

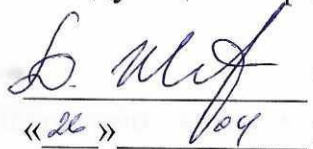
Учреждение образования
«Белорусский государственный университет культуры и искусств»

Факультет культурологии и социокультурной деятельности

Кафедра информационных технологий в культуре

СОГЛАСОВАНО

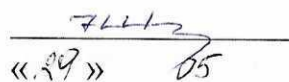
Заведующий кафедрой


«26» 2023 г.

Т.С. Жилинская
2023 г.

СОГЛАСОВАНО

Декан факультета


«29» 2023 г.

Н.Е. Шелупенко
2023 г.

**ЭЛЕКТРОННЫЙ УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС
ПО УЧЕБНОЙ ДИСЦИПЛИНЕ
ТЕХНОЛОГИИ 3D АНИМАЦИИ**

для специальности

*1–21 04 01 Культурология (по направлениям),
направления специальности 1-21 04 01-02 Культурология (прикладная)
специализации 1-21 04 01-02 04 Информационные системы в культуре*

3 курс 5семестр

Составитель:

С.А. Гончарова, доцент кафедры информационных технологий в культуре учреждения образования «Белорусский государственный университет культуры и искусств», кандидат технических наук, доцент

Т.В. Бачурина, старший преподаватель кафедры информационных технологий в культуре учреждения образования «Белорусский государственный университет культуры и искусств».

Рассмотрен и утвержден

на заседании Совета факультета культурологии и социально-культурной деятельности 29.05.2023 г.

протокол № 10

Составители:

С.А. Гончарова, доцент кафедры информационных технологий в культуре учреждения образования “Белорусский государственный университет культуры и искусств”, кандидат технических наук, доцент;

Т.В. Бачурина, старший преподаватель кафедры информационных технологий в культуре учреждения образования “Белорусский государственный университет культуры и искусств”.

Рецензенты:

кафедра дискретной математики и алгоритмики учреждения образования «Белорусский государственный университет»;

В.А. Касан, кандидат педагогических наук, профессор кафедры информационных ресурсов и коммуникаций учреждения образования “Белорусский государственный университет культуры и искусств”, доцент.

Рассмотрен и рекомендован к утверждению:

Кафедрой информационных технологий в культуре
(протокол от 25.05.2023 г. № 9);

Советом факультета культурологии и социально-культурной деятельности
(протокол от 29.05.2023 г. №)

СОДЕРЖАНИЕ

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	4
2. ТЕОРЕТИЧЕСКИЙ РАЗДЕЛ	7
2.1 Конспекты лекций	7
3. ПРАКТИЧЕСКИЙ РАЗДЕЛ	22
3.1 Описание лабораторных работ	22
4. РАЗДЕЛ КОНТРОЛЯ ЗНАНИЙ	130
4.1 Задания для контролируемой самостоятельной работы студентов	130
4.2 Перечень контрольных вопросов по дисциплине	131
4.3 Перечень вопросов к зачету	134
4.4 Критерии оценки результатов учебной деятельности студентов	135
5. ВСПОМОГАТЕЛЬНЫЙ РАЗДЕЛ	136
5.1 Учебная программа	136
5.2 Учебно-методическая карта учебной дисциплины для очной формы получения высшего образования	136
5.3 Список основной литературы	137
5.4 Список дополнительной литературы	137
5.5 Перечень рекомендуемых электронных образовательных ресурсов	137

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

В условиях современного общества неотъемлемым качеством квалифицированного специалиста сферы культуры и искусства становится высокий уровень владения технологиями трехмерного компьютерного моделирования. Это предполагает знание основных программ 3D графики и умение применять их интегрировано на всех этапах разработки художественного продукта и в социально-культурной деятельности.

Цель преподавания дисциплины «Технологии 3D анимации» состоит в том, чтобы сформировать у обучаемых теоретические знания и практические навыки по использованию программных и технических средств трехмерной анимации в профессиональной деятельности менеджера-культуролога при создании и обработке проектов различного назначения в сфере культуры и искусства. Объектом изучения дисциплины являются трехмерные графические изображения. Предметом – методы и технологии их анимации.

Целевая направленность дисциплины обуславливает решение следующих задач: изучение основных понятий и концепций теории анимации, освоение технологий анимации и симуляции трехмерных объектов и визуализации анимированных сцен, знание основных возможностей пакетов прикладных программ и умение применять их интегрировано для решения творческих задач.

Программой предусмотрены преемственные связи со следующими дисциплинами учебного плана: «Технологии компьютерной графики», «Информационные технологии художественного проектирования», «Технологии динамической графики», «Игровая культура и дизайн».

В процессе изучения дисциплины у студента должна быть сформирована СК-24 – Использовать технологии 3D анимации при создании продукта сферы культуры и искусств.

В результате изучения дисциплины студенты *должны знать*:

- основные понятия и концепции геометрического моделирования и анимации;
- особенности конфигурации персонального компьютера для рендеринга анимации;
- технологии анимирования и их особенности;
- основные способы задания анимации трехмерных объектов;
- особенности создания 3D контента для игр;
- этапы создания анимационного ролика;
- особенности рендеринга анимированной сцены;
- основные возможности пакетов прикладных программ 3D-графики.

Должны уметь:

- настраивать устройства персонального компьютера (видеокарту, монитор) для работы с трехмерной графикой;
- работать в среде основных программ трехмерной графики;
- использовать различные методы анимирования трехмерных объектов;
- выстраивать композицию кадра анимированной сцены;
- применять различные схемы и методы освещения в зависимости от требований визуализации сцены;
- работать с графическими ресурсами сети интернет, использовать сетевые средства поиска и обмена графической информацией.
- работать с графическими ресурсами сети интернет, использовать сетевые средства поиска и обмена графической информацией.

Основными методами обучения, отвечающими целям изучения дисциплины, являются:

- информационные (устное изложение, объяснение, мультимедийная демонстрация, консультирование).
- поисково-творческие (наблюдение, творческий диалог, эксперимент).
- методы самостоятельной работы студентов.

В предлагаемом учебно-методическом комплексе представлены материалы четырех разделов: теоретический (материалы для теоретического изучения учебной дисциплины), практический (материалы для проведения лабораторных, практических, семинарских и индивидуальных учебных занятий), раздел контроля знаний (материалы, позволяющие определить соответствие результатов учебной деятельности обучающихся требованиям образовательных стандартов высшего образования и учебно-программной документации образовательных программ высшего образования), вспомогательный раздел (учебно-методическая документация, перечень учебных изданий и информационно-аналитических материалов, рекомендуемых для изучения учебной дисциплины). Методика проведения занятий предполагает использование проблемных и эвристических форм обучения, которые развивают индивидуальность студентов, самостоятельность их мышления. Сотрудничество в группах, коллективные проекты, индивидуальное проектирование – основные методы обучения. Материал излагается на основе современных методических требований с учетом педагогических целей на уровнях представления, понимания, знания, применения и творчества.

Практические и лабораторные занятия направлены на формирование умений использования полученных знаний при решении конкретных задач. Методика их проведения должна содействовать развитию творческих способ-

ностей каждого студента и приобретению навыков самостоятельной работы.

В соответствии с учебным планом на изучение учебной дисциплины «Трёхмерное моделирование и анимация» всего предусмотрено 102 часа, из которых 62 часа – аудиторные занятия (6 часа – лекции, 56 часов – лабораторные занятия). Формы контроля – зачет.

2. ТЕОРЕТИЧЕСКИЙ РАЗДЕЛ

2.1 Конспекты лекций 3 курс 5 семестр (6 часов)

Лекция 1. Введение. Теория анимации

Цель. Знакомство с историей мультипликации, изучение различных видов анимации.

Анимация – это иллюзия движения, созданная за счет последовательного отображения изображений.

Анимация – технические приёмы создания иллюзии движущихся изображений с помощью последовательности неподвижных изображений, сменяющих друг друга с большой частотой.

Эпоха до изобретения кино. Оптические игрушки XIX века. Пионеры кино и анимации. Художники и изобретатели. Уолт Дисней. Студийное производство и первые полные метры. Звук и цвет в анимации. Союзмультфильм. Национальные школы. Послевоенная анимация. Лимитированный стиль, сериалы и авторские короткометражки. Эксперименты 60-х XX века. Японская анимация. История появления компьютерной анимации. Студия PIXAR. Авторская анимация XXI века.

История мультипликации

История анимации начинается задолго до появления кино. Художники всех времен и народов мечтали о возможности передать в своих произведениях подлинное движение жизни. Яркую передачу движения находим мы в искусстве древнего Египта и древней Греции – в скульптурных рельефах, в росписях гробниц и храмов фараонов и в рисунках, украшающих вазы. В дальнейшем стремление получить иллюзию движения от рисунка привело к созданию китайского театра теней, когда тени рисунков, проецируемые на освещенной плоскости, начали двигаться – начали жить. В этом уже был заложен один из основных элементов будущего искусства мультипликации.

70-е годы до н.э. – Римский поэт и философ Лукреций в трактате "О природе вещей" описал приспособление для высвечивания на экране движущихся рисунков.

X-XI вв. – Первые упоминания о китайском театре теней – типе зрелища, визуально близком будущему анимационному фильму.

XV в. – Появились книжки с рисунками, воспроизводившими различные фазы движения человеческой фигуры. Свернутые в рулон, а затем мгновенно разворачивавшиеся, эти книжки создавали иллюзию оживших рисунков.

В средние века также находились умельцы, развлекавшие публику сеансами движущихся картинок при помощи оптических устройств наподобие фильмоскопов, куда вставляли прозрачные пластины с рисунками. Такие аппараты называли волшебным фонарем или по-латински "laterna magica".

1646 г. – иезуитский монах Атанасиус Киршер дал первое описание устройства сконструированного им "волшебного фонаря" – прибора, который высвечивал изображение на прозрачном стекле. С XVII века в бродячих театрах по всей Европе проводились такие представления.

1832 г. – изобретение фенакистископа. Попытки найти способы оживления рисунков посредством специальных аппаратов задолго предшествуют появлению кинематографа. Молодой бельгийский профессор Жозеф Плато построил еще в 1832 году маленький лабораторный прибор – фенакистископ, конструкция которого основана на способности сетчатки человеческого глаза сохранять изображения (название это происходит от греческого слова "фенакс" – обманщик и корня "скоп" – смотреть). Ощущения, возникающие в наших органах чувств, не угасают сразу. Некоторое время глаз сохраняет световое изображение предмета и после того как перестает смотреть на него. Способность сетчатки человеческого глаза сохранять изображение позволяет нам видеть огненный круг в воздухе, когда вращают горящий факел, звездный след и т.п.

Вот как описывает Жозеф Плато свое изобретение: "Мой прибор состоит из черного картонного диска диаметром приблизительно 25 сантиметров, насаженного на ось подобно колесу. Недалеко от внешней окружности диска проделано до двадцати отверстий в виде радиально направленных щелей. Эти щели могут иметь около 2 мм ширины и 2 см длины и должны быть проделаны на равных расстояниях друг от друга. Для наблюдения изменяющихся явлений в их истинном виде поступают следующим образом: приводят диск в достаточно быстрое вращение, закрывают один глаз, а другим смотрят сквозь образующуюся от быстрого вращения щелей прозрачную полосу на движущийся предмет."

1832 г. – Тот же принцип был положен венским профессором Симоном фон Штампефером в основу стробоскопа. Эффект видимого движения при смене неподвижных изображений, сделанных на внутреннем диске, стал называться стробоскопическим эффектом.

1834 г. – Зоотроп, в котором, подобно стробоскопу, двигались наклеенные на ленту рисунки, был сконструирован английским математиком Уильямом Джорджем Хорнером. Впрочем, в оригинале аппарат Хорнера назывался "дедалеум" (в честь Дедала, который, по преданию, создал движущиеся картины людей и животных), и лишь последующие варианты этого аппарата (в частности, аппарат, который сконструировал в 1860 году

француз Дезвинь, и аналогичный аппарат американца Уильяма Е. Линкольна) закрепили за всеми подобными устройствами название "зоотропа".

1853 г. – Австриец барон фон Ухациус, сконструировав стробоскоп, в котором изображения помещались на стеклянном диске и освещались масляной лампой, впервые спроецировал их на экран.

1861 г. – Кинематоскоп американца Колмана Селлера представлял серию наклеенных на барабан фотографий, сменявших друг друга перед глазами зрителя.

1870 г. – Генри Хейл из Колумбуса (штат Огайо) организовал первый в истории публичный сеанс "живой" фотографии. Он применил волшебный фонарь, проецирующий диапозитивы, размещенные на диске. Фазатрон, так назывался аппарат Хейла, отличался от аппарата Ухациуса только тем, что рисунки в нем были заменены фотографиями. На диске Хейла были помещены последовательные фазы движения вальсирующей пары. Главным препятствием в "оживлении" фотографий на экране была сложность воссоздания последовательных фаз движения, и изобретательная мысль привела в дальнейшем к весьма интересным открытиям.

1877 г. – француз Эмиль Рейно, заимствовав вращательный барабан зоотропа Хорнера и усовершенствовав зеркальную систему фенакистископа Жозефа Плато, создал новый, более совершенный прибор, назвав его "праксиноскопом" Рейно. Этот день можно считать днем рождения анимации, а именно 30 августа 1877 года, когда в Париже был запатентован подобный аппарат – праксиноскоп Эмиля Рейно.

1880 г. – Соединив свой праксиноскоп с проекционным фонарем, Эмиль Рейно с большим успехом демонстрировал свои рисунки на экране. Но это еще было простая форма мультипликации. И только когда Эмилю Рейно пришла идея заменить в ленте барабана цикловые рисунки, которые воспроизводили одно и то же движение, начиная и замыкая его, более длинными лентами, на которых в последовательных рисунках разыгрывались занимательные сюжеты маленьких смешных пьесок, буффонады, и пантомимы, тогда он действительно создал искусство мультипликации, искусство "одушевленного рисунка", придав ему форму спектакля.

В то же время, с 1870 года, в Сан-Франциско англичанин Эдвард Мьюбридж начал свою работу по съемке последовательных фаз движения. Его исследования и по сей день представляют огромный интерес для художников и особенно для мультипликаторов. Работами Мьюбриджа заинтересовались такие крупные ученые и художники, как Жюль Маре, Гельмгольц, Мейссонье, Эдгар Дега, Жерико и Делакруа. Когда Мейссонье впервые увидел фотографии бегущих и скачущих лошадей, он был поражен. "Аппарат врет", – заявил Мейссонье. Положение ног лошадей противоречило привычным канонам,

принятым по картинам Рафаэля, Берне и Жерико. Художник Мейссонье, будучи баталистом, серьезно изучал движения лошадей. Его работами восхищались Бодлер и Делакруа. Когда ему нужно было нарисовать лошадь в движении, он следовал в тележке рядом с бегущей лошадьё и делал множество набросков с натуры. Фотографии Мюйбриджа казались безобразными и неправильными. Жюль Маре пишет по этому поводу: "Разве неизвестное не кажется нам безобразным? Разве истина, когда мы видим ее впервые, всегда радует наш глаз? Мы задавали себе этот вопрос, рассматривая моментальные снимки скачущих лошадей. Положение лошади при беге, установленное аппаратом Мюйбриджа, часто казалось неправдоподобным." Мюйбридж решил спор опытным путем. Когда рисунки, сделанные по моментальным фотографиям, показали в движении на экране, самые неверующие сдались.

1884 год – Марей создал первый хронофотографический аппарат. Опыты Мюйбриджа и Марея внесли значительный вклад в изучение движений человека и животных и в развитие техники.

1885 г. – Герман Каствлера создает мутоскоп – прибор с барабаном, в котором помещалась тысяча (и больше) рисунков.

Успешное развитие фотографии позволило в дальнейшем заменить в стробоскопе рисунки фотоснимками, а изобретение прозрачной целлулоидной ленты, покрытой чувствительной эмульсией, и рождение киноаппарата дало возможность увеличить количество отдельных картинок и открыло большие перспективы по "одушевлению" рисунка на экране.

1888 г. – Рейно усовершенствовал свой праксиоскоп, перенес рисунки на целлулоидную ленту (35 мм) и высветил ее на большом экране в кругу своих родных и друзей. С 1892 года Рейно устраивает специальные показы в музее Гревен и дает им название "Оптический театр".

В 1892 году Рейнер создал свой первый мультфильм "Приключения мадам Баттерфляй" (Pauvre Pierrot), который длился всего 15 минут. В 1892 году Этьен-Жюль Маре (1859-1940), создал первый анимационный фильм, который назывался "Прыгающий конь".

Таким образом, Эмиль Рейно считается отцом анимационного кино. Его ленту "Вокруг кабины" ("Autor d'une cabine", 1894), в которой присутствует настоящий, с элементами юмора, сюжет (он разворачивается на морском пляже), французский историк кино Жорж Садуль назвал ключевым моментом в развитии анимационного кинематографа. В ленте было около 500 рисунков, рассчитанных на показ в течение 12 минут. Эмиль Рейно не только изобретатель техники, но прежде всего – создатель элементарных основ мультипликации. Еще до рождения кинематографа он нашел основной метод одушевления рисунков, получивший во Франции название "image par image" т.е. изображение за изображением, рисунок за рисунком, широко

разработанный сейчас в кинематографии искусства рисованного фильма. Интересно, что уже тогда Эмиль Рейно нашел способ отделения рисунков движения своих героев от декораций на фоне которых эти движения воспроизводились. Изготовление рисунков для его пантомим производилось им на прозрачных желатиновых, пластинках, каждая из них вверху и внизу имела по два отверстия, что позволяло Эмилю Рейно, как и в современной мультипликации, точно корректировать местоположение рисунков.

Другими известными художниками и изобретателями в области анимации были Эмиль Коль (1879-1937), создатель первых мультфильмов с персонажами, и Уинсор МакКэй (1867-1934), создатель первых полнометражных мультфильмов.

Первым полнометражным мультфильмом стал "Аврора" (1910) и "Эльдорадо" (1911), созданные американским художником и изобретателем Уинсором МакКэем (1867-1934). В этих фильмах использовалась техника рисованных кадров, каждый из которых создавался вручную художниками.

До момента первой публичной проекции фильмов братьями Люмьер (1895), да и после этого, появилось очень много подобных приборов. Среди их конструкторов были Рейландер, Угатиус, Гудвин, Ньепс, Дагерр, Скотт Арчер, Эдисон. К провозвестникам анимации можно с уверенностью отнести Жоржа Мельеса. Он запатентовал ряд кинематографических приборов, которые позволяли делать задержки, замедления, ускорения, накладывание изображений одно на другое. Тем временем – главным образом в американской и французской печатной периодике – быстро развиваются жанры сатирической иллюстрации и комикса. Именно они стали неиссякаемым источником – и тематическим, и пластическим – для анимации.

Изобретение братьев Люмьер, которые в 1895 г. разработали конструкцию киноаппарата для съемки и проекции движущихся фотографий, назвав его кинематографом, нанесло смертельный удар "Оптическому театру" Эмиля Рейно. Опытная демонстрация фильма, заснятого на киноплёнке, состоялась в марте 1895 г., а в конце декабря того же года в Париже уже начал функционировать первый кинотеатр.

Рождение кинематографа вытеснило мультипликацию. На некоторое время ее предали забвению, увлекшись перспективами быстро развивающейся игровой кинематографии. В 1910 году Рейно, подавленный постигшими его затем неудачами, утопил все оборудование и свои фильмы в Сене. В 1916 году он умер в приюте для бедняков.

В 1923 году Уолт Дисней (1901-1966) основал свою студию анимации. В 1928 году Уолт Дисней и его партнер УБ Иверкс создали персонажа Микки Мауса, который стал одним из самых популярных мультфильмов в истории

анимации. Дисней разработал технологию многоплановой камеры, которая позволяла создавать глубину в кадре.

В 1937 году Дисней выпустил свой первый полнометражный мультфильм "Белоснежка и семь гномов", который стал огромным успехом и открыл новую эпоху в истории анимации.

С появлением компьютеров и развитием компьютерной графики в середине 20 века началась новая эра в развитии анимации.

В 1961 году Иван Сазерленд (Ivan Sutherland) создал первый графический интерфейс для компьютера, что стало началом развития компьютерной графики и анимации.

В 1972 году Эд Кэтмулл (Ed Catmull) создал первую 3D-модель руки. В 1974 году он основал лабораторию компьютерной графики в Университете Юты, где разрабатывался первый в мире 3D-анимационный фильм "Армада" (1978).

Первый полнометражный двумерный мультфильм, созданный с использованием компьютерной графики, стал "Трансформеры: Фильм" (1986).

В 1982 году основана компания Pixar, которую возглавляли Эд Кэтмулл и Альвин Рэй Смит (Alvy Ray Smith). В 1986 году Pixar заключила сделку с компанией Disney на производство трех мультфильмов, которые должны были стать первыми полнометражными 3D-анимационными фильмами.

Первым фильмом был "Кошачий король" (The Cat King, 1987), но из-за технических проблем он не был закончен. Первым успешным 3D-мультфильмом стал "История игрушек" (Toy Story, 1995), который собрал огромную кассу и стал настоящим прорывом в мире анимации.

Стив Джобс является одним из ключевых фигур в развитии трехмерной анимации, особенно в связи с созданием студии Pixar. В 1986 году Джобс купил компанию, которая занималась разработкой графического процессора и графического редактора, и переориентировал ее на создание трехмерной анимации.

Джобс, вместе с Эдом Кэтмуллом и Альвином Рэем Смитом, руководил разработкой технологий, которые позволили создавать качественную 3D-анимацию. В частности, они разработали программное обеспечение RenderMan, которое использовалось для создания визуальных эффектов в таких фильмах, как "Звездные войны" (Star Wars) и "Терминатор 2" (Terminator 2).

Однако настоящий прорыв произошел после того, как Pixar заключила контракт с Disney на производство трех мультфильмов, которые должны были стать первыми полнометражными 3D-анимационными фильмами. Эти фильмы – "История игрушек" (Toy Story), "В поисках Немо" (Finding Nemo), "Тачки" (Cars) и многие другие – стали настоящими шедеврами трехмерной анимации и получили огромный успех у зрителей и критиков.

Стив Джообс играл важную роль в создании успеха Pixar и развитии трехмерной анимации в целом. Его вклад в разработку технологий и управление студией сделали возможным создание таких фильмов, которые изменили восприятие анимации и открыли новые горизонты в этой области.

С тех пор Pixar продолжает выпускать успешные 3D-анимационные фильмы, такие как "В поисках Немо" (Finding Nemo, 2003), "Живая сталь" (Wall-E, 2008), "Тачки" (Cars, 2006) и многие другие.

Кроме Pixar, в развитии трехмерной анимации принимали участие такие студии, как DreamWorks Animation, Blue Sky Studios, Illumination Entertainment и др. В 2001 году вышел первый фильм студии DreamWorks Animation – "Шрек" (Shrek), который тоже получил огромный успех.

Современная трехмерная анимация не ограничивается только полнометражными фильмами. Она используется в различных областях, таких как реклама, музыкальные клипы, игры и многое другое. Благодаря развитию технологий трехмерной анимации, ее возможности продолжают расширяться.

Виды анимации:

Рисованная классическая анимация – это вид анимации, который создается путем рисования каждого кадра отдельно и последующего их быстрого переключения, создавая иллюзию движения.

Пластилиновая анимация – это вид анимации, который создается путем использования пластилина для создания персонажей и объектов. Они затем фотографируются поочередно в разных позах, создавая иллюзию движения.

Кукольная анимация – это вид анимации, который создается путем использования кукол и их манипуляции для создания движения. Куклы фиксируются на специальных основаниях, которые позволяют им двигаться, а затем их фотографируют поочередно.

Переключочная анимация – это вид анимации, который создается путем использования вырезанных из бумаги или других материалов персонажей и объектов, которые затем переключаются поочередно и фотографируются.

Компьютерная анимация 2D – это вид анимации, который создается с помощью компьютерной графики и позволяет создавать двумерные изображения, которые затем могут быть анимированы.

Компьютерная анимация 3D – это вид анимации, который создается путем использования компьютерной графики для создания трехмерных моделей персонажей и объектов. Эти модели затем анимируются и используются для создания анимационных фильмов и игр.

Лекция 2. Создание 3D анимации

Цель. Знакомство с ПО для создания трехмерной анимации, технологиями анимирования, изучение способов создания анимации.

Существует множество программного обеспечения для создания трехмерной анимации, некоторые из них:

- Autodesk Maya
- Blender
- 3ds Max
- Cinema 4D
- Houdini
- ZBrush
- SketchUp
- Modo
- Lightwave 3D
- Poser

Каждая программа имеет свои особенности, преимущества и недостатки, и выбор зависит от конкретных потребностей и задач аниматора.

Pixar использует собственное программное обеспечение для создания анимации, которое называется Pixar Animation Studios Software, или просто Pixar Software. Это программное обеспечение включает в себя множество инструментов для создания и анимации 3D-моделей, управления светом, текстурирования, рендеринга и многого другого. Pixar Software был разработан специально для работы над анимационными фильмами Pixar, такими как "История игрушек", "В поисках Немо", "Хранители снов" и многие другие.

Существуют различные технологии анимирования, которые могут использоваться в создании анимации. Некоторые из них:

1. Процедурная анимация: это технология, которая использует математические алгоритмы и формулы для создания анимации. Эта технология позволяет создавать сложные движения и эффекты, которые трудно или невозможно воссоздать с помощью других технологий.

2. Программируемая анимация: это технология, которая использует программирование для создания анимации. Аниматоры могут написать код, который определяет движения и действия персонажей, объектов и других элементов анимации.

3. Анимация по ключевым кадрам: это технология, которая основывается на создании нескольких "ключевых" кадров, которые определяют движение и положение объектов в конкретные моменты времени. Программа затем автоматически заполняет промежуточные кадры, чтобы создать плавный переход между ключевыми кадрами.

4. Параметрическая анимация – это технология анимации, которая позволяет аниматорам изменять параметры объектов во время анимации, чтобы создавать различные эффекты и движения.

Эта технология основывается на использовании параметров объектов, которые могут быть изменены для изменения их формы, размера, цвета, положения и других свойств. Во время анимации, аниматор может изменять эти параметры на разных этапах анимации, что позволяет создавать плавные и реалистичные движения объектов.

Примером параметрической анимации может служить анимация рыбы, плавающей в воде. Аниматор может настроить параметры рыбы, чтобы изменить ее форму и размер хвоста, форму и размер головы, а также настроить параметры воды, чтобы создать движение волн и течения. Затем, используя эти параметры, аниматор может создать анимацию, в которой рыба движется плавно и реалистично в воде.

Параметрическая анимация может использоваться в различных областях, включая компьютерные игры, фильмы, рекламные ролики и дизайн. Эта технология позволяет аниматорам создавать сложные эффекты и движения, которые могут быть трудно или невозможно создать с помощью других технологий анимации.

5. Технология захвата движения, также известная как Мосар (Motion Capture) – это технология, которая позволяет захватывать движения человека или объектов и использовать их для создания анимации. Данная технология широко используется в киноиндустрии, играх, медицине, спорте и других сферах.

Процесс мосар начинается с надевания на человека специального костюма или наложения маркеров на его тело. Затем камеры, расположенные вокруг человека, снимают его движения в реальном времени. Камеры передают информацию о движениях в компьютер, где эти данные обрабатываются и преобразуются в анимационные данные.

Технология мосар позволяет создавать реалистичную анимацию, потому что движения, захваченные с помощью этой технологии, основаны на реальных движениях человека. Также мосар позволяет экономить время и средства на создание анимации, так как позволяет быстро записывать движения и создавать анимационные данные.

Технология мосар используется в киноиндустрии для создания реалистичных эффектов в фильмах, в компьютерных играх для создания реалистичной анимации персонажей, в медицине для исследования движений тела и реабилитации пациентов, в спорте для анализа движений спортсменов и улучшения их техники.

6. Динамическая симуляция – это технология создания анимации,

основанная на физических законах и принципах. Она позволяет создавать реалистичную анимацию объектов, которые могут двигаться и взаимодействовать друг с другом, подчиняясь физическим законам.

Процесс создания анимации при динамической симуляции начинается с создания 3D модели объекта в программном обеспечении для трехмерной анимации. Затем объект настраивается для динамической симуляции. Это может включать в себя установку свойств, таких как масса, трение, силы и т. д., которые определяют физические свойства объекта.

Далее создаются условия для движения объекта, такие как сила тяжести, сопротивление воздуха, силы трения, взаимодействие с другими объектами и т.д. Затем происходит запуск симуляции, где программа использует физические законы для расчета движения объекта на основе его физических свойств и условий окружающей среды.

После завершения симуляции, полученные данные используются для создания анимации. Это может включать в себя создание кадров, которые отображают движение объекта на каждый момент времени во время симуляции. Кадры затем склеиваются вместе, чтобы создать плавную анимацию движения объекта.

Динамическая симуляция позволяет создавать реалистичную анимацию объектов, которые могут взаимодействовать с другими объектами и окружающей средой, как если бы они существовали в реальном мире. Эта технология широко используется в киноиндустрии для создания спецэффектов, а также в игровой индустрии для создания интерактивных и реалистичных игровых миров.

Каждая технология имеет свои преимущества и недостатки, и выбор зависит от конкретных потребностей и задач аниматора. Например, процедурная анимация может быть полезна для создания сложных эффектов, тогда как анимация по ключевым кадрам может быть лучше подходит для создания плавных движений персонажей.

В различном ПО реализация этих технологий имеет свои особенности.

Ключ – маркер времени, при котором фиксируются положения объектов, значения или настройки

Риггинг – процесс построение скелета модели, чтобы, привязав к нему модель, можно было создавать анимацию

Динамикой называется имитация движения на основе действующих законов физики. Вместо создания ключей анимации вы присваиваете объектам физические характеристики, которые определяют их поведение. Для этого обычные объекты Maya преобразовываются в динамические тела, что приводит к появлению у них набора динамических атрибутов, влияющих на поведение тела при взаимодействии с другими объектами сцены.

Динамические тела приводятся в движение действием внешних сил, называемых полями (fields). В Maya существует целый набор полей, имитирующих самые разные силы – от силы ветра до силы тяжести. Подробную информацию о них вы получите чуть позже.

Динамические объекты в Maya делятся на следующие категории: тела, частицы, волосы и жидкости. Тела (bodies) создаются из геометрических поверхностей и используются в основном при моделировании столкновений. Частицы (particles) представляют собой визуализируемые точки пространства, обладающие динамическими свойствами. Они применяются для создания таких эффектов, как огонь или дым. Появившиеся в текущей версии Maya волосы (hair) представляют собой кривые, обладающие динамическими свойствами. Своим поведением они напоминают веревки. Модуль для работы с жидкостями (fluids) присутствует только в версии Maya Unlimited. Для имитации свойств жидкостей применяются объемные частицы, которые могут проявлять свойства поверхностей. Они позволяют моделировать такие природные явления, как волнистые облака или струйки дыма. Впрочем, рассмотрение динамики мягких тел, волос и жидкостей выходит за рамки данной книги.

MassFX для 3ds Max предоставляет набор инструментов для добавления в проект реалистичных физических симуляций.

В симуляциях MassFX используются твердые тела: объекты, которые не меняют форму во время симуляции. Каждое твердое тело может быть одного из трех типов:

Динамический: движение динамических объектов полностью контролируется симуляцией. Они подвержены гравитации, деформации пространства. Силы и силам, возникающим в результате ударов других объектов в симуляции, включая предметы из ткани.

Кинематика: кинематические объекты можно анимировать стандартными методами, но это не обязательно; они могут быть просто стационарными объектами. Кинематический объект может воздействовать на динамические объекты в моделировании, но не на них. Кинематический объект может переключиться в динамический статус в любой момент моделирования.

Статические: статические объекты похожи на кинематические объекты, но не могут быть анимированы. Однако они могут быть вогнутыми, в отличие от динамических и кинематических объектов. Они полезны в качестве контейнеров, стен, препятствий и так далее.

Лекция 3. Визуализация анимированной сцены

Цель. Изучение особенностей визуализации анимированной сцены.

Важно правильно выставить композицию кадра анимированной сцены. Композиция кадра играет очень важную роль. Это процесс совмещения всех элементов сцены в одну финальную секвенцию, которая затем может быть использована в качестве окончательного результата.

1. Определить главные элементы сцены: сначала необходимо определить, какие элементы будут являться основными и важными для сцены. Это может быть главный персонаж, фоновые объекты, свет и т.д.

2. Создать глубину поля: одна из важнейших задач в композиции кадра – создание эффекта глубины поля, который позволяет создать иллюзию трехмерности и глубины сцены.

3. Расположение элементов: после определения главных элементов и создания глубины поля можно начать располагать элементы сцены в соответствии с их важностью и ролью.

4. Использование цветовой гаммы: цвета могут быть использованы для того, чтобы привлечь внимание к определенным элементам сцены или чтобы создать определенное настроение.

5. Работа со светом: свет может использоваться для создания настроения и подчеркивания главных элементов сцены.

6. Добавление эффектов: эффекты могут быть добавлены для создания дополнительных эффектов и улучшения общей визуальной привлекательности сцены.

7. Анимация камер:

Анимация камеры является важной частью создания анимированных сцен, которая позволяет создавать эффекты движения, путешествия по сцене и создавать настроение. Ниже перечислены некоторые варианты анимирования камеры:

Перемещение по прямолинейной траектории: это один из самых простых способов анимирования камеры, при котором камера перемещается прямо в направлении цели. Этот способ хорошо подходит для создания кратких сцен, таких как вводных сцен.

Осмотр сцены: это способ анимации камеры, при котором камера перемещается вокруг центральной точки сцены, чтобы показать все элементы сцены. Это часто используется в архитектурной визуализации и создании трейлеров.

Слежение за движущимся объектом: в этом способе камера движется вместе с объектом, показывая его движение в пространстве. Этот способ хорошо подходит для создания сцен с автомобильными погонями и другими динамическими действиями.

Облет по криволинейной траектории: этот способ анимации камеры позволяет создать плавное движение камеры вдоль кривой. Этот способ подходит для создания красивых панорамных сцен.

Случайная анимация: этот способ анимации камеры создает случайные движения камеры, чтобы добавить непредсказуемости в сцену. Это может быть полезно, если вы хотите создать абстрактную сцену, в которой нет конкретного действия.

Каждый из этих способов анимации камеры может быть настроен с помощью задания параметров движения, скорости и ускорения, а также добавления эффектов размытия и фокусировки для создания более реалистичной анимации камеры.

Подготовка к визуализации анимированной сцены отличается от статической в нескольких аспектах:

Движение: анимация содержит движущиеся объекты, которые изменяют свою форму, положение и ориентацию во времени. Это требует более тщательной работы с кинематикой и анимацией, чтобы обеспечить правильное движение объектов.

Интерполяция: при создании анимации используется множество ключевых кадров, которые определяют состояние объектов в разные моменты времени. Чтобы создать плавные переходы между ключевыми кадрами, используется интерполяция. Это может потребовать больше времени на рендеринг и обработку данных.

Освещение и тени: при создании анимации нужно обеспечить правильное освещение и создать тени для движущихся объектов. Это может потребовать более сложных алгоритмов освещения и теней, которые занимают больше времени на рендеринг.

Временная линия: для работы с анимацией необходимо использовать временную линию, которая позволяет управлять процессом создания и редактирования анимации. Это может потребовать дополнительной работы с программным обеспечением.

Детализация: при создании анимации необходимо обеспечить достаточную детализацию объектов и сцены, чтобы они выглядели правдоподобно в движении. Это может потребовать большей вычислительной мощности и более длительного времени на рендеринг.

Рендеринг анимации имеет несколько отличий от рендеринга статического кадра: время рендеринга: время, необходимое для рендеринга анимации, может быть значительно больше, чем для статического кадра. Это связано с тем, что при рендеринге анимации требуется рассчитывать большое количество кадров и, зачастую, каждый кадр отдельно; постобработка: при создании анимации может потребоваться дополнительная постобработка, такая

как сборка кадров в видеофайл, корректировка освещения или добавление эффектов.

Рендеринг анимированной сцены включает в себя множество настроек, которые могут быть использованы для оптимизации времени рендеринга и повышения качества итоговой анимации. Рассмотрим этот процесс более подробно.

Настройки рендеринга секвенции

Прежде чем начать рендеринг, нужно определить настройки рендеринга секвенции. Это включает в себя разрешение кадра, соотношение сторон, частоту кадров и формат выходного файла. Кроме того, может быть выбран профиль цветового пространства и настроены другие параметры, включая настройки камеры и освещения.

Настройки семплирования

Семплирование – это процесс определения цвета каждого пикселя изображения. Чем больше семплов используется, тем выше качество изображения, но и время рендеринга будет увеличиваться. При рендеринге анимации, для оптимизации времени рендеринга можно использовать адаптивное семплирование, которое автоматически увеличивает количество семплов в сложных областях изображения.

Оптимизация времени рендеринга

Оптимизация времени рендеринга может быть достигнута путем использования различных техник, включая кэширование геометрии и освещения, использование экземпляров объектов, оптимизацию материалов, а также использование гибридных методов рендеринга, таких как рендеринг в реальном времени.

Использование пресетов

Многие программы для анимации предоставляют готовые пресеты для рендеринга, которые позволяют быстро настроить настройки для определенных типов сцен. Это может существенно ускорить процесс настройки рендеринга.

Эскизный и финальный рендер анимированной сцены

Перед финальным рендером рекомендуется выполнить эскизный рендер. Это позволяет быстро получить промежуточные результаты и проверить, что все настройки рендеринга настроены правильно. После этого можно запускать финальный на большое разрешение.

Рендер Arnold – это мощный и быстрый рендерер, используемый в киноиндустрии и визуальных эффектах. Разработан компанией Solid Angle, которую в 2016 году приобрела Autodesk.

Основными преимуществами Arnold являются его скорость, точность и простота использования. Он основан на алгоритмах Монте-Карло, которые позволяют получать высококачественные изображения с высоким уровнем

детализации и сложной освещенностью. Этот рендерер также поддерживает многопоточность и распределенный рендеринг на нескольких компьютерах.

Arnold интегрируется со многими 3D-пакетами, включая Autodesk Maya, Cinema 4D, Houdini и другие. Он поддерживает широкий спектр функций, включая глобальное освещение, теневые карты, более 60 видов шейдеров, процедурную генерацию текстур, объемный рендеринг и многое другое.

Кроме того, Arnold имеет широкий выбор настроек рендеринга, которые позволяют пользователю оптимизировать процесс и получить наилучшие результаты. Это включает в себя настройки семплирования, параметры света, настройки камеры и многое другое.

Arnold также имеет свою систему постобработки, которая позволяет пользователю настраивать изображения после рендеринга. Это включает в себя коррекцию цвета, настройки контрастности, глубины резкости и многое другое.

В целом, Arnold является одним из лидеров в области трехмерного рендеринга и широко используется в киноиндустрии и других областях, требующих высококачественной визуализации.

3. ПРАКТИЧЕСКИЙ РАЗДЕЛ

3.1 Описание лабораторных работ

Тема 2. Создание 3D анимации

Лабораторная работа 1

Анимация по ключевым кадрам

Цель работы: освоить способ анимации по ключевым кадрам.

Методические рекомендации

1. Откройте программу трехмерного моделирования и возьмите самолетик (либо создайте любой примитив по вашему выбору). Активизируйте окно проекции Perspective (Перспектива). Убедитесь в том, что текущим является кадр № 0. Включите режим анимации, щелкнув на кнопке Auto (Анимация). Кнопка и рамка окна Perspective (Перспектива) окрасятся в красный цвет.

2. Установите в качестве текущего кадр № 100, перетаскив ползунок вправо до появления на нем надписи 100/100. С этой же целью можно щелкнуть на кнопке Go to End. Переместите самолетик. Снова щелкните на кнопке Auto, чтобы выключить режим анимации. Анимация из 100 кадров готова. Обратите внимание на появление двух ключей анимации в начале и конце строки треков.

3. Воспроизведите анимацию и просмотрите ее в окне проекции. Можете попробовать режим покадрового воспроизведения, управляя им кнопками Next Frame (Следующий кадр) и Previous Frame.

4. Примените к самолету какой-либо материал и установите фон сцены в виде фотографии облачного неба. Меню Views – Viewports Background – Configure Viewport Background. В окне Viewports Configuration выберите файл для фона. Сохраните созданную сцену. Самостоятельно дополните созданную анимацию ключами изменения других параметров – масштаба самолета, силы блеска и цвета его материала. Для этого достаточно всего лишь включить режим анимации, выбрать нужный ключевой кадр, внести изменения в параметры материала и выключить режим анимации.

5. Создание рендеринга анимации. Меню Rendering> Render Setup. Renderer – Scanline Renderer, переключится в положение Active Time Segment, выбрать размер кадра Output Size. Покрутить вниз, в группе Render Output нажать Files, задать путь сохранения, имя файла, тип (avi), Save, выбрать тип кодека, их состав зависит от установленных в вашей системе. В появившемся окне диалога выберите в раскрывающемся списке Compressor, например, вариант Microsoft Video1, оставьте ползунок параметра Compression Quality по умолчанию. Выполнить команду Render. Фона нет! Чтобы он считался, откройте Rendering – Environment, кликнете по кнопке Environment Map (слово

None). В окне Material/Map Browser вкладка General – Bitmap. Выберите файл. Если при рендере идет сферическое проецирование фона, открываем окно редактора материалов, тянем карту из окна Environment and Effects на слот с шариком образцом и в свойствах выбираем Mapping – Screen.

6. Визуализация анимаций. Для этого щелкните на кнопке Rendering – Render главной панели. Запустите процесс визуализации и наблюдайте синтез одного кадра за другим. По завершении визуализации перейдите в папку, в которой сохранялся файл анимации Летяга.avi и воспроизведите его.

7. Чтобы видеть пути анимации объекта, выделите его и на вкладке Display панели управления (она справа) поставьте галочку Motion Path в группе Display Properties (или правой кнопкой по объекту, выбираем Object Properties и ставим галочку Motion Path). Если требуется изменить путь, либо двигаем объект на ключевом кадре либо переходим в панели управления на вкладку Motion, нажимаем Motion Path, Sub-object, Key – двигаем и меняем тип ключевых точек на траектории анимации в сцене.

