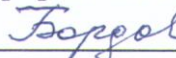


Министерство образования и науки Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Физический факультет
Кафедра астрономии и космической геодезии

ДОПУСТИТЬ К ЗАЩИТЕ В ГЭК
Руководитель ООП
д-р физ.-мат. наук, профессор
 Т.В. Бордовицына
« 2 » июня 2017 г.

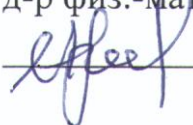
МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

ПРОГРАММНЫЙ КОМПЛЕКС SPACE & PLAYS ДЛЯ ВИЗУАЛИЗАЦИИ ОРБИТАЛЬНОГО ДВИЖЕНИЯ

по основной образовательной программе подготовки магистров
направление подготовки
09.04.02 — Информационные системы и технологии


Тырышкин Дмитрий Сергеевич

Научный руководитель
д-р физ.-мат. наук, профессор

 В.А. Авдюшев

Автор работы

студент группы № 517м

 Д.С. Тырышкин

РЕФЕРАТ

Квалификационная работа на степень магистра 27 с., 26 рисунков,
8 источников.

ПРОГРАММНЫЙ КОМПЛЕКС SPACE & PLAYS ДЛЯ ВИЗУАЛИЗАЦИИ ОРБИТАЛЬНОГО ДВИЖЕНИЯ

В работе предлагается новое программное приложение Space & Plays для визуализации орбитального движения на игровом движке Unity 3D. Описываются основные функции программы, приемы работы с ней и ее архитектура. В заключении перечислены основные результаты работы, а также обсуждаются перспективы развития.

ОГЛАВЛЕНИЕ

Введение	4
1. Unity 3D	6
2. Space & Plays. Ephemerid animator	8
2.1. Инструкция по использованию	8
2.2. Меню «Bodies»	13
2.3. Добавление новых объектов на сцену	17
2.4. Ручное управление камерой	19
3. Архитектура приложения	20
3.1. Структура классов	20
3.2. Интерполяция движения камеры и тел	22
3.3. Алгоритмы автоматизации движения камеры	24
Заключение	26
Список используемых источников	27

ВВЕДЕНИЕ

При моделировании и изучении орбитального движения еще на этапе постановки задачи у исследователя часто возникает необходимость предварительного знакомства с предметом исследования, когда на тот момент уже имеются полученные каким-либо образом эфемериды небесного тела. Информация, заложенная в эфемеридах, вполне достаточная, для того чтобы хотя бы приближенно установить закон движения космического объекта, а также представить его орбиту. Проблема, однако, состоит в том, что у исследователя обычно не оказывается под рукой подходящей и удобной в использовании программной среды для визуализации орбитальной динамики на основе имеющихся эфемерид. Цель квалификационной работы как раз состояла в создании такой программной среды (рабочее название — Space & Plays) на игровом движке Unity 3D.

Для достижения поставленной цели необходимо было выполнить ряд задач: 1) сформулировать общую концепцию предлагаемого приложения Space & Plays; 2) программно реализовать его прототип; 3) протестировать прототип и 4) определить дальнейшее направление в его развитии. Приложение Space & Plays должно отвечать следующим требованиям: 1) обработка входных данных в различной форме; 2) интуитивно понятный и удобный интерфейс; 3) современная графика; 4) высокая степень автоматизации.

Разумеется, существуют также и другие программы для визуализации орбитального движения. Браузерное приложение «Solar System Scope» [1] является частью большого проекта сайта «spacegid.com» по популяризации астрономии. Впрочем, оно непригодно для серьезных научных исследований. Приложение «Eyes on the Solar System» [2] ведомства NASA отображает большое количество объектов Солнечной системы, а также множество искусственных летательных аппаратов, содержит подробную справку обо всех представленных объектах. Так же, как и «Solar System Scope», приложение предназначено для популяризации астрономии. Приложение «Universe Sandbox» [3] позволяет моделировать множество астрономических явлений, добавлять новые объекты, имеет высокое качество визуализации, однако рассчитывает орбиты самостоятельно, используя упрощенную модель сил, что ограничивает её применение в научных целях. Кроме того, для поставленных в нашей работе задач у программы неудобный интерфейс. Следует особо выделить программную среду для визуализации Солнечной системы «Solar Suit Display» [4], разработанную в Томском госуниверситете. Примечательно, что разработчикам удалось совместить визуальную и научную составляющую. В

то же время приложение не является заранее запрограммированной, жёсткой моделью Солнечной системы. Оно может обновляться и дополняться пользователями. На данный момент доступна только Alpha версия от 2014 года.

