

ФГБОУ ВО «АДЫГЕЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

КАФЕДРА ИНОСТРАННЫХ ЯЗЫКОВ

ДНЕВНИК

переводческой практики

студентки III курса

Нурыевой Гозел

квалификации «Переводчик в сфере профессиональной коммуникации»

Начало практики 23 апреля 2019 г.

Конец практики 21 мая 2019 г.

Дневник представлен руководителю практики 21 мая 2019

Дневник проверен _____ А.А. Хатхе

Майкоп 2019

СОДЕРЖАНИЕ

Введение	3
Отчет о работах	6
Аннотация	8
Предпереводческий анализ текста	9
Постпереводческий анализ текста	10
SoftF/X Pro Help	12
The Translation _ SoftF/X Pro Помощь	23
Глоссарий	36
Список литературы	40

ВВЕДЕНИЕ

Основной целью переводческой практики является формирование практических навыков по различным видам перевода:

- совершенствование навыков письменного перевода;
- совершенствование навыков устного перевода в его разновидностях (последовательный, синхронный, реферативный).

Студент, получающий дополнительную квалификацию «Переводчик в сфере профессиональной коммуникации», должен знать и уметь решать задачи, соответствующие данной квалификации, а именно:

- осуществлять и анализировать переводческую деятельность в рамках межкультурной коммуникации;
- проявить достаточный уровень знания иностранного и родного языков;
- уметь выявить социолингвистические особенности ситуаций межкультурной коммуникации;
- осуществлять специальные переводческие операции в речи;
- знать основные правила закономерностей структурирования текста-продукта устной или письменной коммуникации;
- уметь проанализировать условия ситуации коммуникации, определить оптимальные средства её реализации и оценить успешность своей переводческой деятельности;
- проявить творческий подход к работе с конкретным переводческим материалом, изучать и анализировать закономерности и особенности переводческой деятельности с позиции современного переводоведения;
- уметь организовать межкультурное общение в конкретных условиях его реализации.

Кроме того, студент, обучающийся по дополнительной квалификации «Переводчик в сфере профессиональной коммуникации», должен иметь представление о прогрессивных технологиях и способах перевода, особенностях видов перевода, достижениях в области перевода, о принципах, нормах и специфике поведения переводчика в различных ситуациях межкультурной коммуникации, знакомиться с типовыми образцами перевода различных документов. Студент должен знать основные технические нормативы в реализации различных видов перевода, а также основы организации труда переводчика.

Переводческая практика студентов, специализирующихся в области профессионального перевода, преследует ряд целей и задач, основными из которых являются:

- закрепление и углубление теоретических знаний, полученных в ходе изучения языковых дисциплин и дисциплин по переводу и переводоведению;
- приобретение навыков практической работы по специальности, опыта самостоятельной работы на определенной штатной должности или в качестве неоплачиваемых внештатных помощников;

- совершенствование полученных на 1-3 курсах обучения навыков и умений по основам межкультурной коммуникации, практическому владению иностранным языком; дальнейшее развитие и совершенствование умений перекодирования смысловой информации с одного языка на другой, а именно: умение использовать перифразы для передачи прочитанного или услышанного, поиск нужного эквивалента с использованием известных языковых средств; совершенствование умений использования синонимического ряда при дефиците лексических средств и оформления содержания диалога в связный монологический текст;
- совершенствование умения непосредственного общения с иностранцами в аутентичных ситуациях социально-бытовой, социально-культурной и профессиональной сфер общения;
- развитие умений решения проблем, связанных с эффективной переводческой деятельностью;
- подбор и анализ экспериментального и теоретического материала для выполнения учебно-исследовательских и научно-исследовательских работ.

В соответствии с учебным планом длительность практики составила 5 недель. Переводческая практика проводилась в виде перевода письменного текста объёмом 10 страниц, текст представляет собой подборку из нескольких статей, относящихся к информационным технологиям, и взят из википедии.

о работах, выполненных в период переводческой практики

Виды и последовательность выполнения переводческих работ по дням представлены в таблице:

№	Наименование работ	Кол-во рабочих дней	Отметки руководителя
1.	Знакомство с основными целями и задачами практики. Определение смысловой и стилистической направленности письменного текста. Ознакомление с путями преодоления переводческих трудностей для дальнейшего их применения. Определение видов и форм предстоящего перевода. Изучение особенностей письменного перевода и условий его осуществления, а также требований, предъявляемых к его результатам. Изучение формы представления письменного перевода.	20.04. – 24.04.	Выполнено <hr/>
2.	Предпереводческий анализ текста. Общее знакомство с текстом и его источниками. Определение тематики текста. Поиск информации об авторах исследования, его целях и задачах, а также целевой аудитории.	24.04. – 28.04.	Выполнено <hr/>
3.	Работа над переводом текста « Дизайн интерьера ». Первичное осмысление текста. Определение типа авторской аргументации. Анализ организации смысла. Идентификация стилистических аспектов исходного текста, определение использованных образных средств. Идентификация культурно-маркированных элементов, реалий, безэквивалентных лексических единиц и лингвистических лакун.	28.04. – 7.05.	Выполнено <hr/>
4.	Работа над рабочим вариантом перевода.	7.05. –	Выполнено

	Поиск незнакомых слов в словарях общей лексики. Работа с различными словарями для уточнения переводов специальных терминов по теме «Дизайн интерьера». Работа над переводом безэквивалентных единиц и лингвистических лакун.	15.05.	_____
5.	Работа над окончательным вариантом перевода, который в полной мере соответствует лингвистической норме языка перевода. Окончательная правка текста перевода. Проверка стилистического соответствия нормам русского языка. Проверка орфографии и пунктуации. Написание аннотации. Составления списка ключевых слов. Составления глоссария. Оформления списка источника информации. Окончательное оформление текста перевода и подготовка его к печати.	15.05. – 18.05.	Выполнено _____
6.	Постпереводческий анализ текста. Определение переводческих трудностей и переводческих приёмов, примененных для преодоления этих трудностей.	18.05. – 21.05.	Выполнено _____

По итогам переводческой практики были применены все навыки и умения, полученные за три года обучения по дополнительной квалификации «Переводчик в сфере профессиональной коммуникации». Были закреплены теоретические знания, полученные в ходе изучения языковых дисциплин и дисциплин по переводу и переводоведению. Также были усовершенствованы навыки и умения по основам межкультурной коммуникации, практическому владению иностранным языком.

В ходе практики были приобретены навыки практической работы по специальности, были развиты умения решения проблем, связанных с эффективной переводческой деятельностью.

Аннотация

Abstract: *The article discusses the relevance of the use of three-dimensional graphics, 3D-modeling for various purposes. The author proposes a sequence of actions for beginners 3D-artists, which will help to avoid disappointment in this type of activity. The possible difficulties faced by beginners in the field of three-dimensional graphics were analyzed. A large number of areas in the world of 3D, the article considered the language of animation and modeling. Examples of new terms encountered in working with 3D products are given.*

Keywords: 3D, 3D-modeling, model, dimensional, computer, screen, animation, software, monitor, imaginary, modeler, cursor, modeling, triangular, object, surface, edge.

Аннотация: *В статье рассматривается актуальность применения трехмерной графики, 3D-моделирования в различных целях. Автором предлагается последовательность действий для начинающих 3D-художников, которые помогут избежать разочарований в данном виде деятельности. Были проанализированы возможные трудности, с которыми сталкиваются начинающие в сфере трехмерной графики. В мире 3D, большое количество направлений, в статье были рассмотрены язык анимации и моделирования. Приведены примеры новых терминов, с которыми сталкиваются в работе с 3D продуктами.*

Ключевые слова: 3D, 3D-modeling, model, dimensional, computer, screen, animation, software, monitor, imaginary, cursor, modeling, realistic, object, surface, edge, frame.

ПРЕДПЕРЕВОДЧЕСКИЙ АНАЛИЗ ТЕКСТА

Текст оригинала, принятого для перевода и настоящего анализа, представляет собой подборку из нескольких статей, относящихся к информационным технологиям, и взят на сайте <https://ru.wikipedia.org>. Текст не имеет конкретного автора. Тема данной статьи «SoftF/X Pro Help».