Задание. Придумайте и анимируйте объект, в котором будет ускорение, замедление и равномерное движение.

Сохраните сцену в своей папке под именем movement.

Лабораторная работа 2

Таймлайн. Интерполяция ключевых кадров

Цель работы: изучить основные приемы редактирования ключевой анимации.

Методические рекомендации

Менеджер Тамлайн. Кривые анимации. Треки анимации. Работа с ключевыми кадрами на тамлайне в режиме F-кривых. Интерполяция ключевых кадров. Работа с касательным ключевых кадров. Инструмент Участок. Создание анимации в режиме сплайновой интерполяции, ступенчатой и линейной интерполяции.

Для наиболее гибкого редактирования ключевой анимации предназначен менеджер Timeline. Работа с Timeline ведется в основном с анимационными кривыми и ключами.

Таймлайн представляет собой мощный инструмент, предназначенный для контроля, управления и воспроизведения анимации. Временная характеристика всех анимированных элементов сцены, предоставлена здесь в горизонтальном направлении. Основным для анимации являются так называемые ключи, которые в определённый момент времени имеют определённое значение элемента (например, позиция объекта). Анимация, как правило, для своего функционирования должна иметь два ключа. Значения элементов между этими

ключами будут, в данном случае интерполированы. Воспроизведение анимации приведёт к изменению значений - именно так и работает таймлайн!

Timeline представлен тремя окнами (режимами): экспозиционный лист, режим F-кривых, режим движения.

Экспозиционный лист используется в качестве расширенной панели анимации для работы с таймингом – распределением времени между ключевыми кадрами. В нем, как правило, работают на этапе блокинга.

Режим F-кривых предназначен для более тонкой работы - со спейсингом – работой с кривыми анимации и интерполяцией ключевых кадров. В нем, как правило, работают на этапе полировки.

Режим движения используется для работы с системой движения. Здесь возможно производить обработку источников движения, создание клипов движения на слоях и их смешивание, а также производить взаимное смешивание слоёв движения и анимации. (в данном курсе не рассматривается).

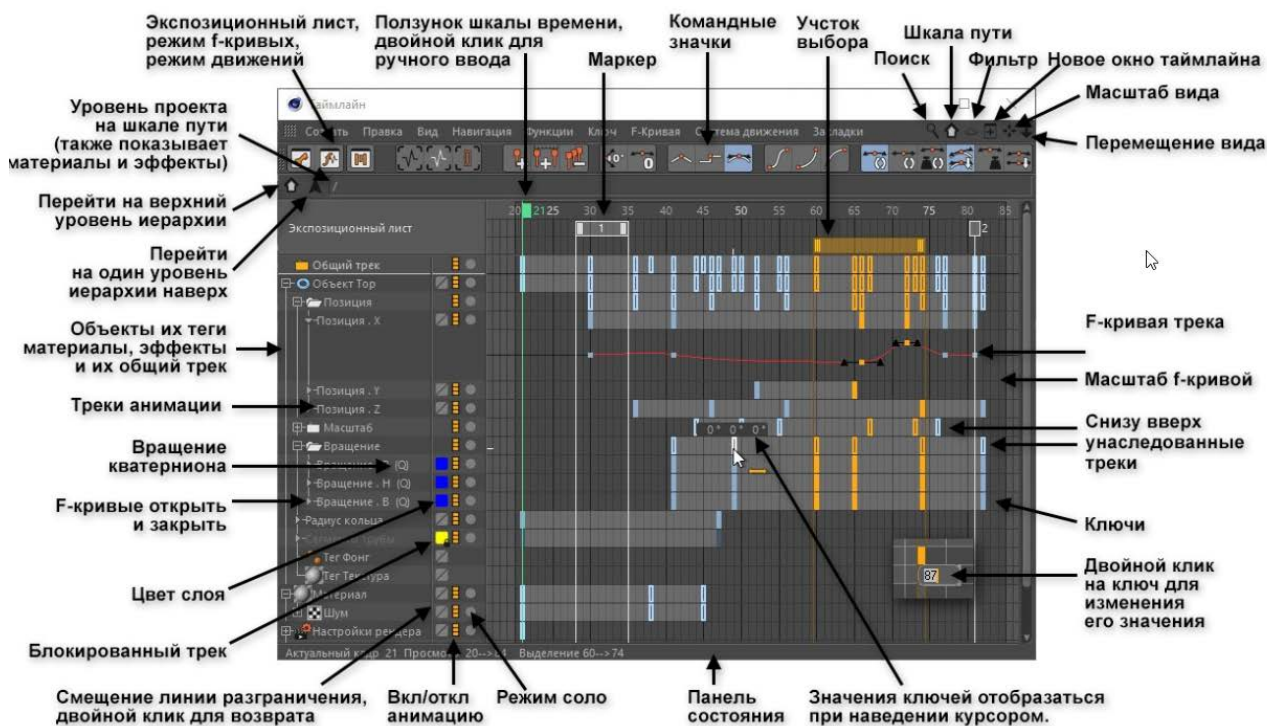


Рисунок 1

Экспозиционный лист

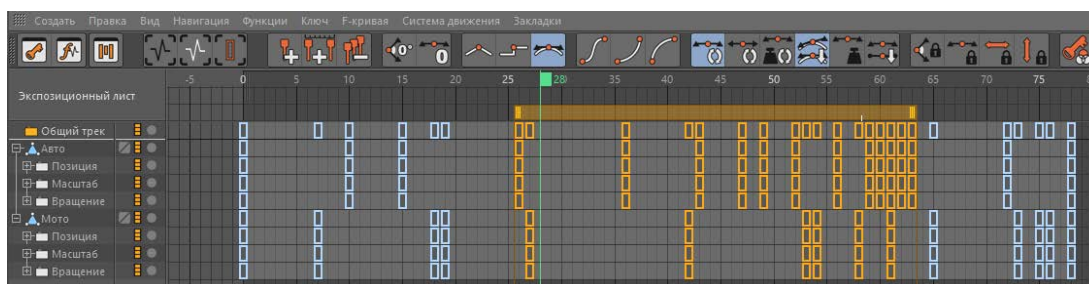


Рисунок 2

Режим F-кривых

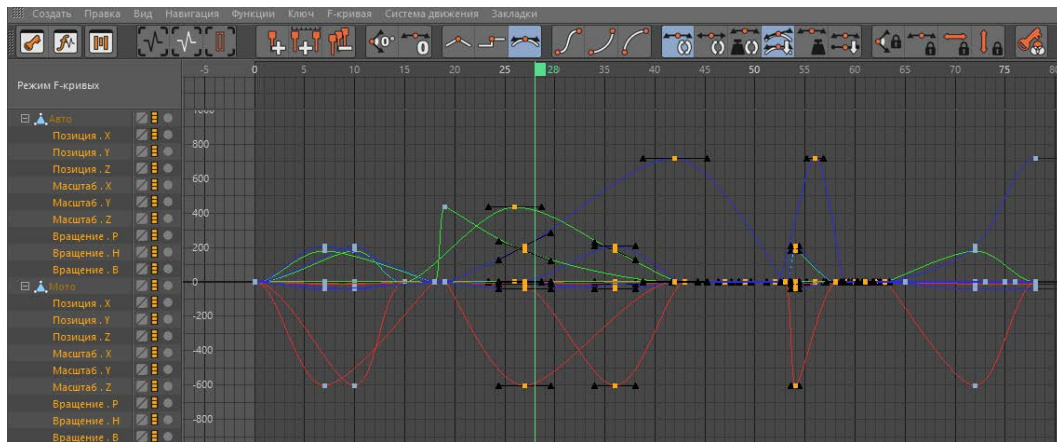


Рисунок 3

Таймлайн состоит из следующих основных областей:

1. Строка меню
2. Панель инструментов
3. Участок объектов
4. Панель слоев
5. Шкала времени
6. Участок ключей

В левой части находится участок объектов, раскрыв который можно увидеть содержащиеся в нем треки анимации. Трек анимации – анимированный параметр объекта. Трек соответствует определенному параметру, который будет изменяться по времени. Треки представляют собой характеристики анимации и могут отвечать за изменение положения, ориентации, масштаба, геометрических параметров, свойств материала, видимости, освещения, при этом каждому параметру соответствует собственный трек.

Слева от названия трека находится панель слоев, определяющие принадлежность трека к определенному слою, а также переключатели режимов, позволяющие отключить анимацию или включить режим Соло.

Трек создается автоматически при создании первого ключа анимации для какого-либо параметра объекта, отображенного в списке анимируемых объектов. При анимации изменения в окне проекции обычно создается девять треков, по три на перемещение, масштабирование и вращение для каждой оси. Чтобы не создавать лишних треков, вы можете выбрать параметры, для которых будут создаваться треки.

Каждый трек имеет ранее упомянутые ключи, которые вы можете создавать в любых количествах. Кроме этого, у каждого трека есть своя f-кривая (если речь при этом идёт о свойстве, для которого может быть

использована интерполяция), которая определяет интерполяцию между ключами. Эта мини-f-кривая расположена под соответствующим ключом трека, и может быть здесь (ограниченно) обработана. Значительно комфортабельнее это выглядит в режиме правки f-кривых, при котором f-кривые будут отображены в полном объёме.

Навигация по таймлайн

Рабочая область ведет себя как обычная ортогональная камера. Вы можете делать наезды отъезды, перемещения с помощью привычных манипуляций с мышью. В таймлайне можно ориентироваться, помимо использования выше описанных значков, следующими кнопками:

Переместить: нажать на клавишу 1 и переместить курсор мыши.

Масштабировать: нажатие на клавишу 2 позволяет производить масштабирование.

Участок объектов

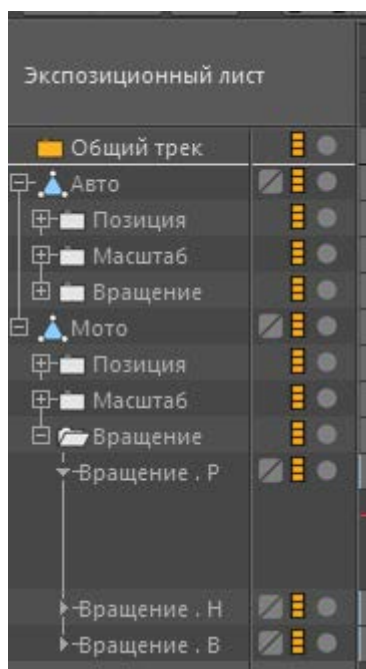


Рисунок 4

На участке объекта вы найдёте все объекты, теги, материалы, шейдеры, пост эффекты, а также соответствующие им треки анимации. Почти все элементы этого участка (за исключением названий папок Объекты, Позиция), вы можете изменять, при совершении двойного клика мыши на соответствующем элементе, или выделении элемента и нажатии клавиши F2.

На основе функции слоёв и использования фильтров, вы можете определять показ элементов для этого участка. Элементы расположены в строгой иерархии, для улучшения визуального просмотра. Все анимированные свойства, принадлежащие для определенного объекта, расположены строго под этим объектом. К этому также принадлежат теги, назначенные для этого объекта. При клике мышью на значке + или - вы можете свернуть \ развернуть

необходимую иерархию объекта. Если вы при этом дополнительно нажмете на клавишу CTRL, иерархия будет полностью развёрнута.

Шкала времени



Рисунок 5

Маркеры могут создаваться посредством клика мыши с нажатой клавишей CTRL. Вы можете в любое время перемещать, созданные таким образом маркеры.

Участок просмотра может быть использован в других частях программы. Например, при расчёте сцены вы можете произвести вывод показа только для этого участка, или анимация должна быть воспроизведённой только в пределах этого участка.

Для установки ползунка на определенное время, кликните просто курсором мыши в нужном месте.

Участок ключей

На участке ключей будут показаны все имеющиеся ключи в хронологической последовательности. Дополнительно вы можете развернуть иерархию F-кривой, посредством клика курсора мыши на маленький треугольник.

Посредством двойного клика на настоящем ключе вы можете изменять значение для этого ключа, непосредственно на шкале таймлайна, без дополнительного использования для этого менеджера атрибутов. При использовании клавиши CTRL + клик мыши на любом треке, вы создаёте к этому промежутку времени дополнительный ключ.

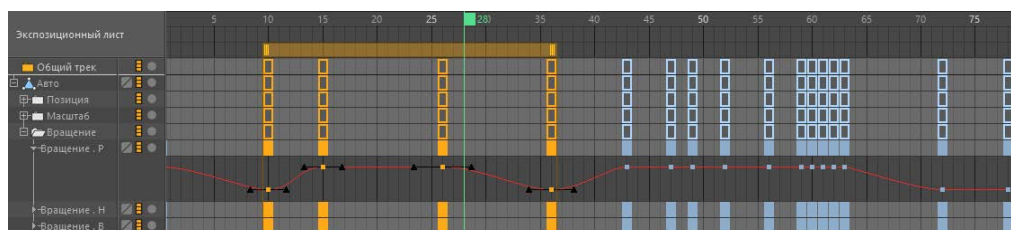


Рисунок 6

Вы можете производить выделение ключей следующим образом: Отдельных ключей посредством клика на них.

Посредством курсора мыши создайте прямоугольное выделение. Для этого просто кликните на пустом участке шкалы таймлайна, и перетащите курсор с нажатой кнопкой мыши. С одновременно нажатой CTRL клавишей (после нажатия кнопки мыши) вы можете производить удаление ключей из имеющегося выделения.

Вы перемещаете общее число выделенных ключей, если:

- перемещаете один из выделенных ключей
- перемещаете участок выделения на линейке таймлайна.
- посредством клавиш перемещения вправо / влево, производите перемещение на 1 кадр в одном из направлений.

Вы масштабируете (лупа) выделенные ключи по времени, если гриф участка выделения перемещаете на линейке шкалы таймлайна.

Если для изложенных выше процессов вы нажмёте дополнительно на клавишу CTRL, выделенные ключи будут дополнительно копированы. Используйте комбинацию клавиш SHIFT+CTRL, если вы намерены произвести копирование ключей по вертикали, или с одного трека на другой.

Режим F-кривой

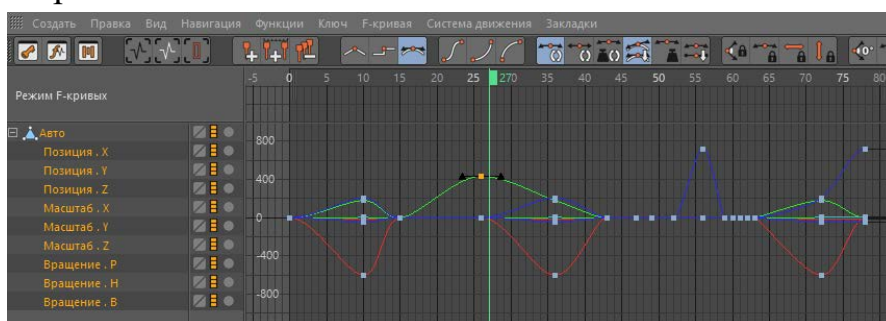


Рисунок 7

Участок ключей в режиме F-кривой представлен анимационными кривыми (в виде графика). Анимационная или F-кривая показывает изменение значений параметров объектов во времени. Время представлено горизонтальной шкалой, значения величин – вертикальной. Редактируя форму этих анимационных кривых, можно управлять анимацией, например, ускорять или замедлять движение объекта: крутой наклон кривой приводит к ускорению движения, а пологий — наоборот, к его замедлению.

Работа на Таймлайне в режиме F-кривой заключается в работе с: F-кривыми; редактировании ключей.



Рисунок 8

Редактировании тангенсов.

Задача F-кривой - регулировка интерполяции между ключами.

Интерполяция — процесс заполнения неизвестных данных между двумя известными значениями, то есть генерацию новых значений между двумя ключевыми кадрами.

Интерполяция – это процесс вычисления программой значений кадров, расположенных между ключевыми кадрами.

F-кривые определяют временную интерполяцию между ключами. С многочисленными функциями, которые предлагает при этом шкала таймлайна, вы можете очень просто производить реализацию процессов ускорения и торможения, для создаваемой анимации.

При создании ключевых кадров программа устанавливает по умолчанию между ними мягкую интерполяцию. Однако вы можете задавать, как будут интерполироваться значения во времени. Другими словами, скорость, с которой изменяются значения от одного ключевого кадра к другому, может оставаться постоянной, возрастая или уменьшаться. Можно даже указать отсутствие интерполяции, так что значение ключевого кадра удерживается до тех пор, пока не будет достигнут следующий ключевой кадр.

Для выбора типа интерполяции ключевых кадров используются кнопки на панели инструментов.



Рисунок 9

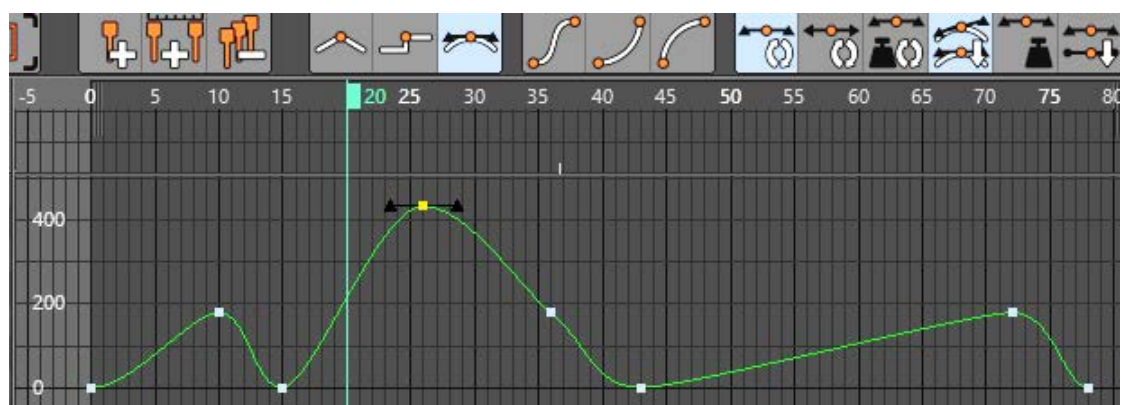


Рисунок 10

Плавная (сплайновая) интерполяция.

Наиболее часто используемый тип, если вам необходимо создать реалистичную анимацию для ускорения или торможения процессов. Только для этого типа существует касательные ключей, на основе которых вы

индивидуально и для каждого ключевого кадра, можете определить форму кривой.

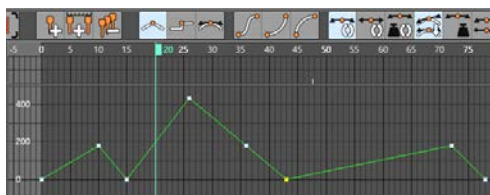


Рисунок 11

Линейная интерполяция

Изменение значений в этом режиме, происходит с постоянной скоростью.

Ступенчатая интерполяция

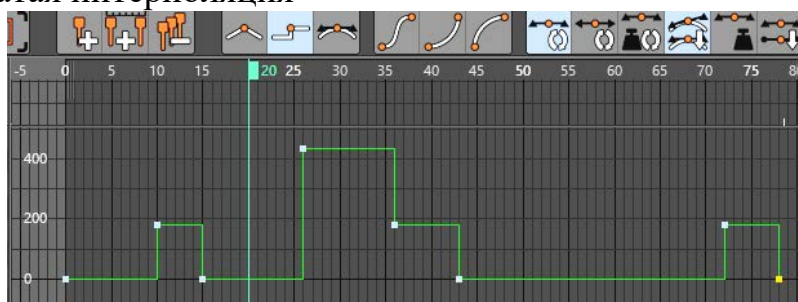


Рисунок 12

Интерполяция при этом отсутствует: значение после ключа остаётся постоянным, и при достижении им нового ключа, изменяется интенсивно до нового значения ключа.

Такая анимация может использоваться для процессов включения / отключения, где необходимо моментальное изменение значения для ключа. Например, интенсивность источника света может быть моментально изменённой с 0% до 100% яркости. Или движение секундной стрелки в часах.

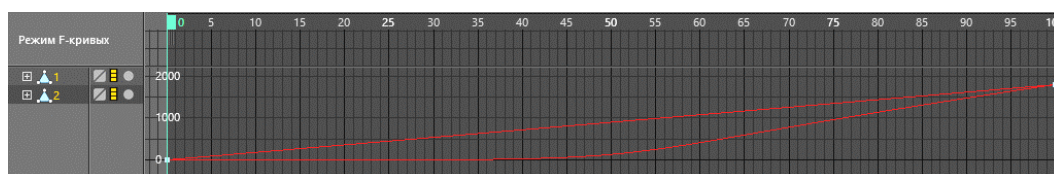


Рисунок 13

Важно понимать, что метод интерполяции влияет не на время между ключевыми кадрами, а только на скорость изменения значений между кадрами. Например, представьте себе, что вы анимируете два объекта так, чтобы они оба перемещались на одинаковое горизонтальное расстояние за сто кадров. Для обоих анимация начинается в одно и то же время, но для одного установлена линейная интерполяция, а для другого – сплайновая. Когда вы воспроизводите анимацию, первый объект сразу же начинает равномерное движение к следующему ключевому кадру, в то время как второй набирает скорость. Хотя второй отстает от первого, он постепенно догоняет его, и оба достигают конечного положения одновременно.

Для назначения определенного типа интерполяции необходимо выделить ключ и выбрать соответствующий тип. Если используется сплайновая интерполяция – то есть возможность изменять форму анимационной кривой, используя тангенсы.

Тангенсы – это отрезки касательных к анимационной кривой в той точке, где находится ключ. Когда вы выбираете ключ и меняете тип интерполяции, этот тип меняется для обоих тангенсов, находящихся справа и слева от ключевой точки. Интерполяцию можно менять индивидуально для каждого тангенса, удерживая нажатой клавишу Shift.

Тангенсы полностью определяют поведение кривой на участке между соседними ключами.

Кроме этого есть ряд функций, позволяющих быстро изменить тип интерполяции ключевого кадра (горизонтальные касательные).



Рисунок 14

Плавные интерполяции уместны в тех случаях, где происходит движение массивных объектов. Так как в реальности не существует бесконечного ускорения, движения при этом начинаются или заканчиваются медленно. Горизонтальные касательные позволяют создавать такую имитацию.

Плавный ввод. Правая касательная при этом остаётся без изменения, в то время как левая касательная, будет повёрнута по горизонтали.

Плавно - плавно: касательные будут расположены горизонтально.

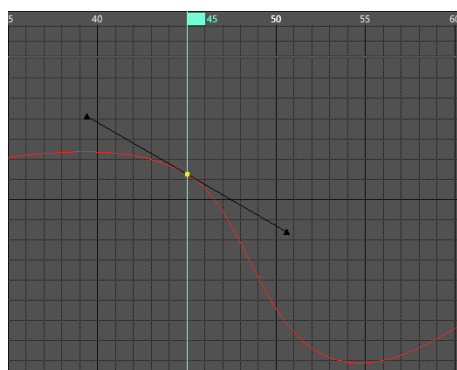


Рисунок 15

Плавный вывод: левая касательная остаётся без изменений, в то время как правая будет развернута по горизонтали.

Инструмент участка является свободным выбором, который может быть изменённым в любое время. Вы можете с его помощью производить обработку групп ключей в режимах правки F-кривой и ключей. Например, большие группы ключей вы можете копировать, и после перемещения участка, абсолютно точно добавлять их для другого участка.

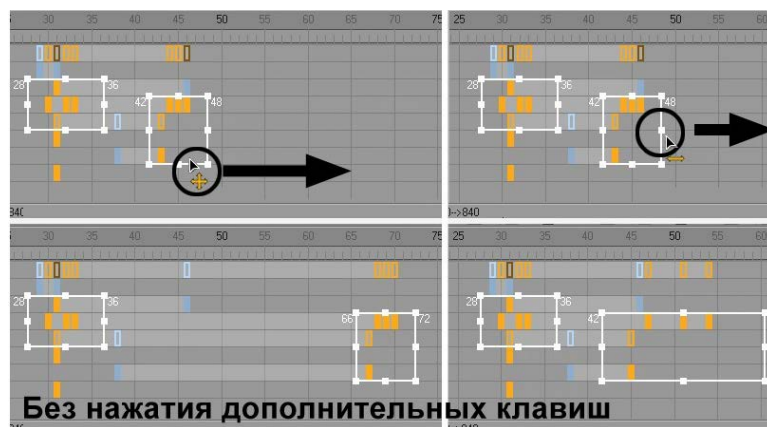


Рисунок 15

Рамка имеет со всех сторон специальные грифы, которые позволяют вам производить масштабирование участка по времени. Справа и слева будут показаны номера для кадров края рамки.

Каждый ключ в пределах показываемой рамки, является выделенным.

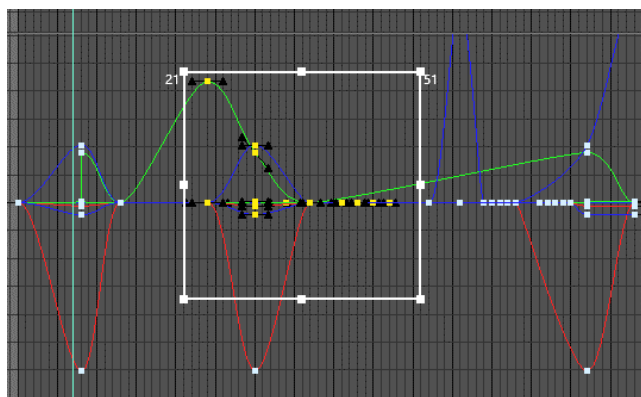


Рисунок 16

Установите активным инструмент и на шкале Таймлайна, создайте на его основе рамку по периметру выделяемых ключей. Вы можете при этом создавать любое количество рамок, которые могут взаимно пересекаться.

При использовании этого инструмента, действительными являются следующие клавиатурные сокращения и условия:

Клик мыши и её перемещение в пределах определенного участка, приводит к перемещению только этого участка находящегося под курсором мыши, с учётом имеющихся ключей.

- клик мыши и перемещение + Alt в пределах одного участка, приводит к перемещению всех участков, с учётом имеющихся ключей.

- клик мыши и перемещение + Shift в пределах одного участка, приводит к перемещению участка находящегося под курсором мыши без учёта ключей.

- клик мыши и перемещение + Shift + Alt в пределах одного участка, приводит к перемещению всех участков без учёта ключей.

- клик мыши и перемещение + Shift за пределами участка: только созданные участки будут созданы дополнительно к уже имеющимся участкам.

- клик мыши и перемещение + Ctrl за пределами одного участка: удаление участков или их частей.

Задание. Измените характер движения объекта с лабораторной работы 1 с помощью редактора кривых.

Лабораторная работа 3

Параметрическая анимация. Контроллеры анимации

Цель работы: познакомиться с контроллерами анимации, освоить их назначение и настройку.

Методические рекомендации

Что такое контроллеры

Контроллеры можно классифицировать несколькими способами: по одному параметру против нескольких, по тому, является ли контроллер параметрическим или ключ-базированным, по типу данных контроллера и по типу интерполяции контроллера.

Контроллеры с одним и с несколькими параметрами

Контроллеры с единственным параметром размещены на самом нижнем уровне иерархии контроллеров. Данные контроллеры хранят значения анимации, указанные пользователем для параметра объекта и для выводных значений во времени. Возвращаемые значения могут иметь как единственный компонент (такой как высота коробки) или несколько компонентов (подобные позициям X, Y и Z объекта). Контроллер с единственным параметром может быть как параметрическим, так и ключ-базированным.

Сложные контроллеры в качестве своего ввода берут вывод других контроллеров. Затем они объединяют полученные данные с любыми данными параметров, ассоциированными со сложным контроллером, манипулируют данными и выводят результаты. Сложными являются контроллеры трансформации Position/Rotation/Scale (PRS) и LookAt, контроллер вращения Эйлера XYZ, контроллер позиции Path и контроллер List.

Параметрические контроллеры против ключ-базированных

Контроллеры с одним параметром можно классифицировать по признаку, является ли контроллер параметрическим или ключ-базированным. Параметрический контроллер принимает в качестве ввода значения данных, указанные пользователем, и затем выводит значения на основании уравнения, реализуемого контроллером, и на значениях вводных данных. Ключ-базированный контроллер воспринимает в качестве ввода значения данных, указанные пользователем в определенных временных точках и затем обеспечивает в качестве вывода интерполированные во времени значения для любой точки.

Примером параметрического контроллера является Noise Rotation. Ввод для данного контроллера указывается в диалоге свойств контроллера и включает частоту и силу помех. Эти данные задаются однажды и не изменяются на протяжении всей анимации. С параметрическим контроллером не связываются ключи, и присутствие контроллера обозначается линией диапазона на дорожке параметра в Track View. Вывод контроллера в заданное время основывается на вводных данных, времени и уравнении, реализуемом функцией помех.

Примером ключ-базированного контроллера является Tension/Continuity/Bias (TCB) Rotation. Вводом в данный контроллер является вращение объекта в заданных временных точках. Эти данные обычно обеспечиваются установкой кадра анимации, включением кнопки Animation и вращением объекта. Каждый раз, когда объект вращается в различных кадрах, генерируются новые данные. Эти данные носят название ключей, а данные, указывающие на величину поворота, называются значениями ключей. Наличие ключа отображается точкой на дорожке параметра в Track View.

Вывод ключ-базированного контроллера основывается на значениях ключей, времени и уравнении, использованном для интерполяции между ключами. Для некоторых контроллеров уравнение, применяемое для интерполяции между ключами, может принимать дополнительный пользовательский ввод. Например, в случае контроллера TCB пользователь может настроить напряжение, непрерывность и смещение каждого ключа. Другие контроллеры, подобные Linear, всегда интерполируют одинаково и не настраиваются.

Типы данных контроллера

Контроллеры можно также классифицировать по типу возвращаемых данных. Типы данных контроллера должны соответствовать типу данных параметра объекта для того, чтобы контроллер мог применяться с данным параметром. Например, контроллер типа данных Scale не может использоваться для параметра позиции объекта ввиду различий в типах данных.

Ключ-базированные контроллеры

Ключ-базированные контроллеры используются для определения значений между ключами. Для всех контроллеров кривая функции всегда проходит через ключевые значения в момент времени, связанный с ключом. Различные контроллеры оказывают влияние только на интерполяцию между ключами, но не на сами ключи.

Типы интерполяции: Linear, Smooth, Bezier, TCB.

Каждая комбинация типов интерполяции и типов данных реализуется посредством уникального контроллера. Метод для варьирования значений

ключей и значений интерполяционных параметров для ключей является одним и тем же внутри группы контроллеров заданного типа интерполяции.

Контроллеры, использующие тип интерполяции Linear, разделяют изменение в значениях ключей между данным ключом и следующим на количество времени между ключами. Значения, возвращаемые из контроллера, следуют прямой линии между ключами и равномерно распределяются во времени, т.е. значения между ключами изменяются с постоянной скоростью. Никакой настройки интерполяции значений выполнить нельзя.

Контроллеры, использующие тип интерполяции Smooth, настраивают касательную кривой, проходящей через значение ключа, и обеспечивают гладкую интерполяцию в ключе. Никакой настройки интерполяции значений выполнить нельзя.

Контроллеры, использующие тип интерполяции Безье, применяют настраиваемый сплайн Безье, проходящий по ключам, и вычисляют значения между ключами. Форма кривой сплайна основывается на значениях ключа и тангенса в ключах.

Контроллеры, использующие тип интерполяции TCB, интерполируют между ключами на основе пяти параметров: Tension, Continuity, Bias, Ease to и Ease from. Форма кривой функции базируется на ключе и значениях параметров в ключе.

Для всех контроллеров можно установить или настроить значения ключа, включая кнопку Animate, устанавливая соответствующее время и затем устанавливая новое значение для варьируемого параметра. Для всех контроллеров, кроме контроллеров вращения, также можно создавать ключи или настраивать значения, связанные с ключом. Для этого применяются инструменты Function Curve в Track View. Как описывалось ранее, ключи вращения нельзя отобразить как Function Curve, следовательно их нельзя настроить описанным способом.

Параметрические контроллеры

Самые распространенные типы параметрических контроллеров – Noise и Expression.

Каждая комбинация параметрического типа и типа данных реализуется уникальным контроллером. Внутри группы контроллеров заданного параметрического типа способ, по которому задаются параметры контроллера, является одним и тем же.

Контроллеры Noise

Параметры для контроллера помех указываются в диалоге Properties контроллера. Для всех типов данных диалог Properties доступен в Track View через выбор параметра, которому назначен контроллер, щелчке на свободной области окна редактирования, чтобы убедиться, что никаких ключей не

выделено, затем щелчке правой кнопкой мыши на названии параметра и щелчке правой кнопкой на линейке диапазона в дорожке параметра или щелчке на Properties в линейке инструментов Track View.

Для контроллеров, связанных с трансформацией на объектном уровне, параметры контроллера можно также настроить из панели команд Motion, выбрав параметр в свитке Assign Controller, щелкнув правой кнопкой на параметре и выполнив из всплывающего меню команду Properties.

Characteristic Graph в диалоге Noise Properties грубо отражает эффект, который окажут на вывод контроллера изменения в параметрах помех.

Поля Strength указывают диапазон выводных значений контроллера помех. Диапазон меняется от $-Strength/2$ до $Strength/2$, если опция >0 отключена, или от 0 до Strength в противном случае. Здесь существует два исключения. Прежде всего для типа данных масштаба значение 100 автоматически прибавляется к каждому из выводных значений помех. Это означает, что вы применяете помехи к 100% фактору масштаба. Во-вторых, если включены фрактальные помехи, выводной диапазон увеличивается, а центральная точка не изменяется. Таким образом могут быть значения меньше нуля даже тогда, когда включена опция >0 . Для Roughness 0.0 диапазон увеличивается приблизительно на 10%, для Roughness 1.0 на 100%.

Поля Ramp In и Ramp Out уменьшают количество помех в начале и в конце диапазона. Данное изменение нелинейно, однако эквивалентно кривой Безье. Для Ramp In кривая определяется вершиной Безье, находящейся в нулевой отметке времени и во времени, указанном в поле Ramp In, с вершиной, имеющей нулевую скорость интерполяции.

Контроллеры выражений (Expression controllers) уникальны в том, что они оценивают определенные пользователем выражения и таким образом определяют свои выводные значения.

Составные контроллеры уровня трансформации (контроллеры Position/Rotation/Scale и LookAt), контроллер, который объединяет вращение вокруг отдельных осей (контроллер вращения Euler XYZ), контроллер для перемещения объекта вдоль сплайна (контроллер Path position) и контроллер, который производит сложение результатов своих вводных контроллеров (контроллер List).

Контроллер Position/Rotation/Scale

Контроллер трансформации Position/Rotation/Scale (PRS) объединяет ввод из контроллеров положения, вращения и масштаба. Контроллер PRS можно применять только на дорожках трансформации объектов и гизмо модификаторов. Никаких настраиваемых пользователем свойств с контроллером PRS не связано.

Контроллер Path

Контроллер Path позиционирует объект так, чтобы точка вращения объекта располагалась на сплайне. Кроме того, создается параметр, подчиняющийся контроллеру Path. Параметр называется Percent. Параметр Percent задает положение на сплайне для использования в конкретный момент времени. Значение Percent автоматически устанавливается в 0 в начале активного отрезка времени и в 100 в конце активного отрезка времени. Сплайн для использования в качестве пути указывается как параметр контроллера Path.

Задание 1. Анимлируйте объект, в котором будет движение с возмущениями (шумом).

Заставим самолет из лабораторной работы 1 в процессе перемещения дрожать и метаться из стороны в сторону, используя контроллеры Noise и List, а затем обеспечим полет по траектории, заданной в виде сплайна, с помощью контроллера Path.

1. Откройте файл `samolet`. Раскройте окно диалога Track View – Curve Editor (Просмотр треков) (Graph Editors) и щелкните в окне дерева иерархии на имени параметра Position (Положение), чтобы выделить его. Щелкните правой кнопкой и выберите `assign Controller` (Назначить контроллер). Появится окно диалога Assign Position Controller. Выделите в списке этого окна строку Position List и щелкните на кнопке ОК.

3. В окне дерева иерархии выделите строку с именем Available, которая представляет собой местозаполнитель для вставки любого допустимого контроллера в список Position List. Снова щелкните на кнопке Assign Controller и выберите в появившемся окне диалога Assign Position Controller строку Noise Position. Трек Available сдвинется на одну позицию вниз, а на его месте появится трек контроллера Noise Position.

4. Обратите внимание на вид траектории в окне Perspective. Из прямолинейной она превратилась в ломанную, отклоняющуюся от прямой во все возможные направления.

5. Настройте параметры контроллера в окне диалога Noise Controller. Используйте для настройки следующие элементы управления окна диалога:

- Seed (Номер выборки) – задает начальное значение для запуска генератора случайных чисел;

- Frequency (Частота) – позволяет управлять характерной частотой случайных возмущений;

- X, Y, Z Strength (Размах по X, Y, Z) – позволяют управлять масштабом возмущений по всем трем осям движения объекта. Справа от счетчиков находятся флажки «О». Если такой флажок сброшен, то выходные значения контроллера по соответствующей оси будут как положительными, так и отрицательными. Если флажок установлен, то шумовые приращения параметра будут принимать значения, лежащие в диапазоне от 0 до величины, заданной в

соответствующем счетчике размаха;

– Fractal Noise – флажок, включающий режим генерации случайных возмущений по фрактальному алгоритму. Если этот флажок установлен, то счетчик Roughness (Шероховатость) позволяет изменять амплитуду фрактальных возмущений функциональной кривой;

– Ramp In (Подъем), Ramp Out (Спад) – задают величины интервалов времени в начале и в конце траектории, в пределах которых случайная функция постепенно нарастает до полного размаха и спадает до нуля.

6. Воспроизведите анимацию в окне проекции Perspective и наблюдайте за ее метаниями из стороны в сторону в процессе перемещения. Постройте и просмотрите эскиз анимации.

7. Теперь попробуйте заставить объект следовать по заданному пути. Активизируйте окно проекции Front (Вид спереди) и создайте сплайн – путь (изогнутую кривую– петлю).

9. Выделите в любом окне проекции объект самолет и перейдите в меню Animation – Constraints – Path constraint, а затем выделите в любом окне проекции траекторию, вдоль которой должен перемещаться самолет. Он мгновенно переместится и прикрепится к началу линии. Воспроизведите анимацию, наблюдая за тем, как самолет перемещается вдоль линии от ее начала к концу. Создайте и просмотрите эскиз анимации.

11. Попробуйте настроить параметры контроллера. На вкладке Motion панели управления, активирован режим Parameters. Положение объекта на траектории (группа Path Parameters) в последовательных кадрах анимации определяется процентной мерой, задаваемой счетчиком % Along Path (% вдоль пути) в разделе Path Options свитка Path Parameters панели Motion. Значение этого параметра численно равно выраженному в процентах отношению времени, прошедшего с начала анимации, к общей продолжительности временного сегмента. При применении к объекту контроллера Path автоматически создаются два ключа анимации. В первом кадре анимации величина параметра % Along Path (% вдоль пути) устанавливается равной 0, а в последнем кадре – равной 100. С помощью данного счетчика можно задавать требуемые положения объекта на траектории. Для этого включите режим анимации, установите нужный кадр в качестве текущего и задайте положение объекта в счетчике % Along Path. Кроме этого, можно настроить следующие параметры:

– Follow (Следовать) – установите флажок и продольная ось объекта всегда будет направлена вдоль траектории. Если флажок сброшен, объект в процессе перемещения вдоль траектории сохраняет свою исходную ориентацию в глобальной системе координат. Переключатель Axis позволяет указать, какая из осей локальной системы координат объекта должна быть

выровнена вдоль траектории; Возможно, потребуется установить флажок Flip (Перевернуть), если полет будет происходить хвостом вперед.

– Bank (Крениться) – установите флажок и объект будет наклоняться на поворотах траектории. Если флажок сброшен, объект сохраняет свою локальную ориентацию при перемещении. Если установлен режим крена объекта, можно указать как величину крена в счетчике Bank Amount, так и степень сглаженности, то есть задержку реакции крена объекта на кривизну траектории, в счетчике Smoothness;

– Allow Upside Down (Разрешить движение вверх ногами) – установите этот флажок, чтобы разрешить объекту перевернуться вверх ногами, следуя по траектории, напоминающей «мертвую петлю». Контроллер Path обычно старается сохранить одну из локальных осей объекта (как правило, ось Z) направленной в ту же сторону, что и ось Z глобальной системы координат, запрещая объекту переворачиваться вверх ногами;

– Constant Velocity (Постоянная скорость) – установите флажок, чтобы движение объекта происходило с постоянной скоростью. Если данный флажок сброшен, то MAX определяет точки траектории, в которых объект будет размещаться в последовательные моменты времени, исходя не из длины сплайна, а из числа его вершин. Если, скажем, используемый в качестве траектории сплайн имеет три вершины (начальную, конечную и промежуточную), то при значении параметра % Along Path (% вдоль пути), равном 50, объект всегда будет располагаться в точке промежуточной вершины вне зависимости от соотношения расстояний между вершинами. В связи с этим движение объекта может происходить с переменной скоростью.

15. Сохраните сцену под именем samolet.

Лабораторная работа 4

Анимация вдоль пути

Цель. Изучить способы создания анимации по сплайну.

Методические рекомендации: создаем анимацию по сплайну, выравниваем объект по сплайну, работаем с использованием тегов ограничения.

Путь анимации

Для того, чтобы создать анимацию объекта по конкретной траектории используются сплайны. Это может быть сплайновый примитив, сплайн, созданный сплайновыми инструментами, сплайн, полученный из поверхности объекта или сплайн, спроецированный на геометрию объекта.

Суть технологии анимации вдоль пути заключается в том, что нужно «посадить» объект на заданный путь. Путь представляет собой заранее созданный сплайн, которому можно придать любую желаемую форму. Он

обеспечивает четкую ориентацию и направление движения объекта и может неоднократно редактироваться путем перемещения существующих вершин, создания новых и удаления ненужных — все это в итоге будет приводить к изменению траектории движения объекта. Возможно изменение времени начала и окончания движения, а также его длительности, что позволяет регулировать скорость перемещения объекта по пути. Возможно также указать, в каком месте кривой объект должен находиться в определенный момент времени, — это позволяет смоделировать изменение направления движения, задержку в одной точке или колебание вокруг нее.