Между тем все представленные программы не позволяют одновременно выполнять высококачественную визуализацию в виде текстурированных моделей, выглядящих «как настоящие», а также работать с объектами не только Солнечной системы, но и других орбитальных систем, в том числе описываемых модельными задачами небесной механики. Таким образом, в рамках проекта Space & Plays мы попытались реализовать все эти возможности в единой программной среде.

1. UNITY 3D

Unity — это программа для создания двух- и трёхмерных приложений и игр (Рисунок 1). Созданные с её помощью приложения работают под различными операционными системами Windows, OSX, WindowsPhone, Android, AppleiOS, Linux, а также на игровых приставках Wii, PlayStation 3, PlayStation 4, Xbox 360, XboxOne. Можно создавать приложения для работы в браузерах с помощью специального дополнительного модуля Unity (UnityWebPlayer), а также с поддержкой технологии WebGL.

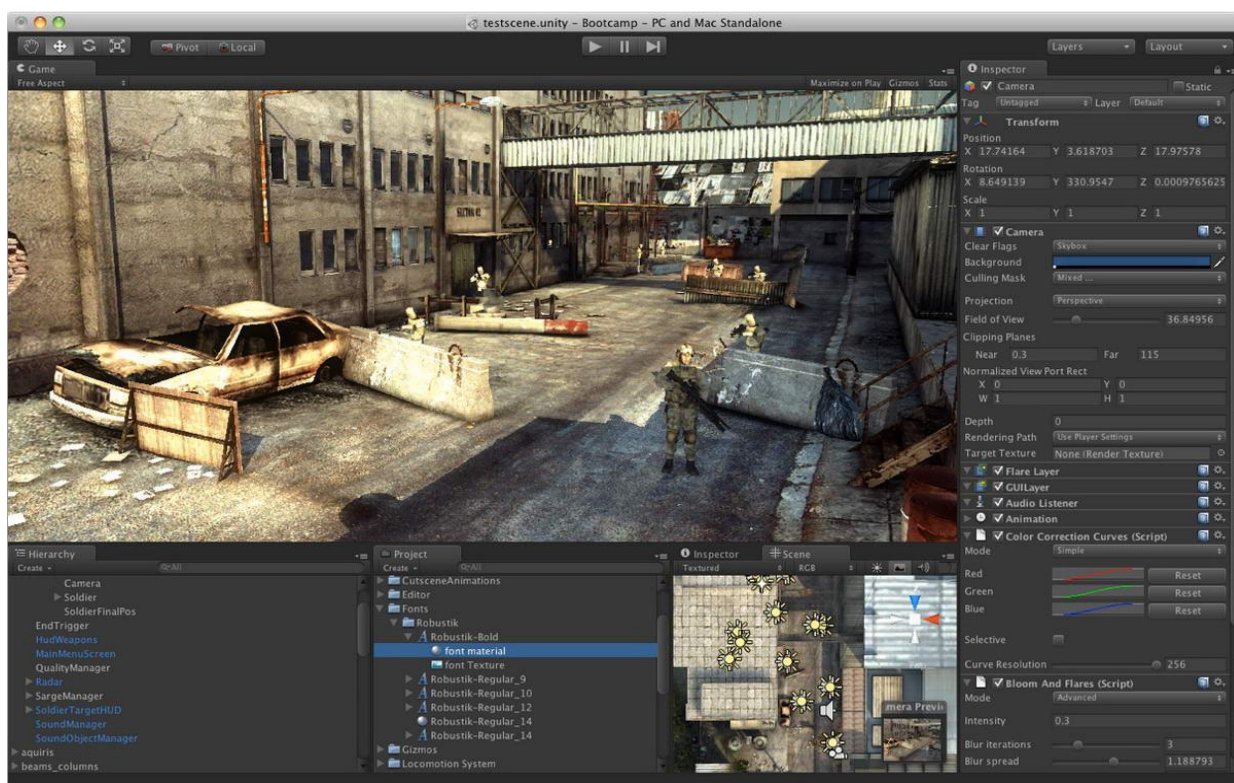


Рисунок 1 — Интерфейс редактора Unity3D

Программы, созданные с помощью Unity, поддерживают DirectX и OpenGL. Движок используется популярностью как крупными разработчиками (Blizzard, EA, Quantic Dream, Ubisoft), так и разработчиками Indie игр.