Текст логически разбит на главы и абзацы, информация упорядочена и изложена точно, двусмысленности в тексте нет. Каждая глава имеет название, в котором отражается смысл. Текст содержит когнитивную информацию и в нем присутствуют логические связи и причинно-следственные отношения. В данной работе лингвистические лакуны и образные средства отсутствуют, а также скрытого смысла нет.

В анализируемом тексте присутствуют как простые, так и сложные предложения, но преобладают сложные предложения. Автор часто использует термины, относящиеся к данной теме. В тексте имеются графические объекты для визуализации рассматриваемых объектов.

В работе присутствует точное описание фактов, описываются некоторые процедуры моделирования и приводятся примеры работы с 3D продуктами. Что касается графических параметров текста, то можно увидеть использование автором сокращений и выделений текста, большое количество скобок и кавычек, используемых для уточнения и раскрытия смысла.

Целевая аудитория: широкий круг читателей, особенно, специалисты сферы 3D-моделирования, а также студенты направления информационных технологий.

Иллокуция – информативная.

Стиль: публицистический.

Жанр: научно-популярная статья.

Ведущая композиционно-речевая форма: повествование.

Вид текста: письменный.

Композиция: стандартизированная.

Тональность: нейтральная.

ПОСТПЕРЕВОДЧЕСКИЙ АНАЛИЗ ТЕКСТА

SoftF/X Pro Help - SoftF/X Pro Помощь – калькирование в лингвистике – заимствование иноязычных слов, выражений, фраз буквальным переводом соответствующей языковой единицы.

You cannot pick it up and move it about on the side of **a computer** monitor! – Вы не можете поднять ее и переместить в сторону монитора! ‘a computer’ – опущение. При переводе опущению подвергаются чаще всего слова, являющиеся семантически избыточными, т.е. выражающие значения, которые могут быть извлечены из текста и без их помощи.

To all intents and purposes, the world we live in is three dimensional. – **В сущности**, мир, в котором мы живем, является трехмерным. Замена – это такой вид переводческих трансформаций, при котором происходит замена слов ИЯ на лексические единицы ПЯ, которые не являются их словарными соответствиями, но может быть выведено из них с помощью определенного типа логических преобразований.

And the brain believes what **the eyes tell** it even though we know the screen is flat! – И мозг верит в то, что **видят глаза**, даже если мы знаем, что экран плоский! Замена – наиболее распространенный и многообразный вид переводческих трансформаций.

For example a low resolution output for a display of 320 pixels across the screen by 200 pixels down the screen requires 64000 pixels **in total**. – Например, низкое выходное разрешение для отображения 320x200 пикселей на экране **в общей сложности** требует 64000 пикселей. Перестановка – это изменение расположения (порядка следования) языковых элементов в тексте перевода по сравнению с текстом подлинника. Одной из причин перестановки является различие в строе (порядке слов) предложения в английском и русском языках.

You just assume that its **pointer** is on the screen in front of you. – Вы просто предполагаете, что его **курсор** находится на экране перед вами. Конкретизация – это замена слова или словосочетания ИЯ с более широким значением словом или словосочетанием ПЯ с более узким значением.

All the user actions take place in a cubic volume called the **WindowBox**. – Все пользовательские действия происходят в кубическом объеме, названном **WindowBox**. Калькирование – буквальный перевод используется для передачи имени собственного.

Image Map - An Image Map is a picture in a GIF or TGA file that is painted on to faces in the model to add decals or other surface detail. – Карта изображения - карта изображений – это изображение в файле GIF или TGA, которое рисуется на гранях модели для добавления надписей или других деталей поверхности. Опущение артикля, так как в русском языке нет такого понятия, как артикль.

Morphing - Enables one model to gradually change shape into another **model**. – Превращение - позволяет одной модели постепенно менять форму на другую. ‘model’ – опущение. При переводе опущению подвергаются чаще всего слова, являющиеся семантически избыточными, т.е. выражающие значения, которые могут быть извлечены из текста и без их помощи.

Phong Shading - A rendering algorithm that shades models smoothly and accurately depicts surfaces, textures, and reflective properties (such as specular) and can simulate mirror reflections. – Закраска **по методу** Фонга - алгоритм рендеринга, который заштриховывает модели гладко и точно изображает поверхности, текстуры, и отражающие свойства (такие как specular) и может моделировать зеркальные отражения. Оправданное добавление – вид грамматических трансформаций, требующий внесения дополнительных слов. Обуславливается различиями в структуре предложения и тем, что более сжатые английские предложения в русском языке требуют более развернутого выражения мысли.

SoftF/X Pro Help

Introduction to 3D

To all intents and purposes, the world we live in is three dimensional. Therefore, if we want to construct a realistic computer model of it, the model should be three dimensional as well.

Unfortunately, the usual display device of a computer (the monitor) is 2 dimensional. However, it seems that 2 dimensions may be enough to convince the viewer that he or she is watching a realistic model of the real world. For example, the images seen on a television.

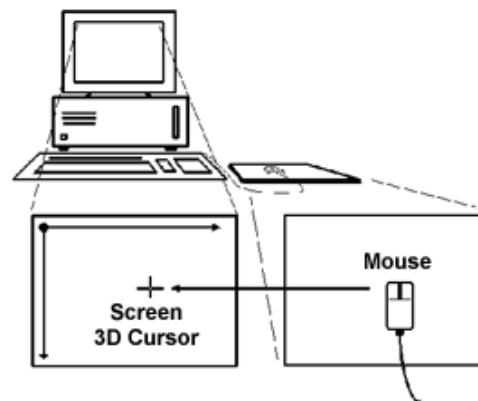
Nevertheless, we have been restricted to two-dimensional television and two-dimensional cinema for many years and seem very happy with it. This is because the images we get on the TV have many subliminal visual cues that tell us "This is 3 Dimensional". And the brain believes what the eyes tell it even though we know the screen is flat! Thus we can conclude that a 2-D screen can do a good job of presenting the 3D world, if we do it right!

We have a not insignificant hurdle in working in 3D on the computer. The screen is flat, it has width and height but we also need depth. Depth is the dimension into and out of the screen. All computer screens are made up from small rectangles of color, each rectangle is called a "pixel" (short for picture element), the more pixels on the screen the higher the resolution of the screen.

It is usual to specify the resolution of the computers display by quoting the number of pixels across the screen and the number of pixels down the screen rather than the very large number of pixels on the screen! For example a low resolution output for a display of 320 pixels across the screen by 200 pixels down the screen requires 64000 pixels in total.

Two dimensional art, animation or drafting (CAD) packages usually use a graphics cursor (an arrow or some other icon) so that you can make a specific action at an appropriate place on the screen. They may also use a co-ordinate reference (rather like a map-reference) to specifically refer to a point on the screen. Like a map reference, you need two numbers to specify a point on the screen, an

across number and a down number. Again like a map reference you need a fixed reference point. This is usually taken as the top left corner of the display and is referred to as (0,0). Moving across the screen points are referenced as (1,0) (2,0) etc. Moving down the screen points are referenced as (0,1), (0,2) etc. So any point can be defined with two numbers. For example the center of the screen might be (160,100).

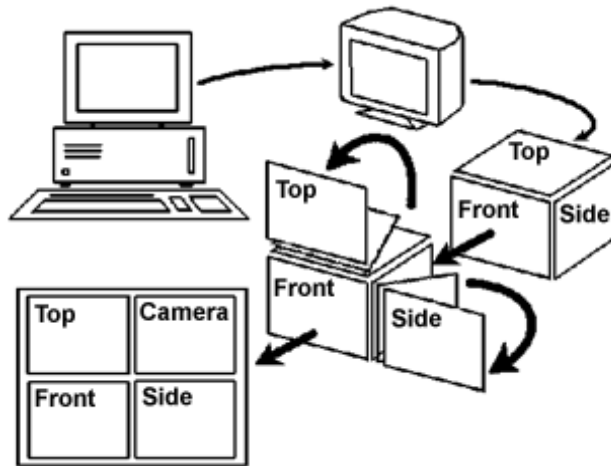


In 3D we need to add a third number to the across and down coordinates (a depth coordinate "in/out") giving us (0,0,0), (0,0,1) etc. We have a problem in visualizing this third coordinate. Because the screen has no depth, we cannot see a move that goes in/out. Ideally we would like a second screen on the side of the monitor to see the depth move.