В Cinema 4D создание анимации по сплайну осуществляется при помощи специальных тегов.

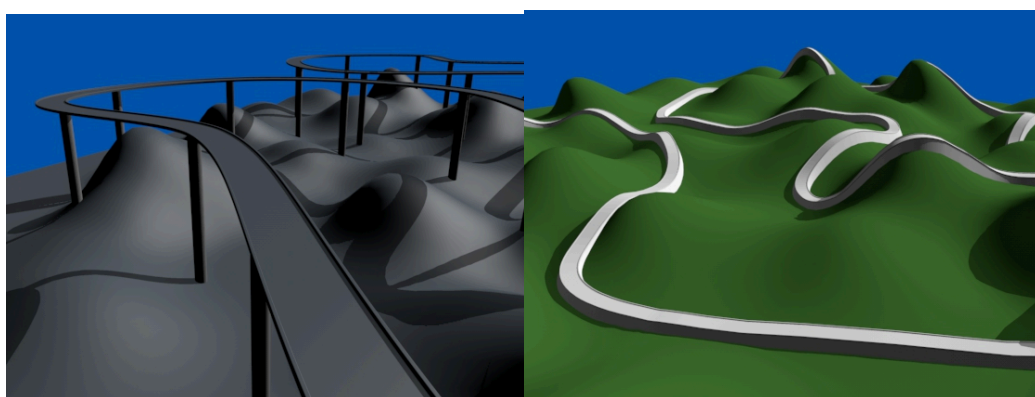


Рисунок 17

Создание анимации по сплайну

Для создания анимации по пути необходимо минимум 2 объекта – сплайн и сам анимируемый объект. Для анимации вдоль сплайна необходимо использовать тег Выровнять по сплайну. Тег назначается на анимируемый объект в менеджере объектов. Для этого необходимо выделить объект и воспользоваться меню Теги в менеджере объектов или кликнуть по объекту правой кнопкой мыши. Среди тегов выбрать Теги Cinema4D – Тег Выровнять по сплайну.

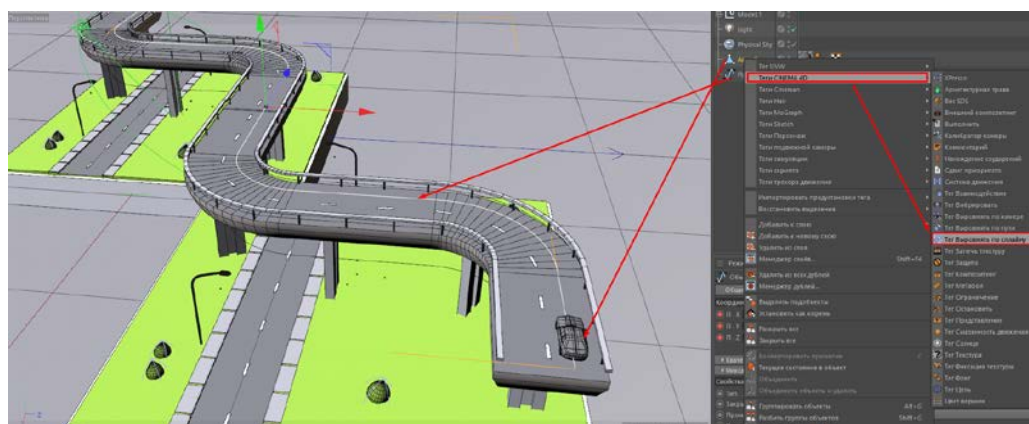


Рисунок 18

После назначения тега необходимо привязать объект к сплайну. Делается это в менеджере атрибутов в атрибутах самого тега. Здесь же и производятся настройки анимации параметров тега и устанавливаются ключевые кадры (не на панели анимации). Для задания конкретного сплайна в качестве пути анимации достаточно перетащить сам сплайн в поле сплайн, находящееся в настройках тега в менеджере атрибутов.

Для создания анимации объекта необходимо анимировать параметр **Позиция**. Для этого необходимо установить ключевой кадр на определенном отрезке времени – кликнуть на кружок возле анимируемого параметра. Следующий ключевой кадр производится в отмеренный вами момент времени таким же образом.

Если вы произведете анимацию параметра **Позиция** для соответствующего выражения, объект будет перемещаться вдоль направляющего пути с осью, определённой вами для параметра **Ось**.

При условии, что объект должен перемещаться только до половины сплайна, произведите анимацию только на 50%.

Для правильной ориентации объекта необходимо установить галочку на параметре **По касательной** – активируется параметр выбора оси.

Имеет значение начало и конец сплайна и его параметр промежуточные точки.

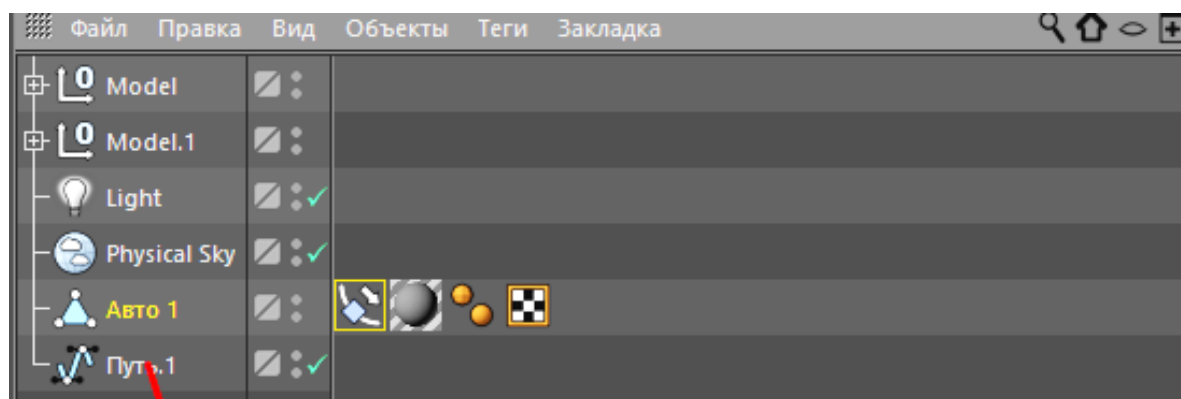


Рисунок 19

Эти параметры влияют на месторасположения объекта начальном положении (0%), направлении движения и равномерность перемещения по всей длине сплайна. Дополнительно вы можете назначить направляющий сплайн, посредством перетаскивания из менеджера объектов в поле **Направляющий сплайн**. Это позволит контролировать положение объекта на протяжении всего пути движения. Кроме того, изменяя положение и форму направляющего сплайна можно регулировать угол наклона объекта.

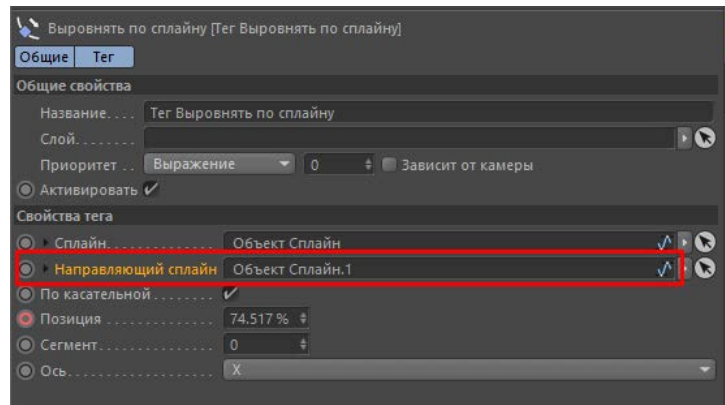


Рисунок 20

Для преобразования анимации в ключевые кадры для ее правки, необходимо воспользоваться командой Запечь объект в меню Функции менеджера Таймлайн. При этом каждый кадр анимации станет ключевым.

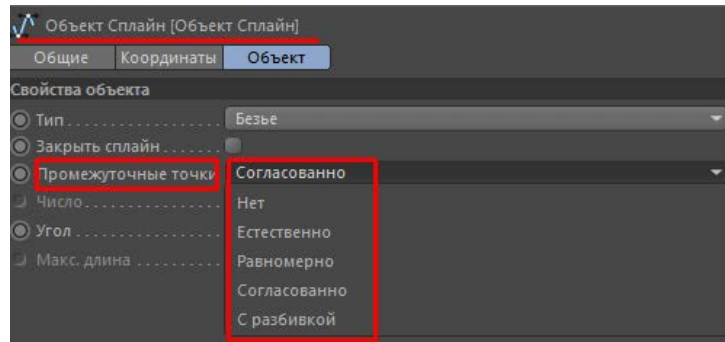


Рисунок 21

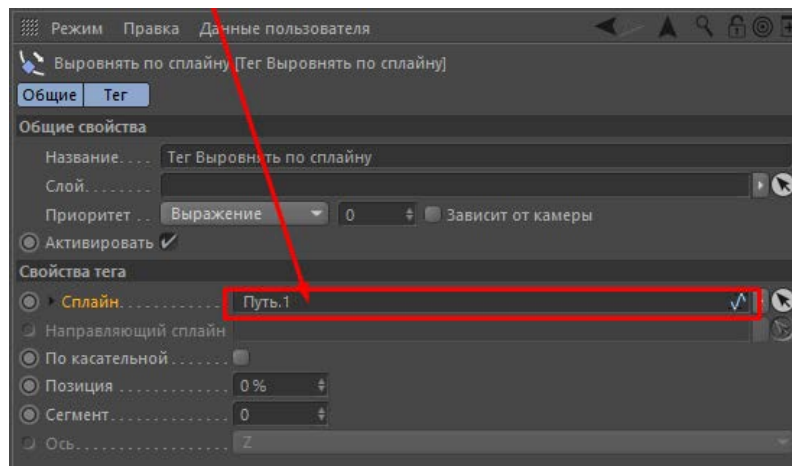


Рисунок 22

Для комфортной работы с ключевыми кадрами необходимо воспользоваться командами Редуцирования или Упрощения ключей. Обе эти функции позволяют сократить количество ключевых кадров.

Задание. Анимировать движение автомобиля по мосту.

Лабораторная работа 5

Работа с анимационными кривыми

Цель работы: освоить навыки работы кривыми функций в редакторе кривых Curve Editor.

Методические рекомендации

Работа с кривыми функций

Следующим режимом редактирования в Track View является режим Function Curve. В других режимах показывается положение ключей и диапазонов с учетом времени, но не указывается действительных используемых значений анимации. Режим Function Curve для выделенных оживляемых элементов отображает и значения анимации в ключах, и интерполированные значения между ключами. В режиме Function Curve при изменении значений ключей эффект изменения показывается на выводе контроллера. Подобная работа предоставляет возможность тонко настраивать значения анимации, подгоняя форму кривых функций во времени. В режиме Function Curves могут отображаться только оживляемые элементы с ключ-базированными контроллерами.

Для отображения кривой функции для элемента выберите один или несколько элементов и щелкните на Function Curves. Окно редактирования Track View изменится с целью отображения формы вывода контроллера элемента во времени, и на линейке инструментов Track View появляется новый набор командных кнопок. Для отображения ключей для кривой щелкните на ней. Значения, ассоциированные с каждым ключом, показываются как вершины кривой, а зеленый треугольник рядом с элементом в иерархии, генерирующем данную кривую, станет подсвеченным. Если данный элемент выводит более одного значения (например, положение, масштаб или цвет), ключевые вершины будут отображены на всех кривых, связанных с данным элементом.

Редактирование кривых функций

Для редактирования кривой прежде всего следует выбрать одну или несколько ее вершин. С целью предотвращения случайной отмены выбора в наборе ключей щелкните на Lock Selection. В случае отображения кривых множества элементов и желании выбрать вершины только некоторых, выберите кривые, с которыми будете работать, и щелкните на Freeze Nonselected Curves. Это предотвратит кривые, которые решено не настраивать, от случайного выбора и модификации.

После этого выбранные вершины можно редактировать по команде Move Keys, масштабировать ключи во времени с использованием Scale Keys по отношению к текущему времени, масштабировать значения, ассоциированные с ключами, по отношению к 0, используя Scale Values, или удалять ключи используя команду Delete Keys. При работе с командами Move Keys или Scale

Keys применительно к вершинам, ассоциированные контроллеры которых выводят более одного значения, соответствующие вершины на других кривых контроллера перемещаются во времени вместе с выбранными вершинами. Обратите внимание на то, что команда Move Keys может настраиваться для перемещения только вертикально или горизонтально. Для этого следует щелкнуть и удерживать кнопку Move Keys и выбрать подходящую кнопку из подменю Move Keys.

Выбранные ключи можно выровнять относительно текущего времени, щелкнув на Align Keys. Если для элемента выбрано более одного ключа, по текущему времени выравнивается самый левый ключ, а остальные выбранные ключи перемещаются на ту же величину, что и самый левый ключ. Невыделенные ключи не перемещаются.

Значения, ассоциированные с отдельными вершинами, настраиваются посредством Key Info и щелчка правой кнопкой на вершине, выбором вершин и щелчком на Properties или выбором вершин и вводом значений в поля в нижней части Track View.

В Function Curves можно также настроить параметры интерполяции, ассоциированные с каждой вершиной.

Применение кривых Ease и Multiplier

Кривые Ease и Multiplier применимы к любому оживляемому параметру. Ease Curve влияет на время ключей контроллера, к которому применяется. Multiplier Curve влияет на выводное значение контроллера, к которому применена. Для применения Ease Curve выберите один или несколько оживляемых элементов и щелкните на Apply Ease Curve. Для доступа к команде Multiplier Curve щелкните и удерживайте Apply Ease Curve и затем выберите из подменю Apply Multiplier Curve. Multiplier Curves присваиваются выбранным элементам таким же образом, как и Ease Curves.

Ease Curves и Multiplier Curves можно удалять, выбрав имя кривой в иерархическом списке и щелкнув на Delete Ease/Multiplier Curve. Ease Curves и Multiplier Curves можно отключать путем выбора имени кривой и щелчка на Ease/Multiplier Curve Enable/Disable Toggle.

Типы Out-of-Range

Вывод контроллера по умолчанию вне диапазона его ключей заключается в выводе значения ближайшего ключа. Подобное действие называется типом Constant Out-of-Range (ORT). Как к правой, так и к левой сторонам диапазона можно применить шесть ORT. Кроме того, сами диапазоны можно разъединить с базовыми ключами, предоставляя возможность создания петель или повторяющихся циклов. Шесть упомянутых ORT таковы: Constant, Cycle, Loop, Ping Pong, Linear и Relative Repeat.

Для изменения ORT элемента, подвергающегося анимации, выделите элемент в иерархическом списке и щелкните на Parameter Curve Out-of-Range Types. Возникающий диалог Param Curve Out-of-Range Types предоставляет возможность применить любой из шести типов к диапазону до или после него. Сплошная часть каждой линии показывает пример кривой функции в диапазоне. Пунктирная часть линии демонстрирует влияние, которое каждый ORT будет оказывать вне диапазона. При Constant и Linear ORT – движение продолжается по прямой линии. Для мяча, прыгающего вниз по лестнице, можно выполнить анимацию одного прыжка и затем зациклить анимацию, чтобы заставить мяч прыгать вниз оставшуюся часть лестницы.

Для отсоединения линии диапазона дорожки выберите дорожку и щелкните на Position Ranges. На линейке инструментов Track View отобразится новый набор командных кнопок. Отображение дорожек изменится так, чтобы показывать линию диапазона для каждой дорожки, проходящей по ключам дорожки. Можно перемещать линию диапазона, транспортируя ее, или настроить длину линии диапазона, транспортируя конечные точки линии.

Работа с траекториями

Траектория изображается в виде линии с точками, обозначающими позиции кадров, и квадратиками, обозначающими позиции ключей.

Работая в Trajectory, можно редактировать ключи кривой траектории, преобразовывать траекторию в сплайн или преобразовывать сплайн в траекторию. Для редактирования ключей траектории щелкните на Sub-Object в ветви Trajectory панели Motion. Это предоставит возможность выбрать один или несколько ключей, а также перемещать, вращать или масштабировать их.

В качестве траектории для объекта можно использовать отредактированный или новый сплайн. Для выбора сплайна, используемого в качестве траектории, нажмите Convert From. Из сплайна берутся экземпляры и конвертируются в положения ключей. Поля Start Time и End Time указывают диапазон времени, внутри которого разместятся эти ключи. Любые существующие в данном диапазоне ключи будут удалены. Любые ключи, существующие вне диапазона, не принимаются во внимание. Экземпляры берутся так, чтобы ключи, представляющие вершины сплайна, равномерно размещались во времени. Это означает, что взятые экземпляры обязательно равномерно располагаются по длине сплайна.

Задание. Создайте анимацию прыгающего мяча по ступенькам лестницы.

1. Нарисуйте сплайн “ступеньки”. Командой Extrude создайте «лестницу».
2. В качестве мяча возьмите примитив Сфера.
3. Включите отображение траектории анимации в окне проекций.
4. Задайте интервал анимации 7 сек.

5. Анимлируйте прыжки мяча по лестнице. Откорректируйте кривую в Curve Editor.

6. Сохраните сцену под именем jumpball.

Лабораторная работа 6

Прямая кинематика

Цель работы: отработать навыки связывания объектов в иерархическую цепочку.

Задание 1. Создание лампы.

Для отработки навыков связывания объектов в иерархическую цепочку понадобится создать несложную сцену в виде настольной лампы.

Выполните следующие действия:

1. Подготовьте объекты, которые нужно будет связать и которые будут образовывать простую сцену под названием Настольная лампа, показанную на рисунке 23.

Подставка лампы представляет собой цилиндр с фаской радиусом 100 см и высотой 15 см (истинные размеры лампы, конечно, великоваты, но масштаб в данном случае не имеет значения). На подставке установлена полусфера радиусом 35 см, играющая роль опорного шарнира стойки лампы. Рычаги – три однотипных параллелепипеда с фаской шириной 25 см и высотой (толщиной) 10 см. Длина первых двух из них равна 195 см, а третий рычаг несколько короче – 125 см. Шарниры между рычагами также сделаны в виде цилиндров с фаской радиусом по 25 см и толщиной 14 см. Отражатель лампы выполнен в виде тела вращения (или конуса), чтобы внизу его имелось отверстие для выхода лучей света.

Внутри отражателя помещен свободный прожектор, размер конуса света которого подогнан так, чтобы он соответствовал размеру отражателя. У прожектора включен режим отбрасывания трассированных теней. Чтобы освещалась внутренняя поверхность отражателя, в центре конической части колпака размещен всенаправленный осветитель. Из всех объектов сцены в число освещаемых данным всенаправленным источником включен только отражатель лампы. Прожектор и всенаправленный источник сгруппированы с отражателем.

Основание сцены выполнено в виде плоского параллелепипеда длиной 700 см, шириной 500 см и высотой 15 см. «Украшение» в виде выключателя можете приделать на свое усмотрение. Рычаги именуются Рычаг01-Рычаг03 в направлении от опоры лампы к ее отражателю. Полусферический шарнир имеет имя Шарнир01, цилиндрические – Шарнир02 и Шарнир03, порядок нумерации – также от опоры лампы к отражателю.

2. Добавьте в состав сцены еще один осветитель – свободный

направленный источник, который будет служить для создания общего уровня освещения сцены. Разумеется, у этого осветителя также следует включить режим отбрасывания теней. При желании примените к объектам материалы, позаимствовав их из библиотеки. Для колпака лампы имеет смысл создать двусторонний материал, применив, например, для лицевой стороны материал Позолота, а для изнаночной – Фаянс. В файле Лампа1 показан результат визуализации получившейся лампы, готовой к связыванию и анимации.

3. Щелкните на кнопке Select and Link (Выделить и связать) главной панели инструментов. Переместите курсор в любое из окон проекций. В данном случае удобно работать в окнах Front (Вид спереди) или Perspective (Перспектива).

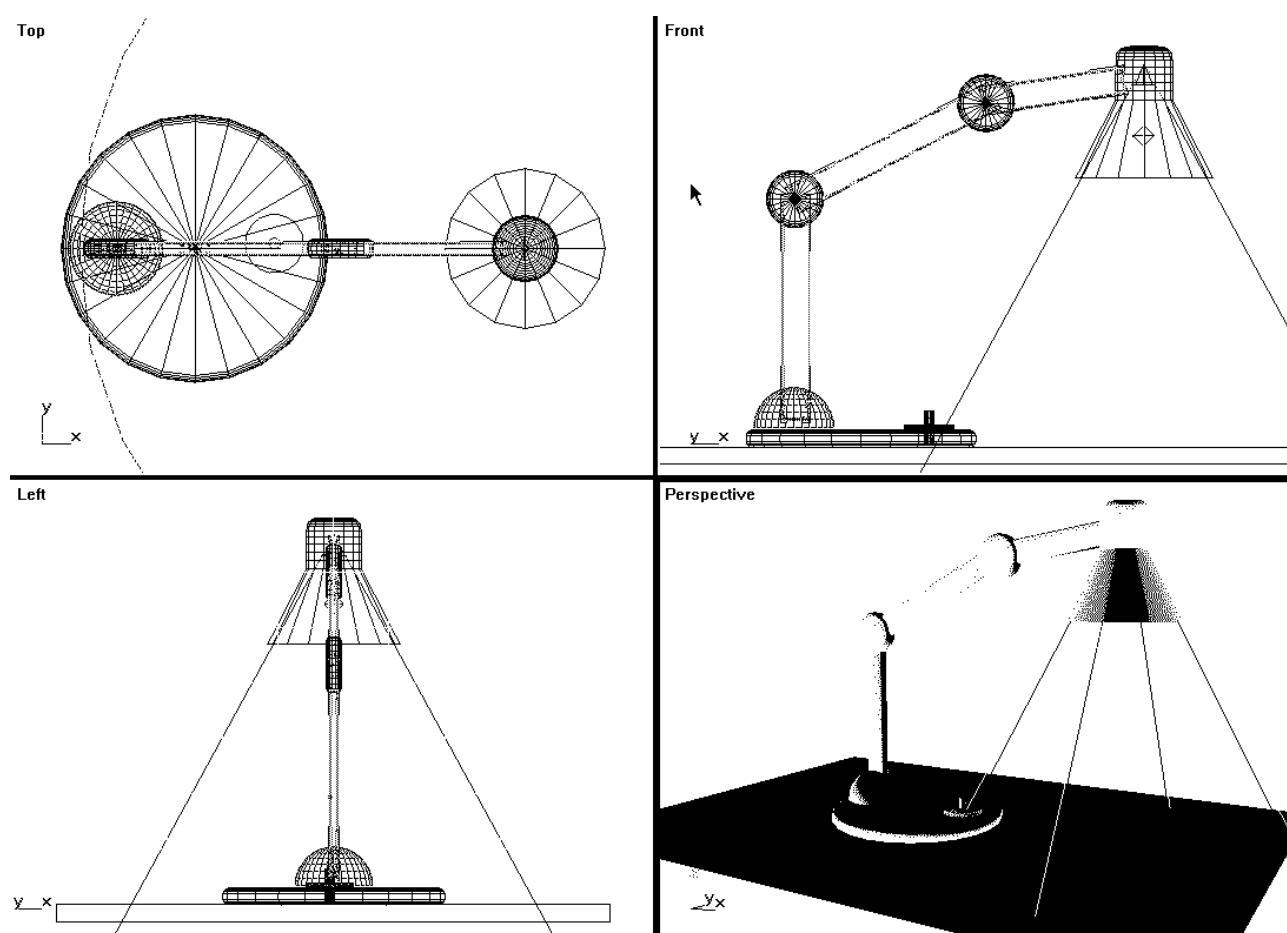


Рисунок 23

4. Связывание следует производить от дочерних объектов к родительским. Расположите курсор над объектом, который должен стать самым младшим дочерним объектом цепочки, – над отражателем лампы. Если этот объект допускает установление связи, курсор примет вид крестика, как в случае обычного выделения объекта. Щелкните на отражателе. Объект выделится, а курсор примет вид значка на кнопке инструмента. Удерживая кнопку мыши, перетащите курсор до объекта, который должен стать предком – до

ближайшего к отражателю рычага лампы (объект Рычаг03), При этом за курсором будет тянуться пунктирная линия. Над объектом, который может стать объектом-предком, курсор снова примет вид значка на кнопке инструмента. Отпустите кнопку мыши. Оба объекта на мгновение выделятся белым цветом, а затем приобретут обычную окраску. Связь установлена. Теперь щелкните на коротком рычаге, перетащите курсор к ближайшему шарниру – объекту Шарнир03. Продолжите работу, последовательно связав Шарнир03 с объектом Рычаг02; Рычаг02 – с объектом Шарнир02; Шарнир02 – с объектом Рычаг01; Рычаг01 – с полусферой Шарнир01, а полусферу – с объектом Опора. Если вы допустите ошибку в ходе установления связей, не отчаивайтесь. Чтобы разорвать ошибочно созданную связь, выделите связанные объекты и щелкните на кнопке Unlink Selection (Разорвать связь с выделенными объектами) панели инструментов, а затем повторите связывание.

5. Теперь все нужные детали настольной лампы связаны между собой и готовы к анимации по методу прямой кинематики. Связи между объектами можно наблюдать в окнах проекций в виде системы рычагов, соединяющих опорные точки выделенных объектов и отдаленно напоминающих кости скелета. Чтобы увидеть такие рычаги, выделите связанные объекты и установите флажок Display Links (Показывать связи) в свитке Link Display (Показ связей), расположенном в нижней части командной панели Display (Дисплей). Если установить еще и флажок Link Replaces Object (Связь заменяет объект), то изображения выделенных объектов цепочки будут заменены изображениями связей. Иногда это облегчает понимание того, как работает метод прямой кинематики.

6. Сохраните сцену под именем Настольная лампа.

Методические рекомендации

Действие преобразований связанных объектов (правила прямой кинематики)

При преобразовании перемещения или поворота любого из объектов, связанных в иерархическую цепочку, действуют следующие правила, определенные для метода прямой кинематики:

– преобразование любого родительского объекта распространяется на все его дочерние объекты, но не касается объектов-предков преобразуемого объекта. Это означает, например, что при перемещении или повороте подставки модели лампы, все дочерние объекты будут сохранять свое положение по отношению к подставке, перемещаясь или поворачиваясь вместе с ней (попробуйте проделать это, чтобы убедиться). Если же применить преобразование перемещения к среднему рычагу лампы, то вместе с ним переместятся только его дочерний шарнир, верхний рычаг и отражатель, а

нижний рычаг и подставка лампы сохраняют свое положение и ориентацию. В результате конструкция «разрушится»;

– самый младший объект-потомок наследует преобразования всех объектов-предков, а преобразования младшего объекта-потомка не влияют на остальные элементы цепочки. В примере с лампой это означает, что отражатель света на конце системы рычагов будет перемещаться и поворачиваться вместе с любым элементом конструкции лампы, однако он сам может быть перемещен и повернут независимо от остальных деталей.

Подобные результаты преобразований связанных объектов объясняются тем, что преобразование всех объектов-потомков производится не относительно их собственных опорных точек, а относительно опорной точки объекта-предка. При анимации связанных объектов правильный выбор положений опорных точек играет ключевую роль.

Расстановка опорных точек связанных объектов

Опорная точка объекта при его создании помещается обычно или в центр основания, или в геометрический центр габаритного контейнера объекта.

Если повернуть средний рычаг лампы, то конструкция «разрушится», так как рычаг повернется относительно своего центра, а не места сочленения с шарниром.

При анимации связанных цепочек объектов часто возникает необходимость поворачивать объект относительно одного из его концов или любой другой заданной точки, а не центра. Например, приводя в движение руку персонажа анимации, вы, конечно, захотите, чтобы она поворачивалась в плече, а не вокруг локтя. С лампой тоже необходимо сделать так, чтобы поворот ее рычагов не приводил к разрыву сочленений в шарнирах. Для решения такой задачи требуется уметь перемещать опорные точки объектов. Доступ к средствам управления Опорными точками объектов можно получить, выделив объекты и щелкнув на вкладке командной панели Hierarchy (Иерархия).

В верхней части командной панели Hierarchy (Иерархия) под полем имени выделенного объекта имеются три кнопки:

– Pivot (Опора) – позволяет изменять положение и ориентацию опорных точек и управлять расположением опорной точки в пределах объекта;

– IK (Inverse Kinematics – Обратная кинематика) – позволяет применять к связанным объектам режим анимации по методу обратной кинематики, а также настраивать параметры связей объектов

– Link Info (Данные о связях) – позволяет задавать характеристики связей объектов друг с другом.

Для управления преобразованиями положения, поворота или масштаба опорной точки объекта в общем случае необходимо выполнить следующие действия:

1. Выделите объект, перейдите на командную панель Hierarchy (Иерархия) и щелкните на кнопке Pivot (Опора). Появятся два свитка: Adjust Pivot (Настройка опоры), который позволяет управлять расположением и ориентацией опорной точки объекта, и Adjust Transform (Настройка преобразования), позволяющий выполнить собственно преобразование и выровнять объект относительно системы координат.

2. Задайте режим преобразования, щелкнув в разделе Move/Rotate/Scale (Перемещение/Поворот/Масштаб) свитка Adjust Pivot (Настройка опоры) на одной из трех кнопок, каждая из которых фиксируется в нажатом положении и подсвечивается синим цветом:

- Affect Pivot Only (Только опора) – обеспечивает возможность перемещать или поворачивать только опорную точку объекта. При этом преобразования не будут касаться ни самого объекта, ни его дочерних объектов. После включения данного режима используйте инструменты преобразований для перемещения, поворота или масштабирования опорной точки обычным образом;

- Affect Object Only (Только объект) – обеспечивает возможность перемещать, поворачивать или масштабировать только геометрическую модель объекта. При этом положение и ориентация опорной точки объекта и его дочерних объектов не изменяются;

- Affect Hierarchy Only (Только дочерние объекты) – обеспечивает возможность перемещать, поворачивать или масштабировать любой дочерний объект текущего выделенного объекта по отношению к его опорной точке. Ни на геометрическую модель этого объекта, ни на его опорную точку эти преобразования не распространяются.

3. Для выравнивания положения опорной точки используйте следующие кнопки раздела Alignment (Выравнивание) свитка Adjust Pivot (Настройка опоры), которые становятся доступны, если нажата одна из кнопок Affect Pivot Only (Только опора) или Affect Object Only (Только объект), и меняют свои названия в зависимости от того, какая из этих кнопок нажата:

- Center to Object/Center to Pivot (Центрировать по объекту/Центрировать по опоре) – перемещает опорную точку в центр объекта или перемещает объект так, чтобы опорная точка оказалась в его центре;

- Align to Object/Align to Pivot (Выровнять по объекту/Выровнять по опоре) – выравнивает направление осей координатной системы опорной точки по осям системы координат неподвижного объекта или выравнивает ориентацию объекта относительно осей координат неподвижной опорной точки;

- Align to World (Выровнять глобально) – выравнивает направление осей координатной системы опорной точки или объекта по осям глобальной системы

координат.

4. Чтобы восстановить исходные положение и ориентацию опорной точки, которые она имела на момент создания объекта, щелкните на кнопке *Reset Pivot* (Восстановить опору). Это бывает полезно, например, если при перемещении или повороте опорной точки была допущена ошибка.

5. Чтобы обеспечить возможность настройки положения, ориентации или масштаба объекта, входящего в иерархическую цепочку, и при этом не оказывать влияния ни на один из дочерних объектов, щелкните на кнопке *Don't Affect Children* (Не трогать дочерние объекты) в разделе *Move/Rotate/Scale* (Перемещение/Поворот/Масштаб) свитка *Adjust Transform* (Настройка преобразования).

6. Чтобы восстановить положение и ориентацию опорной точки объекта относительно осей глобальной системы координат, не оказывая при этом влияния ни на сам объект, ни на его дочерние объекты, щелкните на кнопке *Transform* (Преобразование) в разделе *Reset* (Восстановление) свитка *Adjust Transform* (Настройка преобразования). Щелчок на кнопке *Scale* (Масштаб) переустанавливает коэффициенты матрицы преобразований объекта так, чтобы они соответствовали его новому масштабу, после чего новый размер объекта принимается за 100%.

Попробуем применить только что полученные сведения на практике.

Задание 2. Расстановка по местам опорных точек рычагов настольной лампы.

В исходном положении опорные точки рычагов лампы располагаются в их геометрических центрах. Чтобы рычаги при повороте не вызывали разрыва конструкции, каждый из них должен поворачиваться вокруг центра своего родительского шарнира. Для этого требуется сместить опорные точки каждого из рычагов к тем концам, которые входят в родительские по отношению к этим рычагам шарниры. Так, опорную точку объекта *РычагОЗ* нужно сместить к его концу, входящему в *ШарнирОЗ*, и т. д.

Чтобы проделать это, выполните следующие действия:

1. Откройте файл сцены *Настольная лампа* или продолжите работу над сценой, если она уже загружена в программу. Выделите объект *РычагОЗ* и перейдите на командную панель *Hierarchy* (Иерархия). Кнопка *Pivot* (Опора) должна быть нажата по умолчанию. Щелкните в разделе *Move/Rotate/Scale* (Перемещение/Поворот/Масштаб) свитка *Adjust Pivot* (Настройка опоры) на кнопке *Affect Pivot Only* (Только опора), которая должна зафиксироваться в нажатом положении и подсветиться синим цветом.

2. Выберите в раскрывающемся списке *Reference Coordinate System* (Система координат) главной панели инструментов вариант *Local* (Локальная) и щелкните на кнопке ограничения преобразования осью *Y*, направленной

вдоль продольной оси объекта. Выберите инструмент Select and Move (Выделить и переместить) и аккуратно переместите опорную точку рычага к его левому концу, то есть к центру шарнира этого рычага.

3. Повторите аналогичные действия применительно к объектам Рычаг02 и Рычаг01.

4. Убедитесь в том, что при таком расположении опорных точек поворот рычага не будет вести к «разрушению» конструкции. Поворот среднего рычага при правильном выборе положения опорной точки не разрушает конструкцию модели. Восстановите исходное положение всех рычагов и сохраните сцену под прежним именем.

Настройка параметров связей объектов в цепочках

На связи между объектами в иерархических цепочках можно накладывать ограничения, чтобы обеспечить нужный характер движений связанных объектов.

Для контроля параметров и настройки связей объектов в иерархических цепочках в общем случае необходимо выполнить следующие действия:

1. Выделите дочерний объект и щелкните на кнопке Link Info (Данные о связях) командной панели Hierarchy (Иерархия). Появятся свитки Locks (Блокировки) и Inherit (Наследование), состоящие из однотипных групп флажков.

2. Для блокирования преобразований, которые могут быть применены к дочернему объекту непосредственно, установите флажки нужных координатных осей X, Y или Z в разделах преобразований Move (Перемещение), Rotate (Поворот) и Scale (Масштаб) свитка Locks (Блокировки). Установка любого флажка запрещает непосредственное преобразование выделенного объекта в направлении данной оси координат. Данные ограничения не распространяются на преобразования дочернего объекта вместе с родительским. Это значит, например, что в случае перемещения родительского объекта, дочерний объект будет перемещаться вместе с ним вне зависимости от того, какие из блокирующих флажков установлены в разделе Move (Перемещение).

3. Укажите, какие преобразования наследуются дочерним объектом от объекта-предка, используя флажки координатных осей X, Y или Z свитка Inherit (Наследование), также сгруппированные в разделы по видам преобразований Move (Перемещение), Rotate (Поворот) и Scale (Масштаб). Выделенный дочерний объект будет наследовать преобразование родительского объекта по всем осям глобальной системы координат, флажки которых установлены в разделе соответствующего преобразования. По умолчанию установлены флажки всех преобразований по всем осям координат. Например, если в разделе Move (Перемещение) сброшены флажки осей X и Y, то объект будет

перемещаться вверх и вниз при перемещении родительского объекта, но не будет наследовать движение родительского объекта из стороны в сторону.

Задание 3. Анимация настольной лампы по методу прямой кинематики

Одна из американских кинокомпаний прославилась, выпустив на экран мультипликационный фильм, целиком подготовленный с помощью компьютера, главными персонажами которого были настольные лампы. Выразительная «игра» этих персонажей, их живые характеры воздействовали на зрителя не за счет внешнего сходства с человеком или животным, а только за счет удачно схваченных движений. Попробуйте и вы «оживить» модель лампы по методу прямой кинематики. Заставьте лампу наклоняться вперед и опускать «голову», как бы в попытке заглянуть, что делается у нее «под ногами» и «за спиной».

Выполните следующие действия:

1. Откройте файл сцены Настольная лампа или продолжите работу над ней, если сцена уже загружена в программу. Приведите все детали лампы в положения, которые они должны занимать на момент начала анимации. Удобнее всего это делать с помощью окна диалога Rotate Transform Type-In (Ввод данных преобразования поворота). Исходные значения углов поворота объекта Рычаг01 относительно осей глобальной системы координат должны составлять $(90^\circ; 0^\circ; 0^\circ)$, объекта Рычаг02 – $(90^\circ; 62,5^\circ; 0^\circ)$, Рычаг03 – $(90^\circ; 81,5^\circ; 0^\circ)$, а объекта Отражатель – $(0^\circ; 0^\circ; 0^\circ)$. Установите кадр № 0 в качестве первого кадра анимации, активизируйте нужное окно проекции и щелкните на кнопке Animate (Анимация).

2. Установите в качестве текущего кадр № 25, перемещая ползунок таймера анимации. Поверните нижний из рычагов лампы (объект Рычаг01) на -20° относительно оси Y глобальной системы координат.

Преобразование будет автоматически распространяться на дочерние объекты, так как анимация по методу прямой кинематики применяется к связанным объектам по умолчанию,

3. Переключитесь на кадр № 50. Выделите второй рычаг и установите его угол поворота относительно оси Y глобальных координат равным примерно 67° .

4. Сделайте текущим кадр № 75. Теперь выделите и немного поверните объект Рычаг03, самый ближний к отражателю лампы. Установите его угол поворота относительно оси Y равным примерно 144° .

5. Наконец, установите в качестве текущего кадр № 100. Восстановите Исходные положения всех трех рычагов, начиная с самого старшего – объекта Рычаг01. Закончив создание ключей, выключите режим анимации, щелкнув на кнопке Animate (Анимация). Просмотрите анимацию в окне проекции, создайте ее эскиз или выполните итоговую визуализацию, как описано в предыдущих

главах.

Выполните рендер готовой анимации движений лампы по методу прямой кинематики и сохраните в файле Лампа.avi.

Лабораторная работа 7

Интерактивная обратная кинематика

Цель работы: изучить и освоить способы реализации метода обратной кинематики.

Методические рекомендации

Метод обратной кинематики требует при анимации связанных объектов действовать ровно наоборот по сравнению с методом прямой кинематики. Вместо преобразования объекта-предка следует применять преобразование к самому младшему из дочерних объектов, и действие этого преобразования распространяется на всех предков из иерархической цепочки данного объекта. Это дает возможность, например, переместив всего один палец персонажа, заставить его вытянуть вперед руку. В нашем примере с настольной лампой перемещение ее отражателя вызовет при анимации по методу обратной кинематики соответствующий поворот всех рычагов.

Метод обратной кинематики позволяет:

- реализовывать два типа сочленений объектов: вращающиеся и скользящие;
- ограничивать диапазон действия сочленений объектов любыми осями координат, размером углового сектора поворота или расстоянием перемещения;
- выполнять настройку таких параметров сочленений, как приоритетность, наличие и сила трения и т. п.

Способы реализации метода обратной кинематики

Для анимации цепочки связанных объектов можно использовать один из двух вариантов метода обратной кинематики:

- приложенная обратная кинематика (Applied IK) – метод, основанный на использовании направляющего объекта (follow object). Направляющим объектом может служить любой объект сцены, который подвергнут анимации. Часто в качестве направляющего используется вспомогательный объект-пустышка. Для анимации цепочки какой-то из ее объектов должен быть привязан к направляющему объекту при помощи кнопки Bind (Привязать) свитки Object Parameters (Параметры объекта) командной панели Hierarchy (Иерархия), который рассматривается ниже. После запуска процесса анимации выполняется автоматический расчет ключей сразу для всей заданной серии кадров с учетом ограничений подвижности сочленений объектов. При этом не требуется включать режим анимации кнопкой Animate (Анимация);
- интерактивная обратная кинематика (Interactive IK) – реализуется путем

ручного перемещения и поворота объектов цепочки в окнах проекций при включенном режиме анимации, то есть при нажатой кнопке Animate (Анимация). В этом режиме программа также учитывает ограничения на подвижность сочленений, но позволяет создать ключ анимации только в текущем кадре. Результаты, полученные с помощью данного метода, могут отличаться от результатов анимации по методу приложенной ИК, при использовании которого движение оптимизируется сразу на всем сегменте времени анимации.

Программа позволяет использовать еще один подход к анимации по методу обратной кинематики, основанный на использовании контроллера ИК (Обратная кинематика). Применение этого контроллера требует создания системы объектов типа Bones (Кости) с нужным числом элементов-«костей» и настройки параметров контроллера ИК (Обратная кинематика) в свитке IK Controller Parameters (Параметры контроллера обратной кинематики) командной панели Motion (Движение).

Реализация методов интерактивной и приложенной обратной кинематики

Для реализации методов интерактивной или приложенной обратной кинематики необходимо выполнить ряд действий, одинаковых для обоих методов и связанных с настройкой свойств объектов, входящих в цепочку, и их сочленений, и РЯД специфических действий, связанных собственно с анимацией объектов по методам интерактивной или приложенной обратной кинематики.

Подготовка цепочки объектов для анимации по методам интерактивной или приложенной обратной кинематики

Чтобы создать анимацию объектов, используя методы интерактивной или приложенной обратной кинематики, нужно выполнить следующие общие действия по настройке свойств объектов и их сочленений:

1. Связать объекты в иерархические цепочки, как и в случае прямой кинематики. При связывании всегда начинайте с последнего, самого младшего дочернего объекта цепочки и перемещайтесь к самому старшему объекту-предку.