UnityEditor имеет простой DragandDrop интерфейс, который легко настраивать, состоящий из различных окон, благодаря чему можно производить отладку игры прямо в редакторе. Движок поддерживает три языка программирования: C#, JavaScript (модификация), Boo (диалект Python). Интегрирован физический движок PhysX от NVIDIA. Проект в Unity состоит из сцен (уровней) — отдельные файлы, содержащие свои игровые миры с набором объектов, подпрограмм, и настроек. Сцены могут содержать в себе как, собственно, объекты (модели), так и пустые игровые объекты — объекты пу-

стышки. Объекты, в свою очередь состоит из компонентов, с которыми и взаимодействует пользовательский код. У объектов есть имена, может быть тег и слой, на котором он будет отрисовываться. Так, у любого объекта на сцене обязательно имеется компонент Transform, хранящий информацию о координатах, повороте и размере объекта по трём осям.

Движок поддерживает как физику твёрдых тел и ткани, а также физику вида Ragdoll (тряпичная кукла). В редакторе имеется система наследования объектов; дочерние объекты поворачиваются, перемещаются и растягиваются вместе с родительским. Пользовательский код в редакторе представляется также в виде отдельных компонентов.

При импорте графики в Unity можно генерировать alpha канал, mip-map текстуры, карты нормалей и света, light-map, однако непосредственно на модель текстуру прикрепить нельзя — будет создан материал с шейдером, и затем материал будет назначен модели. Движок Unity поддерживает шейдеры. Редактор Unity имеет компонент для создания анимации, но также анимацию можно создать предварительно в 3D-редакторе и импортировать вместе с моделью, а затем разбить на файлы [5].

Unity 3D поддерживает систему Level of Detail, которая заключается в том, что на дальнем расстоянии от камеры высоко детализированные модели заменяются на слабо детализированные, и наоборот, а также систему Occlusionculling, суть которой в том, что у объектов, не попадающих в поле зрения камеры не визуализируется геометрия. Это значительно снижает нагрузку на компьютер и позволяет оптимизировать проект.

Движок поддерживает большое количество популярных форматов, например моделей 3ds, max, obj, fbx, dae, ma, mb. Различные звуковые форматы: mp3, wmv, ogg. Самые распространенные форматы компьютерной графики: bmp, gif, png, tga, psd, tif, dds. А так же видео файлы: mov, ogv.

Модели, звуки, графику, материалы, исходный код можно поместить в формат .unityassets и передавать другим разработчикам, или выкладывать в свободный доступ. Этот же формат используется в специальном магазине UnityAssetStore, в котором разработчики могут бесплатно и за деньги выкладывать в общий доступ элементы, необходимые для создания различных графических приложений на движке Unity. Чтобы использовать UnityAssetStore, разработчику необходимо иметь аккаунт на сайте Unity. В программе имеются все нужные компоненты для создания простого мультиплеера. Можно также использовать подходящую пользователю систему контроля версий, например, TortoiseSVN или SourceTree [6,7].

2. SPACE & PLAYS. EPHEMERID ANIMATOR

2.1. ИНСТРУКЦИЯ ПО ИСПОЛЬЗОВАНИЮ

Space & Plays. Ephemerid animator — это приложение, работающее под операционной системой Microsoft Windows, предназначенное для визуализации орбитального движения космических тел: звезд, планет, астероидов, спутников и др. Начать знакомство с программой лучше всего запустив демонстрационную сцену. Для этого нажмите на кнопку «Generate» в меню «Project» (Рисунок 2).