We can't expect computer manufactures to make monitors with screens on the side as well as the front, even if we (as users) were prepared to move round to the side to look at it!

However, we have designed our 3D software so that it folds this 'imaginary' screen on the side of the monitor round, until it lies in the same plane as the monitor's screen. It is also helpful to fold an imaginary screen from the top of monitor so that we have three views arranged on the screen and all visible at the same time.

Our next difficulty arises because the mouse only works when it is on a flat



surface. You cannot pick it up and move it about on the side of a computer monitor! But wait, you don't pick the mouse up and move it about the screen for a 2-D drawing either. You just assume that its pointer is on the screen in front of you.

Therefore, let's assume that when the mouse pointer is in a part of the screen which we are using as one of the folded views. It moves as if it were on the side or top of the monitor.

There we are; that's got the 3D screen and mouse sorted out.

SoftF/X Pro Metaphor

The Main metaphor of the SoftF/X Pro program is to observe the standard conventions of:

- Click on an item to select it.
- Double click for extended information
- Click and drag to move the item.

All the user actions take place in a cubic volume called the WindowBox. A 3D Cursor may be placed anywhere inside the WindowBox by moving it in one or more of the view windows.

The Animator and Modeler modules have two main user interface components, Tools and Actions. Tools are appropriate for repeated use and actions are usually applied one at a time, When a tool is activated it stays active until

another tool is selected. In the Modeler, actions are normally prohibited unless the default tool is in use.

The language of the Animation and Modeling

We use a "Triangular Faceted" model of the objects that are to be animated.

Objects can be anything you like, a car, a plane, a spaceship, a cup, a box of breakfast cereal, a dog or a human figure.

The triangular faces are joined together to form the surface of the models and the more numerous the triangles, the more realistic the representation of the object the model is.

Deciding how many triangles to use in the model is not an exact science. The triangular faces can be given additional attributes such as color and texture to make the models look realistic. They can even have a picture or another animation painted onto them.

The triangular facets (faces of the model) are positioned using a vertex at each corner. Each triangular face is surrounded by three edges. The edges make up a "wireframe" description of the model.

You use the mouse to move the vertices of the model in 3D space and thus change the shape of the model. There are many tools and actions to help you.

The action takes place over a number of 'frames', a movie usually plays 24 frames per second, conventional cartoon animation usually shows 12 different frames per second. A reasonable animation will consist of about 60 frames. But it could stretch to nearly 1000 frames for just one shot!

Models are built inside a cubic region of space called the "WindowBox". The size and position of the "WindowBox" is easily changed.

A Keyframer specifies costume, movement, orientation and scale in "Keyframes". Animations are rehearsed in "wireframe" in the Animation module before the drawing is rendered in full color, frame by frame. Each frame may be compressed together into an FLI or FLC animation file.

In an animation the term "Object" is used to describe each directable element. An Object can be a "Star" which performs in front of the camera, the Camera from which the animation is viewed, one of a number of lights, or a path along which smooth and accelerated motion is possible.

The "Star" Objects wear a "Costume" which is an model created in the Modeler. Objects can change costume during the animation. (This allows model morphing - changes of shape, for example, a fish can turn into a bird).

3D Views of Objects

Let's imagine we are looking through the front face of a cubic fish tank, with some goldfish swimming inside. The other sides, the top and the bottom of the tank are covered up. Our view into the tank shows the goldfish swimming up and down, turning, going behind a rock and moving behind and in front of one another.

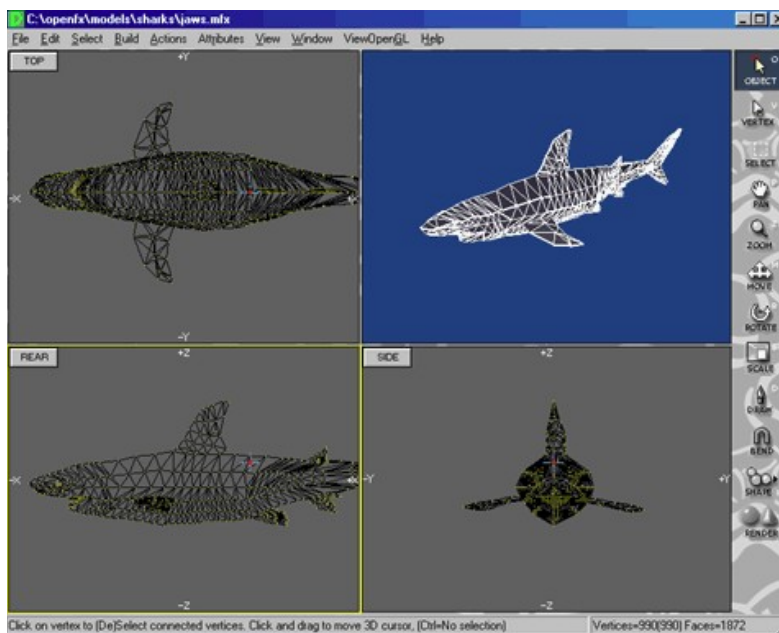
As a fish turns we can see its left side and then its right side. If we are going to make an animation of the goldfish in the tank we need to know what the fish looks like from the left side, the right side and from any other point of view we care to choose.

If we had a 2D animation package we would need to be very good artists to make the necessary drawings as our fish turns, even if the computer helps by doing the 'inbetweens'. A 3D package is very clever because it will draw the appropriate view of the goldfish, but you don't get something for nothing, you will still have to tell the computer what the goldfish looks like.

There is a very subtle difference between drawing a goldfish and defining what a goldfish will look like! You can think of the difference between an architects plan and an artists drawing of a house. This analogy with the plan of the house is quite a good one. The plan tells you everything about the house but it doesn't show you what it will look like. A plan enables the builders to put the house together, you can then walk round and through the house to see what it looks like from wherever you want. However every time an artist wants to draw the house from a different view point he or she will have to start from scratch.

The same thing applies to goldfish or any other object. A plan contains much more information than a drawing but it is not what we like to look at, a plan doesn't sell many houses, an artists impression may do!

The essence of 3D animation is that it takes a plan for objects and draws them as an artist would. The plan that we use is not quite the same as an architect's plan and thus we prefer to call it a model. In the example of our goldfish, the model provides information on what a goldfish looks like and it is stored in the computers memory.



You are now probably asking yourselves; what do we mean by a model that we can store in the computers memory? It can't be anything physical such as clay, plastic or balsa wood!

It has got to be numbers. What numbers? How many? What do the numbers mean? How do we use them to get the computer to draw a lifelike

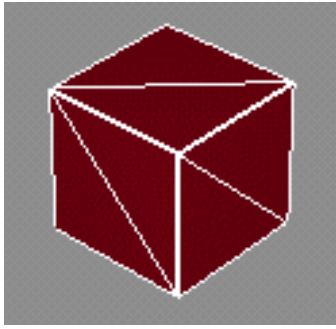
goldfish?

We will now attempt to answer these questions, but first, here is another question. What do you see when you look at a goldfish?

You see its skin (its surface). Thus the minimum that the model must do is to represent the surface of the fish. There is no point in modeling internal bits, they won't be visible. This surface is likely to be quite irregular (fins, eyes etc.). So how do we represent the surface in terms of computer data?

3D Models of Objects

Ultimately we must know where every point on the surface is, what color it is and, what it is made of and how much light is falling on it. We might think of recording a 3D co-ordinate for every point on the surface. But how many points are there on the surface of a goldfish, or put another way, what is the size of a point?