2. Произвести тщательную настройку свойств всех объектов иерархической цепочки и их сочленений. Для этого выделите объект, входящий в состав иерархической цепочки, применительно к которому следует настроить параметры обратной кинематики, а затем щелкните на кнопке ИК командной панели Hierarchy (Иерархия). Обычно начинать настройку следует с самого младшего дочернего объекта. Появятся свитки параметров обратной кинематики Inverse Kinematics (Обратная кинематика), Object Parameters (Параметры объекта), Auto Termination (Автоограничение), Sliding Joints

(Скользящие сочленения) и Rotational Joints (Вращающиеся сочленения).

3. Задать степень привязки выделенного объекта к его исходному положению и ориентации, а также настроить приоритетность объектов, используя следующие параметры свитка Object Parameters (Параметры объекта).

– Bind Position (Привязать к положению) – установка этого флажка заставляет выделенный объект иерархической цепочки стремиться сохранить свое текущее положение в глобальной системе координат или занять положение направляющего объекта, если таковой назначен. Щелчок на кнопке R (Relatively – Относительно) справа от флажка заставляет выделенный объект сохранять свое положение относительно направляющего объекта. Если кнопка R не нажата, то объект иерархической цепочки будет стремиться переместиться в положение, занимаемое направляющим объектом;

– Bind Orientation (Привязать к ориентации) – установка этого флажка заставляет выделенный объект иерархической цепочки стремиться сохранить свою текущую ориентацию в глобальной системе координат или принять ориентацию направляющего объекта. Щелчок на кнопке R справа от флажка заставляет выделенный объект сохранять свою ориентацию относительно направляющего объекта. Если кнопка R не нажата, то объект иерархической цепочки будет стремиться повернуться так, чтобы соответствовать ориентации направляющего объекта;

– Axis (Ось) – флажки этой группы задают оси координат, на которые распространяется привязка;

– Weight (Вес) – позволяет задать степень влияния направляющего объекта на связанный с ним объект цепочки, при этом значение 0 устраняет привязку;

– Precedence (Приоритетность) – позволяет установить приоритеты (важность) объектов в кинематической цепочке. Этот параметр определяет степень влияния объекта на результат расчетов в ходе вычисления ИК-решения. Чем выше значение параметра, тем сильнее влияние объекта на результат. Обычно приоритетность определяется и назначается самой программой. Текущее значение приоритетности объекта указывается в счетчике, если щелкнуть на одной из кнопок Child-Parent (Потомок-Предок) или Parent-Child (Предок-Потомок);

– Child-Parent (Потомок-Предок) – устанавливает для всей цепочки более высокий приоритет объектов-потомков по отношению к предкам. Если щелкнуть на этой кнопке, то все объекты-потомки выделенного объекта получат приоритет выше, чем у текущего объекта;

– Parent-Child (Предок-Потомок) – устанавливает для всей цепочки более высокий приоритет объектов-предков по отношению к потомкам. Если

щелкнуть на этой кнопке, то все объекты-предки выделенного объекта получат приоритет выше, чем у текущего объекта;

– Copy (Копировать), Paste (Вставить) – позволяют копировать параметры скользящих и вращающихся сочленений от одного объекта к другому. Выделите объект с настроенными параметрами сочленения и щелкните на кнопке Copy (Копировать), затем выделите другой объект и щелкните на кнопке Paste (Вставить);

– Mirror Paste (Вставить зеркально) – используйте этот переключатель для выбора оси координат, относительно которой будет производиться зеркальное отражение параметров сочленений ИК-цепочки при вставке.

4. Присвоить сочленению выделенного объекта с родительским объектом тип скользящего или вращающегося, а также задать ограничения подвижности сочленений, которые будут учитываться в ходе решения задачи обратной кинематики, используя три однотипные группы параметров – X Axis (Ось X), Y-Axis (Ось Y) и Z-Axis (Ось Z) – свитков Sliding Joints (Скользящие сочленения) и Rotational Joints (Вращающиеся сочленения):

– Active (Активно) – флажок, разрешающий выделенному объекту скользить вдоль или поворачиваться вокруг той оси локальной системы координат объекта-предка, к которой относится данный флажок. Если флажок сброшен, то перемещение или поворот относительно данной оси запрещены. По умолчанию флажки всех трех осей в свитке Sliding Joints (Скользящие сочленения) сбрасываются, а в свитке Rotational Joints (Вращающиеся сочленения) – устанавливаются, то есть соединению по умолчанию назначается тип вращающегося по всем трем осям;

(Замечание. Сброс флажков Active (Активно) никак не ограничивает возможностей перемещения или поворота объекта путем применения к нему средств преобразований. Состояние этих флажков учитывается только при решении задачи обратной кинематики.)

– Limited (Ограниченно) – установка этого флажка включает режим ограничения движения относительно выбранной оси координат. Допустимые верхний и нижний пределы движения сочленения вдоль заданных осей задаются счетчиками From (От) и To (До). Для вращающихся сочленений значения данных параметров измеряются в градусах;

– Ease (Плавно) – включает режим плавного замедления скольжения по мере приближения к границам допустимого диапазона;

– Spring Back (Пружинный буфер) – если данный флажок установлен, то сила упругости воображаемой буферной пружины, заставляющей объект возвращаться назад после того, как он минует точку равновесного состояния, будет возрастать по мере удаления объекта от этой точки. Положение точки равновесного состояния задается в счетчике справа от флажка;

– Spring Tension (Натяжение пружины) – задает силу натяжения воображаемой буферной пружины;

– Damping (Демпфирование) – позволяет успокаивать движение объекта в сочленении, увеличивая сопротивление силам обратной кинематики. Допустимые значения параметра лежат в диапазоне от 0 до 1, где 1 соответствует наибольшей демпфирующей силе. Успокоение движения позволяет имитировать реальные процессы типа действия сил трения или инерции.

5. Установить в цепочке объектов ограничитель. Если этого не сделать, то анимация распространится на все родительские объекты, даже самый старший из которых не останется в неподвижности. С этой целью выделите в составе цепочки объект, который должен служить ограничителем анимации, и установите флажок Terminator (Ограничитель) в самом верху свитка Object Parameters (Параметры объекта). В результате анимация будет действовать только на объекты, являющиеся дочерними по отношению к ограничителю.

6. Установить флажок Auto Termination (Автоограничение) в свитке Auto Termination (Автоограничение), чтобы решение ИК-цепочки ограничивалось автоматически. Счетчик # of Links Up (Число связей вверх) задает, насколько далеко от начала цепочки будет размещаться ограничитель.

На этом настройка цепочки для применения метода обратной кинематики завершается и наступает время реализовывать собственно анимацию.

Интерактивная анимация объектов по методу обратной кинематики

Чтобы создать анимацию предварительно настроенной цепочки связанных объектов по методу интерактивной обратной кинематики, нужно выполнить следующие действия:

1. Придать всем связанным объектам исходные положения, применяя к ним преобразования перемещения или поворота. Включите режим обратной кинематики, щелкнув на кнопке Interactive IK свитка Inverse Kinematics (Обратная кинематика) командной панели Иерархия. Установите в качестве текущего нулевой кадр анимации. Щелкните на кнопке Animate (Анимация).

2. Установить в качестве текущего кадр, в котором будут созданы ключи анимации. Выберите нужный инструмент преобразования и вручную переместите или поверните младший дочерний объект, располагающийся на конце кинематической цепочки. Наблюдайте за тем, как все родительские объекты вплоть до ограничителя будут изменять свое положение и ориентацию, следуя за младшим дочерним объектом, с учетом ограничений на подвижность сочленений. Для этих родительских объектов будут созданы ключи анимации в текущем кадре.

3. Сменить номер кадра и повторить действия по перемещению или повороту младшего дочернего объекта. Продолжайте настраивать ключи

анимации пока не достигнете конца временного сегмента анимации. Выключите режим анимации, щелкнув на кнопке Animate (Анимация),

4. Воспроизвести анимацию в окне проекции, создать ее эскиз или выполнить визуализацию обычным порядком.

Задание. Анимация настольной лампы по методу интерактивной обратной кинематики.

Отработаем на практике навыки применения методов интерактивной и приложенной обратной кинематики. Использование этих методов требует выполнить настройку свойств отдельных объектов, входящих в иерархическую цепочку, и их сочленений.

Чтобы создать анимацию настольной лампы, выполните следующие действия:

1. Откройте файл Настольная лампа или продолжите работу над файлом, если он уже загружен в программу. Удалите все ключи анимации всех деталей лампы, созданные ранее методом прямой кинематики.

2. Настройте ограничения подвижности вращающихся сочленений рычагов лампы, начав с самого ближнего к отражателю рычага – объекта РычагОЗ. Чтобы правильно выбрать ось вращения этого рычага, выделите его родительский объект – ШарнирОЗ. Выберите в раскрывающемся списке Reference Coordinate System (Система координат) главной панели инструментов вариант Local (Локальная) и выполните команду основного меню Views > Show Transform Gizmo (Проекция > Показывать контейнер преобразования), чтобы стала видна тройка координатных векторов выделенного объекта. Перпендикулярно плоскости шарнира ориентирована его ось Z (если у вас иначе, внесите коррективы). В связи с этим нам необходимо разрешить вращение объекта РычагОЗ только относительно оси Z локальных координат его родительского объекта ШарнирОЗ. Выделите объект РычагОЗ, перейдите на командную панель Hierarchy (Иерархия) и щелкните на кнопке IK. Прокрутите область свитков вверх, чтобы стал виден свиток Rotational Joints (Вращающиеся сочленения). Сбросьте флажки Active (Активно) в разделах X Axis (Ось X) и Y Axis (Ось Y) свитка Rotational Joints (Вращающиеся сочленения), но оставьте установленным аналогичный флажок в разделе Z Axis (Ось Z).

3. Выполните аналогичные действия применительно к объекту РычагО2, также разрешив для него вращение только вокруг оси Z локальных координат его родительского объекта ШарнирО2. Чтобы правильно настроить ось вращения нижнего рычага – объекта РычагО1, – выделите его родительский объект -полусферический ШарнирО1. У этой полусферы ось Z направлена вертикально вверх, а перпендикулярно плоскости рычага направлена ось Y (внесите коррективы, если у вас не так). В связи с этим для объекта РычагО1

следует оставить установленным только флажок Active (Активно) в разделе Y Axis (Ось Y), сбросив флажки активизации осей Z и X

4. Назначьте ограничитель цепочки обратной кинематики. Выделите полусферу, установленную на подставке лампы – объект Шарнир01, – и установите флажок Terminator (Ограничитель) в верхней части свитка Object Parameters (Параметры объекта).

5. Проверьте, действует ли цепочка обратной кинематики. Включите режим обратной кинематики, щелкнув на кнопке Interactive Kinematics свитка Inverse Kinematics. Выберите в раскрывающемся списке Reference Coordinate System (Система координат) главной панели инструментов вариант View (Оконная). Щелкните на кнопке Select and Move (Выделить и переместить), в окне проекции Front (Вид спереди) щелкните на объекте Отражатель и попробуйте перемещать его влево-вправо. В соответствии с правилами обратной кинематики за отражателем, как за самым младшим объектом цепочки, будут тянуться все три рычага и соединяющие их шарниры. Перемещение отражателя лампы в режиме обратной кинематики увлекает за собой все родительские объекты вплоть до терминатора – полусферического шарнира. Закончив «испытания», отмените произведенные перемещения объектов с помощью инструмента Undo (Отменить) и убедитесь, что все объекты лампы находятся в исходных состояниях. Установите в качестве текущего кадр № 0 и включите режим анимации, щелкнув на кнопке Animate (Анимация).

6. Переключитесь на кадр № 25. С помощью инструмента Select and Move (Выделить и переместить) переместите отражатель лампы в окне проекции Front (Вид спереди) сначала вперед, а затем вверх и несколько назад, так чтобы лампа стала «смотреть вперед», выгнув «шею». Если это не получится у вас с первого раза, отмените результат преобразований и повторите попытку.

7. Установите в качестве текущего кадр № 50. Переместите отражатель еще дальше влево по окну проекции, чтобы лампа по-гусиному «закинула голову на спину».

8. Установите в качестве текущего кадр № 75. Теперь сначала сместите отражатель вправо и вверх, чтобы вытянуть всю цепочку рычагов в линию, а затем опустите отражатель вниз, заставив лампу «посмотреть под ноги». Кстати, можно дополнительно повернуть Рычаг01 – все дочерние объекты повернутся вместе с ним, так как в режиме обратной кинематики соглашения прямой кинематики продолжают действовать.

9. Перейдите к последнему кадру сегмента анимации – кадру № 100 – и переместите отражатель так, чтобы все детали лампы заняли положения, близкие к исходным. Это совсем не обязательно, но при многократном повторе анимации обеспечивает более гладкий показ движений без видимых «скачков»

при переходе от кадра № 100 к кадру № 0. Выключите режим анимации. Воспроизведите полученную анимацию в окне проекции Perspective (Перспектива), создайте ее эскиз, а при желании и выполните полную визуализацию.

(Замечание. Принципиальной в данной работе является последовательность действий, а не точное положение объектов в каждом кадре. Вы можете реализовать собственный сценарий движений лампы).

Лабораторная работа 8

Приложенная обратная кинематика

Цель работы: изучить и освоить способы реализации метода обратной кинематики.

Методические рекомендации

Приложенная анимация объектов по методу обратной кинематики

Чтобы создать анимацию предварительно настроенной цепочки связанных объектов по методу приложенной обратной кинематики, нужно выполнить следующие действия:

1. Создать направляющий объект, за перемещениями и поворотами которого будет следовать младший дочерний или иной объект кинематической цепочки. В качестве такого объекта часто используют вспомогательный объект-пустышку (Dummy), который не включается в изображение сцены при визуализации. Выполните анимацию объекта, который будет играть роль направляющего при анимации цепочки по методу приложенной обратной кинематики, обычным методом ключевых кадров. При этом старайтесь заставить его перемещаться так, как в последствии должен будет перемещаться дочерний объект цепочки.

2. Настроить общие параметры метода приложенной обратной кинематики в целях создания ключей всех объектов иерархической цепочки, используя элементы управления свитка Inverse Kinematics (Обратная кинематика). Используйте для настройки следующие параметры свитка:

Apply Only To Keys (Применять только к ключам) – если этот флажок установлен, то ИК-решение будет рассчитываться только для тех кадров, в которых имеются ключи для направляющего объекта. Если флажок сброшен, ИК-решение будет рассчитываться для всех кадров анимации, каждый из которых в итоге станет ключевым кадром;

Update Viewports (Обновить окна) – установка этого флажка обеспечивает последовательное, кадр за кадром, обновление изображений в окнах проекций по мере нахождения ИК-решения;

Clear Keys (Удалить ключи) – установка этого флажка ведет к удалению всех ключей кинематической цепочки до начала расчетов ИК-решения;

Start (Начало), End (.Конец) – два счетчика, позволяющие указать интервал времени анимации, для которого рассчитывается решение задачи обратной кинематики.

3. Привязать выделенный объект цепочки к направляющему объекту. Для этого щелкните на кнопке Bind (Привязать) в разделе Bind to Follow Object (Привязать к направляющему объекту) свитка Object Parameters (Параметры объекта). Кнопка зафиксируется в нажатом положении и подсветится желтым цветом. Щелкните на выделенном объекте иерархической цепочки и перетащите курсор к направляющему объекту. За курсором будет тянуться пунктирная линия. Когда курсор окажется над направляющим объектом, он примет вид канцелярской кнопки. Отпустите кнопку мыши. Направляющий объект на миг выделится, а затем примет обычную окраску. В разделе Bind to Follow Object (Привязать к направляющему объекту) над кнопкой Bind (Привязать) появится, имя направляющего объекта, а вокруг всех объектов цепочки появятся белые габаритные контейнеры, указывающие на формирование ключа анимации для этих объектов в текущем кадре. Чтобы отменить привязку объекта цепочки к направляющему объекту, выделите объект цепочки и щелкните на кнопке Unbind (Отменить привязку).

4. Закончив настройку параметров, щелкните на кнопке Apply IK (Применить IK) в свитке Inverse Kinematics (Обратная кинематика). Это приведет к запуску процесса расчета IK-решения, то есть нахождения необходимых преобразований положения и вращения для всех объектов цепочки. Щелкать на этой кнопке следует только после того, как будут заданы все остальные параметры и ограничения обратной кинематики. В ходе расчета ключей в строке состояния появляется прогресс-индикатор, отображающий ход выполнения расчетной задачи, ползунок таймера анимации перемещается по шкале времени, а в строке треков появляются значки рассчитанных ключей. Дождитесь завершения расчетов и воспроизведите анимацию.

(Замечание. Если запустить процесс решения задачи обратной кинематики раньше, чем будут тщательно настроены ограничения подвижности сочленений всех объектов, составляющих иерархическую цепочку, то результат может оказаться непредсказуемым и, во всяком случае, совсем не таким, на который вы рассчитываете.)

Окончательные итоги:

– Умелое использование контроллеров позволяет разнообразить виды движений (да и не только движений, а любых изменений во времени) объектов сцены за счет использования различных способов управления объектами при анимации;

– для замены любых типов, контроллеров служит окно диалога Track View (Просмотр треков), а контроллеры преобразований положения, поворота и

масштаба можно менять с помощью командной панели Motion (Движение);

– объекты сцены можно связывать друг с другом в цепочки, в которых действуют отношения «потомок–предок» и к которым легко применять анимацию по методу прямой кинематики;

– при использовании метода прямой кинематики любое преобразование, примененное к одному из родительских объектов, распространяется на все связанные с ним дочерние объекты, то есть действует правило: «потомки следуют за предком»;

– в режиме обратной кинематики связанные объекты ведут себя в точности наоборот по сравнению с методом прямой кинематики – преобразование дочернего объекта распространяется на все его родительские объекты, то есть действует правило: «предки следуют за потомком»;

– обратную кинематику можно реализовывать двумя способами: как интерактивную, когда преобразования младшего дочернего объекта производятся вручную, и как приложенную, когда младший дочерний объект связывается с направляющим объектом, анимация которого выполняется простым методом ключей.

Задание 1. Анимация настольной лампы по методу приложенной обратной анимации.

Продолжите работу над файлом из предыдущей лабораторной работы и выполните следующие действия:

1. Создайте объект Dummy (Пустышка) из категории Helpers (Вспомогательные объекты) командной панели Create (Создать). Этот объект создается точно так же, как обычный параллелепипед. Расположите пустышку правее отражателя лампы. Это будет направляющий объект для реализации метода приложенной обратной кинематики. Выполните анимацию перемещений этой пустышки методом ключей, заставив ее двигаться по собственному усмотрению, но примерно так, как должен будет двигаться отражатель. Просмотрите анимацию пустышки в окне проекции, чтобы убедиться в правильности ее настройки.

2. Выделите отражатель лампы и настройте параметры его привязки к направляющему объекту. Для этого в свитке Object Parameters (Параметры объекта) командной панели Hierarchy (Иерархия) установите флажок Bind Position (Привязать к положению) и щелкните на кнопке R (Relatively – Относительно), чтобы заставить отражатель стремиться сохранить свое положение относительно направляющего объекта. Оставьте установленными все три флажка осей координат из группы Axis (Ось). Проследите за тем, чтобы флажок Bind Orientation (Привязать к ориентации) не был установлен. Свяжите отражатель с направляющим объектом. Для этого щелкните на кнопке Bind (Привязать) в разделе Bind to Follow Object (Привязать к направляющему

объекту), перетащите курсор к объекту-пустышке и, когда он примет вид канцелярской кнопки, отпустите кнопку мыши. Убедитесь в том, что связывание произошло: над кнопкой Bind (Привязать) должно появиться имя направляющего объекта – DummyO1.

3. В свитке Inverse Kinematics (Обратная кинематика) установите флажок Update Viewports (Обновить окна), чтобы иметь возможность наблюдать обновление изображений в окнах проекций по мере нахождения ИК-решения, а также флажок Clear Keys (Удалить ключи), чтобы удалить все имеющиеся ключи кинематической цепочки до начала расчетов ИК-решения.

4. Щелкните на кнопке Apply IK (Применить ИК) в свитке Inverse Kinematics (Обратная кинематика). Это приведет к запуску процесса расчета ИК-решения. В ходе расчета ключей в строке состояния будет демонстрироваться прогресс-индикатор, отображающий ход выполнения расчетной задачи, ползунок таймера анимации будет перемещаться по шкале времени, а в строке треков будут появляться значки ключей, рассчитываемых в каждом кадре анимации. Дождитесь завершения расчетов и воспроизведите анимацию в окне проекции. Если движения происходят так, как нужно, выполните визуализацию анимации.

5. Готовый эскиз анимации движений лампы по методу приложенной обратной кинематики сохраните в виде файла Лампа (приложенная ИК).avi.

Лабораторная работа 9

Прямая кинематика. Сравнение интерфейсов

Цель работы: освоить использование прямой кинематики при создании анимации, сравнить интерфейсы различных программ трехмерного моделирования.

Задание. Создать цепь иерархии в меню Персонаж, предварительно выбрав объекты в иерархической цепочке, которые будут связаны между собой кинематикой.

Методические рекомендации: научиться работать с прямой кинематикой на основе родительских и дочерних объектов.

В основе методов прямой и обратной (инверсной) кинематики, а также скелетной анимации лежит создание объектов, связанных друг с другом в иерархические цепочки (системы различных механизмов, взаимодействие частей тела и т.д.).

Иерархия – это набор объектов со связями между ними, при условии, что объекты объединены между собой по принципу «родитель/дочерний объект (потомок)».

Родительский объект – это объект, который контролирует поведение одного или более дочерних объектов (потомков). Некоторые родительские

объекты сами могут контролироваться другими родительскими объектами, имеющими более высокий уровень иерархии.

Дочерний объект (потомок) – объект, контролируемый родительским объектом, который одновременно может оказаться родительским для других дочерних объектов, находящихся по иерархии ниже его. У каждого родительского объекта может быть много дочерних, но всякий дочерний имеет только один родительский. Самый верхний объект иерархии (ни для одного другого объекта он не является дочерним) управляет всеми объектами иерархии и называется корневым, или Root-объектом.

Для понимания устройства иерархических цепочек отлично подойдет объект Фигура из набора примитивов (выполните команду Конвертировать примитив, горячая клавиша «С»).

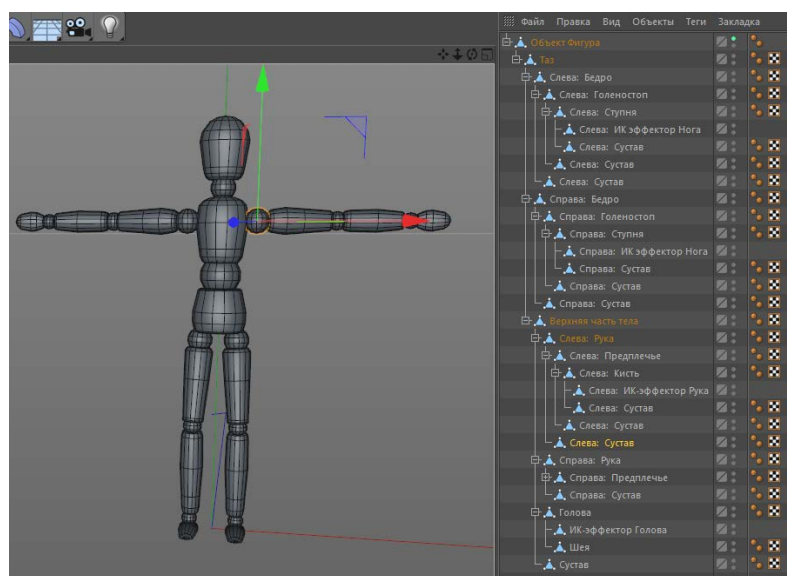


Рисунок 24

Прямая кинематика (FK - Forward Kinematics) представляет собой движение дочерних объектов относительно родительских. При манипуляции иерархическими FK цепочками в результате перемещения родительских объектов перемещаются дочерние объекты.

Перемещению дочернего объекта, например, сучка дерева, положение элемента-родителя (ветки или ствола) остается неизменным. Но если вы перемещаете ствол, все остальные элементы, в том числе ветки и сучки, следуют за ним.

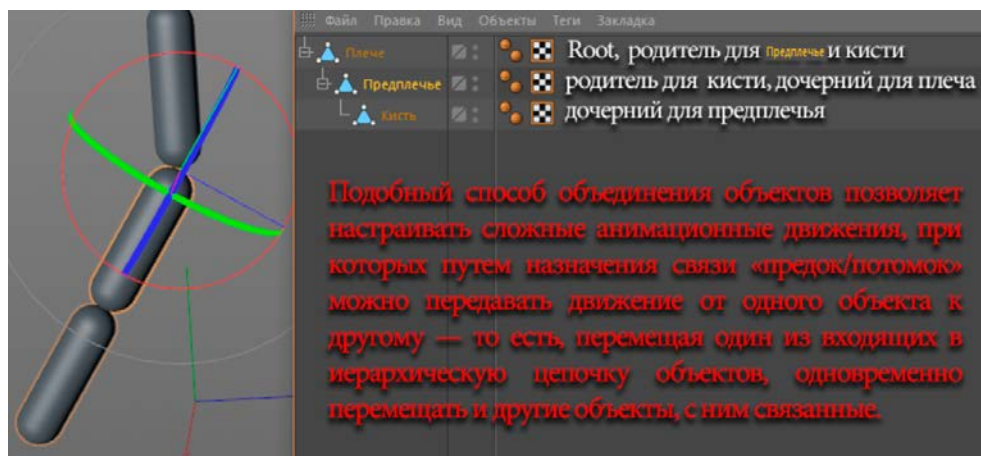


Рисунок 25

Подобно дереву, иерархия должна иметь ствол (или Корень) - единственный объект, который является родителем всех остальных элементов и управляет всеми ветвями иерархии.

При использовании метода прямой (FK) кинематики любое воздействие передается по иерархической цепочке сверху вниз. Если переместить таз персонажа, движется все тело; если повернуть локоть, движется только рука. Чтобы герой взял в руку кофейную чашку, вначале необходимо повернуть плечо, затем локоть, запястье и пальцы, перемещаясь по иерархической цепочке сверху вниз. В результате каждого поворота рука будет приближаться к чашке. Нельзя просто переместить руку к чашке - она не выполнит это движение.

В примере ниже, чтобы наклонить «голову», нужно последовательно (или нет) повернуть все входящие в «персонаж» части. Для правильного вращения частей необходимо, чтобы опорные точки (центр трансформации) находились в центрах вращения.

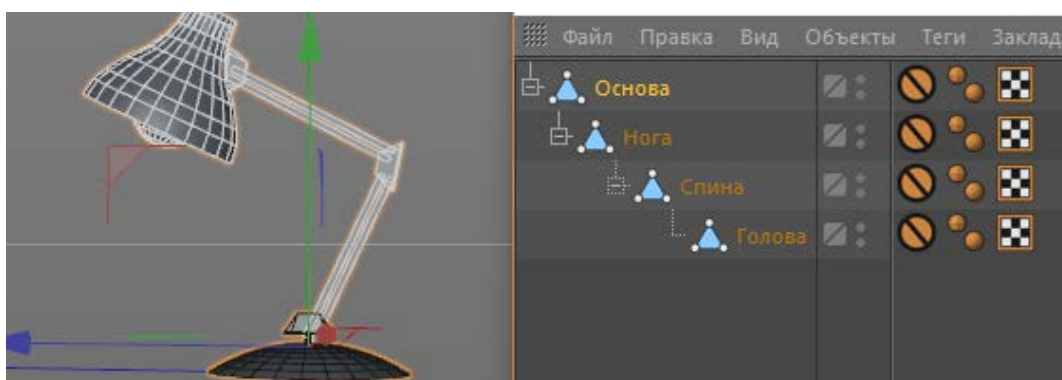


Рисунок 26

Тег защита (опционально)

Для удобства работы при анимации, а также с целью исключить ошибки, можно использовать тег Защиты. Тег находится в группе общих тегов Cinema 4D и назначается также, как и другие – ПКМ по объекту в МО.

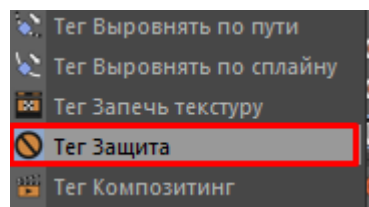


Рисунок 27

Тег Защита предназначен для защиты объектов от случайного изменения на предмет их позиции, геометрических размеров и углов вращения. Этот тег также используется для защиты камеры после окончательной настройки её параметров и позиции.

Дополнительно тег Защита может производить ограничение определённых параметров ПМВ. Например, интерактивное перемещение в окне редактора программы, может быть ограничено только для оси Y.

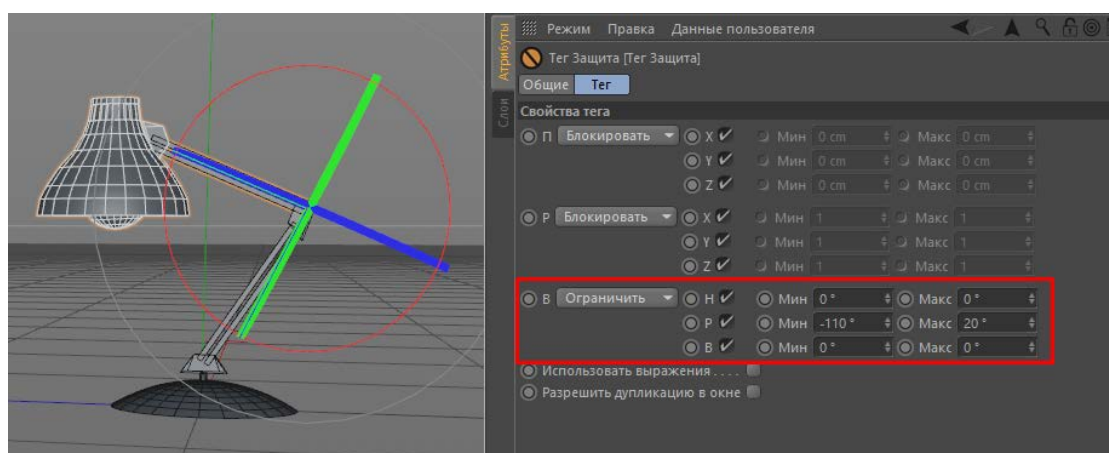


Рисунок 28

В данном примере этот тег используется для блокировки осей вращения, по которым вращение не будет производиться. Кроме того, для осей, по которым предполагается вращение – устанавливается ограничение угла от и до, с целью избежать неестественных поворотов (так, например, у персонажа ограничивают углы вращения в локтевых и прочих суставах, так как руки в локтях не сгибаются в обратную сторону, если их не сломать, что тоже имеет место в персонажной анимации).

Лабораторная работа 10

Инверсная кинематика. Сравнение интерфейсов

Цель работы: освоить использование ИК инверсной кинематики при создании анимации, сравнить интерфейсы различных программ трехмерного моделирования.

Задание. Создать цепь ИК в меню Персонаж, предварительно выбрав объекты в иерархической цепочке, которые будут связаны между обратной кинематикой.

Методические рекомендации: используя перемещение корня в конце цепочки привести в движение всей иерархии.

Обратная кинематика (ИК - Inverse Kinematics) – прямо противоположна прямой кинематике, поскольку перемещение дочерних объектов приводит к изменению положения родительских объектов. Он используется для создания плавного движения различных связанных и зависимых друг от друга систем объектов. Так, например, Вы перемещаете кисть героя к кофейной чашке, и остальные части руки автоматически следуют за ней.

Один из недостатков метода инверсной кинематики заключается в том, что в распоряжении аниматора имеется лишь один концевой элемент управления (называемый, случае Cinema 4D - Цель) для всей иерархической цепочки, с помощью которого контролируется движение каждого элемента цепи. Если анимации подвергаются конечности, имеющие по два элемента (сустава, в случае скелетной анимации), например, ноги или руки человека, возможно, особых проблем не будет. Однако при работе с конечностями, имеющими множество суставов (такими как хвост), контроль над формой объекта утрачивается. Вероятно, в подобной ситуации лучше выбрать метод прямой кинематики.

Первый элемент в цепочке называется корневым (Root), или Корнем иерархии. При его перемещении движется вся иерархическая цепочка. Создавая ее, всегда начинайте с самого верхнего сустава иерархии. Например, если вы конструируете ногу, начинайте с бедра, а затем переходите к лодыжке и стопе.

Последний элемент (при необходимости любой другой, который является элементом Цель) цепочки управляет положением конца иерархической цепочки инверсной кинематики. Программа всегда стремится расположить элементы цепочки таким образом, чтобы они находились между Корнем и Целью.

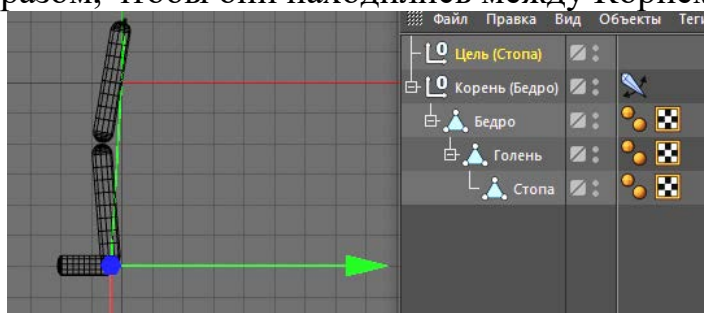


Рисунок 29

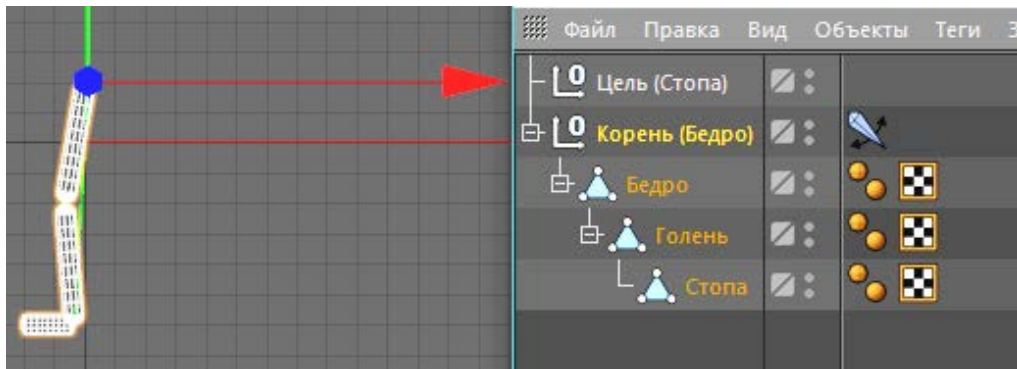


Рисунок 30

Манипулирование цепочкой осуществляется объектом Цель. При его перемещении соответственно поворачиваются суставы. Данное свойство позволяет легко придавать различные позы персонажу и выполнять его анимацию, поскольку разработчик должен следить только за положением эффектора. Возможности по управлению, в сравнении с методом прямой кинематики, уменьшаются, но зато упрощается процесс анимации, который теперь напоминает работу с марионетками.

Перемещение Корня на другом конце цепочки обычно приводит к движению всей иерархии, включая объект Цель в случае если он является дочерним Корня. Например, если цепочка инверсной кинематики представляет собой модель ноги персонажа, корнем является бедро, а Цель - стопа. При перемещении верхушки двигается вся нога. Если необходимо, чтобы в подобной ситуации стопа оставалась неподвижно стоящей на полу, скелет следует строить иначе – объект Цель не должен быть дочерним объекту Корень.

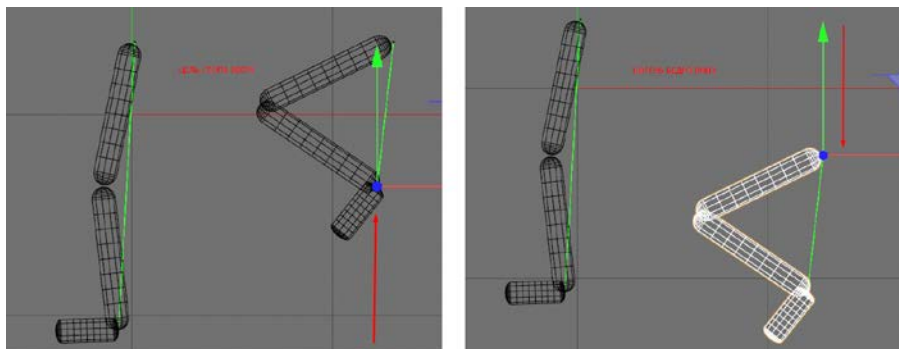


Рисунок 31

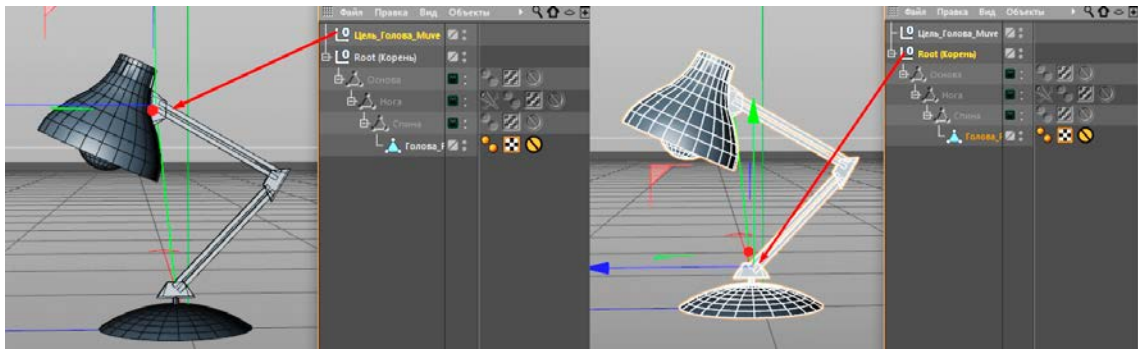


Рисунок 32

При методе инверсной кинематики процесс анимации происходит быстрее. В то же время прямая кинематика очень проста в применении. Поэтому выбор типа кинематики зависит от того, анимация какой части скелета будет выполняться.

Метод прямой кинематики чаще всего используют для верхней части туловища: позвоночника, плеч и предплечий. Он весьма удобен, если персонаж просто идет или стоит на одном месте и лишь жестикулирует – его руки при этом не взаимодействуют с другими объектами (пол, дверь, стена, чашка и т.п.).

Если же герой, к примеру, открывает дверь, отжимается от пола, толкает гирию или снимает и бросает шляпу или каким-то образом общается с другими действующими лицами, может потребоваться метод инверсной кинематики. При этом не исключается прямая кинематика для анимации позвоночника и кистей рук.

Почти всегда инверсная кинематика применяется для анимации ног и стоп (взаимодействие с поверхностью), особенно если необходимо показать, что стопы персонажа стоят неподвижно на земле. Однако если герой плывет, возможно, придется использовать приемы прямой кинематики не только для анимации рук, но и для ног и стоп. Как видите, выбор подходящего метода зависит от конкретных обстоятельств. Поэтому часто подготавливают несколько вариантов начальных установок для одного и того же персонажа.

Построение ИК цепи

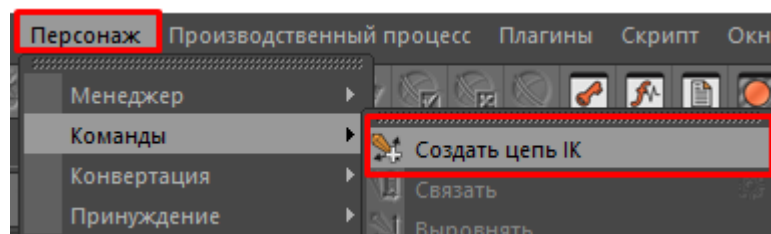


Рисунок 33

Для построения цепочки обратной кинематики необходимо выполнить команду Создать цепь ИК в меню Персонаж, предварительно выбрав объекты в иерархической цепочке, которые будут связаны между обратной кинематикой.

При этом нужно учитывать, что самый нижний по иерархии выбранный дочерний объект станет местом положения объекта Цель – объектом, перемещение которого будет влиять на все родительские объекты в иерархии вплоть до объекта Корень (Root).

Самый верхний выбранный Вами объект – станет местом положения объекта Корень (Root) – объект, который будет влиять на всю цепочку сверху вниз.

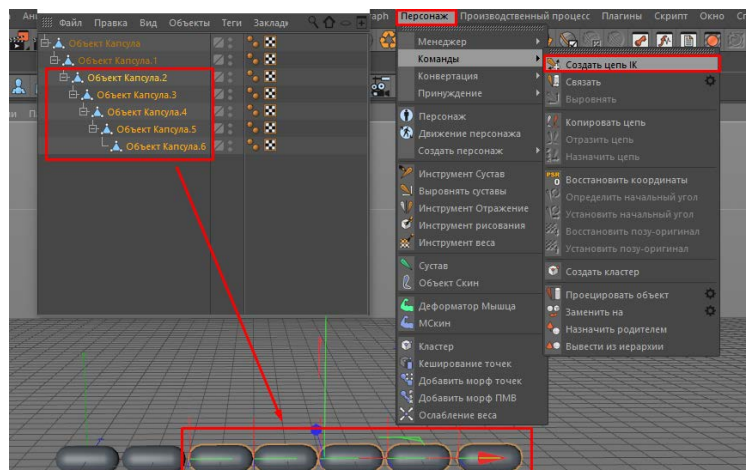


Рисунок 34

Если есть необходимость в том, чтобы объект Цель не оказывал влияние на некоторый участок цепи (объекты, расположенные вверху или (и) внизу иерархии), достаточно не выделять эти объекты перед выполнением команды Создать цепь IK.