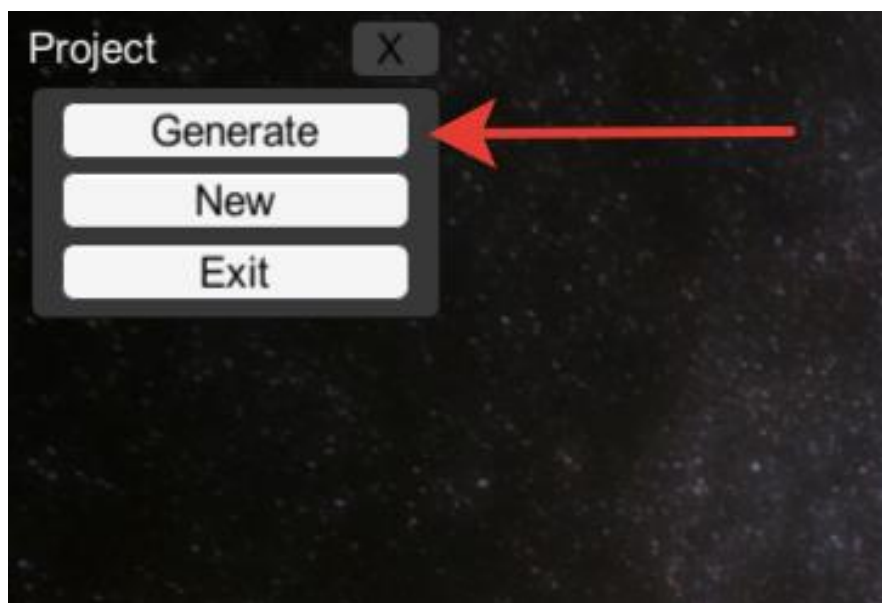


Рисунок 2 — Меню «Project» и кнопка «Generate»

На экране появятся система космических тел. Камера автоматически начнет двигаться вокруг звезды так, чтобы было удобно наблюдать за динамикой планет и спутников. В программе присутствует проигрыватель, который позволяет перематывать и останавливать время в сцене (рисунок 3). Он имеет классический для проигрывателей интерфейс: кнопка «Stop\Play», полоса перемотки и поле с числовым значением текущего времени. Использовать кнопку «Stop\Play» можно не только нажатием мышки, но и клавишей клавиатуры «Пробел». Перемотка также доступна во время паузы.



Рисунок 3 — Проигрыватель сцены

Для открытия меню «Camera» нажмите соответствующую кнопку на левой панели (Рисунок 4). Здесь находятся различные настройки связанные с камерой и ее поведением (Рисунок 5).

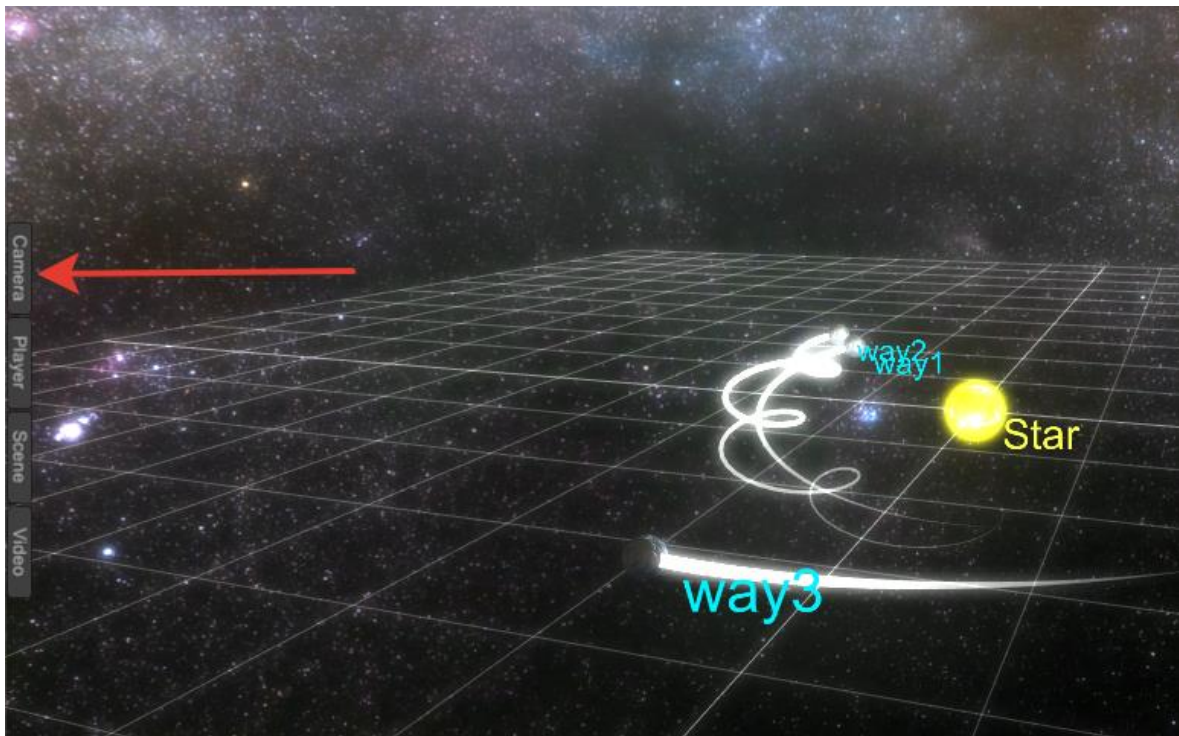


Рисунок 4 — Левая панель и кнопка «Camera»