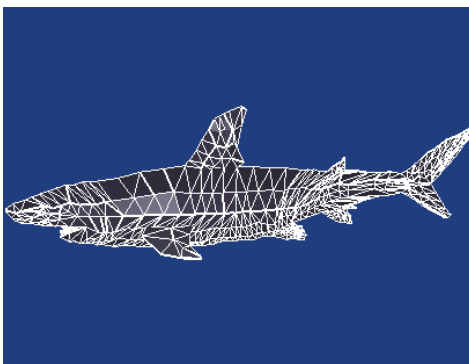


A point has no size, therefore we would need an infinite number of points to specify the surface of our goldfish (well one per molecule anyway). No computer can store an infinite number of points so we must give our point some size, some area. The simplest shape that has area is a triangle. A triangle has three edges, two of which meet at a point (a vertex).

Therefore there are three vertices per triangle. If we specify a 3D co-ordinate for each vertex we have completely specified the position and orientation of a triangle. We can even find the co-ordinates of points within a triangle if we need to do so.

If we imagine a goldfish's surface covered with triangles and each triangle defined by the co-ordinates of its vertices we have a set of numbers that model the shape of a goldfish. The triangular pieces (which we will call faces) don't have to be the same size or shape (they must of course be triangular).

A cube of any size can be described exactly using 8 3D co-ordinates, one for each vertex. The 12 triangular faces are recorded by storing the identity of the vertex at each corner of the triangle, 36 numbers. A cube made up from triangular pieces has 18 edges and since each edge is joined to 2 vertices, two numbers are needed to specify an edge. Therefore 36 numbers are needed to define the edges in a cube made up from triangular faces. As mentioned earlier, 3 numbers are needed to specify a 3D co-ordinate, thus the shape of a cube can be fully specified by $24+36+36=96$ numbers.



The computer loves manipulating numbers and from a numerical model it can reproduce a view of a cubic goldfish tank, goldfish or anything else. Building a numerical model for an object means that you won't have to do any additional modeling works to change you view point or the lighting conditions?

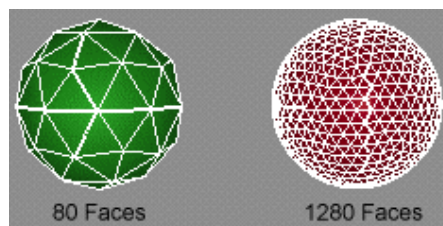
There is one drawback to using triangular faces to describe the surface of an object and that is that objects are rarely made up of triangular pieces! The surface

of a sphere is not made up of triangular pieces, it is perfectly curved. Some objects can be exactly modeled by triangular pieces, the pyramids at Gisa for example. Any object that has part of its surfaces curved cannot be represented exactly by triangular pieces, but don't despair, if we choose the right number of triangles and make them the right size and shape we will get surfaces that look as smooth and curved as the aerodynamic shape of a high performance sports car.

The pyramid example cited above would need 6 triangles to get the shape exactly right. If there had been any curved surfaces on the pyramid we would have modeled this by increasing the number of triangles, and consequently decreasing the size of each triangle so that they cover the same surface area.

To imagine how increasing the number of triangles leads to better and better models for the real thing consider a Sphere. We can get pretty good looking spheres if we use 80 triangles covering the surface, and spheres that will be impossible to tell apart from the real thing if we use 1280 triangles to cover the surface.

One of the first skills that you will need to acquire is in deciding how many vertices and faces are needed to model an object. The more vertices and faces you use the better looking the model, but you will run out of memory on the computer and it will take longer and longer to draw (or render) the images if you just go on adding vertices, edges and faces. It is usually possible to reach a good compromise



long before the machine lets you down.

It is good practice to build models, especially complex ones in pieces. Store each piece in a separate file and when all the components have been built they can be brought together to make a single model. (Much the same thing happens in any modern factory.) For example a push-bike can be made by building the frame, the front and rear wheels, the saddle, the brakes, etc. as separate bits.

Once we have build our model using the vertex, edge and face data structure its a very simple procedure to change how the model will look like, just by moving the vertices. You could for example turn our goldfish into a shark by making it larger and moving the vertices in the fins so that they look more likes those of a shark.

If you keep the number of vertices the same but change their positions you will be able to "morph" the model so that it changes from a goldfish into a shark before you very eyes. You might have more difficulty in changing the goldfish into an airplane, but it could still be done.

Painting for Realism

You might think that with the specification of the triangular model for an objects surface we have achieved our goal. However, as we have defined it, this model only describes the geometry (the shape of the object) like any good model we need to paint the surface to make it look like the real thing.

It turns out that this is not too difficult. You just have to give a few extra numbers for each face to tell the computer the color, or the reflectivity or whatever you think is important.

Terms used in SoftF/X Pro

3D Cursor - To be able to work in three dimensions it must be possible to specify a point in space where your tools do their work and the actions take effect. A two dimensional computer screen is one dimension short therefore in addition to the normal mouse pointer a second "Cursor" is provided. This cursor can be placed anywhere in 3D space by dragging it (with the mouse pointer) in one, two or three of the WindowBox view windows.

Active Window - The Active Window is the WindowBox view window with input focus. It is highlighted by being surrounded with a focus rectangle. When the multi-views are visible pressing the Tab key activates the next window in a cycle, clicking the mouse in any view window activates it.

Center Point - Each Model has a point which is used in the Animator to "hold" the model. For example if a model of an aircraft is following a path then the point in the model that is exactly on the path is the Center Point. For an aircraft it should be the center of gravity.

Channel - To direct the activity of an Object during an animation requires the use of four channels, one each for costume, movement, rotation, and scale. A channel spans a number of frames and has a Keyframe associated with it. The Keyframe is the frame in which the action of the Object is exactly specified, other frames spanned by the channel have the action of the Object determined by interpolation.

The Keyframe is always the last frame in the channel A channel therefore is a range of frames where an action of the Object is determined from the details specified in the Keyframe at the end of that channel.

Channel Span - A Channel spans the current frame if the first frame in the channel is less than or equal to the current frame and if the last frame in the channel is greater than or equal to the current frame.

Costume - To be visible in an animation an Object must wear a costume. The type of Object dictates what sort of costume an Object can wear, but Objects may change their costume during an animation.

Current Frame - The current frame is the frame in the animation that is displayed in the view windows and in which any changes take effect.

Curve - A set of vertices connected together by edges without forming any faces. Curves may be open or closed.

Edge - A line segment connecting two vertices (the plural of vertex).

Face - A triangular surface area composed of three edges joining exactly three vertices.

Follow Point - A location (specified in the Modeler module) in a model which other objects can follow.

Example: The tail rotor of a helicopter can be simulated by making models of the fuselage and rotor, setting the fuselage follows point at the axis of the tail rotor and then instructing the tail rotor to follow the fuselage.

Frame - Like a conventional cinematic film an animation is build up from a large number of still pictures (images) shown for approximately 1/24th sec. Each of these pictures is called a frame.

Gourand Shading - A rendering algorithm that shades models smoothly by blending and blurring adjacent surfaces together. Gourmand shading is the rendering method used by the OpenGL Shaded Camera View.

Image Map - An Image Map is a picture in a GIF or TGA file that is painted on to faces in the model to add decals or other surface detail.

Indicate - To choose one particular vertex, edge, face, or object to be affected by an operation by placing the 3D Cursor on or very near it (within one or two pixels). If more than one vertex, edge, face, or object is indicated in the current WindowBox view window, then the one closest to the 3D Cursor in the other Views will be used.

Keyframe - A Keyframe is a frame in which some information is specified about the activity of an Object. It always occurs at the end of a channel. The action of the Object in other frames spanned by the channel is determined by interpolation.

Model - A model is the main unit of structure. A model is made up of faces and each face is bounded by three edges. The size, position and orientation of a face is determined by the vertices that define it.

Modifier Keys - The Shift, Ctrl and Alt keys are used to modify the action of the Modeler tools. If one or other is held down as the tool is used then the action of the tool is modified in some manner.