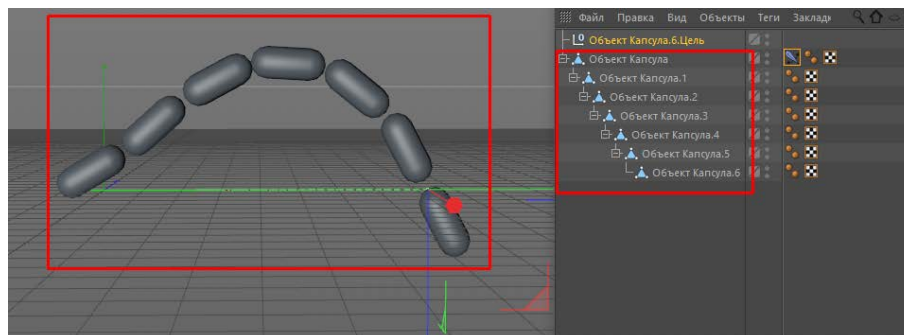


Рисунок 35

В случае с Лампой, при ее анимации будет необходимо вращать ее основание во время прыжка. Если поместить все объекты в цепь IK, то сделать это будет невозможно:

- объект, на который назначается тег IK (Корень) не поддерживает вращение
- кроме того, невозможно повернуть саму геометрию этого объекта – она входит в часть цепи и влияние на нее могут оказывать только объекты Цель и Корень

– нежелательное вращение основания Лампы при перемещении объекта Цель вследствие того же влияния.

Выходом из ситуации станет исключение из цепи ИК объекта Основа. В этом случае на него не будет оказывать влияние объект Цель, плюс его можно будет вращать при анимации.

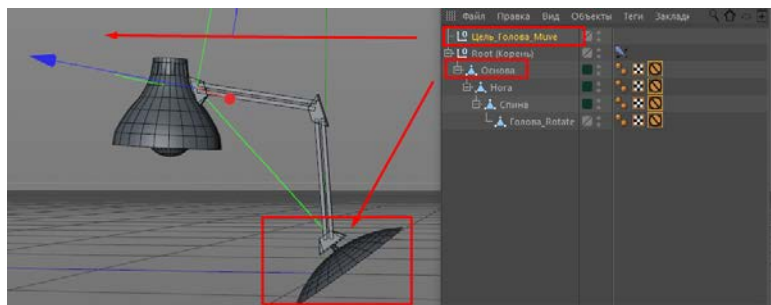


Рисунок 36

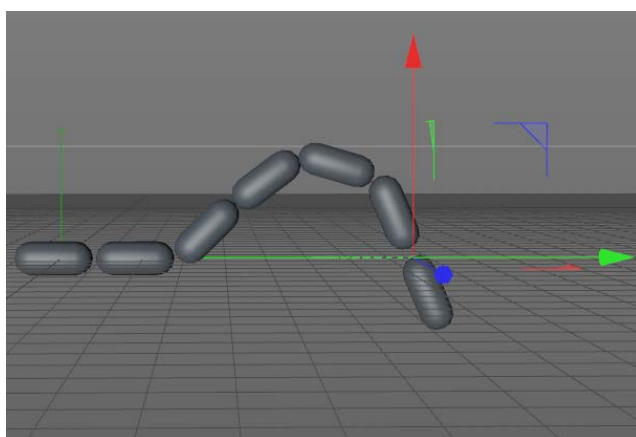


Рисунок 37

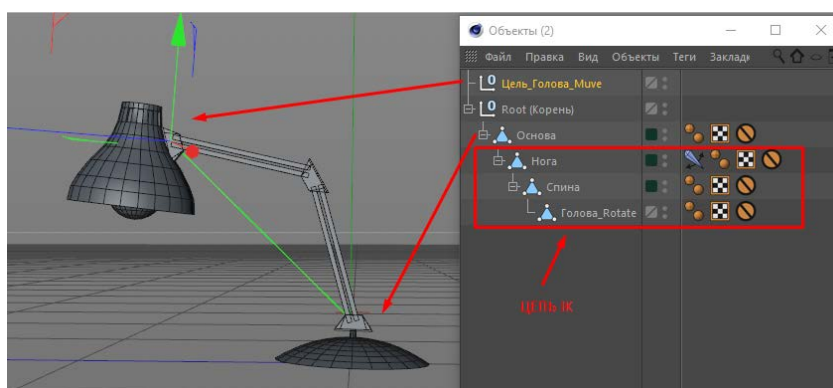


Рисунок 38

В результате получаем настроенный для анимации объект, у которого есть два управляющих элемента:

- объект Цель, конечный в иерархической цепи объект, управляющий «головой» лампы. При этом Основа остается на месте неподвижной.

- объект Корень - объект, управляющий нижней частью лампы при перемещении с возможностью вращения вокруг собственного центра без влияния на всю цепь ИК.

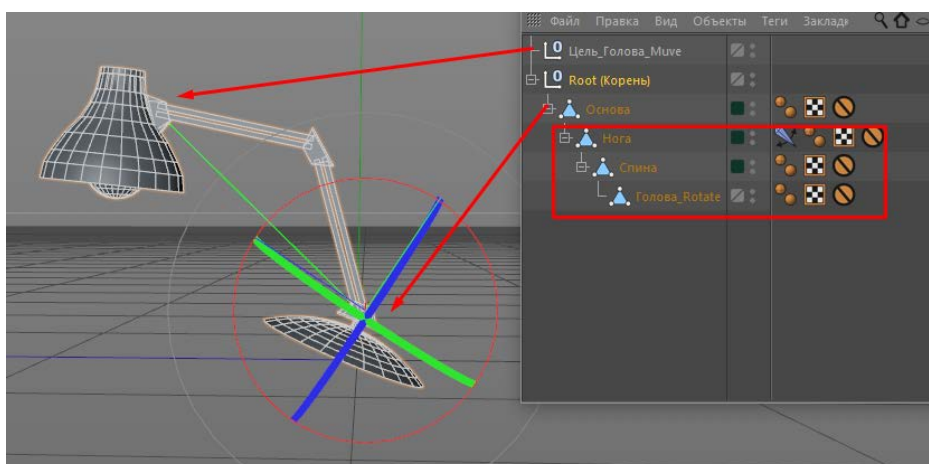


Рисунок 39

В цепи ИК объект Цель не поддерживает вращение, поэтому вращение «Головы» Лампы будет осуществляться непосредственным вращением геометрического объекта Голова.

В данных примерах были рассмотрены способы создания FK/IK цепи модели, состоящих из отдельных частей. Осуществляется это путем простого создания правильной иерархии отдельных объектов и назначении соответствующего FK/IK тега.

В случае, когда модель (персонаж) представляет собой единую геометрию (единый Меш, объект), необходимо дополнительно создавать скелет в виде иерархии суставов (риггинг) с последующей привязкой скелета к геометрии (скиннинг). Эти приемы настройки (Setup) персонажа в данном курсе не рассматриваются, т.к. это тема отдельного курса, наравне с персонажной анимацией.

Анимация объектов в режиме ИК

Любая анимация (более-менее сложная) состоит из нескольких этапов.



Рисунок 40

Планирование – это получение четкого представления о том, что вы хотите видеть в будущей анимации, перед началом непосредственной работы. Независимо от того, какой процесс анимации: 3D, 2D, кукольная анимация, stop motion анимация, самый быстрый способ планировать анимацию - делать быстрые и информативные наброски (скетчи).

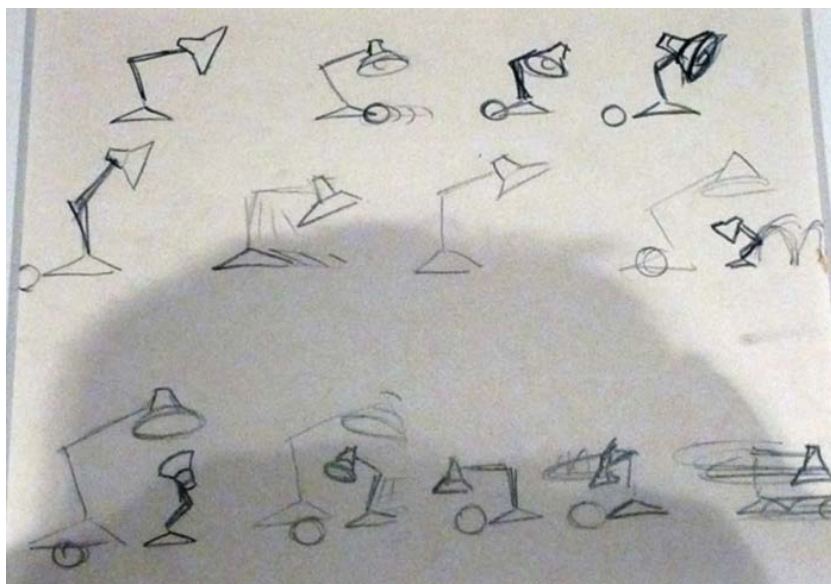


Рисунок 41

Правила этапа планирования:

– вы должны знать, что и кого вы анимируете. Досконально. Нужно знать, как может двигаться этот сустав, как поворачивается голова и т.п. В должны знать все о персонаже, даже если это обычный мячик. Даже если это самый простой объект, то с ним отрабатываются все тонкости и правила анимации (12 законов Диснея).

– вы должны знать всю информацию по таким понятиям, как Action/Acting (Действие/актерская игра). (в идеале)

В основу анимации лег короткометражный мультфильм Люксо-младший, созданный на студии Pixar в 1986 году. Этот мультфильм является первой компьютерной анимационной картиной, номинированной на высшую кинопремию.

Блокинг – это одна из техник анимации, при которой создаются наиболее важные для сюжета ключевые позы, чтобы выставить расположение персонажа или объекта в пространстве и обозначить, как он будет двигаться в сцене.

Эта техника используется на ранней стадии процесса анимации и помогает рассказать “историю” анимации. На данном этапе прорабатывается тайминг – расчет времени на выполнения действий. Аниматор, должен умело использовать тайминг в анимации, так как можно один и тот же жест показать с разным смыслом. К примеру взмах руки может быть энергичен и полон решительности, но если увеличить время на выполнение взмаха, то он будет выглядеть как вялый и вельможный. В результате получится совсем не то, что требовалось. Поэтому необходимо внимательно следить за таймингом.

Ключевые кадры необходимо ставить на управляющих объектах – Цель, Корень (+ голова для вращения).

Позинг. (Средняя поза, Breakdown – разбивка на составляющие). Средняя поза показывает переход персонажа или объекта из одной ключевой позы в

другую. Среднюю позу можно считать промежуточным этапом, однако он имеет свои особенности. Когда у вас стоит одна поза на первом кадре, а другая на десятом, следующим вашим шагом будет создание раскрывающей средней позы посередине этого временного отрезка.

Полировка – финальный этап анимации, на котором прорабатывается спейсинг, контакты, сквозные движения, эмоции, дуги. Спейсинг (от англ. spacing “интервал, промежуток”) – это определение интервалов между ключевыми кадрами /анимационными ключами на линейке времени. В зависимости от способа расстановки ключей, физика движения объекта может меняться. Начиная с определенного кадра скорость движения может замедляться при увеличении интервала между ключевыми кадрами или ускоряться при более плотной их расстановке. Также объект может двигаться равномерно, если ключи на линейке выставлены с одинаковым интервалом.

Тема 3. Динамические взаимодействия

Лабораторная работа 11

Моделирование ткани

Цель работы: изучить имитацию действия силы тяжести.

Задание. Моделирование скатерти.

Методические рекомендации

1. Для этого давайте построим примитивную модель стола и скатерть. В качестве стола используйте примитив Cylinder (Цилиндр), радиусом 60 см и высотой 3 см. с количеством сторон от 20 до 40. Для скатерти подойдет примитив Plane (Плоскость), с размерами 200 на 200 и количеством сегментов по длине и ширине по 50. Для тех объектов, которые будут выступать в роли тканей, старайтесь не превышать 3000 полигонов для их общего количества, в противном случае возможны сбои в работе программы.

2. Понятно, что скатерть должна иметь больший, нежели стол размер, чтобы покрыть его и достаточно большую плотность сетки для того, чтобы складки, образованные свободно свисающими краями, выглядели естественными. Кроме того, расположите эти два объекта так, чтобы скатерть находилась на некотором расстоянии выше стола и была в окне проекции вида сверху ровно посередине его. Выровняйте ее по центру цилиндра командой Align.

3. Сейчас нужно указать программе роли, в каких телах взаимодействия будут выступать наши объекты. Для этого выполните следующие действия:

- выделите «стол»;
- в меню Animation выберите Mass FX – Rigid Bodies – Set selected as

static rigid body;

- выделите «скатерть»;
- в меню Animation выберите Mass FX – Cloth – Create mCloth;
- запустите анимацию Mass FX – Simulation – Play simulation;
- если «не хватило кадров», увеличьте их, Reset simulation и заново.
- превратим получившуюся скатерть в меш Tools – Snapshot.

4. Модификатор Cloth Modifier (Ткань) задает определенные параметры объектов, скажем, массу и жесткость. Объекты, которым назначается этот модификатор, должны иметь открытые ребра – хорошим примером является объект-примитив Plane (Плоскость) – и достаточное количество вершин для создания деформации.

5. Поэкспериментируйте: в свитке Properties (Свойства) модификатора mCloth (реактор, Ткани) выставьте значение 0.2 для Damping (Затухание); выберите в качестве стола параллелепипед...

6. Выберите понравившийся кадр и воспользуйтесь инструментом Snapshot (Снимок) для создания копии скатерти, зафиксированной в данный момент анимации. Объекты, полученные таким способом, уже не являются анимированными и не подвергаются обсчету со стороны модуля MassFX. Они представляют собой редактируемые сетки, которые сложно было бы получить другим путем. «Отодвиньте» объект-сетку в сторону.

7. Поверх модификатора mCloth (Ткань) можно назначить модификатор MeshSmooth (Сглаживание сетки) или Tessellate (Разбиение). В этом случае для получения более детальной деформации не потребуются дополнительные вычисления MassFX.

8. Сохраните файл под именем tablecloth.

Лабораторная работа 12

Моделирование силы тяжести, упругости и трения

Цель работы: изучить принципы работы с твердыми и мягкими телами, освоить настройки физических свойств.

Методические рекомендации

В выбранной для MassFX модели расчета все тела рассматриваются как твердотельные (Rigid Body), т.е. не подвергаются деформации в процессе просчета. Различают три вида таких тел.

Dynamic. Тела Dynamic rigid подобны объектам в реальном мире. Они попадают под действием гравитации, испытывают силы трения при взаимодействии с другими телами и могут быть сдвинуты другими телами. Физические свойства объекта спроецированы на расчетную среду и расчет ведется с учетом формы объекта. Тела с вогнутыми поверхностями не поддерживаются (если объект расчета представить в виде одного тела, а не

нескольких, но об этом ниже). В случае с вогнутой поверхностью реальная форма объекта приводится к какой-то идеализированной геометрической форме.

Kinematic. Тела Kinematic rigid подчиняются анимационным параметрам. Они не падают под действием силы гравитации. Эти тела могут двигать другие динамические объекты, но сами не могут быть сдвинуты другими телами. Геометрия объекта контролируется 3ds Max (с анимацией или без) для изменения геометрии объекта. Тела с вогнутыми поверхностями не поддерживаются.

Static. Тела Static rigid подобны телам Kinematic rigid за исключением того, что они не могут быть анимированы. Если тело Dynamic rigid, например, врежется в Static rigid то произойдет взаимодействие, например отскок, но тело Static rigid никак не среагирует на взаимодействие. Тела Static rigid полезны для оптимизации производительности, а также потому, что с ними можно корректно просчитывать вогнутую поверхность.

Таким образом, следует различать геометрическую поверхность и физическую поверхность. Для физической поверхности создается отдельный объект, который можно изменять отдельно от геометрической модели. Например, если рассматривать стол, то с точки зрения геометрии это плоскость с четырьмя ножками, между ножками свободное пространство. Что-то может закатиться под стол. Если же представить стол как Rigid Body, то получится, грубо говоря, куб (помним, что не поддерживается вогнутая геометрия). И просчитать корректно мячик катящийся под столом не получится. Для таких случаев используют композитное представление объекта. Так, стол можно представить как 5 разных простых объектов – столешницы и четырех ножек, что позволяет производить более-менее корректный расчет в рамках модели Rigid Body.

Задание. Анимация прыгающего мяча.

1. Создание прыгающего мячика с помощью MassFX в 3ds max.

Создаем поверхность «земли». Простой бокс, на котором будет прыгать мячик. Create>Standard Primitives>Box с параметрами: Length (Длина): 200; Width (Ширина): 200; Height (Высота): 5. Параметры могут быть разные, т.е. какие вы захотите. Далее делаем Сферу или Геосферу, параметры тоже могут быть разные, например: Radius (Радиус): 25.

2. Теперь нам надо сделать, так, чтобы мячик прыгал на земле (земля это Box, а мячик это Sphere или GeoSphere). Для того, чтобы мячик прыгал, ему и земле нужно присвоить физические свойства.

Выделите «землю»; в меню Animation выберите Mass FX – Rigid Bodies – Set selected as static rigid body;

Выделите «мяч»; в меню Animation выберите Mass FX – Create mSoft Body. Можно добавить в коллекцию ещё один Box, чтобы мячик прыгал с одного на другой. Все дальнейшее зависит от вашей фантазии и варьирования параметров.

3. В меню Animation запустите анимацию Mass FX – Simulation – Play simulation; если «не хватило кадров», увеличьте их, Reset simulation и заново.

Видим, как мячик падает на землю, но он не прыгает, и не лежит на земле (т.е. висит очень близко над землёй), сейчас мы это исправим.

4. Выделяем мячик и редактируем Properties. Там мы видим такие параметры как Mass (Масса), Stiffness (Упругость), Damping и Friction (трение).

Параметры могут быть такие: Mass: 2 (кг). Stiffness: желательно побольше, 6. Damping: как можно меньше, 0. Friction: 0.

Во время просмотра анимации (симуляции), мячик будет крутиться, и при прикосновении мячика с землёй происходит трение, т.е. если поставить трение на максимум (1), то мячик будет отскакивать от земли, как и в реальности, и прыгать он будет не на одном месте, мы поставили 0 для того, чтобы мячик прыгал на одном месте, т.е. трение между мячиком и землёй не происходит.

Включаем нашу анимацию (Preview Animation), и видим, что мячик стал прыгать, но с землёй он по прежнему не сталкивается. Для того, чтобы исправить это, нужно пойти в меню Utilities (на этом меню изображён молоток), потом нажать кнопку “Reactor” и открыть подменю World, изменить значение “Col. Tolerance” на самое низкое, 0.2. Теперь мячик отскакивает от земли.

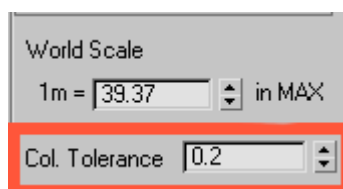


Рисунок 42

5. Сохраните файл под именем ball.

Лабораторная работа 13

Динамика жидкости

Цель. Изучить способы создания и динамики жидкостей.

Методические рекомендации

Универсальный объект Space Warp под названием Motion Field сочетает в себе гравитацию, ветер, шум/турбулентность, сопротивление, а также может деформировать геометрию. Кроме того, обновленный плагин рендеринга MAXtoA Arnold поддерживает рендеринг жидкостей, добавляет поддержку облака точек и устраняет дефекты.

Моделирование жидкости состоит из следующих компонентов:

Излучатель : определенная форма, которая непрерывно испускает частицы жидкости. Вы можете выбрать из стандартных эмиттеров, таких как сферы, коробки и плоскости, или выбрать пользовательский объект в качестве формы.

Контейнер : тип эмиттера, который содержит фиксированное количество жидкости в начале симуляции. Вы можете выбрать один из стандартных контейнеров, таких как сферы, коробки и плоскости, или выбрать пользовательский объект в качестве формы.

Решатель : механизм, который вычисляет симуляцию на основе текущих параметров и связанных объектов. Вы можете использовать несколько решателей в каждом моделировании жидкости, чтобы помочь настроить результаты, не теряя результатов предыдущих итераций.

Коллайдер : объект, настроенный на столкновение с жидкостью, разбрызгивание и/или изменение направления жидкости.

Пена : частицы, имитирующие пузыри, пену и брызги.

Направляющая : система, которая создает жидкую поверхность, такую как часть океана, используя полигональную сетку или симуляцию с низким разрешением. Это ограничивает более высокую детализацию верхним поверхностным слоем, чтобы уменьшить требования к памяти и вычислениям.

Kill plane : поверхность, которая удаляет частицы из симуляции, как только они ее пересекают. Kill planes полезны для предотвращения ненужных вычислений частиц, которые больше не видны.

Оптимизация производительности

При работе над любой симуляцией жидкости цель состоит в том, чтобы как можно быстрее создать визуально приятный результат. Для этого важно понимать доступные варианты балансировки деталей и точности с памятью и временем вычислений.

Основной настройкой для управления деталями симуляции является параметр Master Voxel Size в группе Solver Properties на панели Solver Parameters. Этот параметр также оказывает большое влияние на точность. Как

правило, вы бы использовали более высокое значение для быстрого и менее подробного предварительного просмотра вашей работы, а затем более низкое значение для более детальной симуляции, когда вы удовлетворены эффектом.

При любом размере основного вокселя атрибуты Transport и Time Stepping в группе Simulation Parameters панели Solver Parameters могут повысить точность моделирования. Это, однако, потребует дополнительных вычислений и займет больше времени.

Панель «Создать» > «Геометрия» > «Жидкости»

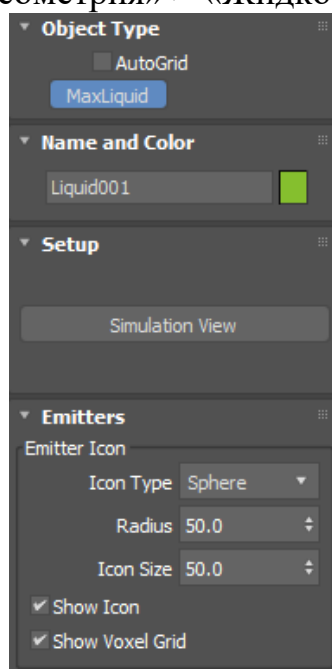


Рисунок 43

Ниже приведен стандартный рабочий процесс для создания моделирования жидкости:

На панели «Создать» выберите «Геометрия» > «Жидкости», затем нажмите кнопку «MaxLiquid».

В свитке Emitters в разделе Icon Type выберите тип используемого эмиттера: Sphere, Box, Plane или пользовательскую форму.

Отрегулируйте параметры по мере необходимости.

Перетащите в окне просмотра, чтобы разместить и изменить размер логотипа Fluids, который показывает жидкий объект. Объект добавляется в сцену, а также отображаются излучатель и воксельная сетка (если она включена).





В свитке Setup нажмите Simulation View, чтобы открыть окно Simulation View.

В окне просмотра моделирования:

Используйте панель «Настройки отображения», чтобы указать, как отображать жидкости и пену в окне просмотра.

Используйте панель Liquid Attributes, чтобы настроить излучатель и добавить коллайдеры, плоскости уничтожения, пенные маски и поля движения. Используйте эту панель, чтобы также создать систему направляющих, если она подходит.

Используйте панель Solver Parameters для настройки свойств решателя, созданного для жидкости. Настройте параметры моделирования, такие как масштаб и размер вокселя, а также параметры жидкости, такие как поверхностное натяжение и вязкость. Вы также можете настроить дополнительные параметры для объектов и помощников, участвующих в моделировании.

В области Management System запустите решатель,  чтобы вы могли просмотреть симуляцию и внести дополнительные корректировки. Вы можете решить жидкий компонент , пенный компонент , сетку  или любую их комбинацию. Как правило, вы сначала решаете жидкость, пену и, наконец, сетку, как только вы готовы к рендерингу.

Добавляйте и клонируйте решатели, чтобы экспериментировать с симуляцией и устранять такие проблемы, как утечка частиц. По мере улучшения симуляции вы можете уменьшить размер вокселя, чтобы увидеть предварительный просмотр с более высоким разрешением и внести окончательные настройки.

Когда вы будете удовлетворены результатом, используйте панель Render Settings, чтобы установить параметры перед визуализацией моделирования.

Вы можете остановить решение в любое время или приостановить моделирование, а затем возобновить его, когда будете готовы.

Создание пены

Создавайте эффекты пузырьков, пены и распыления, добавляя частицы пены к моделированию жидкости.

Пена может образовываться со всеми типами жидкостей, а также может образовываться путем выделения жидкости в уже существующую жидкость с другой плотностью, например, для создания эффекта джакузи.

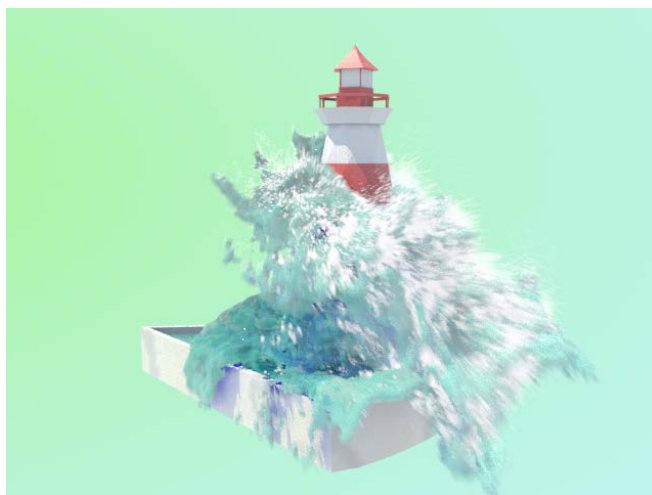


Рисунок 44

Частицы пены непрерывно выделяются в жидкости с исходным значением плотности, которое со временем рассеивается. Плотность частиц пены и скорость диссипации зависят от близости частиц к поверхности жидкости:

Пена создается частицами на поверхности с плотностью 1 и рассеиванием, основанным на значении скорости рассеивания.

Пузырьки создаются частицами под поверхностью с плотностью 1 плюс значение максимальной глубины жидкости. Эти частицы наследуют установленное значение плавучести и не рассеиваются.

Распыление производится частицами над поверхностью с плотностью менее 1. Эти частицы рассеиваются быстрее, чем установленная скорость рассеивания.

Процедура


Для создания пены:

Создайте симуляцию жидкости.

Откройте окно просмотра моделирования.

На панели Solver Parameters настройте параметры пены, чтобы получить желаемый результат.

На панели «Параметры отображения» задайте любые параметры предварительного просмотра пены в окне просмотра.

В области управления системой включите компонент пены  и запустите решение.

На панели Render Settings выберите способ визуализации пены.

Использование коллайдеров

Коллайдеры действуют как препятствия для потока симуляции. Вы можете использовать коллайдеры для создания барьеров, бассейнов или волн и брызг.

Панель «Создать» > «Геометрия» > «Жидкости» > «Настройка» > «Вид моделирования» > панель «Атрибуты жидкости»

Добавление и удаление коллайдеров

Чтобы добавить коллайдер в симуляцию:

На панели «Создать» создайте объект для использования в качестве коллайдера в сцене.

Выберите объект Liquid, а затем нажмите Simulation View в свитке Setup, чтобы открыть окно Simulation View.

На панели Liquid Attributes в разделе Colliders/Kill Planes > Add Colliders нажмите Pick, чтобы выбрать объект в окне просмотра, или Add, чтобы выбрать объект в диалоговом окне Add Colliders.

Задание. Создайте поверхность воды с волнами. Добавьте пену. Сохраните файл и предъявите преподавателю.

Тема 4. Визуализация анимированной сцены

Лабораторная работа 14

Анимация камеры

Цель работы: изучить виды камер, их параметры; научиться анимировать камеру.

Методические рекомендации.

Контроллер LookAt

Контроллер трансформации LookAt объединяет вывод из контроллеров положения, вращения (тип данных Float) и масштаба. Контроллер LookAt вращает объект так, чтобы негативная локальная ось Z всегда указывала на точку вращения другого объекта. Параметр вращения задает угол поворота объекта вокруг локальной оси Z.

Целевой объект, за которым необходимо следить, определяется в качестве параметра контроллера LookAt. Целевой объект контроллера LookAt можно увидеть и отобразить только в разделе Parameters панели команд Motion.

Задание. Облет объекта.

1. Откройте файл с текстурированным объектом. Активизируйте окно проекции Perspective (Перспектива).

2. Щелкните по слову Perspective и в раскрывшемся списке выберите Show Safe Frames. В окне вы увидите установленные размеры кадра в параметрах Render Setup. Можете их поменять.

3. В окне Top создайте нацеленную камеру. Поменяйте в окне Перспектива вид на вид с камеры (клавиша C). Перемещая камеру и линию визирования камеры, добейтесь композиционно выставленной картинки.

4. В окне Top нарисуйте окружность с центром на объекте и радиусом до

камеры, поднимите ее на высоту камеры.

5. Выделите камеру и привяжите ее к пути меню Animation – Constraints – Path constraint. Просмотрите анимацию.

6. Сохраните сцену в своей папке под именем oblet.

Лабораторная работа 15

Рендеринг анимированной сцены. Рендер Arnold

Цель работы: изучить параметры рендера Arnold.

Методические рекомендации.

Arnold Renderer - одна из наиболее широко используемых программ 3D-рендеринга для визуализации 3D фильмов. Она принадлежит компании Autodesk, которая использует ее в качестве стандартного рендера в своем программном обеспечении 3D.

Чтобы улучшить качество картинки при рендеринге с помощью Arnold, можно изменять следующие параметры:

Sample settings (Настройки сэмплирования): увеличение числа сэмплов (samples) для primary и secondary rays может уменьшить шум и улучшить качество изображения. Однако, это может также увеличить время рендеринга, поэтому необходимо балансировать между качеством и скоростью.

Ray depth (Глубина лучей): увеличение глубины лучей (ray depth) может улучшить качество теней, преломлений и отражений. Однако, слишком большая глубина может также увеличить время рендеринга и потребление памяти.

Sampling quality (Качество сэмплирования): изменение параметров качества сэмплирования, таких как contrast threshold и variance threshold, может помочь уменьшить шум и улучшить качество изображения.

Filter settings (Настройки фильтрации): использование различных типов фильтров, таких как Gaussian или Catmull-Rom, может помочь улучшить качество изображения и сгладить шум.

Light settings (Настройки света): изменение настроек освещения, таких как мощность и цвет света, может помочь улучшить качество и настроить нужное освещение.

Camera settings (Настройки камеры): изменение настроек камеры, таких как угол обзора, фокусное расстояние и диафрагма, может помочь улучшить глубину резкости и перспективу.

Texture settings (Настройки текстур): использование более высокого разрешения текстур и настройка параметров фильтрации текстур может помочь улучшить качество текстур.

Задание. Создайте сцену в программе трехмерного моделирования, содержащую простое здание (например, дом) и окружающую среду. Погода –

дождь. Визуализируйте с применением рендера Arnold. Длительность анимации – 10 сек.

1. Настройте освещение сцены, используя светильники Arnold и регулируя их параметры для достижения нужного эффекта. Можно использовать различные типы светильников, такие как точечные, направленные или прожекторные, чтобы создать интересные эффекты света и тени. Светильники от ветра раскачиваются.

2. Создайте материалы для здания и окружающей среды, используя стандартные текстуры Arnold.

3. Настройте параметры рендеринга Arnold, такие как разрешение изображения, глубина поля (depth of field), экспозиция и т.д. Проверьте настройки рендеринга на предмет возможных ошибок.

4. Запустите рендеринг Arnold и дождитесь окончания процесса.

5. Оцените результат и внесите необходимые изменения в настройки сцены, освещения, материалов или рендеринга, если это необходимо.

6. Выполните рендер сцены и сохраните в файле Rain.

Тема 5. Создание анимационного ролика 8 часов

Лабораторная работа 16 Анимация системы частиц

Цель работы: изучить настройки и способы задания анимации систем частиц.

Методические рекомендации

Система частиц (Particle System) представляет собой совокупность малоразмерных частиц, с использованием которых можно создать множество анимационных эффектов. Частицы представляют собой средства анимации большой группы сходных объектов по определенным процедурным правилам. Подобно твердым телам частицы имитируют физические процессы. При этом они не являются частью анимации в привычном смысле этого слова, следовательно, ими нельзя манипулировать непосредственно как объектами. Управление ими происходит за счет изменения параметров или посредством добавления внешних сил. С помощью частиц можно имитировать различные предметы и явления природы, например дождь, снег, стаю птиц и многое другое.

Различают системы частиц, *неуправляемые* событиями (например, система Super Spray и т. п.) и системы частиц, *управляемые* событиями (система Particle Flow). Система частиц состоит из трех основных элементов: излучателя, частиц и внешних сил.

Стандартные системы частиц, в том числе Blizzard (Метель), Snow (Снег), Spray (Распылитель), Super Spray (Суперраспылитель), PCloud (Облако частиц) и PArray (Массив частиц), относятся к неуправляемым событиям, т.е. их свойства обычно не меняются на протяжении всей анимации. Такие системы частиц предназначены для создания, например, эффекта дождя. Но, если количество частиц должно уменьшаться в ответ на открытие окна, более пригодной оказывается управляемая событиями система частиц.

Неуправляемые событиями системы частиц имеют разную сложность. К самым простым из них относятся системы частиц Snow и Spray, а более сложными их вариантами являются системы частиц Blizzard и Super Spray. Поэтому Snow и Spray относятся к устаревшим системам частиц.

Частицы могут иметь практически любой внешний вид (сферы, четырехгранники, кубы, геометрические фигуры и лицевые плоскости). При создании эффекта на основе большого числа мелких частиц, например пара или дождя, старайтесь пользоваться самой элементарной формой частиц, чтобы сократить объем расчетов лишней геометрической формы частиц в сцене.

Задание. Создание эффекта водопада (используя систему частиц).

1. Создание основы. В окне Perspective создаем Box (коробку): Length (Длина) = 300 Width (Ширина) = 100 Height (Высота) = 150.

В окне Front создаём cylinder (цилиндр): Radius (Радиус) = 30 Height (Высота) = 400. Примените к цилиндру модификатор Noise с параметрами, как на рис. 1.

Теперь выберите Box (коробку), и затем Boolean (*Create>Geometry>Compound Objects>Boolean*). Щелкните Pick Operand B и затем в окне выбираем цилиндр. У вас должно получиться примерно так, как на рис. 45, 46.

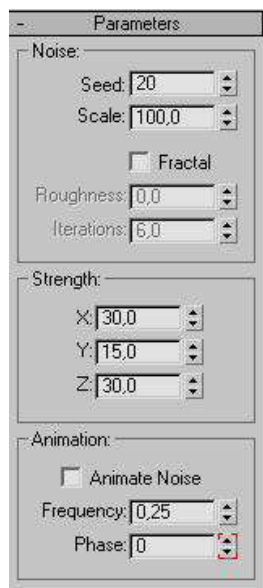


Рисунок 45

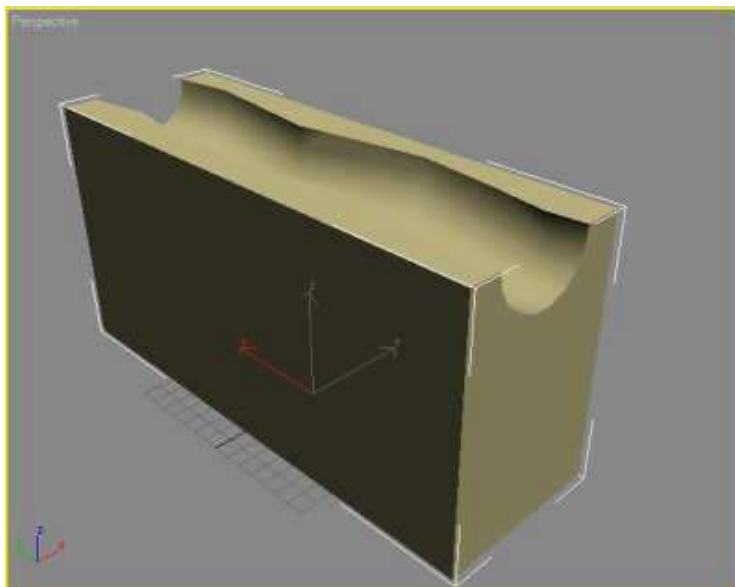


Рисунок 46

2. Затем создайте Plane в главном окне с Length = 300 and with = 90 . Поместим эту плоскость в желоб в Box.

Создайте второй Plane с Length = 40 и with = 90. Разворачиваем нашу плоскость на 90 градусов и помещаем в передней части желоба.

Теперь создайте третий Plane в главном окне с Length = 275 и with = 250. Помещаем ниже нашего Box.

У Вас должно получиться, как на рис. 47.

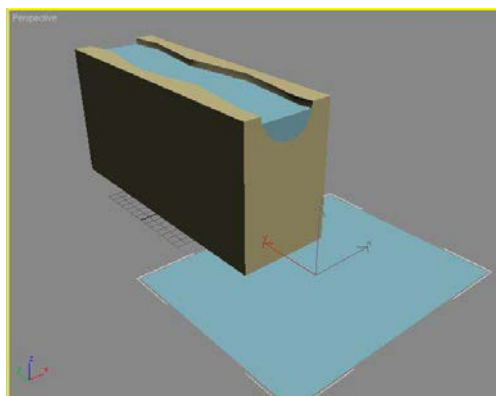


Рисунок 47

3. Создание материала воды.

Войдите в редактор материалов, нажимаем "М". Используйте параметры для материала:

Blinn

Для Ambient: Red = 5 / Green = 119 / Blue = 88.

Для Diffuse : Red = 99 / Green = 161 / Blue = 141.

Specular Level 40, Glossiness 50, Soften 0,1

Для Filter color в секции Extended parameters: Red = 0 / Green = 112 / Blue = 71. Amt 40 Index of Refraction 1,5

Теперь для bump map выбираем noise. Используйте параметры, как показано на рис. 5. Установите amount на 30 %. Теперь щелкните на пустой кнопке рядом с Reflection и используйте falloff map (карту спада).

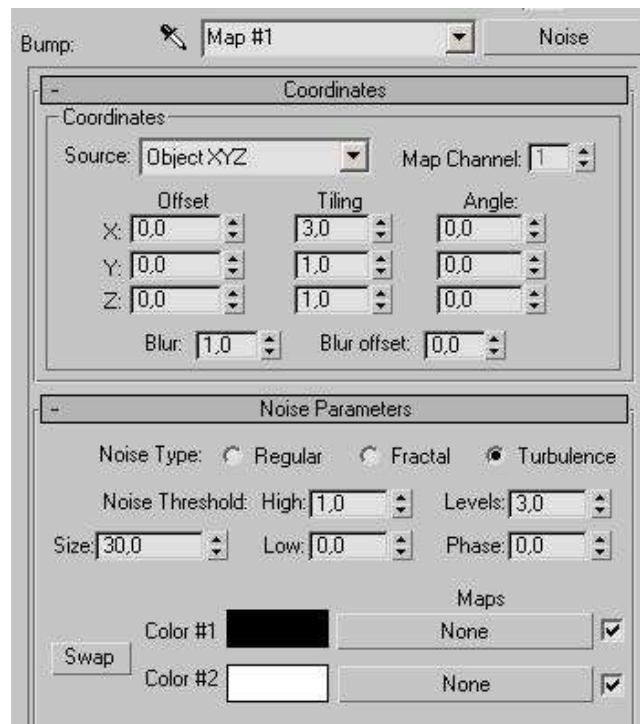


Рисунок 48

Теперь примените материал к трём своим Plane. Это должно смотреться как на рис. 49.

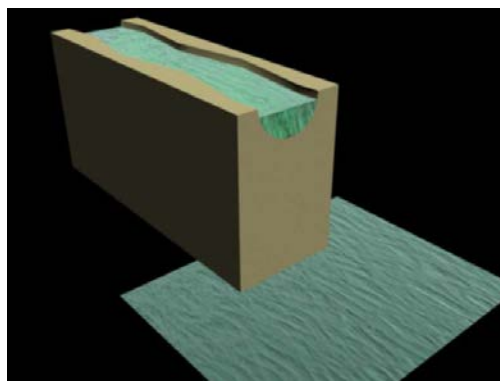


Рисунок 49

4. Создание реки.

Придадим воде движение. Для этого используем системы частиц. Выбираем create>geometry>particle systems.

Используем систему частиц Spray. Нажимаем кнопку Spray. И рисуем две плоскости с параметрами

Width = 60, Length = 3.

Задаём параметры, как на рис. 50, затем поворачиваем плоскости на 90 градусов и помещаем в Plane 1, одну за другой. Можете проанимировать сцену.

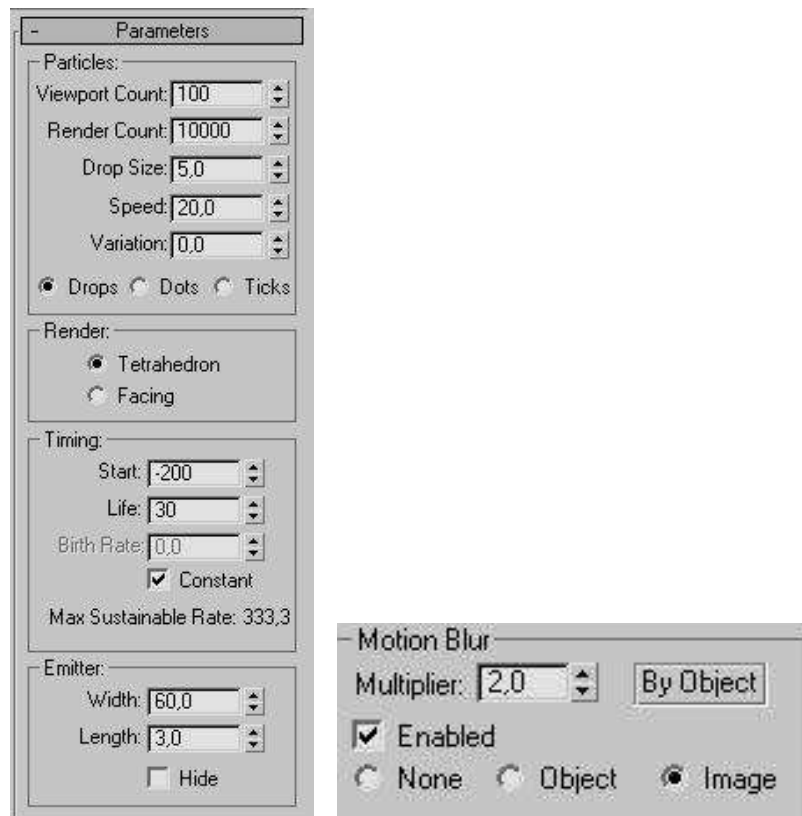


Рисунок 50

Теперь создаем еще два Sprau для нижнего Plane 3 с параметрами With = 250 и Length = 3 и помещаем их друг за другом в Plane 3. Должно получиться примерно так, как на рис. 51.