Опция «Position» определяет способ движения камеры и может находиться в одном из трех состояний. «Around» и «Tracking» — два режима полностью автоматического позиционирования камеры, во время которых пользователь может только менять дистанцию камеры до цели при помощи колесика мыши. «Free» — режим, в котором положение камеры определяется состоянием меню «Bodies» (оно будет рассмотрено ниже), и колесиком мыши, а поворот задается движением мыши при зажатой правой кнопкой мыши. Это гибкий режим. В нем, например, можно настроить камеру так, чтобы она находясь на определенном смещении следовала за одним телом, а смотрела на центр масс двух других. Для перехода в этот режим не обязательно открывать меню. Если зажать правую кнопку мыши и начать двигать её, камера сразу переключится в режим «Free». «Field of view» определяет угол обзора камеры. Оптимальным значением является 60 градусов. В режимах «Around» и «Tracking» опция «Key frames» устанавливает количество ключевых кадров при расчете положения камеры, а «Angle» — угол камеры относительно сетки плоскости событий.

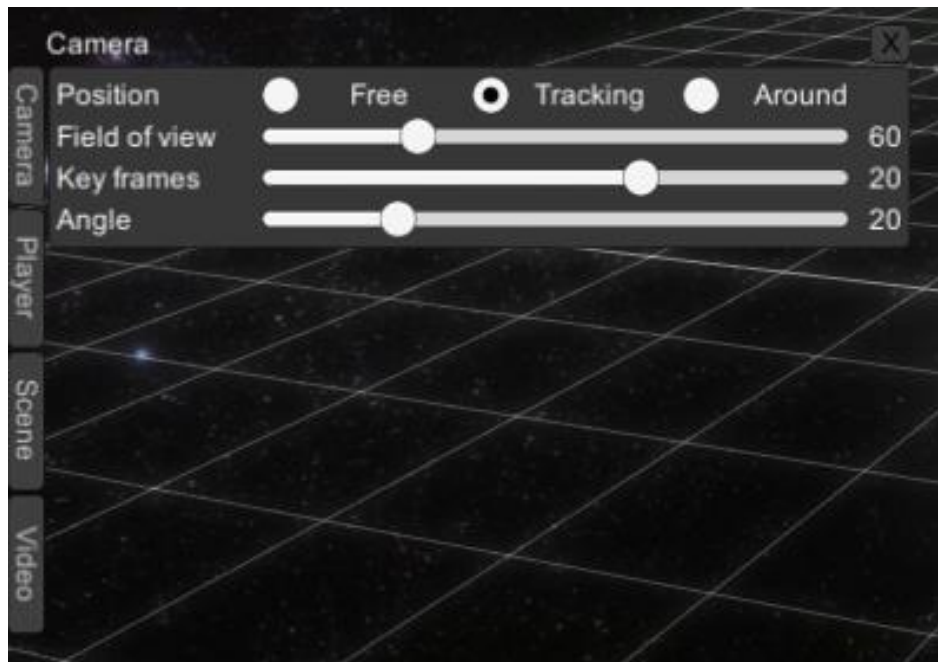


Рисунок 5 — Меню «Camera»

Следующая кнопка на левой панели открывает меню «Player», в котором находятся настройки плеера сцены (Рисунок 6). Галочка «Loop playback» зацикливает анимацию. Опция «Playback time» определяет время в секундах, которое будет занимать воспроизведение сцены. Меняя это значение, можно регулировать скорость анимации.