Morphing - Enables one model to gradually change shape into another model

Node - A node is a point on a path or robot that can be manipulated. Represented by a small square on the path or the robot.

Object - An Object is the term used for each directable element in the Animator. An Object is given a costume, the costume is implied by the type of Object and costumes can change from frame to frame. An Object on the other hand is constant throughout the animation. There are 11 types of Objects. The most important of which is the "Model". A Model Object uses models from the Modeler module as its costumes. Models are stored in files with the name extension. SCE.

Object's track - The track of an Object is an implied path that the Object will follow throughout the whole animation. A track will only be visible if the Object has at least two contiguous movement channels.

Origin - The default origin is coordinate values of (0, 0, 0).

Path - A Curve that used in the animator as templates for Path Objects (.PTH) and may be used in the Modeler as paths for the Extrude along function.

Phong Shading - A rendering algorithm that shades models smoothly and accurately depicts surfaces, textures, and reflective properties (such as secularity) and can simulate mirror reflections. SoftF/X Pro uses Phong rendering by default unless ray tracing is enabled. See Phong smoothing.

Phong Smoothing - This is a procedure that helps to remove the faceted appearance from the curved surfaces of models.

Ray Tracing - A complex rendering method that accurately renders mirror effects, transparency, shadows, and refraction (bending of light through clear surfaces). Ray tracing is the most realistic rendering available in SoftF/X Pro but is also the most time consuming.

Selection Status - Each vertex in a model can be in one of three states, Selected, Deselected or Hidden. Selected vertices are manipulated with Tools and Actions. Deselected Vertices are not affected by tools or actions even if connected to selected vertices. Hidden vertices are never drawn but remain as part of the model until revealed.

Shader - A Shader is an algorithmic procedure that can add fine detail to the appearance of surfaces in the model. Examples are Wood and Marble finishes.

Skeleton - The Skeleton and the Names Hierarchy are one and the same thing. The Skeleton in addition to facilitating a hierarchical naming system for set of vertices provides a way of deforming parts of the model.

Skeleton Joint - The Skeleton is made up from a number of Joints, each Joint has a "long bone" associated to it and a "node" at one end. Each bone is attached to one other Joint at it's node or the "Root node".

Specular Highlight - Shininess is simulated by what is known as a specular highlight. Basically, the more shiny a surface the smaller the specular highlight. For example, plastic material has a less shiny appearance than tin, so a plastic bottle will have a larger specular highlight than a tin can.

Vertex - A distinct point in 3D space represented by a yellow dot if selected or a purple dot if deselected.

WindowBox - The WindowBox is a cubic volume of space which is used as the working environment for building models and setting up animations. It can be moved around and changed in size so that details on a model can be constructed without having the screen cluttered up by every vertex and edge in the whole model in view at all times. The three right angle view windows are views into the WindowBox, normally they are a Top view which looks down, a Side view and a view from the front or rear.

SoftF/X Pro Modules

The SoftF/X Pro logon dialog box gives access to the most commonly used starting points. There are three main modules, an Animator, a Modeler, and a Renderer.

The Animator module is where animations are staged and Objects perform in front of the camera. The animations are rehearsed and previewed in wireframe or shaded mode in near real time. The other modules are loaded from the Animator and remain active until explicitly closed.

The Modeler is used to design and build costumes for Objects, it uses a conventional triangular faceted 3D representation with facets attached to a network

of vertices. Like the Animator the user is given three view windows in which a wireframe for the model is built up.

The Renderer is used by both the Animator and Modeler and will continue to execute in the background if activity continues in the other modules.

Other External Libraries are available that add functionality and if suitable program development tools are available it is possible to build additional libraries. Shaders, image processors, and animation effects are examples of external libraries.

The Translation

SoftF/X Pro Помощь

Введение в 3D

В сущности, мир, в котором мы живем, является трехмерным. Поэтому, если мы хотим создать реалистичную компьютерную модель, то модель также должна быть трехмерной.

К сожалению, обычный дисплей компьютера (монитор) 2-мерный. Однако, кажется, что 2-мерности может быть достаточно, убедить зрителя, что он или она наблюдает реалистичную модель реального мира. Например, изображения, просмотренные по телевидению.

Однако, мы были ограничены двумерным ТВ и двумерным кино много лет и кажемся очень довольными этим. Это потому, что изображения, которые мы получаем по телевизору, имеют много подсознательных визуальных сигналов, которые говорят нам: "это трехмерно". И мозг верит в то, что видят глаза, даже если мы знаем, что экран плоский! Таким образом, мы можем прийти к заключению, что 2-D экран может сделать хорошую работу представления 3-D мира, если мы сделаем это правильно!

У нас есть незначительное препятствие в работе с 3D на компьютере. Экран в квартире, он имеет ширину и высоту, но нам также нужна глубина. Глубина-это измерение вне экрана. Все экраны компьютеров состоят из маленьких цветных прямоугольников, каждый прямоугольник называется "пиксель" (сокращение от элемента изображения), чем больше пикселей на экране, тем выше разрешение экрана.

Обычно, для указания разрешения компьютеров дисплей разделяют на количество пикселей сторон экрана и количество пикселей вниз по диагонали экрана, в то время как, не очень большое количество пикселей находится на

экране! Например, низкое выходное разрешение для отображения 320x200 пикселей на экране в общей сложности требует 64000 пикселей.

Двумерные художественные, анимационные или чертежные (CAD) пакеты обычно используют графический курсор (стрелку или какой-либо другой значок), чтобы вы могли сделать определенное действие в соответствующем месте на экране. Они также могут использовать координатную ссылку (скорее, как карта-Ссылка), чтобы конкретно ссылаться на точку на экране. Как ссылка карты, Вы нуждаетесь в двух числах, чтобы определить точку на экране, через X и Y. Опять же, как ссылка на карту, вам нужна фиксированная точка отсчета. Это обычно берется как верхний левый угол дисплея и называется (0,0). Перемещение по экрану точки обозначаются как (1,0) (2,0) и т. д. Перемещение вниз по экрану точки обозначаются как (0,1), (0,2) и т. д. Поэтому любая точка может быть определена двумя числами. Например, в центре экрана может быть (160,100).

В 3D нам нужно добавить третье число к координатам поперек и вниз (координата глубины "in / out"), давая нам (0,0,0), (0,0,1) и т. д. У нас проблема с визуализацией этой третьей координаты. Поскольку экран не имеет глубины, мы не можем видеть движение, которое входит / выходит. В идеале мы хотели бы второй экран на стороне монитора, чтобы увидеть движение глубины.

Мы не можем ожидать, что компьютерные производители сделают мониторы с экранами сбоку, а также спереди, даже если мы (как пользователи) были готовы двигаться вокруг сторон, чтобы посмотреть на него!

Тем не менее, мы разработали наше 3D-ПО так, что оно складывает этот "воображаемый" экран на стороне монитора, пока он не окажется в той же плоскости, что и экран монитора. Также полезно свернуть воображаемый

экран с верхней части монитора так, чтобы у нас было три вида, расположенных на экране и видимых одновременно.

Наша следующая трудность возникает потому, что мышь работает только тогда, когда он находится на плоской поверхности. Вы не можете поднять ее и переместить в сторону монитора! Но подождите, вы не поднимаете мышь и не перемещаете ее по экрану для рисования 2-D. Вы просто предполагаете, что его курсор находится на экране перед вами.

Поэтому, давайте предположим что, когда курсор находится в части экрана, который мы используем в качестве одного из сложенных видов. Он движется, как если бы он был сбоку или сверху монитора.

Там мы получили 3D экран и мышь, в которой они взаимодействуют.

Метафора SoftF/X Pro

Основная метафора SoftF/X Pro программа должна наблюдать стандартные соглашения:

- Кликнуть по элементу, чтобы выбрать его;
- Двойной клик для расширенной информации;
- Кликнуть и зажать, чтобы переместить элемент.