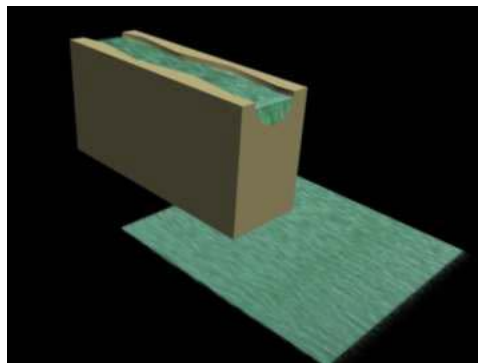


Рисунок 51

5. Создание водопада.

Теперь создаем водопад, для этого нам понадобятся частицы Snow. Выбираем Snow с параметрами, как на рис.52, затем перемещаете туда, откуда падает вода, направьте так, чтобы частицы Snow двигались параллельно Plane 1 и вперед.

Создайте плоскость с Длиной = 15 и Шириной = 40.

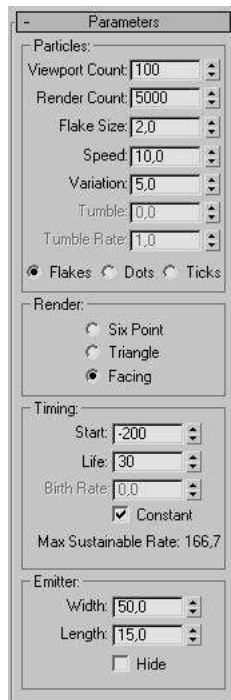



Рисунок 52

Теперь создаем Create>Space Warps>Forces>Gravity. Создаем объект со стрелкой вниз.

Выбираем систему Snow, щелкните этой кнопкой  в панели. Щелкаем на Snow и тянем к изображению со стрелкой.

Выбираем изображение со стрелкой, устанавливаем для него strength 10, и вы увидите, что частицы Snow падают вниз.

6. Создадим материал для водопада. Водопад часто выглядит белым, так что передадим водопаду белый вид. Создаём новый материал, как на рисунке и назначаем его для Snow (рис. 53).

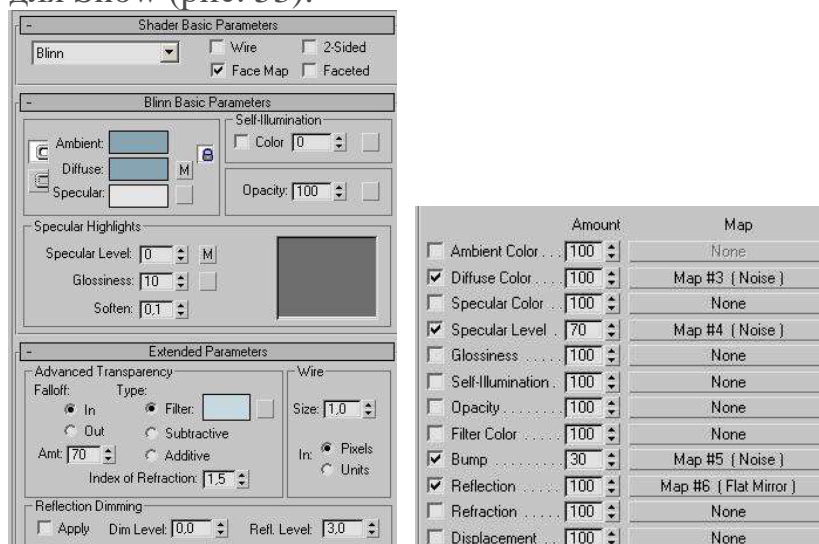


Рисунок 53

7. Создайте deflector с Шириной = 250 и Длиной = 275. Create>Space Warps>Deflectors>Deflector.

Теперь разместите его в Plane3. Используйте параметры как на рис. 54. Свяжите Snow с Deflector. Ваша мультипликация должна выглядеть так, как на рис.55.

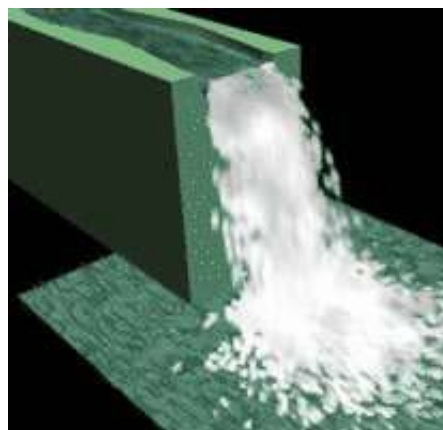
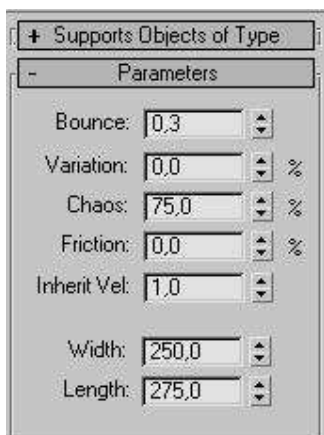


Рисунок 54

Рисунок 55

8. Сохраните файл под именем waterfall.

Лабораторная работа 17

Использование деформаций с системами частиц

Цель работы: изучить настройки и способы задания анимации систем частиц.

Задание. Создадим горящую спичку без применения эффекта Fire.

Методические рекомендации.

При создании большинства сложных визуальных эффектов в трехмерной графике используются частицы. Одно из предназначений частиц – имитация атмосферных эффектов, таких как огонь и дым, снег и дождь. С помощью частиц можно получить более реалистичное пламя. Создадим горящую спичку без применения эффекта Fire.

1. Создайте модель спички. Как известно, она состоит из двух частей: серы и деревянной щепки. Для серы используйте объект Capsule. Чтобы этот примитив был похож на нужный объект, в его настройках необходимо выбрать для параметра Height значение 26, а для параметра Radius – 7. Повернем его на девяносто градусов.

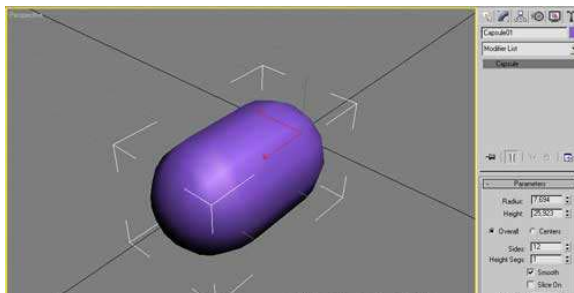


Рисунок 56

2. Создайте объект Box со значениями параметров Height – 5,5, Length - 129 и Width – 10. Подберите его положение относительно верхней части спички.

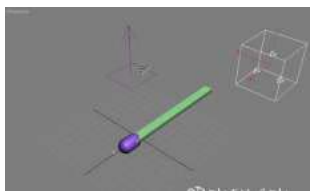


Рисунок 57

3. Для того чтобы создать эффект огня, необходимо превратить объект Capsule в источник частиц и назначить частицам соответствующий материал. Добавьте в сцену источник частиц PArray. Для этого перейдите на вкладку Create командной панели, в категории Geometry выберите строку Particle Systems и нажмите кнопку PArray. В свитке Basic Parameters нажмите кнопку Pick Object и укажите в сцене объект-эмиттер Capsule.

4. Для того чтобы пламя поднималось вверх, необходимо добавить в сцену некоторую силу, которая направит частицы в нужную сторону. Для этого используем объемную деформацию Wind (Ветер). Перейдите на вкладку Create командной панели, в категории Space Warps выберите строку Forces, нажмите кнопку Wind и нарисуйте значок объекта в любом месте окна проекции. Свяжите объемную деформацию с источником частиц PArray. Для этого используется кнопка Bind to Space Warp на основной панели инструментов. Чтобы привязать объемную деформацию к объекту, нажмите эту кнопку, щелкните на объемной деформации и, не отпуская кнопку мыши, переместите указатель мыши на объект.

Если прокрутить ползунок анимации, можно увидеть, что частицы теперь стремятся вверх.

5. Перейдите в настройки источника частиц. В свитке Basic Parameters переключатель Particle Formation установите в положение At Distinct Points. Благодаря этому частицы будут испускаться со всей поверхности источника в случайном порядке. Увеличьте значение параметра Total, определяющего общее число точек испускания частиц. Переключатель Viewport Display установите в положение Mesh. После этого в окне проекции частицы станут отображаться в виде поверхностей.

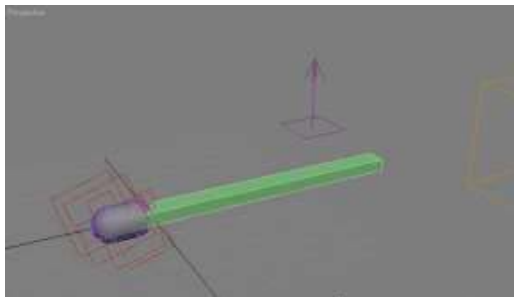


Рисунок 58

В области Particle Quantity свитка Particle Generation задайте значение параметра Use Rate равным ста шести. Это положение переключателя означает, что в каждом кадре будет испускаться определенное количество частиц, в нашем случае – сто шесть.

Движение частиц определяется объемной деформацией, поэтому значение параметра их собственной скорости установите равным нулю, выбрав это значение для параметра Speed в области Particle Motion. В области Particle Timing значение параметра Emit Stop, определяющее последний кадр, в котором будут испускаться частицы, установите равным 100.

Продолжительность жизни частиц задайте при помощи параметра Life, установив значение 15. Эта цифра означает, что частицы будут жить на протяжении пятнадцати кадров после испускания. В области Particle Size подберите подходящий размер частиц, используя параметр Size. Установите значение этого параметра равным примерно семи, а также увеличьте до ста значение параметра Variation, который определяет степень разброса частиц. На протяжении своей недолгой жизни каждая частица может увеличиваться или уменьшаться в размерах. Поскольку в данном случае нужно, чтобы частицы становились меньше, установите для параметра Grow For (Расти до) значение 0, а для параметра Fade For (Уменьшаться до) – значение 22.

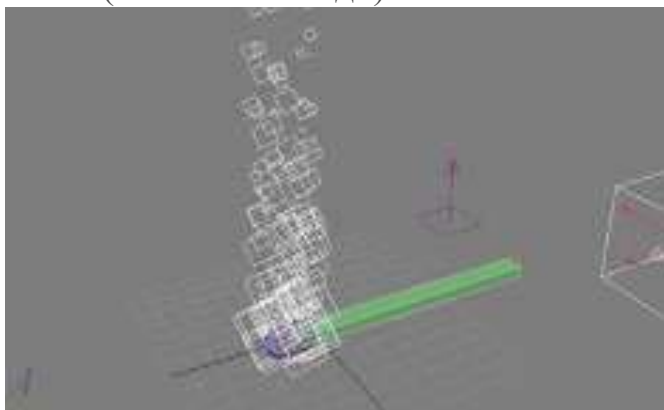


Рисунок 59

Перейдите к свитку Particle Type и определите форму частиц. Она важна, поскольку в будущем к частицам будет применен материал. Установите переключатель Standard Particles в положение Facing. Благодаря этому каждая частица будет иметь квадратную форму и всегда будет повернута к виду камеры.

Наконец, последний параметр, который нужно установить для частиц, это Influence в свитке настроек Object Motion Inheritance. Он определяет процент частиц, которые наследуют движения эмиттера в момент их создания. По умолчанию это число равно ста, благодаря чему все создаваемые частицы перемещаются вслед за движущимся объектом. Уменьшите значение этого

параметра до нуля, обеспечив тем самым их независимость от преобразований эмиттера. Конечные параметры источника частиц показаны на рисунке.

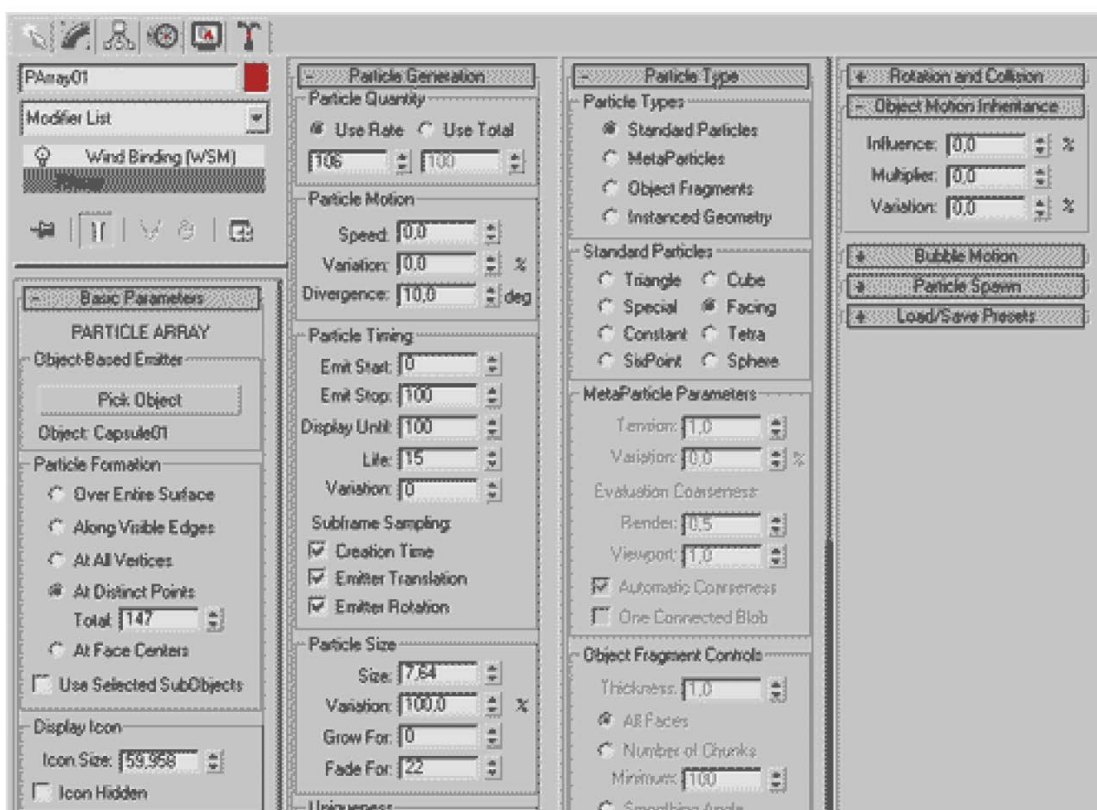


Рисунок 60

6. Теперь задайте параметры материала для частиц. Откройте редактор материалов (горячая клавиша M) и установите флажок Face Map. Благодаря этому материал будет накладываться на каждую поверхность. Назначьте материал объекту, перетащив его из ячейки. Пламя имеет неоднородный окрас. Возле спички оно более светлое, а по мере удаления от эмиттера становится темнее. В качестве карты диффузного цвета (Diffuse) установите Particle Age, которая дает возможность воссоздать эффект изменения цвета частиц во времени. Карта Particle Age позволяет определять характеристики цвета каждой частицы в зависимости от ее возраста. Определите в настройках карты цвета пламени – желтый и разные оттенки коричневого.

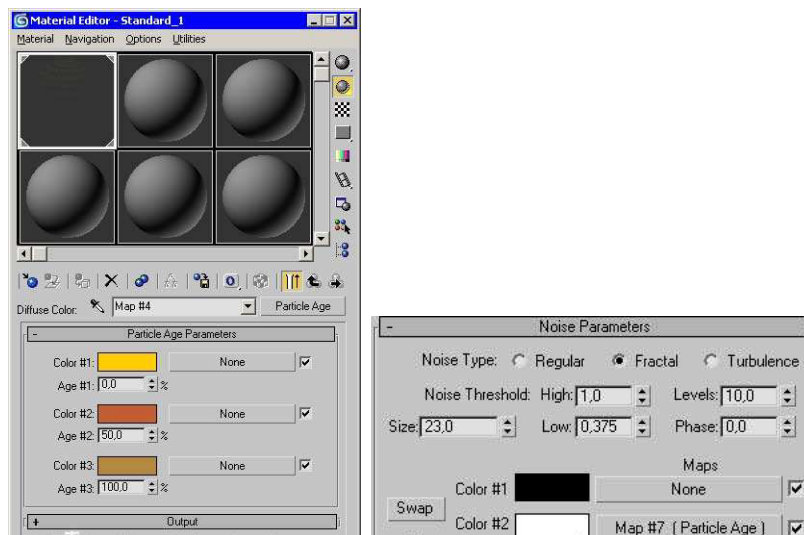


Рисунок 61

7. Вернитесь к настройкам материала. Пламя должно быть отчасти прозрачным, поэтому установите для параметра Opacity значение ноль и назначьте карту Mask. Такая карта позволит воссоздать неоднородную прозрачность пламени. В качестве карты, которая будет видна через маску (слот Map), выберите Noise. В свитке Noise Parameters установите переключатель Noise Type в положение Fractal и настройте характер неоднородности пламени. Для параметра Size установите значение равным 23, Low – 0,375, Levels – 10. Чтобы зашумленность зависела от возраста частицы, выберите в качестве одного из смешиваемых цветов карты Noise карту Particle Age. Поскольку мы настраиваем карту непрозрачности, для нее имеют значение только светлые и темные участки используемого рисунка. Поэтому установите в настройках Particle Age белый и черный цвета. На определенном этапе карта будет менять свой рисунок.

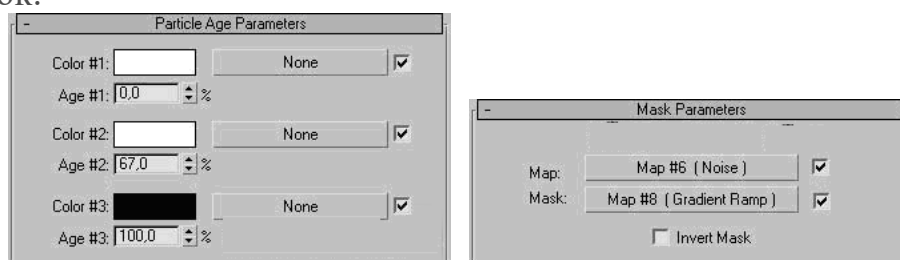


Рисунок 62

8. Вернитесь к параметрам карты Mask и выберите в качестве карты, которая будет использоваться как маска, Gradient Ramp. При помощи этой карты вы сможете создать градиентный переход от белого цвета к черному. Установите такие параметры, как показано на рисунке. Для изменения цвета градиента дважды щелкните по маркеру и выберите соответствующий цвет в окне Object Color Selector: Color.

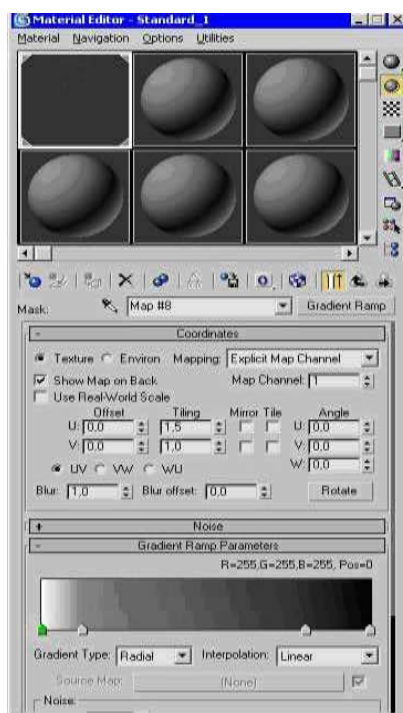


Рисунок 63

На этом настройка материала завершена. Чтобы сделать пламя более ровным, необходимо включить эффект смазанного движения в настройках источника частиц. Для этого выделите его в окне проекции, вызовите контекстное меню, выберите строку Properties и в группе Motion Blur окна Object Properties установите переключатель в положение Image. Это означает, что эффект будет применен ко всему изображению. Уменьшите величину эффекта, изменив значение параметра Multiplier до 0,7. Можно визуализировать анимацию.

9. Сохраните сцену в своей папке под именем, например, Fire.

Лабораторная работа 18

Анимация взрыва

Цель работы: освоить танимацию взрыва.

Задание. Анимация взрыва бомбы.

Методические рекомендации

Рассматривая разные варианты анимации объектов, можно заметить, что для ее создания не обязательно прибегать к каким-то ухищренным способам, очень часто достаточно использования стандартных средств. В настоящем упражнении рассмотрим один из вариантов анимации взрыва.

1. Создадим сферу и цилиндр, выполняющие роль бомбы. Из этих 2-х объектов с помощью булевских операций создадим один составной объект. Это будет наша модель бомбы, которая будет взорвана. Создадим сплайн пути для запального шнура. Сплайн начинайте строить **от запала** бомбы.

2. Создадим сферу и цилиндр, выполняющие роль бомбы. Из этих 2-х объектов с помощью булевских операций создадим один составной объект. Это будет наша модель бомбы, которая будет взорвана. Создадим сплайн пути для запального шнура. Сплайн начинайте строить от запала бомбы. • Теперь построим запальный шнур и анимируем его. Создайте в окне Top цилиндр с параметрами: $R=1.8$ $H = 200$ $Segment=40$ • примените к цилиндру модификатор Path Deform и выберите в одном из окон проекций ваш сплайн пути для запального шнура; • щелкните на кнопке Move to Path (Передвинуть на путь) для того, чтобы цилиндр занял на сплайне пути правильное положение (рис.2).



Рисунок 64

Можно, изменяя параметр Stretch, растянуть цилиндр и тем самым увеличить длину запального шнура. Прделайте это. Пусть цилиндр закроет весь сплайн.

3. Настройте количество кадров анимации, равное 200. Создадим анимацию:

- активизируйте автоматическую запись ключей анимации, для чего в правом нижнем углу интерфейса программы щелкните на кнопке Auto Key (Авто ключ);
- передвиньте ползунок таймера анимации в кадр 100;
- перейдите к свитку Parameters (Параметры) цилиндра, для чего в списке модификаторов щелкните на строке с именем Cylinder (Цилиндр), в результате чего строка подсветится цветом;
- измените значение поля Height (Высота) с двухсот до нуля;
- выключите запись ключей анимации, для чего повторно щелкните на кнопке Auto Key (Авто ключ).

Проверьте правильность выполненных действий, запустив воспроизведение кнопкой Play Animation (Воспроизвести анимацию). На протяжении ста кадров запальный шнур должен уменьшаться пока совсем не исчезнет в сотом кадре.

4. Для того, чтобы усилить эффект горения, можно добавить на окончание запального шнура систему частиц, которая будет имитировать разлетающиеся в стороны искры. Постройте сферу диаметром такого же размера, как и запальный шнур. В нашем случае это 1.8.

5. Примените к построенной сфере модификатор Path Deform и выберите в одном из окон проекций сплайн пути для запального шнура.

6. Щелкните на кнопке Move to Path (Передвинуть на путь) для того, чтобы сфера заняла на сплайне пути правильное положение. В счетчике Percent

(Проценты) установите такое значение, при котором сфера в нулевом кадре шкалы анимации переместится в начало запального шнура. У нас это значение равно 93.3.

8. Активизируйте автоматическую запись ключей анимации, щелкнув на кнопке Auto Key (Авто ключ). Передвиньте ползунок таймера анимации в кадр 100. Введите в поле счетчика Percent (Проценты) значение равное нулю.

9. Проверьте анимацию. Сфера должна перемещаться по сплайну – пути вместе с запальным шнуром. Шнур сгорает и становится короче и короче пока не достигнет запала.

10. Создадим искры. Будем имитировать сгорание шнура. Для этого построим простую анимацию системы частиц. Выполните из главного меню команду Create>Particles>PArray (Создать>Частицы>Массив частиц) и в произвольном месте любого окна проекции создайте значок массива частиц. В свитке Basic Parameters (Базовые параметры), построенной системы частиц PArray щелкните на кнопке Pick Object (Указать объект) и выберите в одном из окон проекций построенную ранее сферу. Перейдите к свитку Particle Generation (Генерация частиц) и выставьте параметры создаваемых частиц:

User Rate =150, Emit Start =1, Creation Time – вкл., Speed=4, Emit Stop = 100, Emitter Translation – вкл., Variation =100, Display Until = 100, Emitter Rotation – вкл., Divergence = 30, Life = 2, Variation = 10, Particle Size 1, 40, 0, 20

Из всех параметров данного свитка существенным являются время окончания генерации частиц, которое должно быть продлено до окончания горения запального шнура (в нашем случае до 100) и флажок на Emitter Rotation (Поворот источника), что позволит задать направление движения частиц.

В группе Standard Particles (Типовые частицы) свитка Particle Type (Тип частиц) установите переключатель на Special (Специальный), что позволит задать форму частиц в виде пересекающихся плоскостей.

В группе Spin Axis Controls (Управление осями вращения) свитка Rotation and Collision (Поворот и столкновения) установите переключатель на Direction of Travel (Направление движения/Размывание движения) и в счетчике Stretch (Растяжение) установите значение равное 50. Изменение этих параметров позволит получить протяженное размытие частиц по мере удаления от источника.

Настроив основные параметры системы частиц, выполните визуализацию статичного кадра в середине анимации и убедитесь в том, что все выполнено правильно. Если результат вас не удовлетворит, проведите дополнительную настройку параметров системы частиц.

11. Следующим шагом будет создание анимации самого взрыва. В данном случае нам поможет объект Bomb (Бомба), относящийся к

пространственным деформациям (Space Warps). Для создания и настройки анимации взрыва выполните следующие действия.

Выполните из главного меню команду Create>SpaceWarps>Geometric / Deformable>Bomb (Создать >Пространственные деформации>Деформируемая геометрия>Бомба).

Разместите созданный значок бомбы внутри нашего объекта.

Свяжите объект с воздействием. В одном из окон проекций выделите значок объекта Mesh Bomb (Бомба) и перейдите к свитку Bomb Parameters (Параметры бомбы), расположенного во вкладке Modify (Изменить) командной панели.

В группе Explosion (Взрыв) установите значения счетчиков Strength (Мощность), Spin (Вращение) равным 1.5.

В группе Fragment Size (Размер фрагментов) установите значения счетчиков Min (Минимум) и Max (Максимум) в 1 и 5 соответственно.

В счетчике Detonation (Детонация) группы General (Общие) запишите номер кадра, в котором произойдет взрыв. В нашем случае это сотый кадр, в котором запальный шнур догорит до конца.

Проверьте анимацию. Начиная с первого по сотый кадр будет гореть запальный шнур, а начиная с 101-го кадра, начнется процесс взрыва бомбы.

12. Прделанными выше действиями мы построили простой взрыв. Повысить его реалистичность можно, воспользовавшись эффектом огня (Fire Effect). Для этого необходимо предварительно построить габаритный контейнер, который будет определять зону действия атмосферного эффекта Fire Effect (Эффект огня). Выполнить это можно из главного меню – Create>Helpers>Atmospherics>Sphere Gizmo (Создать>Вспомогательные объекты>Атмосферные эффекты>Сферический контейнер).

После построения расположите габаритный контейнер так, чтобы бомба находилась в середине этого контейнера. При активном габаритном контейнере SphereGizmo перейдите на вкладку Modify (Изменить) командной панели и в свитке Atmospheres & Effects (Атмосфера и эффекты) щелкните на кнопке Add (Добавить), в результате чего откроется окно диалога Add Atmosphere (Добавить атмосферный эффект). В этом окне выберите из списка Fire Effect (Эффект огня) и щелкните на кнопке Ok для подтверждения выбора. В результате этих действий в список эффектов габаритного контейнера добавится эффект огня.

13. Пришло время настройки параметров эффекта для взрыва. Для этого выполните следующее:

В свитке Atmospheres & Effects (Атмосфера и эффекты) выделите строку с именем добавленного эффекта и щелкните на кнопке Setup (Настройки). В

результате этих действий откроется окно диалога Environment and Effects (Окружающая среда и эффекты).

Убедитесь в том, что в группе Shape (Форма) открывшегося окна диалога Environment and Effects (Окружающая среда и эффекты) установлен переключатель на Fireball (Облако пламени).

В этой же группе установите значение счетчика Regularity (Регулярность) на 0.6 что позволит придать эффекту форму близкую к форме габаритного контейнера (значение, равное единице создаст абсолютно круглый огненный шар).

В группе Explosion (Взрыв) установите флажок на Explosion (Взрыв). Это позволит программе рассматривать эффект огня как взрыв. В этой же группе щелкните на кнопке Setup Explosion (Настройки взрыва) и в открывшемся окне диалога Setup Explosion Phase Curve (Настройки кривой фаз взрыва) установите значение полей Start Time (Время начала) и End Time (Время окончания) в 100 и 180 соответственно. Эти значения не что иное, как начало и окончание действия эффекта взрыва. Остальные параметры эффекта взрыва могут оставаться по умолчанию либо в случае, если вам не понравился результат, вы всегда можете поменять их на свои.

14. Активизируйте окно перспективного вида, переместите ползунок таймера анимации в 102 кадр (начало взрыва) и выполните визуализацию.

Визуализация эффекта возможна только в окне Perspective (Перспектива) или окне вида из камеры. В окнах проекций Front (Спереди), Left (Слева), Top (Сверху), Right (Справа) и других эффект не визуализируется.

Как видно из визуализации, размера эффекта взрыва недостаточно. Сделать его большим можно, увеличив габаритный контейнер SphereGizmo, для чего выделите этот объект и в поле счетчика Radius (Радиус) свитка Sphere Gizmo Parameters (Параметры сферического габаритного контейнера) установите значение, равное 130. Повторите визуализацию эффекта и убедитесь в том, что размер облака пламени близкий к радиусу разлетающихся осколков бомбы. Для проверки выполните визуализацию в других кадрах анимации и при желании внесите изменения в настройки эффекта или размер габаритного контейнера.

15. Выполните анимацию всех кадров и сохраните ее как последовательность или как готовый видеоролик.

Лабораторная работа 19

Разработка сюжета ролика

Цель работы: изучить шаги, которые помогут вам создать увлекательный сюжет для ролика.

Методические рекомендации

Определите цель ролика: Прежде чем начинать работу над сюжетом, нужно понять, какая цель у вашего ролика. Например, вы можете хотеть рассказать историю о продукте, привлечь внимание к своему бренду, продемонстрировать новый сервис и т.д.

Определите главного героя: Выберите героя, который будет главным персонажем вашего ролика. Главный герой может быть реальным человеком, мультяшным персонажем или даже животным.

Создайте конфликт: Чтобы сюжет был увлекательным, ему нужен конфликт. Конфликт может быть маленьким или большим, но он должен быть достаточно интересным, чтобы удерживать внимание зрителя.

Разрешите конфликт: Научите главного героя преодолевать препятствия и разрешать конфликты. Это позволит зрителю восхищаться героем и почувствовать сопереживание.

Добавьте визуальные эффекты: Чтобы ролик выглядел более увлекательным, добавьте визуальные эффекты, такие как анимация, компьютерная графика или спецэффекты.

Сократите сюжет: Важно помнить, что у вас есть всего 60 секунд, чтобы рассказать свою историю. Поэтому сократите сюжет до минимума и сосредоточьтесь на ключевых моментах.

Сделайте финал запоминающимся: Закончите ролик так, чтобы зритель запомнил его надолго. Например, вы можете добавить неожиданный поворот сюжета или закончить ролик забавной шуткой.

Чем отличается сценарий от литературного произведения? Сценарии не пишутся, они конструируются. Главный проводник между автором и зрителем – герой. Каким должен быть он и его вселенная? Чем характеристика героя отличается от его характера? Зачем нужна тайна? Любая история рождается из события. Какова структура любой истории? События должны менять историю. "Что-то пошло не так!" – главная драматургическая формула.

Задание. Придумайте необычного интересного героя и выстройте с ним историю как минимум из двух событий: завязка и кульминация. И если вы будете вводить других персонажей, не забывайте, что они обязаны быть связаны с главным героем и фактически его выражать. Тема истории – море, путешествие.

Тема 6. Игровая анимация

Лабораторная работа 20

Анимация материалов

Цель работы: изучить основные способы анимирования материалов.

Методические рекомендации

Существует несколько способов анимации материалов в 3d программах.

Это:

Использование текстурной анимации: можно создать последовательность текстур, изображающих различные состояния материала, и затем анимировать переключение между ними с помощью ключевых кадров и анимационной кривой.

Использование параметрической анимации: можно анимировать параметры материала, такие как цвет, прозрачность, блеск и т.д., используя ключевые кадры и кривые анимации.

Использование специальных эффектов: можно добавлять эффекты, такие как свечение, мерцание, отражение и т.д., и анимировать их параметры для создания интересных эффектов.

Использование плагинов и скриптов: существуют плагины и скрипты для 3ds Max, которые позволяют создавать более сложные и динамические анимации материалов, такие как анимация текстуры волос или анимация воды.

Анимация материала может быть выполнена на основе типа материала Blend, у которого анимируется параметр Mix Amount (Рисунок 65).

Есть разные способы анимирования параметра Mix Amount.

1. С помощью ключей (Bezier Float).
2. С помощью Wire parameters (Float Wire).
3. С помощью Reaction Manager (Float Reaction).
4. С помощью скрипта (Float Script).

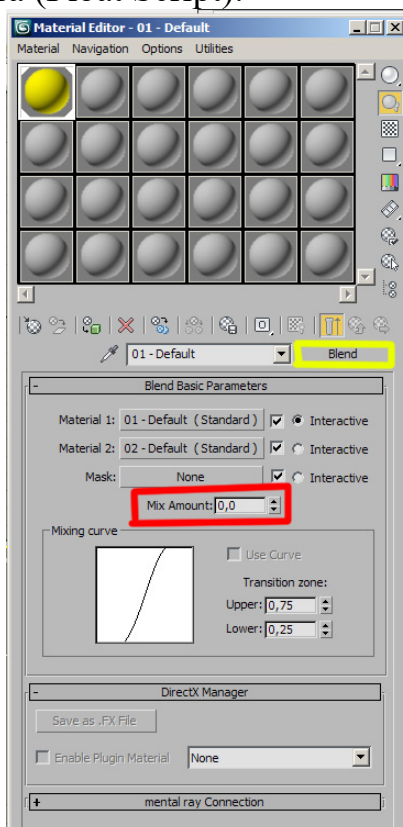


Рисунок 65

Пример анимации с помощью ключей:

1. В окне редактора материалов создаем 2 материала с разным цветом Diffuse.
2. Создаем Blend материал и в слоты материалов назначаем 2 ранее созданных материала.
3. Устанавливаем Time Slider в нужном кадре (например, 50). Включаем кнопку Auto Key.
4. В поле Mix Amount материала Blend устанавливаем значение 100. Выключаем кнопку Auto Key.
5. Создаем один объект Box и применяем к нему материал Blend.

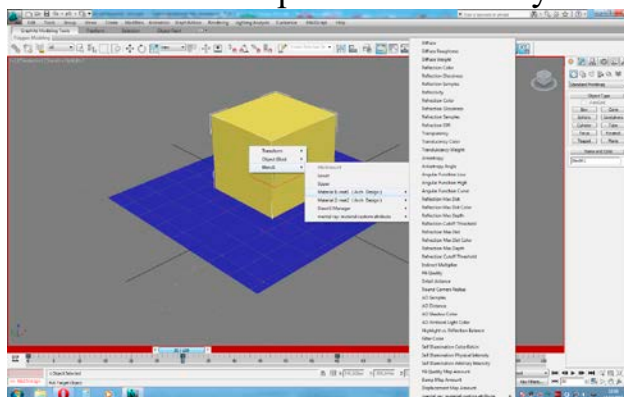


Рисунок 66

Задание. Создать анимацию со стеной здания, штукатурка у которого превращается в кирпичную кладку.

1. Сменить материал в разных ключах анимации. Речь идет о стене здания. В одном ключе, например, штукатурка, далее в другом ключе она превращается в кирпичную кладку.
2. Визуализируйте сцену с кадра 0 до последнего кадра и сохраните файл.

Лабораторная работа 21

Морфинг

Цель работы: освоить использование морффинга при создании анимации.

Методические рекомендации по выполнению: познакомиться с использованием инструмента кисть при анимации.

Морфинг (англ. morphing, трансформация) – технология в компьютерной анимации, визуальный эффект, создающий впечатление плавной трансформации одного объекта в другой. Таким образом имея два объекта разной формы (но одинаковой топологии!!) можно получать любые промежуточные формы. Морфинг часто используется для анимации мимики персонажей.

Для осуществления Морфинга в Cinema 4D используется тег Pose Morph.

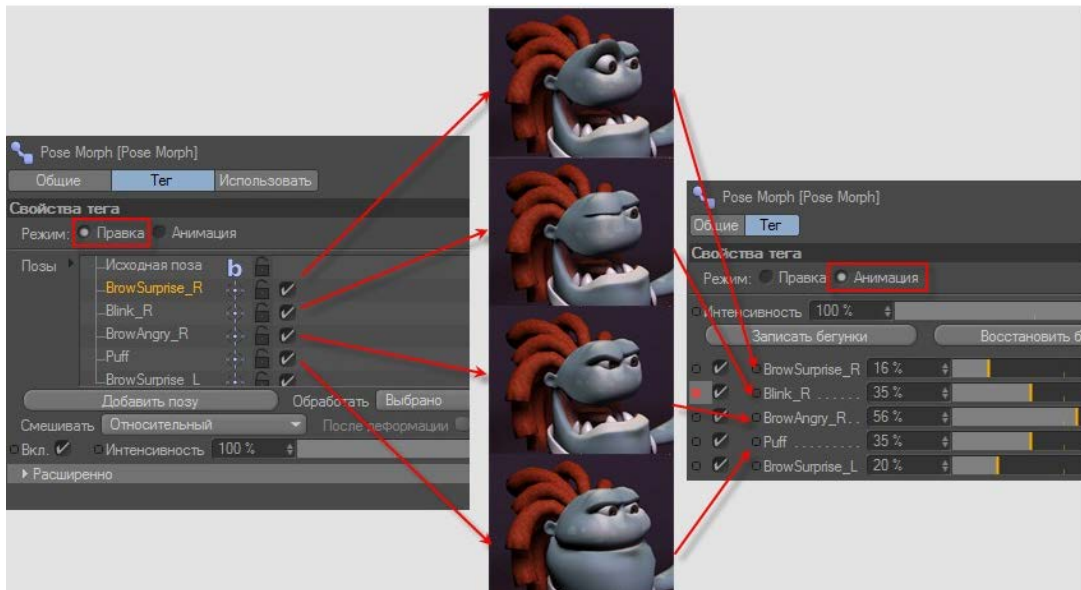


Рисунок 67

Целевая геометрия – это геометрия, представляющая собой отдельные объекты – вариации одного исходного объекта. И на основании этих вариаций форм и получаются плавные деформации – морфинг. Это морфинг целевой геометрии.

Исходная геометрия – это оригинальный объект, который изменяется (деформируется) посредством любых инструментов, тем самым позволяя получить нужные вариации поз морфинга. Объект при этом один. Это морфинг исходной геометрии.

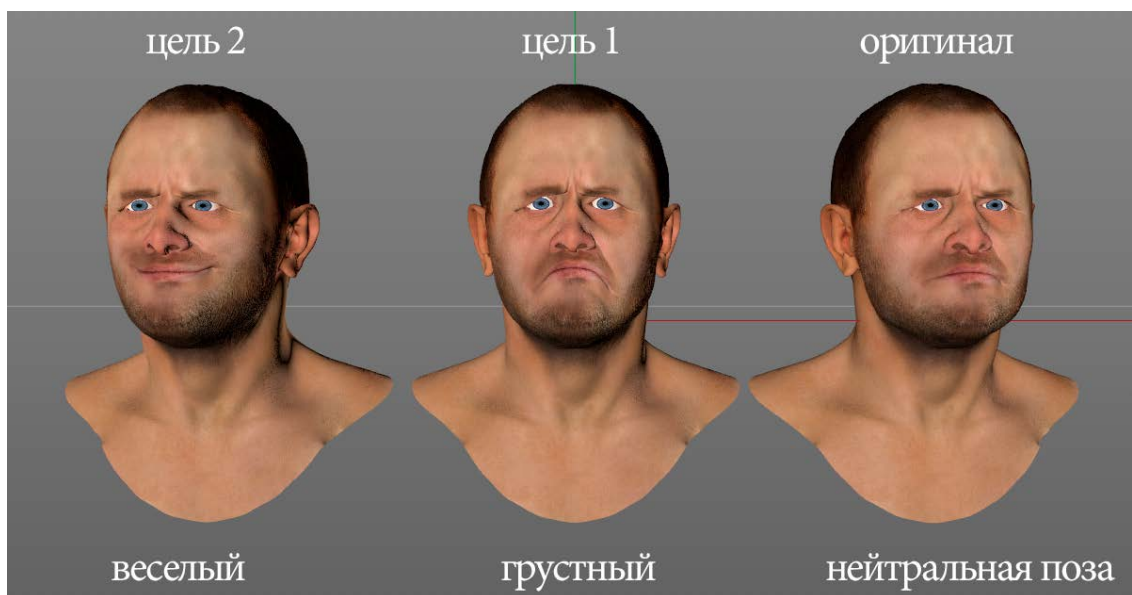


Рисунок 68

Морфинг целевой геометрии

Для создания морфинга необходимо назначить тег Pose Morph из группы тегов Персонаж на исходную (нейтральную) геометрию. В настройках тега Pose

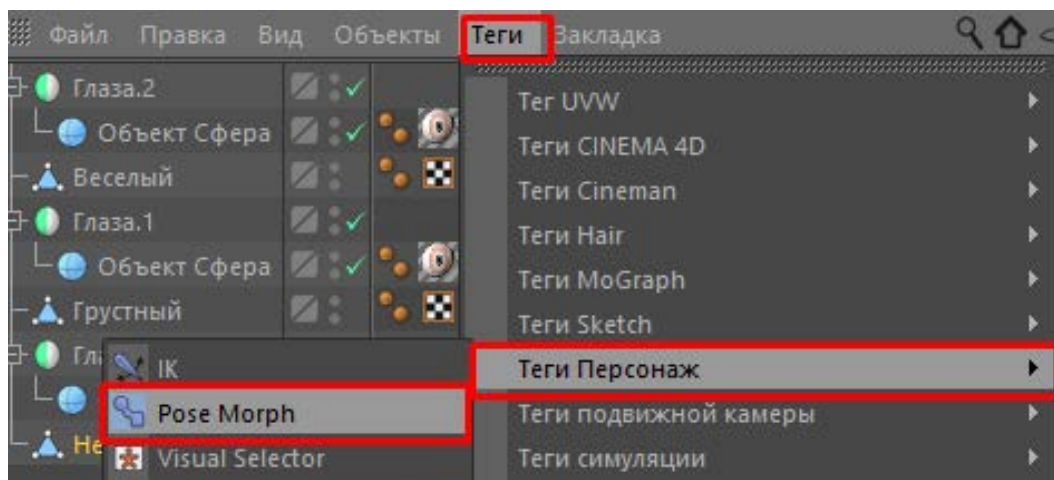


Рисунок –69

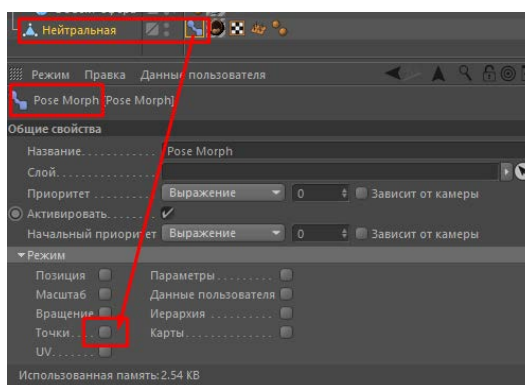


Рисунок –70

Morph назначить режим Точки.