Рисунок 6 — Меню «Player»

В меню «Scene» (Рисунок 7) доступны настройки изменяющие внешний вид самих тел на сцене. Опция «Background» позволяет изменять задний план (звездное небо). Доступно несколько текстур космоса и черный фон. Галочка «Grid» включает прорисовку сетки плоскости событий на сцене. Важно отметить, что эта сетка не имеет отношения к системе координат. Она присутствует исключительно для облегчения визуального восприятия сцены и движения тел в трехмерном пространстве. Параметр «Size scale» изменяет размер всех тел на сцене. Если небесные тела в вашей системе визуально касаются друг друга или даже находятся внутри звезды, то следует значительно уменьшить их размер. Стандартное значение 8. «Trail length» определяет нормализованную длину следа, который оставляют тела при движении. Опция «Tail density» регулирует плотность следа, но в текущей версии программы не работает.

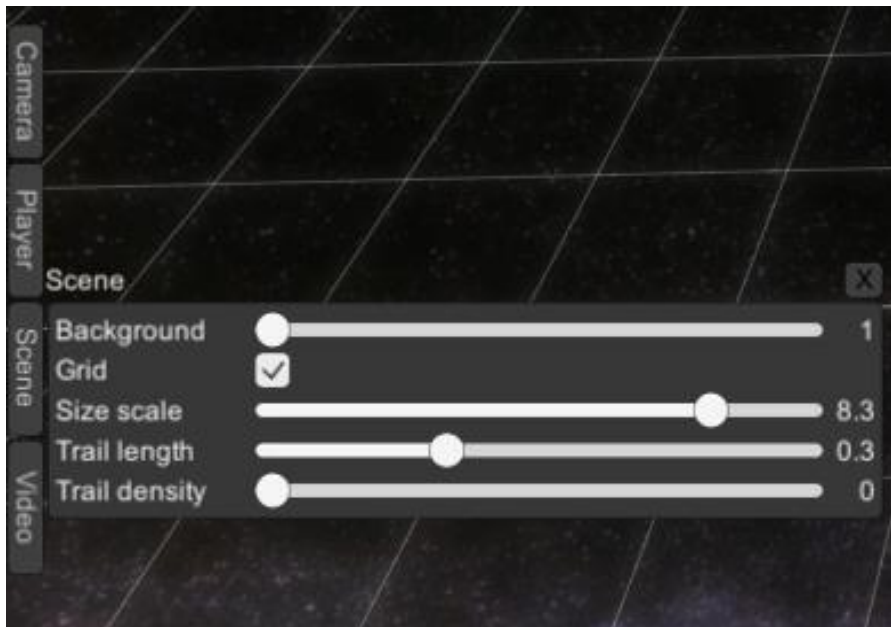


Рисунок 7 — Меню «Scene»

Меню «Video» (Рисунок 8) позволяет создавать видео с изображением вашей сцены. Параметры «Time start» и поле «Time end» означают время первого и соответственно последнего кадра будущего видеоролика. Поставьте плеер на паузу, перемотайте время на предполагаемый последний кадр, запишите время указанное в плеере в поле «Time end». Изначально в нем указано максимально возможное время для сцены. Затем перемотайте время на планируемый первый кадр, при этом значение параметра «Time start» будет меняться. Нажмите кнопку «Start record».