Все пользовательские действия происходят в кубическом объеме, названном WindowBox. 3D курсор может быть помещен куда угодно в WindowBox, перемещая ее в один или больше окон просмотров.

У модулей Аниматора и Разработчика модели есть два основных компонента пользовательского интерфейса, Инструменты и Действия. Инструменты являются подходящими для повторных действий, и они обычно применяются по одному. Когда инструмент активирован, он остается активным, пока другой инструмент не выбран. В Разработчике модели

обычно запрещают действия, если инструмент по умолчанию не используется.

Язык Анимации и Моделирования

Мы используем "Треугольную Граненую" модель объектов, которые должны быть анимированы.

Объектами могут быть все, что угодно: автомобиль, самолет, космический корабль, чашка, коробка с завтраком, собака или человеческая фигура.

Треугольные грани соединены вместе, чтобы сформировать поверхность моделей и чем больше треугольников, тем более реалистичным является представление объекта модели.

Решение о том, сколько треугольников использовать в модели, не является точной наукой. Треугольным граням можно придать дополнительные атрибуты, такие как цвет и текстура, чтобы сделать модели реалистичными. Они могут даже иметь изображение или другую анимацию, нарисованную на них.

Треугольные грани (грани модели) располагаются в вершине каждого угла. Каждая треугольная грань окружена тремя краями. Края составляют "каркасное" описание модели.

Вы используете мышь для перемещения вершин модели в 3D-пространстве и, таким образом изменить форму модели. Есть много инструментов и действий, которые помогут вам.

Действие происходит над несколькими "кадрами", фильм обычно играет 24 кадра в секунду, обычная мультипликационная анимация обычно показывает 12 разных кадров в секунду. Разумная анимация будет состоять

примерно из 60 кадров. Но он может растянуться до почти 1000 кадров только для одного выстрела!

Модели созданы в кубической области пространства, названного "WindowBox". Размер и позиция "WindowBox" легко изменены.

Keyframer определяет скин, движение, ориентацию и масштаб в "ключевых кадрах". Анимации обрабатываются в "каркасе" в модуле Анимации прежде, чем рисунок будет представлен в насыщенном цвете, кадр за кадром. Каждый кадр может быть сжат вместе в файл анимации FLI или FLC.

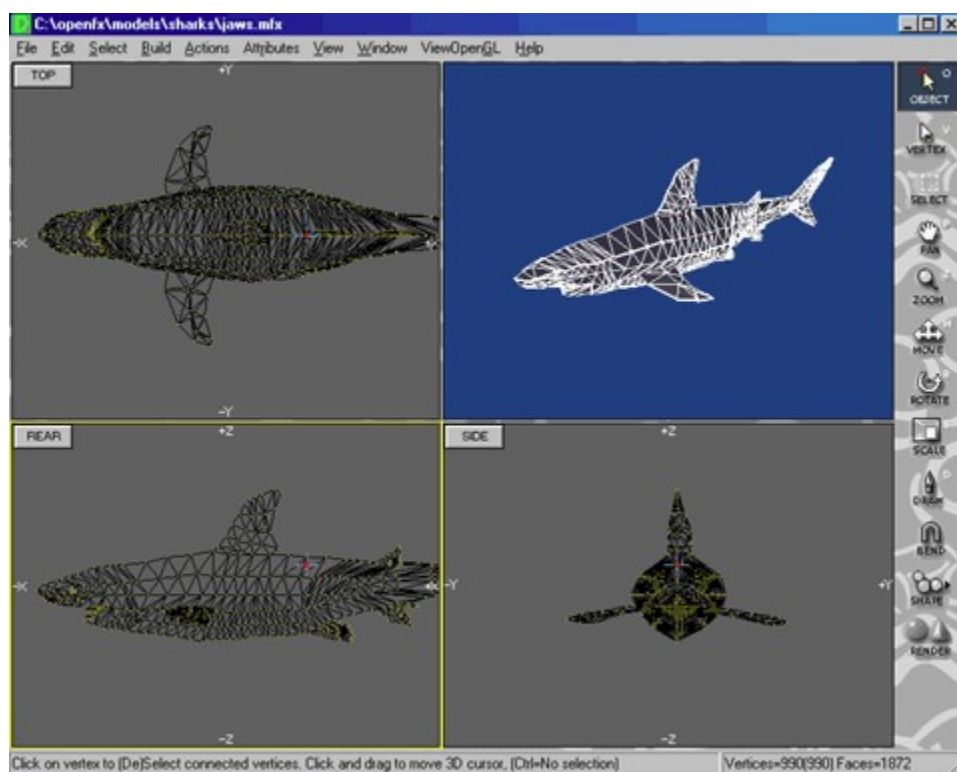
В анимации термин "Объект" использован, чтобы описать каждый управляемый элемент. Объект может быть "Звездой", которая располагается перед камерой. Камера, от которой анимация просматривается - один из многих огней, или путь, вдоль которого возможно плавное и ускоренное движение.

Объект "Звезда" имеет оболочку "Скин", который является моделью, создаваемой в Разработчике модели. Объекты могут изменить скин во время

анимации. (Это позволяет морфинг модели - изменение формы, например, рыба может превратиться в птицу).

3D Просмотр Объектов

Давайте представим, что мы просматриваем переднюю сторону кубического садка для рыбы с некоторой



золотой рыбкой, плавающей внутри. Другие стороны - вершина и нижняя часть корпуса покрыты. Наше представление в корпус показывает золотую рыбку, плавающую вверх и вниз, превращение, движение позади скалы и перемещение сзади и перед друг другом.

Поскольку рыба поворачивается, мы можем видеть ее левую сторону и затем ее правую сторону. Если мы собираемся сделать анимацию золотой рыбки в корпусе, мы должны знать то, на что рыба похожа от левой стороны, правой стороны, и с любой точки просмотра мы хотим выбрать.

Есть очень тонкие различия между рисованием золотой рыбки и определением, на что будет похожа золотая рыбка! Вы можете думать о различии между планом архитекторов и рисунком художников дома. Эта аналогия с планом дома - вполне хорошая. План говорит Вам все о доме, но это не показывает Вам, на что это будет похоже. План позволяет разработчикам соединить дом, Вы можете тогда идти через дом, чтобы видеть то, на что это похоже от того, везде, где Вы хотите. Однако каждый раз, когда художник хочет потянуть дом из точки другого представления, он или она должен будет запустить с нуля. Та же самая вещь применяется к золотой рыбке или любому другому объекту. План содержит намного больше информации чем рисунок, но это не то, на что нам нравится видеть, план не показывает многие содержания, воображение художников может сделать это!

Сущность 3D анимации - то, что она берет план относительно объектов и тянет их, как художник тянул бы. План, который мы используем, является не совсем тем же самым как планом архитектора и таким образом мы предпочитаем вызывать это модель. В примере нашей золотой рыбки модель предоставляет информацию о том, на что похожа золотая рыбка, и это сохранено в компьютерной памяти.

Вы теперь, вероятно, спрашиваете: Что мы подразумеваем под моделью, которую мы можем сохранить в компьютерной памяти? Это не может быть ничто физическое, такое как глина, пластмассовое или пробковое дерево!

Это должны быть числа. Какие числа? Сколько? Что означают числа? Как мы используем их, чтобы заставить компьютер нарисовать живую золотую рыбку?

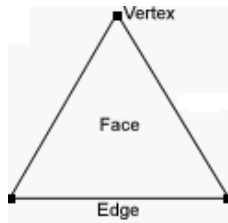
Мы теперь попытаемся ответить на эти вопросы, но сначала, здесь другой вопрос. Что Вы видите, когда смотрите на золотую рыбку?

Вы видите, что это - кожа (поверхность). Таким образом, минимум, который должна сделать модель - представить поверхность рыбы. Нет никакого смысла в моделировании внутренних битов, они не будут видимы. Эта поверхность, вероятно, будет довольно неправильна (плавники, глаза и т.д.). И так, как мы представляем поверхность с точки зрения компьютерных данных?