После выбора режима тега в закладке Тег осуществляется назначение поз морфинга. В случае с целевой геометрией в поле Цель помещается соответствующая геометрия (цель 1, цель 2 и т.д.). Изменяя значения параметра интенсивность – регулируется степень плавного перехода от исходной позы к целевой. Таким образом добавляется (кнопка Добавить позу) любое количество целевых поз. В закладке Анимация осуществляется анимация морфинга.

Для создания морфинга необходимо назначить тег Pose Morph из группы тегов Персонаж на исходную (нейтральную) геометрию. В настройках тега Pose Morph назначить режим Точки.

После выбора режима тега в закладке Тег осуществляется назначение поз морфинга. В случае с исходной геометрией вся работа производится с одним объектом – исходной позой.

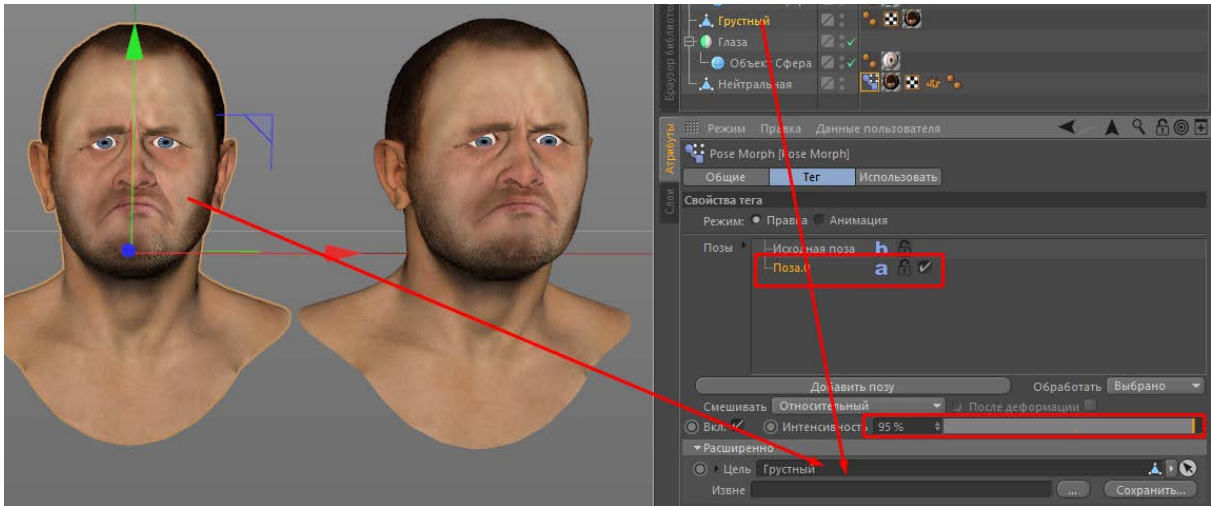


Рисунок 71

Для добавления поз морфинга нужно активировать новую позу (не исходную!) и используя любые инструменты работы с геометрией придать модели нужную форму. Но!! При этом количество полигонов, ребер и точек должно оставаться таким – же, топология тоже не должна меняться. Изменяется только форма объекта! Изменения вносятся на одну сторону симметричной модели, для последующего отражения на второй стороне. Это делается для возможности отдельно анимировать каждую половину.

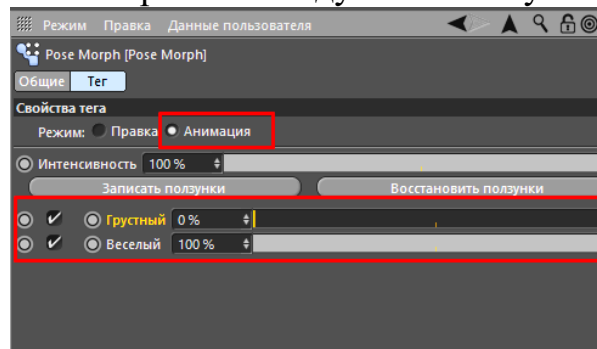


Рисунок 72

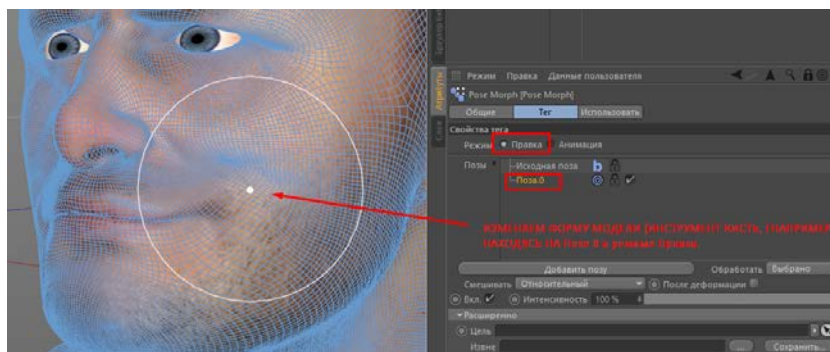


Рисунок 73

После создания первой новой позы достаточно нажать Добавить позу и внести соответствующие изменения. Повторить операцию нужное количество раз.

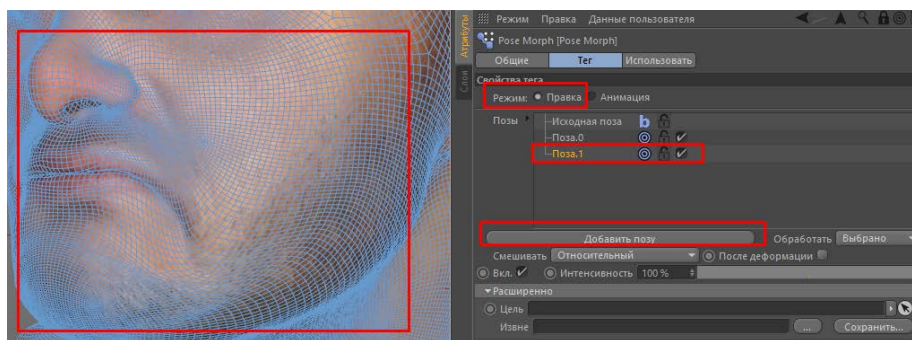


Рисунок 74

Название позы можно задавать двойным кликом по имени позы. После создания всех поз, необходимо создать их зеркальную копию. Для этого:

1. Кликаем правой кнопкой мыши по соответствующей позе и выбираем Копировать.
2. Кликаем правой кнопкой мыши по соответствующей позе и выбираем Вставить. Переименовываем позы.

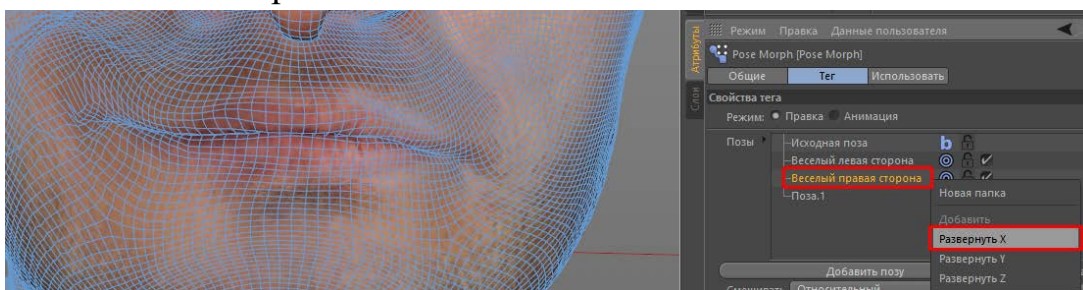


Рисунок 75

3. Кликаем правой кнопкой мыши по соответствующей позе и выбираем Развернуть (по нужной оси). Прodelываем со всеми позами. Для создания анимации переходим в закладку Анимация. Анимация осуществляется записью ключевых кадров на различных значениях соответствующих полей (поз).

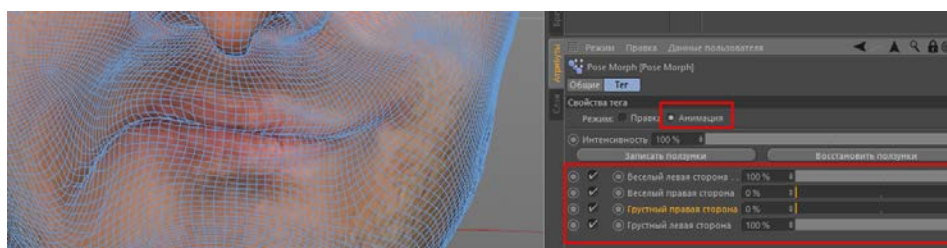


Рисунок 76

Для плавных деформаций при создании различных вариаций исходной позы удобно пользоваться опцией инструментов выделения – Мягкое выделение. Это позволит вносить изменения на больших участках при этом сглаживая края области выделения.

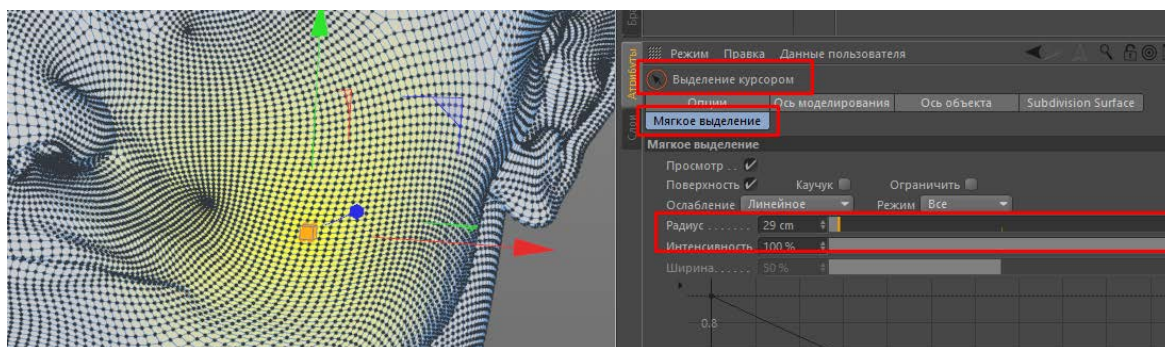


Рисунок 77

Задание: используя инструмент кисть, создать морфинг эмоций радости и удивления на лице.

Лабораторная работа 22

Изучение интерфейса Unity3d. Создание простых моделей. Добавление массы, гравитации

Цель работы: изучить интерфейс Unity3d и освоить приемы моделирования трехмерных геометрических объектов и их физических свойств.

Методические рекомендации

1. Для создания нового проекта необходимо запустить среду Unity3d и создать новый проект (File – New Project).

Окно создания нового проекта в Unity3d Unity3d содержит набор базовых пакетов с различными физическими свойствами. Для создания стандартных геометрических моделей (куб, сферы и т.д.) и затем задания гравитационных свойств этим объектам нам потребуются, соответственно, пакеты PhysicMaterials.UnityPackage и StandartAssets (Mobile).UnityPackage. Для добавления пакетов в новый проект в открывшемся окне (рис. 78) выбираем вкладку Create New Project, указываем путь к текущей рабочей папке с проектом, и отмечаем необходимые пакеты.

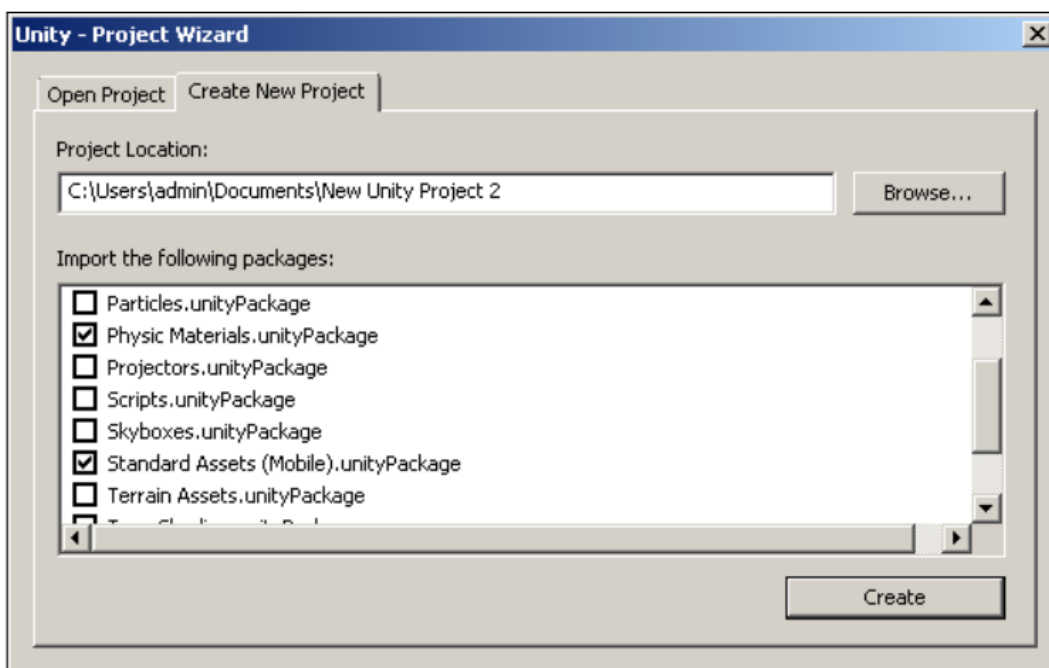


Рисунок 78 – Окно выбора стандартных пакетов

Рассмотрим окно пустого проекта в Unity3d и его основные элементы (рис. 79).

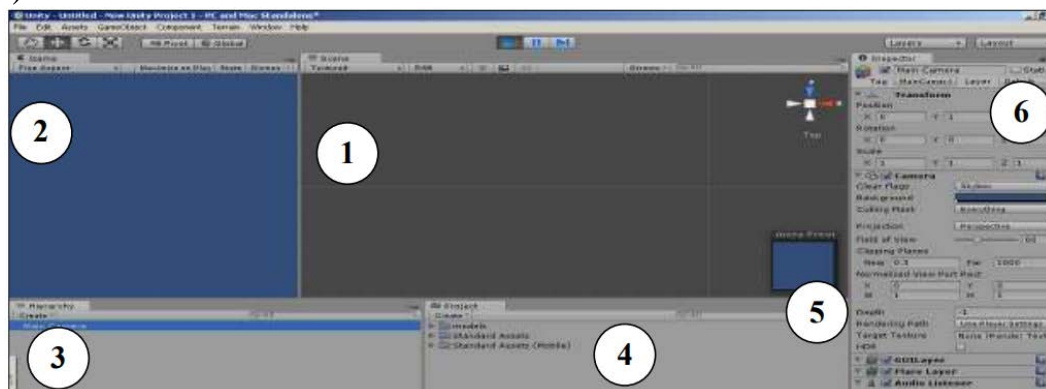


Рисунок 79 – Окно пустого проекта в Unity3d:

- 1 – окно сцены (Scene);
- 2 – окно запуска и просмотра игры (Game);
- 3 – окно иерархии объектов (Hierarchy);
- 4 – окно инспектора префабов и ресурсов (Project);
- 5 – окно предварительного просмотра сцены (Camera Preview);
- 6 – окно инспектора компонентов и их свойств (Inspector)

Интерфейс Unity3d является гибко настраиваемым под нужды разработчика. Так, при выделении в любом элементе окна текущей вкладки с помощью контекстного меню, она может быть дополнена новой (Add Tab) или удалена (Close Tab)

Инспектор префабов и используемых ресурсов (Project) содержит ссылки на файлы добавленных в проект моделей, текстур, звуков, скриптов, а также специальные объекты – префабы, предназначенные для дальнейшего повторного использования объектов в проекте.

Иерархия объектов (Hierarchy) – это список всех объектов на текущем уровне, показывающий также их отношения наследования (Parent-Child).

Инспектор компонентов и их свойств (Inspector) характеризует выделенный в данный момент объект – модель, текстуру, префаб и т.д., отображаемый в окне иерархии объектов (Hierarchy).

Добавление к проекту стандартного геометрического объекта Unity3d осуществляется на вкладке GameObject в главном меню. В качестве простейших объектов в сцену проекта могут быть добавлены геометрические объекты (куб, сфера, цилиндр и т.д.), источники освещения, камера.

2. Выберем на этой вкладке пункт Cube для создания в текущей сцене объекта «куб» (GameObject – Create Other – Cube).

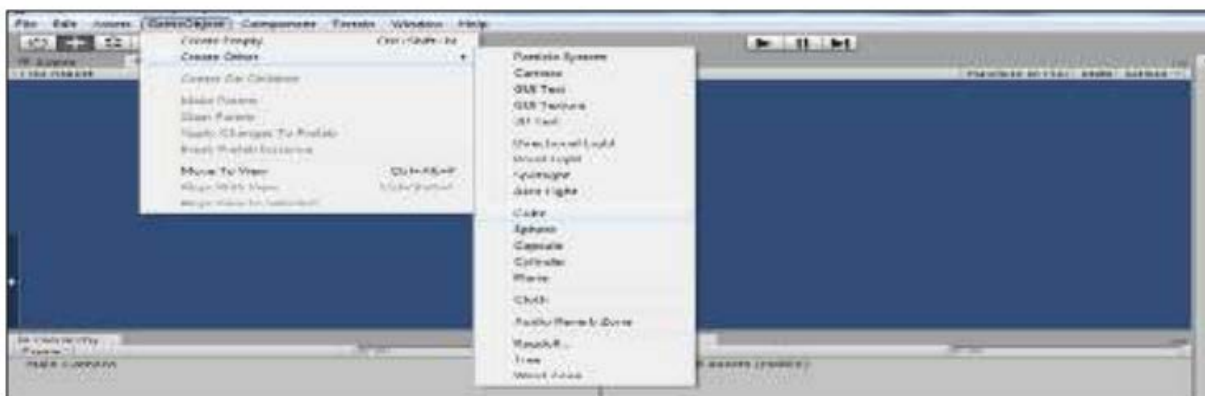


Рисунок 80 – Добавление к проекту простейшего геометрического объекта

3. Рассмотрим, что представляет собой только что созданный геометрический объект. Если выделить его в окне иерархии объектов или в окне сцены, то в окне инспектора мы увидим список связанных с объектом компонентов.

Transform – это компонент, отвечающий за положение объекта в пространстве в мировой системе координат и имеющийся у любого игрового объекта. Он хранит в себе переменные – координаты объекта в трехмерном пространстве x , y , z ; данные о повороте его относительно осей координат, а также методы для изменения этих параметров. Также он отвечает за иерархию объектов на сцене: свойство parent может содержать ссылку на другой аналогичный компонент Transform, являющийся родительским по отношению к данному. Если родительский компонент при этом двигается или вращается, вместе с ним двигается и вращается и дочерний. Дочерний же может свободно двигаться и вращаться, при этом его перемещения не оказывают влияние на родительский компонент.

Mesh filter – компонент, хранящий в себе трехмерную модель объекта, в данном случае – это кубический примитив.

Mesh Renderer – компонент, который дополняет геометрические данные о трехмерной модели из вышеуказанного компонента Mesh filter текстурой

(текстурные координат) и применяет шейдеры, в результате чего мы получаем на экране полноценную модель такой, как она должна выглядеть. Также этот компонент позволяет включить/отключить свойство отбрасывания теней объектом и на объект.

Box Collider – компонент, хранящий в себе трехмерную модель «коллизий», то есть ту модель, по которой физический движок рассчитывает столкновения между объектами или со средой. Модель, указанная в данном случае как Box Collider, отображается на сцене в виде зеленого каркаса. Варьируя переменными x, y, z центра масс (Center) и его размера (Size), возможно существенным образом менять характер взаимодействия объектов.

Для того чтобы созданный куб имел под собой основание в виде плоской поверхности, необходимо создать для него объект «плоскость» (GameObject – Create Other – Plane) (рис. 81).

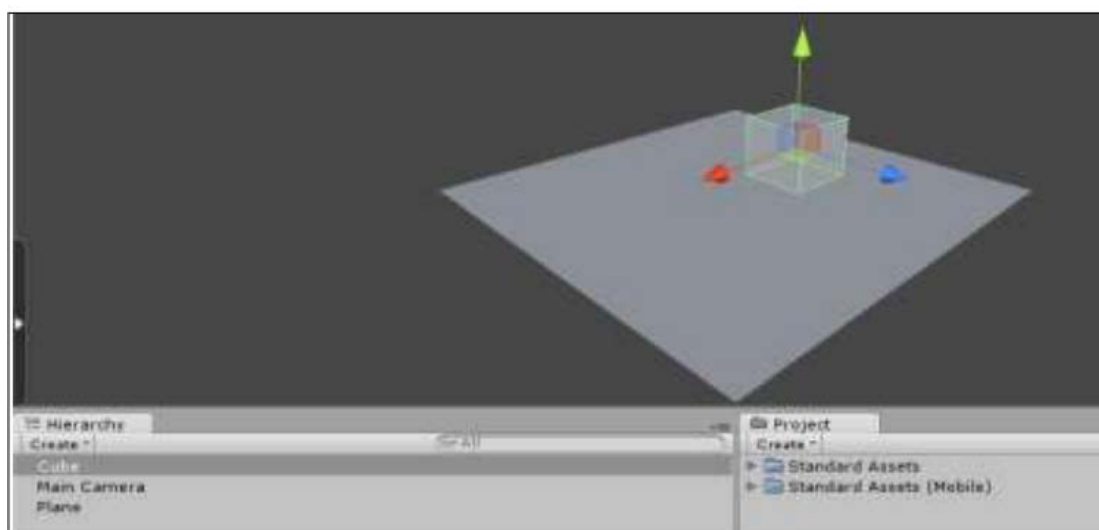


Рисунок 81

5. Как видно из рисунка, цвет куба теряется в пространстве сцены, т.е. сливается по цвету с вновь созданным объектом плоской поверхности. Для исключения этого необходимо создать для объекта освещение. Добавление к проекту точечного освещения объектов также осуществляется в том же меню (GameObject – Create Other – Point Light).

6. В сцене не хватает еще одного объекта – камеры. Для того, чтобы разместить камеру не просто в «пространстве среды», а на конкретной плоскости, необходимо в окне иерархии созданных объектов проекта указать созданную нами ранее плоскость, после чего создать камеру (GameObject – Create Other – Camera):

Воспользовавшись кнопкой перетаскивания объектов, расположим их примерно так, как это показано на рис. 82.

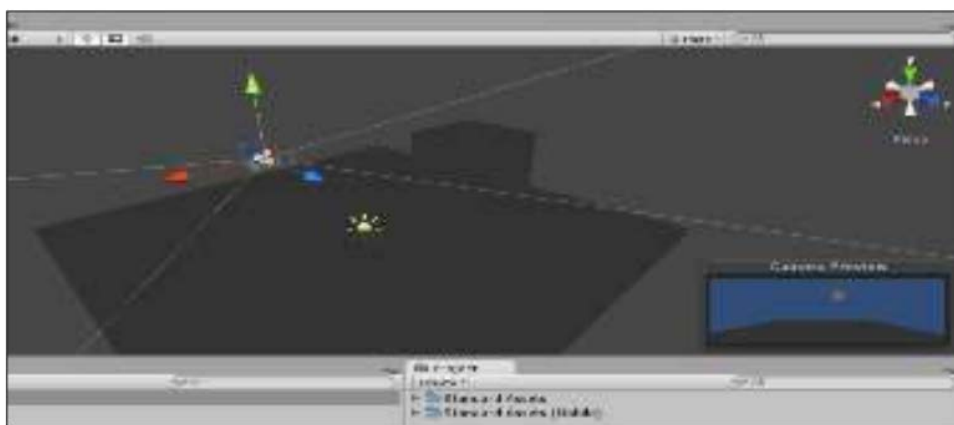


Рисунок 82

Теперь при нажатии управляющей кнопки “Play” в окне проекта Game отобразится статическая сцена, в которой кубический объект нависает над плоскостью (рис. 83).

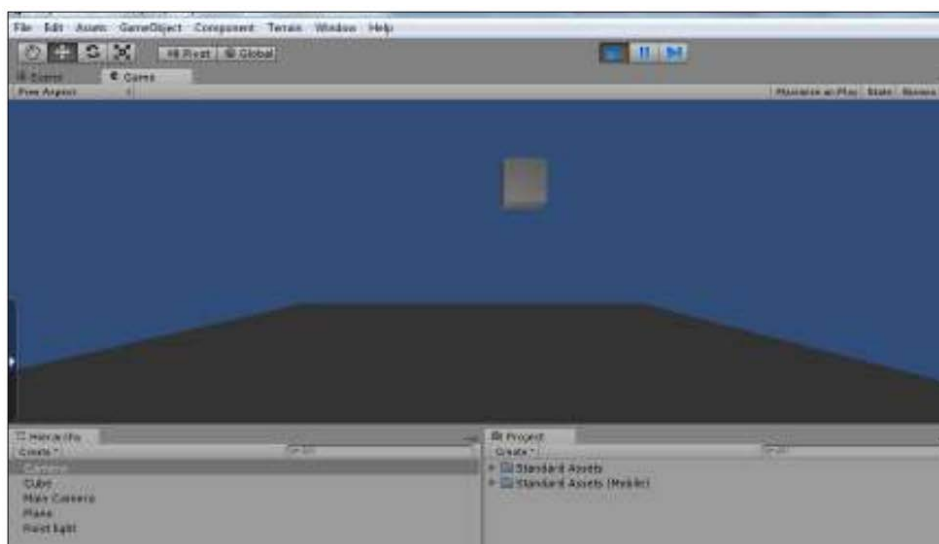


Рисунок 83

7. Для установления взаимодействия между созданными геометрическими объектами необходимо добавить к ним соответствующие физические свойства. Физический движок среды используют систему динамики твердых тел (Rigid Body), для создания реалистичного движения. Это означает, что вместо статичных объектов, находящихся в виртуальном пространстве сцены, мы имеем объекты, у которых могут быть следующие свойства: масса (Mass), гравитация (Use Gravity) и другие. Так, свойства массы или гравитации добавляются к объекту с помощью компонента Rigidbody (Component – Physics – Rigidbody). В соответствующих полях переменных этого компонента mass и drag в окне инспектора компонентов (Inspector) выставляются величины массы и скорости падения (притяжения к земле) согласно рис. 84, при этом в окне иерархии куб должен оставаться выделенным:

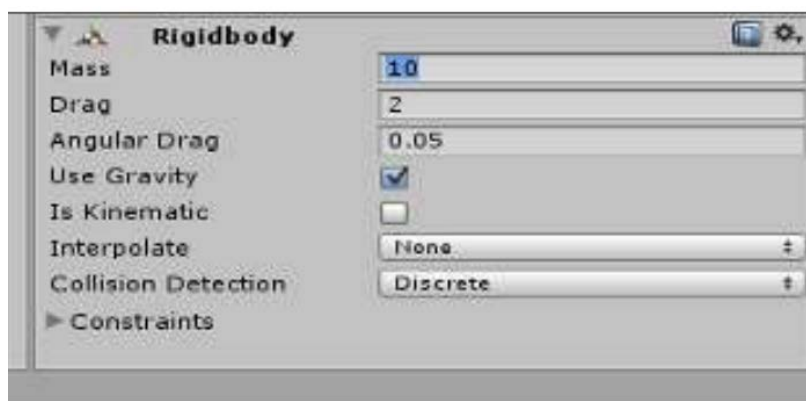


Рисунок 84

Теперь при запуске проекта во вкладке Game можно увидеть, как с высоты на созданную плоскость падает куб под действием силы тяжести. Причем падение это после соприкосновения с плоскостью прекращается, и куб остаётся на плоскости. Изменение физических свойства куба можно осуществить в компоненте `BoxCollider`, одним из параметров которого является материал (`Material`). По умолчанию значение этого параметра выставляется в значении `None`. Поскольку при создании проекта был подключен стандартный набор ресурсов `PhysicMaterials.UnityPackage`, то в проекте появляется возможность использования нескольких стандартных физических материалов `Unity3d`. Указание в качестве параметра упругого материала (`Bouncy`) позволяет добиться отскока куба от плоской поверхности за счет указанного физического свойства.

Задание 1. Создать физическую модель падающего шара, скатывающегося по наклонной поверхности.

1) Познакомьтесь с особенностями моделирования простейших трехмерных геометрических моделей в среде `Unity3d`.

2) Освойте приемы моделирования трехмерных геометрических объектов и их физических свойств с помощью компонентов `Transform`, `Rigidbody`, `BoxCollider` в новом проекте в среде `Unity3d`.

3) Освойте приемы организации взаимодействия объектов за счет добавления гравитации и массы к 3d-объектам в этом проекте.

4) Сохраните файл проекта и предоставьте преподавателю. Последовательность моделирования описать в плане-отчете по лабораторной работе.

Лабораторная работа 23

Основы взаимодействия (столкновения) между объектами. Применение скриптов на языке C#. Ввод данных с использованием скриптов

Цель работы: познакомится с особенностями разработки и внедрения скриптов на языке программирования `C#` в `Unity3d`; изучить способы организации взаимодействия трехмерных моделей, как с использованием

функций и скриптов Unity3d; освоить приемы организации взаимодействия объектов за счет столкновений между 3d-объектами на основе прикрепления скрипта на языке программирования C#.

Методические рекомендации

Collision detection – это способ, с помощью которого анализируется 3D-пространство сцены на предмет столкновений между объектами. Присваивая объекту компонент Collider, мы фактически размещаем вокруг него невидимую сетку – так называемый коллайдер, который имитирует форму объекта и информирует о наличии столкновения с другим объектом.

Информация о столкновении коллайдеров поступает в физический движок, который сообщает столкнувшимся объектам их дальнейшую реакцию на это столкновение, основанную на направлении и силе удара, скорости и других факторах. Отметим, что использование коллайдеров, повторяющих форму меша модели, с одной стороны, дает более точное определение столкновений, но в то же время приводит к увеличению затрат на их вычисление.

Рассмотрим особенности столкновений объектов как с использованием непосредственного функционала Collision detection, предоставляемого Unity3d.

Для этого добавим в созданную ранее сцену новый кубический объект, выполняющий роль некоторого препятствия для падающего куба, смоделированного ранее (GameObject - Create Other - Cube), и придадим ему форму параллелепипеда (рис. 85).

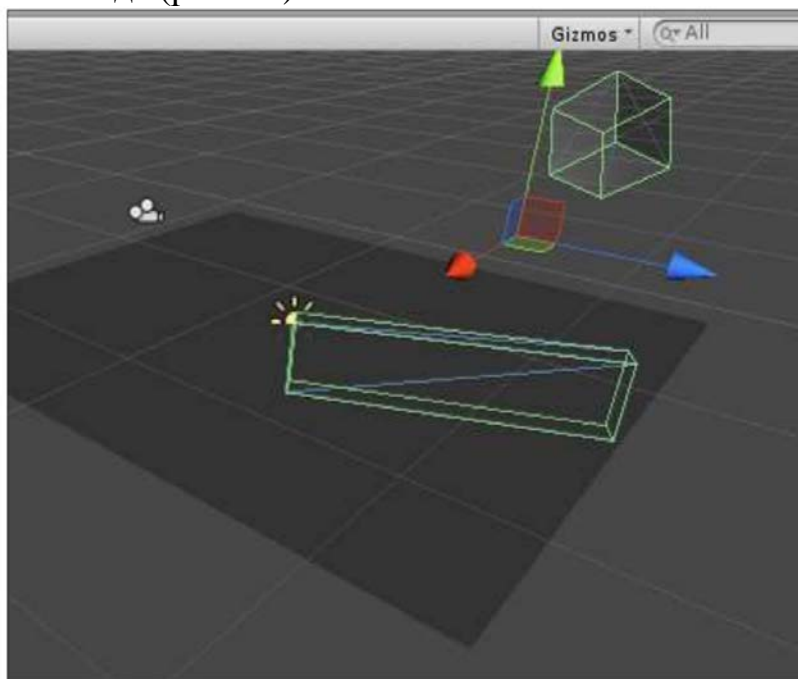


Рисунок 85

Чтобы лучше увидеть картину взаимодействия куба и прямоугольного препятствия, необходимо развернуть исходное положение падающего куба – на ребро. Вращения выбранного объекта (куба) осуществляется в Unity3d с

помощью инструмента «Rotate» (вращение).

Далее переключившись в режим просмотра Game, можно наблюдать следующую сцену (рис. 86):

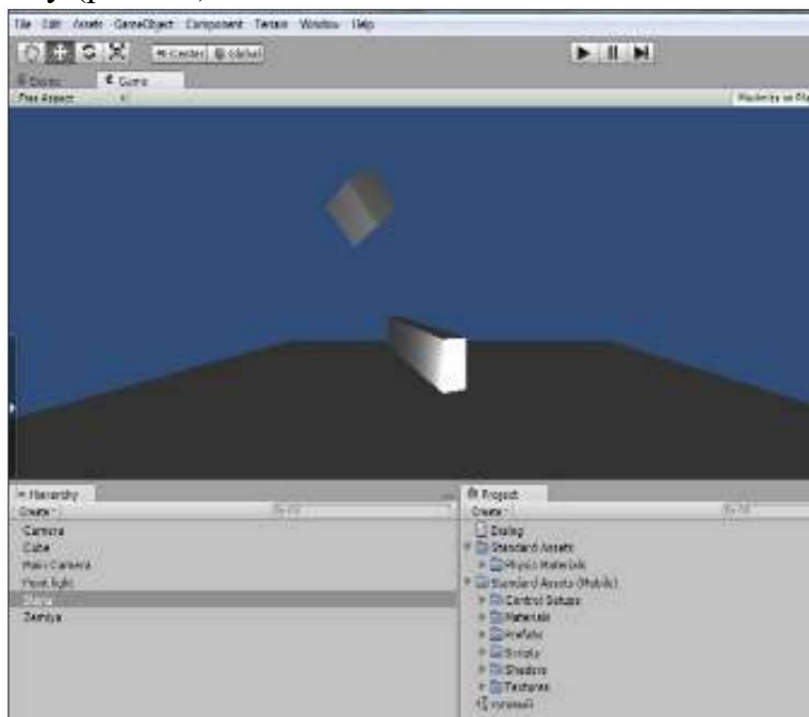


Рисунок 86 – Сцена взаимодействия объектов

Для определения столкновений необходимо отличать объекты по их названию. Переименовать объект «Plane» (Плоскость) в объект «Earth», а параллелепипед, представляющий стену (препятствие) в объект «Wall» можно непосредственно выбрав объект в окне иерархии и применив к нему команду «Rename» из контекстного меню, после чего задать новое имя объекта. Отметим, что среде Unity3d разрешается ввод русскоязычных символов, однако при работе со скриптами необходимо использовать латиницу.

Для того, чтобы заставить взаимодействовать между собой имеющиеся в сцене трехмерные модели, создадим скрипт на языке программирования C# и назовем его «Dialog». Для этого на вкладке ресурсов и префабов «Project» выбираем «Create – C# Script». Двойной щелчок позволяет открыть скрипт в редакторе скриптов MonoDevelop.

Создание скриптов – один из важнейших моментов в разработке игр. Код, написанный для использования в Unity, опирается на ряд готовых встроенных классов, о которых вы должны думать, как о библиотеках команд или поведения. При написании скриптов, вы будете создавать свои собственные классы, на основании существующих в Unity Engine. При создании нового скрипта в Unity3d автоматически создается каркас для C# скрипта, который на первых порах программирования не понадобится, и все его содержимое можно удалить. Рассмотрим следующий код:

```

using UnityEngine;
using System.Collections;
public class Dialog : MonoBehaviour {
    // Метод-функция, вызываемая при столкновении объектов
void OnCollisionEnter() {
    Debug.Log("Hit Something"); // Передаем сообщение в
консоль Unity
    }
}

```

Первые две строки подключают к скрипту используемые пространства имен. Далее необходимо запомнить, что главным классом в Unity3d является MonoBehaviour. Любой пользовательский скрипт (в описанном случае это Dialog) должен быть его наследником, и не спроста – ведь именно этот класс реализует интеграцию всех объектов в основной цикл программы. Именно это наследование позволяет пользовательскому скрипту (классу) выполнять роль компонента и быть привязанным к игровому объекту.

Здесь метод «OnCollisionEnter» определяет столкновение объекта с другими объектами. А статический метод «Log» класса «Debug» пишет сообщение "Hit Something" в консоль Unity.

После сохранения скрипта, добавляем его в качестве компонента для падающего куба. Для этого необходимо сначала выбрать соответствующий объект в окне иерархии и перетащить на него вновь созданный нами скрипт «Dialog». При этом обратите внимание, что добавленный скрипт также отобразится внизу в окне «Inspector» в качестве компонента объекта.

Теперь перейдя в режим Play можно наблюдать, что в тот момент, когда созданный объект куб коснется плоской поверхности, в консоли среды Unity3d (Window – Console) появится соответствующее сообщение. Заметьте, что такое сообщение будет выдаваться при каждом столкновении объектов. Причем последнее консольное сообщение отображается в статус (внизу окна).

Для выяснения того, с какими именно объектами столкнулся исходный объект, к которому прикреплен скрипт, необходимо использовать значение параметра класса «Collision», которое будет принимать метод «OnCollisionEnter». Открываем редактор скрипта и вставляем в него следующий код:

```

// Теперь метод принимает объект класса Collision, с которым
происходит столкновение
void OnCollisionEnter(Collision myCollision) {
    // определение столкновения с двумя разноименными объектами

```

```

if (myCollision.gameObject.name == "Earth") {
    // Обращаемся к имени объекта, с которым столкнулись
    Debug.Log("Stolknulsya s Earth");
} else if (myCollision.gameObject.name == "Wall") {
    Debug.Log("Stolknulsya s Earth ");
}
}

```

После открытия консоли (Window – Console) мы увидим, с какими именно объектами столкнулся объект. Таким образом Unity3d позволяет нам оценить возможности взаимодействия объектов внутри среды.

Задание. Измените скрипт Dialog на языке C# так, чтобы при столкновении куба со стеной происходило разрушение стены. Составить план-отчет по лабораторной работе (используйте функции Destroy(gameObject) и Destroy (collision.gameObject)).

Сохраните файл проекта и представьте преподавателю.

Лабораторная работа 24

Префабы. Копирование и удаление объектов среды в Unity3D

Цель работы: познакомиться с назначением префабов в среде Unity3d, особенностями их создания и взаимодействия, в частности, с целью их последующего удаления (разрушения).

Методические рекомендации

Префаб (Prefabs) – это конструкция подготовленных объектов и компонентов, предназначенная для их многократного использования в проекте. Экземпляр префаба может быть добавлен в любое количество сцен, а также многократно в одну сцену. Все экземпляры являются ссылками на оригинальный префаб и, фактически, его «клонами»; имеют те же свойства и компоненты, что и оригинальный объект.

Префабы полезны, когда нужно создать несколько экземпляров сложного объекта.

Прежде всего, необходимо добавить префаб в проект (Project – Create – Prefab). В результате на панели Project в окне проекта появится префаб с именем «New Prefab»; переименуем его в «UprugostCube». Обратите внимание, что на данном этапе ссылка на префаб на панели Project отображается серым цветом. Для того чтобы добавить объекты к префабам, достаточно выбрать объект в окне иерархии объектов и перетянуть его на соответствующий префаб («UprugostCube») на панели Project. Свойства созданного префаба «UprugostCube» описываются на панели инспектора, а его предварительный вид доступен в окне «Preview» (рис. 87).

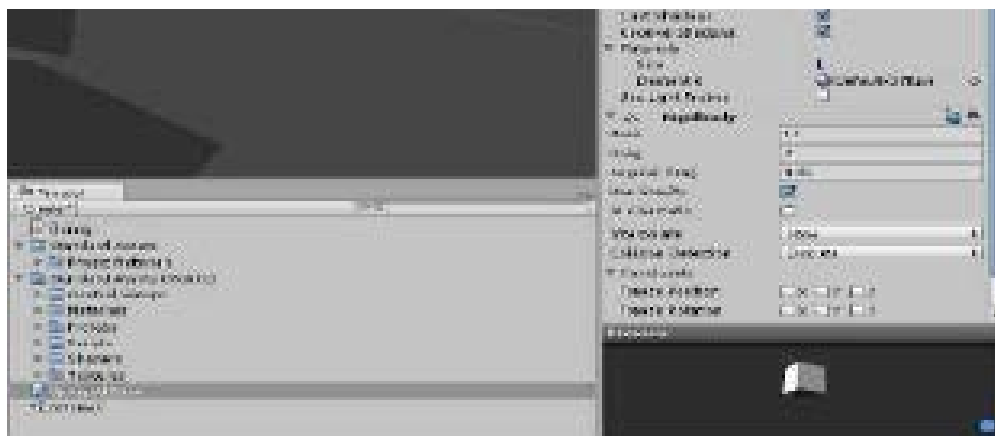


Рисунок 87 – Просмотр свойств префаба в unity3d

Для создания пары экземпляров префаба на плоскости, необходимо просто его перетащить. В результате на плоскости появятся два куба, а в окне «иерархии» добавится объект с именем соответствующего префаба «UprugostCube» (рис. 88).