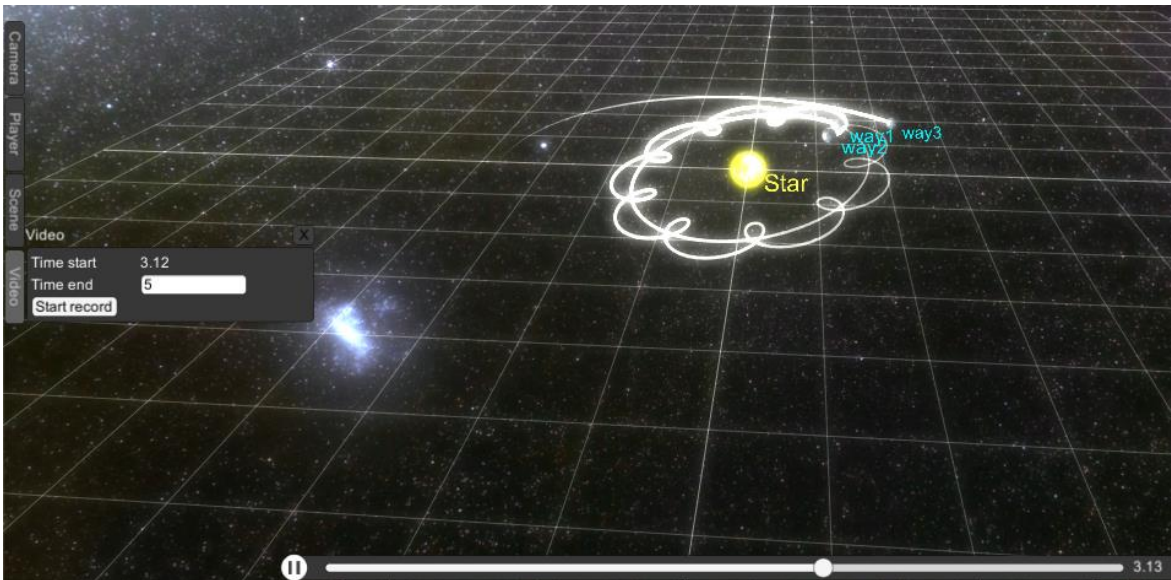


Рисунок 8 — Меню «Video»

После этого начнется процесс при котором каждый кадр будущего видео будет временно сохраняться на диск. Это может потребовать несколько Гбайт свободного места для короткого видео. Когда последний кадр будет удачно сохранен, запустится утилита «CreateVideoFromPath», которая распространяется вместе с программой «Space & Plays» и позволяет создавать видеоролики из последовательности картинок. На экране появиться подсказка с информацией о процессе создания видео. Желтым текстом указан путь, по которому можно будет найти создаваемое видео (Рисунок 9). Быстро открыть эту папку можно нажатием кнопки «Open video folder».

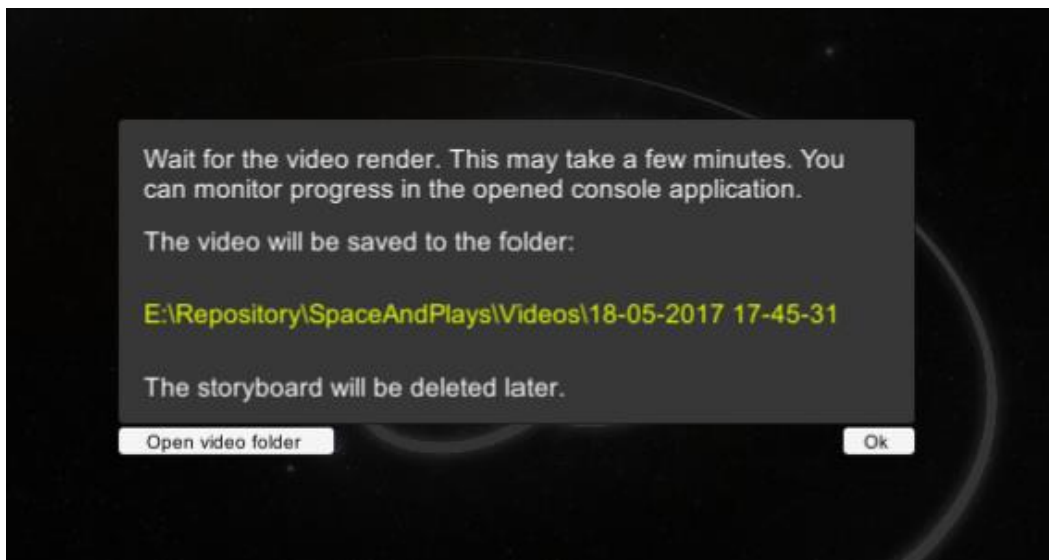


Рисунок 9 — Подсказка при генерации видеоролика

Кроме этого, в виде консольного приложения (Рисунок 10) запуститься сама утилита «CreateVideoFromPath». В случае каких-либо проблем при создании видео в консоли можно будет найти информацию об ошибках. Если генерация видео закончится успешно, то подсказка и утилита закроются, а по указанному пути появится видео файл и все кадры ролика. Кадры автоматически удаляются при закрытии приложения. Для корректной работы утилиты может потребоваться установка кодеков, в частности кодека H264.

```

E:\Repository\VideoCreator\VideoCreator\CreateVideoFromPath\bin\Debug\CreateVideoFromPath.exe
E:\Repository\SpaceAndPlays\Videos\15-05-2017 19-31-23
1920
1080
30
[libx264 @ 00D49020] using cpu capabilities: MMX2 SSE2Fast SSSE3 FastShuffle SSE4.2 AVX
[libx264 @ 00D49020] profile High, level 4.0
File 00000000 Success
File 00000035 Success
File 00000070 Success
File 00000105 Success
File 00000140 Success
File 00000174 Success
File 00000210 Success
File 00000245 Success
File 00000280 Success
File 00000314 Success
File 00000349 Success
File 00000384 Success
File 00000420 Success
File 00000455 Success
File 00000490 Success
File 00000524 Success
File 00000560 Success
File 00000595 Success

```

Рисунок 10 — Утилита «CreateVideoFromPath», запущенная в виде консольного приложения.