3D Модели Объектов

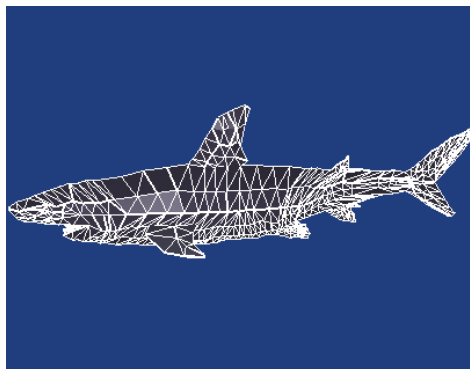
В конечном счете, мы должны знать, где каждая точка на поверхности что окрашивает, из чего сделана и насколько свет падает на нее. Мы могли бы думать о записи 3D координаты для каждой точки на поверхности. Но сколько точек находится там, на поверхности золотой рыбки или помещены иначе, каков размер точки?

У точки нет никакого размера, поэтому мы нуждались бы в бесконечном числе точек, чтобы определить поверхность нашей золотой рыбки (хорошо один на молекулу, так или иначе). Никакой компьютер не может сохранить бесконечное число точек, таким образом, мы должны дать нашей точке некоторый размер, некоторую область. Самая простая форма, у которой есть область, является треугольником. У треугольника есть три края, два из которых встречаются в точке (вершина).



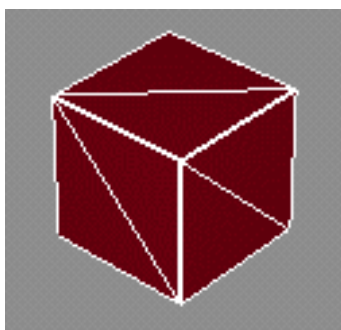
Поэтому есть три вершины на треугольник. Если мы определяем 3D координату для каждой вершины, мы полностью определили позицию и ориентацию треугольника. Мы можем даже найти координаты точек в пределах треугольника, если нам надо сделать так.

Если мы представляем поверхность золотой рыбки крытой треугольниками и каждым треугольником определенными координатами его вершин, у нас есть ряд чисел, которые моделируют форму золотой рыбки. Треугольные части (назовем их лицом) не должны быть тем же самым размером или формой (они должны быть, конечно, треугольными).



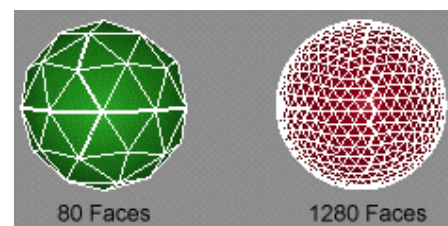
Например, предположите, что мы хотим смоделировать куб, 12 треугольников дадут совершенный куб независимо от того, каков размер куба. Поскольку несколько из вершин треугольных поверхностей происходят в том же самом месте, мы не можем сохранить 3 вершины на поверхность, 8 вершин сделают задание для целого куба. Таким же образом, если у двух треугольников есть общий край, мы должны только сохранить единственный край, который может использоваться для обоих треугольников.

Компьютер любит управлять числами, и от численной модели он может воспроизвести представление кубического корпуса золотой рыбки, золотой рыбки или чего-либо еще. Создание численной модели для объекта означает, что Вы не должны будете сделать никакой дополнительной работы моделирования, чтобы изменить Вас точка наблюдения или условия освещения.



Есть один недостаток к использованию треугольных поверхностей, чтобы описать поверхность объекта, и это - то, что объекты редко состояются из треугольных частей! Поверхность сферы не составлена из треугольных частей, она отлично изогнута. Некоторые объекты могут быть точно смоделированы треугольными частями, пирамидами в Gisa, например. Любой объект, у которого есть часть его изогнутых поверхностей, не может быть представлен точно треугольными частями, но не отчаивается, если мы выберем правильное число треугольников и сделаем тогда правильный размер и форму, то мы получим поверхности, которые выглядят столь же гладкими и кривыми как аэродинамическая форма высокопроизводительного спортивного автомобиля.

Пример пирамиды, приведенный выше, нуждался бы в 6 треугольниках, чтобы получить форму, точно правильную. Если бы были какие-либо кривые поверхности на пирамиде, то мы смоделировали бы это, увеличивая число треугольников, и следовательно



при уменьшении размера каждого треугольника так, чтобы они покрыли ту же самую площадь поверхности.

Чтобы представить, как увеличение числа треугольников приводит лучше и лучшие модели для реальной вещи рассматривают Сферу. Мы можем получить довольно хорошо выглядящие сферы, если мы используем 80 треугольников, покрывающих поверхность, и сферы, которые будет невозможно сказать кроме реальной вещи, если мы будем использовать 1280 треугольников, чтобы покрыть поверхность.

Одни из первых навыков, которые Вы должны будете получить, находятся в решении, сколько вершин и поверхностей необходимо, чтобы смоделировать объект. Больше вершин и поверхностей, Вы используете лучше выглядящее модель, но Вы исчерпаете память на компьютере, и это займет больше времени и дольше потянуть (или рендеринг) изображения, если Вы только продолжите добавлять вершины, края и поверхности. Обычно возможно достигнуть хорошего компромисса задолго до того, как машина подводит Вас.

Это - хорошая практика, чтобы создать модели, особенно сложные в частях. Сохраните каждую часть в отдельном файле и когда все компоненты были созданы, они могут быть объединены, чтобы сделать единственную модель. (Почти такая же вещь происходит на любой современной фабрике.) Например велосипед может быть сделан, создавая фрейм, передние и задние колеса, седло, тормоза, и т.д. как отдельные биты.

Как только у нас есть сборка, наша модель, используя вершину, край и дату поверхности структурирует очень простая процедура, чтобы измениться, как модель будет похожа, только перемещая вершины. Вы могли, например, превратить нашу золотую рыбку в акулу, делая это больше и при перемещении вершин в плавники так, чтобы они больше походили на таковые из акулы.

Если Вы сохраните число вершин тем же самым, но измените их позиции, то Вы будете в состоянии "превратить" модель так, чтобы это изменило от золотой рыбки в акулу перед Вами очень глаза. Вы могли бы испытать больше затруднений в изменении золотой рыбки в самолет, но это могло все еще быть сделано.

Рисование для Реализма

Вы могли бы думать, что со спецификацией треугольной модели объекты появляются - мы достигли нашей цели. Однако, поскольку мы определили это, эта модель только описывает геометрию (форма объекта) как любая хорошая модель, мы должны нарисовать поверхность, чтобы заставить это быть похожим на реальную вещь.

Оказывается, что это не является слишком трудным. Вы только должны дать несколько дополнительных чисел для каждой поверхности, чтобы сказать компьютеру цвет, или коэффициент отражения или независимо от того, что Вы думаете, важно.

Термины, использованные в SoftF/X Pro

3D Курсор - Чтобы быть в состоянии работать в трех измерениях, должно определить точку в пространстве, где Ваши инструменты делают свою работу и действия, вступает в силу. Двумерный монитор - одна размерность, короткая в дополнение к нормальному указателю мыши обеспечен, второй "Курсор". Этот курсор может быть помещен куда угодно в 3D пространстве, перетаскивая это (с указателем мыши) в один, два или три из окон представления WindowBox.

Активное окно - окно представления WindowBox с фокусом ввода. Это выделено, будучи окруженным прямоугольником фокуса. Когда мультипредставление – видимое - нажатие клавиши Tab, активируется,

следующее окно в цикле. Щелкая мышью в любом окне – активирует представление.

Центральная точка - у каждой Модели есть точка, которая используется в Аниматоре, чтобы "содержать" модель. Например, если модель самолета следует за путем тогда точка в модели, которая находится точно на пути, Центральная точка. Для самолета это должен быть центр тяжести.

Канал - Чтобы направить действие Объекта во время анимации требует использования четырех каналов, один каждый для костюма, перемещения, вращения, и масштаба. Канал охватывает много фреймов и связал Ключевой кадр с ним. Ключевой кадр - фрейм, в котором точно определено действие Объекта, у других фреймов, заполненных каналом, есть действие Объекта, определенного интерполяцией.