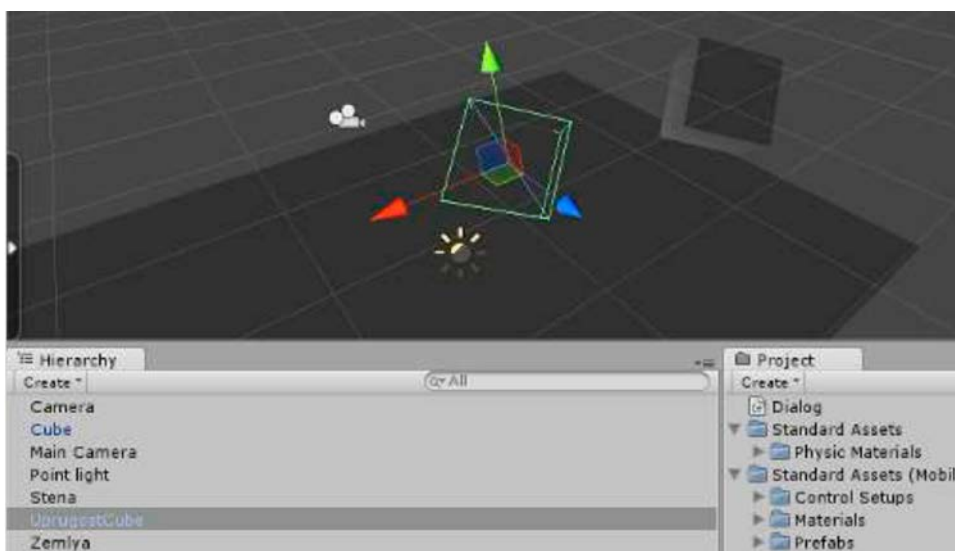


Рисунок 88

Переключившись в режим просмотра «Game» и запустив сцену (рис. 24), можно увидеть, что добавленный куб ведет себя точно так же, как и оригинальный, обладая одинаковыми физическими свойствами.

Уничтожение объектов

Рассмотрим, каким образом можно удалять объекты в Unity3d в скриптах на языке C#. Для этого в окне «Project» создадим новый скрипт с именем «Destroy» и откроем его в редакторе скриптов MonoDevelop.

Как уже говорилось, при создании C#-скрипта Unity создает некий каркас, состоящий из подключенных библиотек и основного класса (используемого скриптом) с методами Start() и Update().

В предыдущих случаях нами рассматривался метод Update, который вызывается каждый раз в новом кадре для каждого компонента всех объектов на сцене.

В данном случае мы воспользуемся методом Start(), который выполняется единожды для каждого компонента сразу после нажатия на кнопку «Play». и, соответственно, должен использоваться для инициализации переменных и придания им каких-то начальных значений. Добавим в тело метода Start() функцию Destroy() и передадим в нее gameObject, указав таким образом, что скрипт должен уничтожить объект, компонентом которого он является:

```
// метод Start() выполняется единожды, сразу после окончания загрузки сцены
Start() {
    // уничтожить объект, к которому прикреплен данный скрипт
    Destroy(gameObject)
}
```

Добавим этот скрипт к кубическому объекту, который должен удаляться, с помощью меню компонентов (рис. 25) Теперь после запуска сцены можно убедиться, что добавленный куб при запуске программы сразу пропадает.

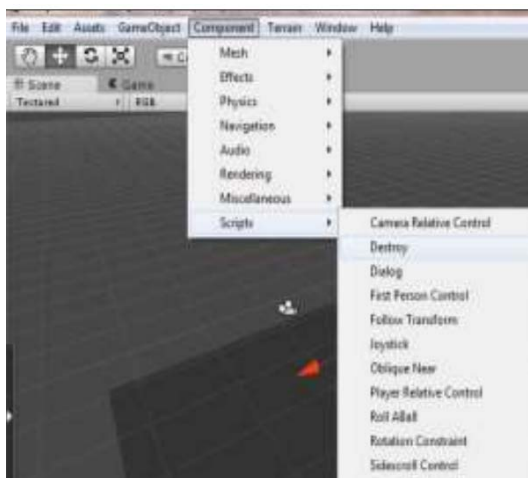


Рисунок 89 – Добавление скрипта C# к объектам с помощью меню компонентов

Теперь попробуем уничтожить другой объект с помощью его поиска в среде. Для этого воспользуемся статическим методом Find основного класса GameObject:

```
// ищем объект с именем Stena и если таковой есть – уничтожаем его Destroy(GameObject.Find("Stena"));
```

В случае необходимости уничтожения объекта не сразу, а спустя какое-то время, можно задать значение во 2-ой параметр функции Destroy:

```
// Теперь уничтожаем стену спустя 2 секунды после загрузки сцены Destroy(GameObject.Find("Stena"), 2);
```

Задание. Добавить в проект новый объект – сферу с физическими свойствами твердого тела из металлического материала. Наклонить поверхность и расположить объект (сферу) так, чтобы при падении с высоты происходил накат сферы на объекты среды, с их последующем удалением. Изменить материал сферы на упругий. Описать разницу.

Лабораторная работа 25

Работа с анимированными моделями в Unity

Цель работы: научиться импортировать и настраивать анимированные модели в Unity.

Методические рекомендации

Чтобы вставить анимированные 3D модели в Unity, необходимо выполнить следующие шаги:

Найдите и загрузите 3D модель в формате, поддерживаемом Unity (например, FBX).

Создайте новый проект в Unity и откройте окно проекта.

Перетащите 3D модель в окно проекта, чтобы импортировать ее в проект.

Выберите импортированную модель в окне проекта и перейдите в окно инспектора.

В окне инспектора установите настройки импорта для модели, такие как размер, масштаб, поворот и другие свойства.

Чтобы добавить анимацию, включите импорт анимаций для модели в настройках импорта.

Выберите модель в сцене Unity и перейдите в окно аниматора.

Создайте новый аниматор для модели и добавьте анимационные состояния для различных действий модели.

Импортируйте анимации в аниматор, перетащив их в окно аниматора.

Настройте параметры анимаций, такие как скорость, повторение и длительность.

Сохраните проект и запустите игру, чтобы убедиться, что модель и анимация работают должным образом.

Это лишь общие шаги для импорта анимированных 3D моделей в Unity, но они могут быть уточнены в зависимости от конкретных потребностей вашего проекта.

Задание. Импортировать анимированные модели и задать их взаимодействие.

Лабораторная работа 26

Анимация деформации

Цель работы: освоить использование параметров деформаторов для создания анимации.

Задание. Используя параметр деформаторов интенсивность, создать анимацию по образцу.

Методические рекомендации по выполнению: познакомиться с использованием настроек при анимации деформаторов.

Анимация значений

При использовании кнопки Запись активных объектов на панели анимации (F9), программа создает ключевой кадр на каждом значении объекта, характеризующем его положение, масштаб и поворот в пространстве. Проще говоря – записывает 9 ключевых кадров – по три на каждый атрибут (положение, вращение, масштаб - PSR). Можно сказать, что таким образом мы фиксируем (анимируем) положение, масштаб и вращение объекта. При этом другие параметры объекта остаются не тронутыми.



Рисунок 90

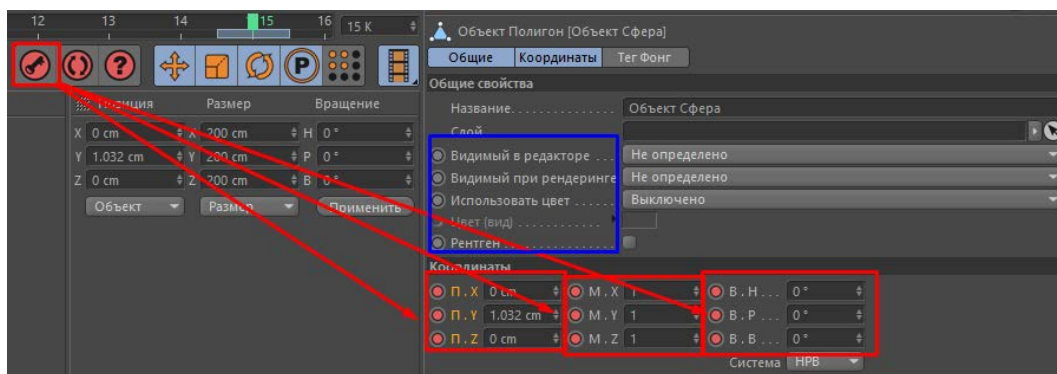


Рисунок 91

Ключевые кадры, значения которых не изменяются со временем, представляют собой прямые линии в режиме F-кривых и называются пустыми.

Для того, чтобы создать анимацию других параметров объекта (или инструмента, генератора и т.п.) – необходимо записывать ключевые кадры непосредственно в менеджере атрибутов на соответствующем параметре. Для этого достаточно нажать на кружок слева от параметра. Красный цвет кружка свидетельствует о том, что в этот момент времени записан ключевой кадр. При

этом ключевой кадр будет отражен и на шкале времени. На шкале времени эти ключи можно перемещать, масштабировать, копировать и удалять. Но! Изменять значения (параметры) – в менеджере атрибутов или на панели Таймлайн.

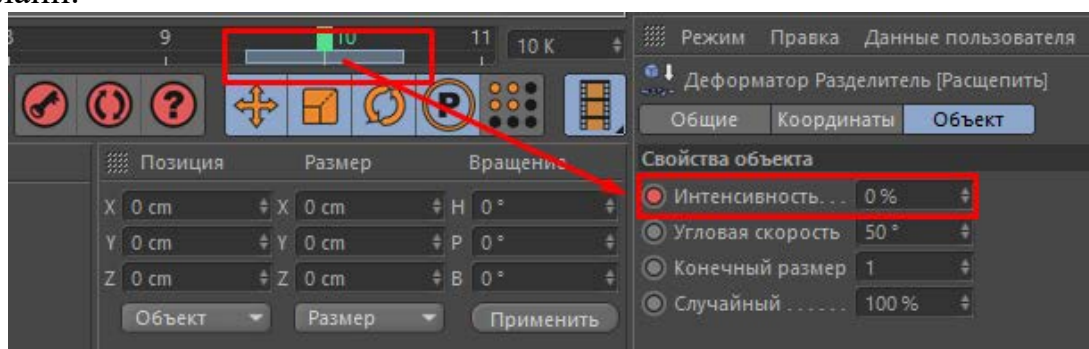


Рисунок 92

При создании анимации с использованием деформаторов – отдельно записываются (на панели анимации, например) ключевые кадры объекта (PSR) и отдельно (в менеджере атрибутов) – значения различных параметров деформатора (генератора, инструмента и т.д.). Все ключевые кадры отображаются на шкале времени при активном объекте.

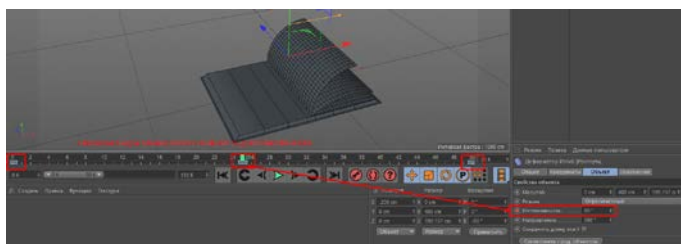


Рисунок 93

Работа с ключевыми кадрами параметров инструментов, деформаторов, тегов – аналогична работе с ключевыми кадрами объекта – PSR. Они обладают всеми свойствами КК – изменяемая интерполяция, цикличность и т.п. Кроме анимации параметров деформатора, можно анимировать и значения – PSR, как любого другого объекта. Помните, что можно применять несколько деформаторов к одному объекту и наоборот – один деформатор к нескольким объектам. Для этого важно правильно выстроить иерархическую связь объект – деформатор.

Имеются 3 режима, которые Деформатор может использовать при работе с объектами.

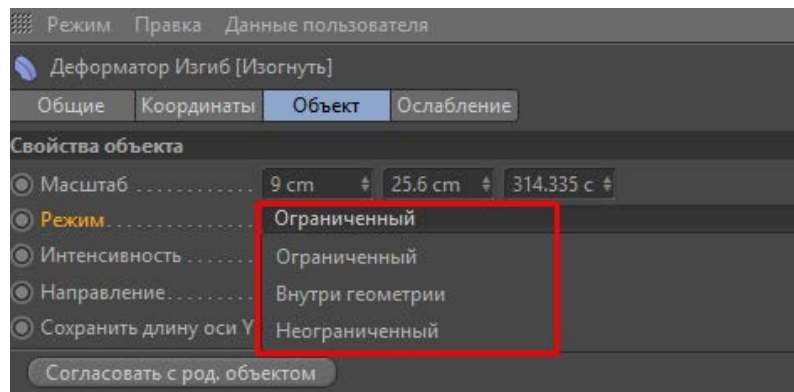


Рисунок 94

Ограниченный. Деформатор создаёт воздействие для всего объекта. Деформированы будут при этом только те части объекта, которые находятся в пределах границ синего бокса.

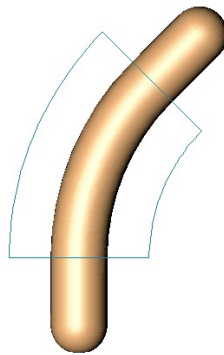


Рисунок 95

Внутри геометрии

Деформатор создаёт воздействие только в пределах синего бокса. Все поверхности, находящиеся за пределами этого каркаса, будут просто отсечены.

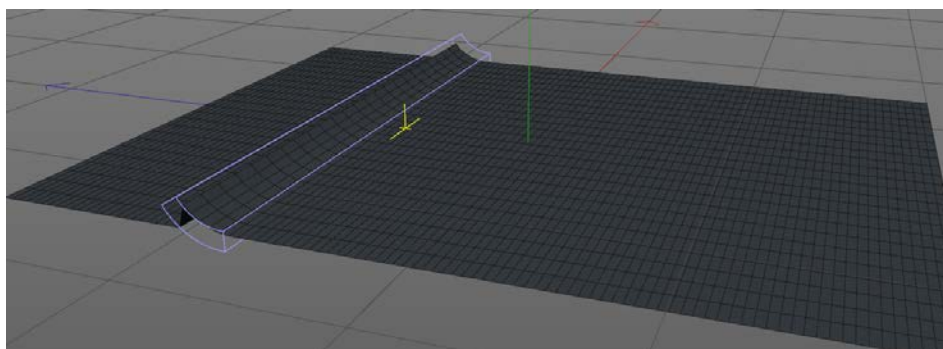


Рисунок 96

Неограниченный

Деформатор создаёт воздействие для всего объекта. Деформированы будут при этом также и те участки, которые находятся за пределами границ синего бокса. На основе этого параметра вы можете определить пространственную форму расширения для силового поля воздействия.

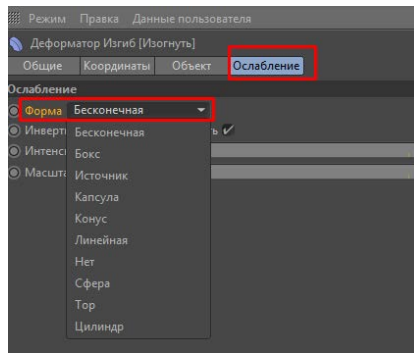


Рисунок 97

Здесь вы можете выбрать следующую форму:

Бесконечный

Поле воздействия при этом не имеет ограничения и имеет 100% влияния на всю сцену (объект).

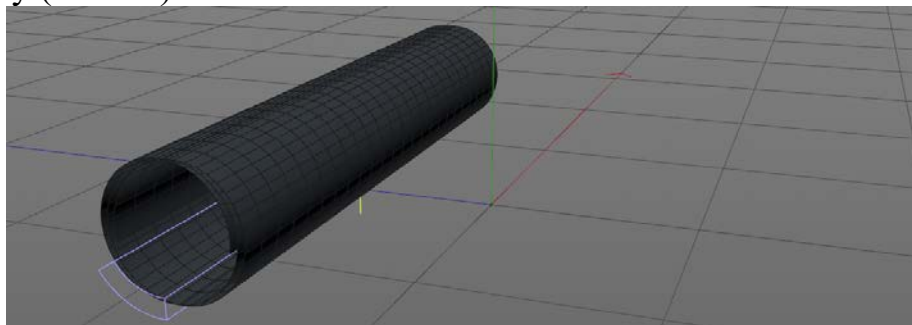


Рисунок 98

Капсула

Форма поля ослабления при этом соответствует форме примитива капсулы. В менеджере атрибутов при этом будут показаны оба параметра – Вверху и Внизу. Вы можете их изменять в менеджере атрибутов или в окне редактора программы, используя для этого специальные контроллеры (оранжевые точки). Оба этих параметра определяют верхний и нижний радиус капсулы.

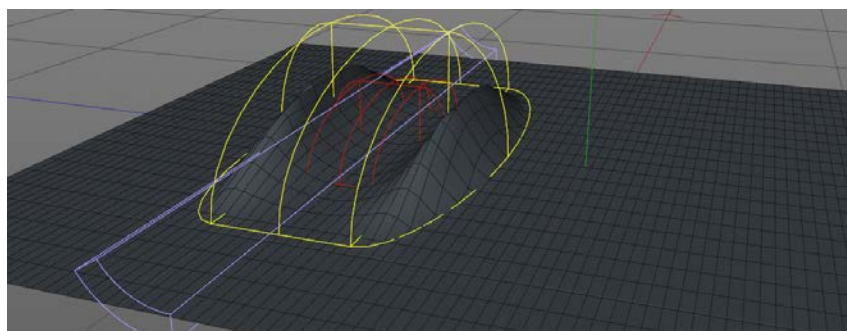


Рисунок 99

Бокс, Сфера, Цилиндр, Конус, Тор

Приводят к созданию соответствующих форм.

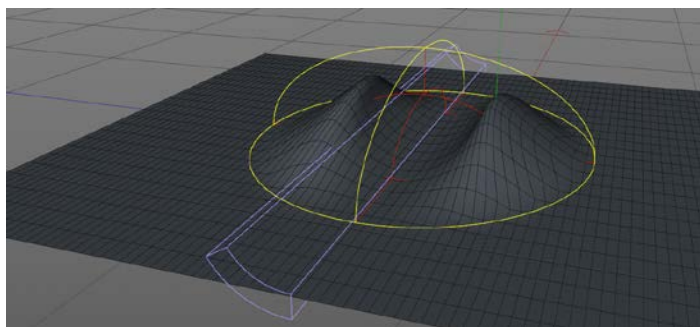


Рисунок 100

Нет. Этот режим является актуальным только для объекта Сустав. При его активном положении функция ослабления для объекта Сустав является не актуальной.

Линейный

Ослабление произойдёт по длине оси Z для поля воздействия. При этом имеется следующее правило: В отрицательном направлении Z за пределами поля воздействия, 100% влияния. В положительном направлении за пределами поля воздействия, 0% влияния.

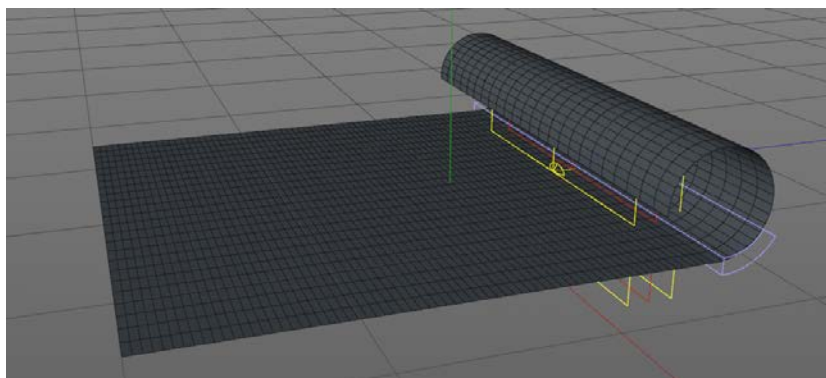


Рисунок 101

Источник

Возможно применение сплайнов, частиц и полигональных объектов. Воздействие в этом случае будет происходить на определённом расстоянии вокруг сплайна или частицы (или в случае полигонального объекта по периметру точек объекта).

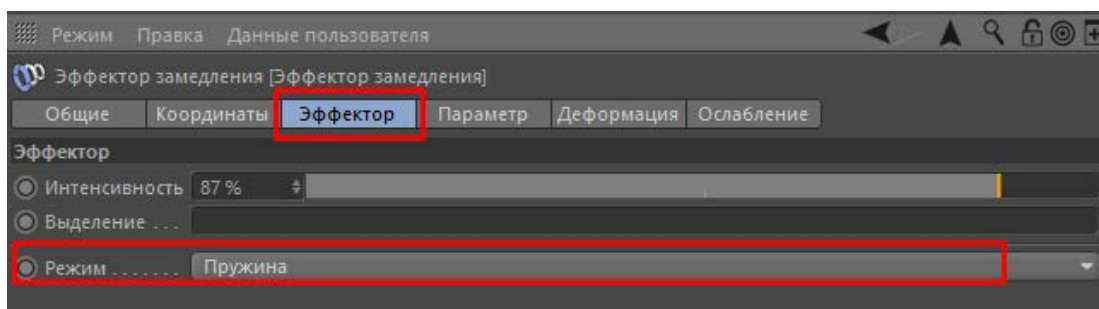


Рисунок 102

Лабораторная работа 27

Производство трехмерного видеоролика

Цель работы: закрепить полученные навыки по трехмерной анимации.

Задание. Разработать авторский видеоролик.

Требования: продолжительность (не более): 30-48 сек. Программная среда – программы трехмерного моделирования и композинга. Используйте сюжет из лабораторной работы 19.

Лабораторная работа 28

Представление трехмерного анимационного ролика

Цель работы: освоить технологию описания и представления трехмерного анимационного ролика.

Задание. Описать и представить выполненную работу по трехмерной анимации в стандартизированном виде.

Методические рекомендации по выполнению

Описать авторский видеоролик.

Стандартные требования к проектам (пример):

1. Хронометраж

Продолжительность (не более): 42сек

Из них:

Вступление – 6сек

Ролик – 20-30сек

Финальные титры – 6сек

Не допускается превышение тайминга на вступление и титры!

2. Требования к вступлению

Во вступлении должно быть обязательно указано:

– Название ВУЗа

– Номер группы автора

– Название ролика

– Автор ролика

Вся текстовая информация должна легко читаться.

3. Требование к финальным титрам

В титрах должно быть обязательно указано:

– Автор ролика

– Автор моделей/ источник моделей

– Автор текстур/ источник текстур

– Аниматор

- Используемое ПО
- Используемая музыка в формате «Автор – Название композиции»

Завершаются титры текстом:

© текущий год, аббревиатура ВУЗа, ФИО автора, номер группы автора
Вся текстовая информация должна легко читаться.

4. Технические требования к видео

Разрешение кадра: 1024x576 (HDTV, 16:9, Image aspect 1,77778)

Кадров в секунду (fps): 25

5. Технические требования к звуку

Звук в формате mp3, lame, OGG, wav.

В интернете можно легко найти эти кодеки (ogg, lame – бесплатные)

6. Используемое ПО

Используйте только 3ds max 11.

Использование других версий 3ds max запрещено!

7. Рендер

Mental Ray или Default Scanline Render. Это стандартные рендеры, встроенные в 3ds max. Использование других (внешних) рендеров запрещено!

8. Сохранение результатов рендера

Результат рендера должен сохраняться в последовательность кадров (image sequence).

Формат файлов: JPG (максимальное качество).

9. Сборка последовательности кадров

Для сборки используется программа Virtual Dub. Это простая, небольшая и бесплатная программа. Скачать можно отсюда: <http://www.virtualdub.org/>

10. Требования к сжатию

Сжатие Xvid MPEG-4 – это простой, бесплатный, широко распространенный кодек.

Качество – 1.

Идет в составе K-Lite Codec Pack (бесплатная сборка). Скачать можно отсюда: http://www.codecguide.com/download_kl.htm

11. Требования к текстурам

Исходные текстуры в формате PSD (со слоями).

Финальные текстуры в формате JPG (максимальное качество).

12. Требования к именам файлов

Простые, короткие и понятные имена с цифровыми суффиксами. Только на английском языке (латиница). Без пробелов и спецсимволов. Допускается использование букв латинского алфавита, цифр, знаков _ и – Относится ко всем файлам в проекте (текстуры, модели и т.д.).

13. Название финального файла ролика (только латинские символы):

фамилия_имя_группа_год_«название ролика».avi

14. Требования к раскадровке

Один кадр должен занимать не менее ½ листа А4.

Все кадры должны быть пронумерованы и содержать приблизительную информацию о времени (тайминг). Под кадром может находиться любая дополнительная текстовая и графическая информация.

Финальные файлы (фото – 2048пикс в длину минимум, скан – 72-150dpi) с раскадровкой сохранить в формат JPG (максимальное качество) в папку «Storyboard».

Вся текстовая информация должна легко читаться.

Названия файла раскадровки:

№ п/п фамилия имя группа год «название ролика».avi

До 10 файлов раскадровки нумерация: 1, 2, ..., 9

10 и более файлов раскадровки: 01, 02, ..., 10, 11, ...

15. Таблица трудозатрат

Желательно использовать MS Excel или Google Docs. Расширение файлов – *.xls

Название файла с таблицей трудозатрат:

фамилия_имя_группа_год_«название ролика»_work.xls

Столбцы:

1. № п/п
2. Наименование работы (на русском)
3. Ф.И. Автора (на русском)
4. Краткое описание работы (название модели, текстуры, анимации)
5. Размер текстуры в пикселях (только для текстур)
6. Кол-во треугольников (только для моделей)
7. Продолжительность анимации в секундах (только для анимации)
8. Временные затраты (в часах, округление до 1 часа)

Временные затраты (в часах) должны быть просуммированы в конце.

В финале файл должен находиться в папке Docs.

16. Сценарий

Желательно использовать MS Word или Google Docs. Расширение файлов – *.doc, *.rtf

Краткое, не более 1 листа, текстовое описание проекта.

Название файла сценария:

фамилия_имя_группа_год_«название ролика»_scenario.xls

В финале файл должен находиться в папке Docs.

17. Структура папок в проекте

Название корневой папки (латиница):

фамилия_имя_группа_год_«название ролика»

В ней, в корне, находится финальный ролик. В корневой папке находятся

следующие папки:

1. src – все текстуры/ модели используемые в проекте.
2. docs – сценарий и таблица трудозатрат. Документы в формате doc или rtf.
3. storyboard – раскадровка для ролика
4. wip – все временные модели (стадии, «мусорка»)
5. ref – референс фото, изображения и т.д.

Предоставить преподавателю:

Ролик

Все текстуры и исходники в PSD

1. Все модели и стадии работы над ними (не менее 3-х стадий)
2. Сценарий, раскадровка и таблица трудозатрат
3. Референс

По моделям – нужна исходная сцена в 3d или Unity, которая работает. Все должно работать, никаких потерянных текстур, анимации, моделей и т.д.

4. РАЗДЕЛ КОНТРОЛЯ ЗНАНИЙ

4.1 Задания для контролируемой самостоятельной работы студентов

Самостоятельная работа студентов направлена на совершенствование их умений и навыков по дисциплине «Технологии 3D анимации». Цель самостоятельной работы студентов – способствование усвоению в полном объеме учебного материала дисциплины через систематизацию, планирование и контроль собственной деятельности. Преподаватель дает задания по самостоятельной работе и регулярно проверяет их исполнение.

Содержание и формы контролируемой самостоятельной работы студентов рекомендуется непосредственно связывать с использованием метода проектов, что позволяет реализовывать индивидуальный подход к обучению. В ходе работы над проектами студенты лучше углубляются в предметную область. В результате каждый студент создает в процессе самостоятельной работы несколько проектов (выполняет моделирование и анимацию заданных объектов, производство видеоролика и т. п.) под руководством преподавателя. Такая организация работы способствует развитию как информационной, так и профессиональной компетенции.

Вопросы и творческие задания

1. Тема 2 Создание 3D анимации (2 часа)

1. Для лампы из лабораторной работы Прямая кинематика попробуйте дополнительно разнообразить движения лампы, заставив ее покрутить «головой» из стороны в сторону, по-гусиному вытянуть вперед «шею», оглянуться «через плечо».

Форма отчета – электронная версия (файл).

1. Тема 2 Создание 3D анимации (2 часа)

1. Создайте на основе системы частиц водопад.

Форма отчета – электронная версия (файл).

2. Создайте эффект разлетающегося текста при помощи частиц.

Форма отчета – электронная версия (файл).

3. Тема 2 Создание 3D анимации (2 часа)

1. Создайте на основе системы частиц снег и метель.

Форма отчета – электронная версия (файл).

4. Тема 2 Создание 3D анимации (2 часа)

1. Создайте простого лего робота и анимируйте его, используя Кости.
Форма отчета – электронная версия (файл).

5. Тема 3. Динамические взаимодействия (2 часа)

1. Создайте сцену с двумя роботам, добавьте динамику взаимодействия между объектами, например, движение, столкновение, вращение и т.д. (с помощью физической симуляции в программе или путем создания анимации).

2. Используя рендер Arnold, визуализируйте сцену.

Форма отчета – электронная версия (файл).

6. Тема 3. Динамические взаимодействия (2 часа)

1. Реализуйте одну из сцен:

– мяч и стенка, о которую он ударяется;

– кубики домино, которые падают друг на друга;

– автомобиль, который движется по дороге и сталкивается с препятствием;

– кубик, который вращается и сталкивается с другими объектами на сцене.

Форма отчета – электронная версия (файл).

4.2 Перечень контрольных вопросов по дисциплине

Тема 1. Теория анимации

1. Направления использования трехмерной анимации в культуре и искусстве.

2. История анимации: от рисованной до трехмерной.

3. Виды анимации (рисованная классическая, пластилиновая, кукольная, перекладочная, компьютерная (2D и 3D)).

4. Программное обеспечение для создания трехмерной анимации.

5. Технологии анимирования: процедурная, программируемая и анимация по ключевым кадрам.

6. Способы анимации: анимация по ключевым кадрам; анимация по траектории; создание анимации при динамических симуляциях; анимация, полученная методом захвата движения и др.

7. Этапы создания анимированной сцены.

Тема 2. Создание 3D анимации

1. Базовые инструменты управления анимацией.

2. Панели управления анимацией.

3. Шкала треков. Редактор треков Track View, два режима: редактор

кривых Curve Editor, диаграмма ключей (редактор дескрипторов) Dope Sheet.

4. Временная шкала. Анимация по ключевым кадрам.
 5. Изменение продолжительности анимации.
 6. Изменение характера протекания анимации.
 7. Создание анимации в режиме автоматической установки ключевых кадров.
 8. Создание анимации в режиме ручной установки ключевых кадров.
 9. Параметрическая анимация.
 10. Контроллеры анимации. Виды контроллеров. Контроллеры, основанные на ключах. Процедурные контроллеры.
 11. Дополнительные техники анимации.
 12. Скелеты, кинематика и иерархия.
 13. Анимация связанных объектов.
 14. Связывание объектов в иерархические цепочки. Окно Schematic View.
 15. Проверка расположения опорных точек связанных объектов.
- Контроль параметров и настройка связей.
16. Анимация по методу прямой кинематики.
 17. Анимация по методу обратной кинематики. Способы реализации.
 18. Специализированные контроллеры обратной кинематики (IK solvers).
 19. Встраиваемая система костей. Создание систем объектов типа Bones (Кости).
 20. Анимация персонажа.
 21. Анимация формы (морфинг).
 22. Анимация материалов.

Тема 3 Динамические взаимодействия

1. Анимация с учетом законов физики.
2. Моделирование физики твердых и мягких тел.
3. Создание реалистичных физических симуляций.
4. Наборы инструментов для добавления в проект реалистичных физических симуляций в различных программах трехмерного моделирования.
5. Динамические тела, их атрибуты.
6. Категории динамических объектов: тела (твердые и деформируемые), частицы, нити и жидкости. Активные и пассивные тела.
7. Внешние силы, поля.
8. Ограничения.
9. Определение границ соударения объектов.
10. Столкновение с последующим разрушением.
11. Имитация ткани.
12. Динамика жидкости.

13. Анимация систем частиц. Источники частиц. Создание эффектов с частицами.

14. Деформации пространства (силы).

15. Визуализация симуляции.

16. Запекание результатов симуляции.

Тема 4. Визуализация анимированной сцены

1. Выстраивание композиции кадра анимированной сцены.

2. Анимация камер. Перемещение по прямолинейной траектории.

Осмотр сцены.

3. Анимация камер. Слежение за движущимся объектом.

4. Анимация камер. Облет по криволинейной траектории.

5. Рендеринг анимированной сцены.

6. Настройки рендеринга секвенции.

7. Настройки семплирования.

8. Оптимизация времени рендера.

9. Использование пресетов.

10. Эскизный и финальный рендер анимированной сцены.

11. Сборка последовательности кадров.

12. Пакетный и сетевой рендеринг.

13. Постобработка.

Тема 5. Создание анимационного ролика

1. Спецификация. Требования к ролику.

2. Этапы создания анимационного ролика.

3. Постобработка ролика.

4. Презентация ролика.

Тема 6. Игровая анимация

1. Отличие CGI-анимации от игровой.

2. Особенности создания 3D контента для игр.

3. Поиск референса. Библиотеки ассетов.

4. Требования к 3D модели для компьютерной игры.

5. Топология модели.

6. Разработка высокополигональных и низкополигональных моделей.

Бюджет полигонов.

7. UV-развертка и текстурирование 3D моделей для компьютерных игр.

8. Теория физического шейдинга (Physically Based Shading, PBS).

9. Карты текстур Diffuse, Normal, Specular и Occlusion.

10. Импорт 3D модели (статической и анимированной) в игровой движок.

11. Сбор игровой локации. Расстановка камер.
12. Настройка анимации. Алгоритмы манипулирования объектами и управление анимационной секвенцией.
13. Настройка текстур и шейдинга.
14. Настройка освещения.
15. Тестирование игровой локации.

4.3 Перечень вопросов к зачету

1. Направления использования трехмерной анимации в культуре и искусстве.
2. История анимации.
3. Виды анимации.
4. Программное обеспечение для создания трехмерной анимации.
5. Технологии и способы анимирования.
6. Этапы создания анимированной сцены.
7. Базовые инструменты управления анимацией.
8. Анимация по ключевым кадрам.
9. Параметрическая анимация.
10. Контроллеры анимации. Виды контроллеров.
11. Дополнительные техники анимации. Скелеты, кинематика и иерархия.
12. Анимация связанных объектов.
13. Анимация по методу прямой кинематики.
14. Анимация по методу обратной кинематики.
15. Специализированные контроллеры обратной кинематики (IK solvers).
16. Анимация персонажа. Встраиваемая система костей.
17. Анимация формы (морфинг).
18. Анимация материалов.
19. Анимация с учетом законов физики. Моделирование физики твёрдых и мягких тел.
20. Создание реалистичных физических симуляций в различных программах трехмерного моделирования.
21. Категории динамических объектов: тела (твердые и деформируемые), частицы, нити и жидкости. Активные и пассивные тела, их атрибуты.
22. Внешние силы, поля. Ограничения. Деформации пространства (силы).
23. Определение границ соударения объектов.
24. Динамические взаимодействия.
25. Имитация ткани.
26. Динамика жидкости.
27. Анимация систем частиц. Создание эффектов с частицами.

28. Визуализация симуляции. Запекание результатов симуляции.
29. Анимация камер.
30. Рендеринг анимированной сцены. Настройки рендеринга секвенции.
31. Оптимизация времени рендера. Эскизный и финальный рендер анимированной сцены.
32. Пакетный и сетевой рендеринг.
33. Постобработка.
34. Спецификация и технические требования к видеоролику.
35. Отличие CGI-анимации от игровой.
36. Особенности создания 3D контента для игр. Требования к 3D модели для компьютерной игры.
37. Топология модели. Высокополигональные и низкополигональные модели. Бюджет полигонов.
38. Теория физического шейдинга (Physically Based Shading, PBS).
39. Импорт 3D модели (статической и анимированной) в игровой движок.
40. Настройка анимации. Алгоритмы манипулирования объектами и управление анимационной секвенцией.
41. Сбор игровой локации. Тестирование игровой локации.

Практические задания:

1. Защита лабораторных работ.
2. Выполнение КСР.

4.4 Критерии оценки результатов учебной деятельности студентов

Для выявления и исключения пробелов в знаниях студентов рекомендуется использовать следующие средства:

- 1) фронтальный опрос на лекциях, лабораторных и семинарских занятиях;
- 2) критериально-ориентированные тесты для контроля теоретических знаний современных технологий трехмерного моделирования и анимации, основных определений, терминологии и методов анимации трехмерных моделей и их визуализации (рендеринга).
- 3) выполнение тестовых заданий с произвольной формой ответа для контроля умения анализировать и грамотно излагать и формулировать свои соображения и выводы в данной предметной области;
- 4) выполнение творческих заданий, которые предполагают эвристическую деятельность и поиск неформальных решений.

5. ВСПОМОГАТЕЛЬНЫЙ РАЗДЕЛ

5.1 Учебная программа

Технологии 3D анимации: Учебная программа по специальности 1-21 04 01 Культурология (по направлениям) направления специальности 1-21 04 01-02 Культурология (прикладная) специализации 1-21 04 01-02 04 Информационные системы в культуре ре / [сост.: С. А. Гончарова]. – Минск : БГУКИ, 2022. – 11 с., вкл. обл. : табл. ; 20x15 см. – Библиогр.: с. 9 (8 наим.).

5.2 Учебно-методическая карта учебной дисциплины для дневной формы получения высшего образования

Название раздела, темы	Количество аудиторных часов				Количество часов УСР	Форма контроля знаний
	Лекции	Семинарские занятия	Лабораторные занятия	Иное		
Введение. Тема 1. Теория анимации	2					УО
Тема 2. Создание 3D анимации	2		20			
Тема 3. Динамические взаимодействия			6			Реферат
Тема 4. Визуализация анимированной сцены	2		4			
Тема 5. Создание анимационного ролика			8			Проект
Тема 6. Игровая анимация			18			
Всего...	6		56			

5.3 Список основной литературы

1. Ложкина, Е. А. Проектирование в среде 3ds Max : учебное пособие / Е. А. Ложкина, В. С. Ложкин. – Новосибирск : НГТУ, 2019. – 180 с. – ISBN 978-5-7782-3780-3. – Текст : электронный // Лань : электронно-библиотечная система. – Режим доступа : <https://e.lanbook.com/book/152241> (дата обращения: 31.03.2022).

2. Никитиных, Е. И. Анимация графических объектов и зависимостей в программе Autodesk 3ds Max : учебное пособие / Е. И. Никитиных. – Москва : РГУ им. А.Н. Косыгина, 2019. – 74 с. – ISBN 978-5-87055-692-5. – Текст : электронный // Лань : электронно-библиотечная система. – Режим доступа : <https://e.lanbook.com/book/167017> (дата обращения: 21.06.2022).

3. Хохлов, П. В. Технологии трехмерного моделирования и визуализации изображений в визуализаторе Арнольд (Arnold, 3ds Max) : учебное пособие / П. В. Хохлов, В. Н. Хохлова ; RU. – Новосибирск : СибГУТИ, 2021. – 160 с. – Текст : электронный // Лань : электронно-библиотечная система. – Режим доступа : <https://e.lanbook.com/book/257282> (дата обращения: 03.10.2022).

5.4 Список дополнительной литературы

1. Васильков, Э.И. Самоучитель по программе 3D Studio Max. Часть 3. Анимация / Э.И. Васильков. – М.: Академия медиаиндустрии, 2014. – 76 с. – Режим доступа : <http://www.ipk.ru/ftpgetfile.php?id=155> (дата обращения: 03.10.2022).

2. Васильков, Э.И. Самоучитель по программе 3D Studio Max. Часть 4. Системы частиц / Э.И. Васильков. – М.: Академия медиаиндустрии, 2014. – 95 с. – Режим доступа : <http://www.ipk.ru/ftpgetfile.php?id=156> (дата обращения: 03.10.2022).

3. Васильков, Э.И. Самоучитель по программе 3D Studio Max. Часть 5. Динамика в трехмерных сценах / Э. И. Васильков. – М.: Академия медиаиндустрии, 2014. – 83 с. – Режим доступа : <http://www.ipk.ru/ftpgetfile.php?id=157> (дата обращения: 03.10.2022).

4. Семькин, В. Простота движений: принципы создания игровой анимации [Электронный ресурс] / В. Семькин Gamedev-DTF. – 2019. – Режим доступа: <https://dtf.ru/gamedev/50227-prostota-dvizheniy-principyu-sozdaniya-igrovooy-animacii>.

5. Особенности создания 3d моделей для игровых движков [Электронный ресурс] / 2D-model.net. – Режим доступа: <https://3d-model.net/uroki-videokursi/3d-grafika/6041-osobennosti-sozdaniya-3d-modeley-dlya-igrovyyh-dvizhkov.html>.

5.5 Перечень рекомендуемых электронных образовательных ресурсов

1. 3ds Max. Support and leaning [Электронный ресурс] / Autodesk leaning network. – Режим доступа: <https://knowledge.autodesk.com/support/3ds-max/learn>.
2. Arnold for 3DS Max User Guide [Электронный ресурс] / <https://docs.arnoldrenderer.com/display/A5AF3DSUG>.
3. 3dtotal – ресурс по цифровому искусству [Электронный ресурс]. – – Режим доступа: <https://3dtotal.com>.
4. RENDER.RU – ресурс по 3D графике и анимации. [Электронный ресурс]. – Режим доступа: <http://www.render.ru>.
5. Библиотека текстур для художников [Электронный ресурс]. – – Режим доступа: <https://www.3d.sk>.