установки, и количество которых растет с каждым днём. К примеру, в стандартный набор системы K-3D входят плагины для поддержки таких форматов как RIB, GTS, MFX, OBJ, OFF, RAW, K3D.

Гордостью разработчиков является также внешний вид программы. Очень удобный, интуитивно-понятный интерфейс подходит как новичкам, так и профессионалам. По

словам разработчиков[FAQ, вопрос „*How does K-3D compare to Blender?*“, <http://www.k-3d.org/wiki/FAQ>], они часто получали похвалы за хороший дизайн и за философию "Всё - плагины" ("*everything is a plugin*").

Ülevaade.

Selles töös ei olnud jutustanud kõikidest K-3D modelleerimise võimalustest, aga kasutades selle töö teadmised saab teha kergemad suured mudelid. Oma töös autor märgis põhilised teadmised 3-D modelleerimisest K-3D baasil ja oma näidise abil näitas, kuidas neid saab kasutada.

Meie ajal on olemas palju erinevaid 3D modelleerimise programmid. K-3D aga saab konkureerida nii kommerts kui ka vabavaraliste programmidega oma unikaalse omaduste abil. *Plug-in-oriented* arhitektuur ja selle programmi litsensioon(*Open Source*) annavad kasutajatele vabadust.

Tuleb ka märkida, et K-3D saab kasutada paljudes süsteemides, näiteks GNU/Linux, Win32, MacOSX, BSD, Solaris. Nõudmised arvutile on väga madalad.

K-3Dga saab avada ja muuta tööd, mis on tehtud teiste programmide abil. See võimalus on realiseeritud plug-in'ite abil. Standardised plug-in'id annavad võimalust töötada niisuguste formaatidega, nagu RIB, GTS, MFX, OBJ, OFF, RAW, K3D.

Список использованных в работе источников.

1. <http://www.k-3d.org/wiki/Features> - общий обзор особенностей К-3D
2. <http://www.k-3d.org/wiki/FAQ> - часто задаваемые вопросы пользователей
3. http://www.k-3d.org/wiki/Video_Tutorials - обучающие видео по К-3D
4. <http://www.k-3d.org/wiki/Category:Tutorials> - текстовые туториалы по К-3D
5. http://www.k-3d.org/wiki/Surface_Color_Tutorial - документация по использованию цветов в К-3D
6. <http://www.k-3d.org/wiki/Scripting> - документация о скриптах
7. http://www.k-3d.org/wiki/User_Documentation - различные пользовательские документации
8. Брюс Фрейзер, Реальный мир Camera RAW и Adobe Photoshop CS2
9. http://shape.cs.princeton.edu/benchmark/documentation/off_format.html - статья о формате OFF
10. <http://lesscan.info/5.php> - статья об использовании цифровых и файловых форматов
11. <http://msdn.microsoft.com/en-us/library/bb173014.aspx> - все о формате файлов X
12. <http://local.wasp.uwa.edu.au/~pbourke/dataformats/obj/> - статья о формате OBJ