Ключевой кадр - последний фрейм в канале, канал поэтому - диапазон фреймов, где действие Объекта определено от деталей, определенных в Ключевом кадре в конце того канала.

Промежуток канала - Канал охватывает текущий фрейм, если первый фрейм в канале меньше чем или равен текущему фрейму и если последний фрейм в канале больше чем или равен текущему фрейму.

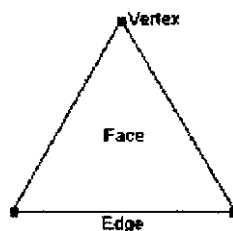
Костюм - Чтобы быть видимым в анимации Объект должен носить костюм. Тип Объекта диктует, какой костюм Объект может носить, но Объекты могут изменить свой костюм во время анимации.

Текущий Фрейм - фрейм в анимации, которая выведена на экран в окнах представления и в котором вступают в силу любые изменения.

Кривая - Ряд вершин, соединенных вместе краями, не формируя поверхностей. Кривые могут быть открытыми или закрыты.

Край - линейный сегмент, соединяющий две вершины (множественное число вершины).

Поверхность - треугольная площадь поверхности сочиняла трех краев, присоединяющихся точно к трем вершинам.



Следование за Точкой - расположение (определенный в модуле Разработчика модели) в модели, за которой могут следовать другие объекты.

Пример: ротор хвоста вертолета может быть моделирован, делая модели фюзеляжа и ротора, установка фюзеляжа следует за точкой в оси ротора хвоста и затем того, чтобы давать команду ротору хвоста следовать за фюзеляжем.

Фрейм - Как стандартный кинематографический фильм, который анимация, растет от большого количества фотоснимков (изображения), показанные в течение приблизительно 1/24-ой секунды. Каждое из этих изображений вызывают фреймом.

Закраска по методу Гуро - алгоритм рендеринга, который заштриховывает модели гладко, смешиваясь и при размывании смежных поверхностей вместе. Закраска по методу Гуро - метод рендеринга, используемый OpenGL Теневое Представление Камеры.

Карта изображения - Карта Изображения - изображение в GIF или в TGA, который подрисован к поверхностям в модели, чтобы добавить переводные картинки или другую поверхностную деталь.

Указание - Чтобы выбрать одну определенную вершину, край, поверхность, или объект, который будет влиять на работу, помещая 3D Курсор рядом (в пределах одного или двух пикселей). Если больше чем одна вершина, край, поверхность, или объект будут обозначены в текущем окне

представления WindowBox, то одно самое близкое к 3D Курсору в других Просмотре будет использоваться.

Ключевой кадр - фрейм, в котором некоторая информация определена о действии Объекта. Это всегда происходит в конце канала. Действие Объекта в других фреймах, заполненных каналом, определено интерполяцией.

Модель - основной модуль структуры. Модель составлена из поверхностей, и каждая поверхность ограничена тремя краями. Размер, позиция и ориентация поверхности определены вершинами, которые определяют это.

Модифицирующие клавиши - Сдвиг, клавиши Ctrl и клавиши Alt используются, чтобы изменить действие инструментов Разработчика модели. Если один или другой удерживается, поскольку инструмент используется тогда, действие инструмента изменено некоторым способом.

Превращение - Морфинг-позволяет одной модели постепенно менять форму на другую.

Узел - точка на пути или работе, которым можно управлять. Представляется небольшим квадратом на пути или работе.

Объект - термин, использованный для каждого directable элемента в Аниматоре. Объекту дают костюм, костюм подразумевает типом Объекта, и костюмы могут измениться от фрейма до фрейма.

Объект, с другой стороны, является постоянным в течение анимации. Есть 11 типов Объектов. Самым важным из которых является "Модель". Объект Модели использует модели от модуля Разработчика модели как его костюмы. Модели сохранены в файлах с расширением имени. SCE.

Дорожка объекта - подразумеваемый путь, за которым Объект будет следовать в течение целой анимации. Дорожка только будет видима, если у Объекта будет по крайней мере два непрерывных канала перемещения.

Источник - источник по умолчанию - координатные значения (0, 0, 0).

Путь - Кривая, которая использовалась в аниматоре в качестве шаблонов для Объектов Пути (.PTH) и может использоваться в Разработчике модели в качестве путей для Экструзии вдоль функции.

Закраска по методу Фонга - алгоритм рендеринга, который заштриховывает модели гладко и точно изображает поверхности, текстуры, и отражающие свойства (такие как specularity) и может моделировать зеркальные отражения. SoftF/X Pro использует Phong, представляющий по умолчанию, если трассировка лучей не включена. См. сглаживание Phong.

Сглаживание Phong - Это процедура, которая помогает удалить фасетное появление из кривых поверхностей моделей.

Трассировка лучей - сложный метод рендеринга, который точно представляет зеркальные эффекты, прозрачность, тени, и преломление (изгиб света через четкие поверхности). Трассировка лучей - самый реалистичский рендеринг, доступный в SoftF/X Pro, но является также самой трудоёмкой.

Состояние выбора - Каждая вершина в модели может быть в одном из трех состояний, Выбранных, Невыбранных или Скрытых. Выбранными вершинами управляют с Инструментами и Действиями. На невыбранные Вершины не влияют инструменты или действия, даже если соединенный с выбранными вершинами. Скрытые вершины никогда не оттягиваются, но остаются частью модели пока не показано.

Программа построения теней - Программа построения теней - алгоритмическая процедура, которая может добавить мелкие детали к появлению поверхностей в модели. Примеры - Лес и Мраморные концы.

Скелет - Скелет и Иерархия Имен - одна и та же вещь. Скелет в дополнение к облегчению иерархической системы именования для набора вершин обеспечивает способ деформировать части модели.

Скелетное Соединение - Скелет составлен от многих Соединений, у каждого Соединения есть "длинная кость", связанная с этим и "узел" в одном конце. Каждая кость присоединена к одному другому Соединению в, это - узел или "Корневой узел".

Зеркальное Выделение - Блеск моделируется тем, что известно как зеркальное выделение. В основном, более солнечное поверхность меньше зеркальное выделение. Например, у пластмассового материала есть менее солнечное появление чем олово, таким образом, у пластмассовой бутылки будет большее зеркальное выделение чем консервная банка.

Вершина - отличная точка в 3-D пространстве, представленном желтой точкой если выбрано или фиолетовой точкой если не выбранный.

WindowBox - кубический объем пространства, которое использовано как рабочая среда для создания анимации установки и модели. Это может быть перемещено и изменено в размере так, чтобы детали о модели могли быть созданы, не загромождая экран каждой вершиной и краем в целой модели в поле зрения всегда. Три прямоугольных окна представления - представления в WindowBox, обычно они - Вид сверху, который смотрит вниз, Вид сбоку и представление от передней стороны или задней части.

SoftF/X Pro Модули

SoftF/X Pro входят в систему, диалоговое окно предоставляет доступ к обычно используемым начальным точкам. Есть три основных модуля, Аниматор, Разработчик модели, и Средство рендеринга.

Модуль Аниматора - то, где анимации подготовлены, и Объекты выполняют перед камерой. Анимации репетируются и предварительно просмотрены в каркасе или теневом режиме в псевдореальное время. Другие модули загружены от Аниматора и остаются активными пока явно не закрыто.

Разработчик модели используется, чтобы разработать и создать костюмы для Объектов, это использует стандартное треугольное фасетное 3D представление с фасетами, присоединенными к сети вершин. Как Аниматор пользователю дают три окна представления, в которых создан каркас для модели.

Средство рендеринга используется и Аниматором и Разработчиком модели и будет продолжать выполняться в фоновом режиме, если действие продолжается в других модулях.

Другие Внешние Библиотеки доступны, которые добавляют функциональность и если подходящие средства разработки программы доступны, возможно, создать дополнительные библиотеки. Программы построения теней, процессоры изображения, и эффекты анимации - примеры внешних библиотек.