2.2. МЕНЮ «BODIES»

Меню «Bodies» (Рисунок 11) представляет собой таблицу, где строка соответствует конкретному телу в сцене. Изменения строке будут отражаться на самом теле, либо на поведении камеры. Поставив галочку в столбце со значком перечеркнутого глаза, можно временно сделать соответствующее тело невидимым на сцене. След от объектов отрисовывается относительно центра масс тел, помеченных в столбце F (Frame). Изначально траектории изображаются относительно звезды, так как галочка стоит только в её строке. Галочки в столбцах P (Position) и T (Target) определяют поведение камеры, если она находится в режиме «Free». Быстро перейти в него можно зажав правую кнопку мыши и сделав ей движение. Камера будет следовать за центром масс небесных тел, отмеченных галочками в столбце P, а смотреть на центр масс тел столбца T.

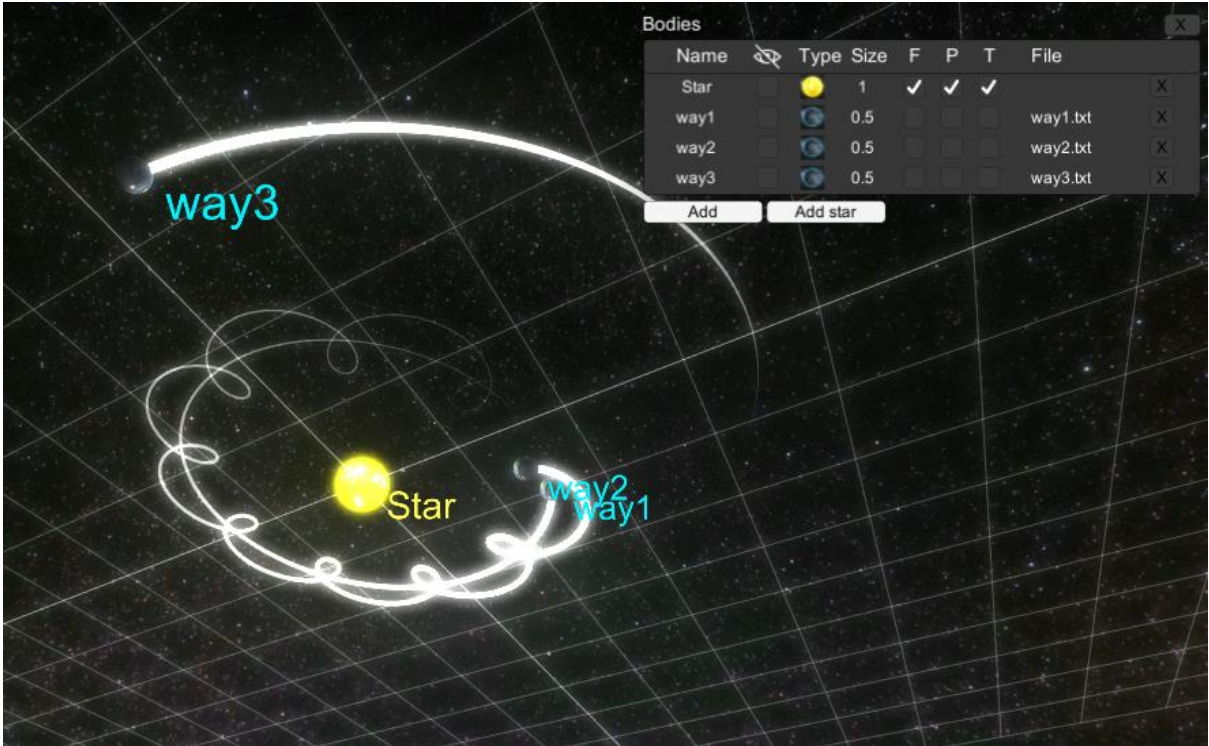


Рисунок 11 — Меню «Bodies»

Например, чтобы в демонстрационной сцене наблюдать попятное движение тела «way3» относительно тела «way1», необходимо переключить камеру в режим «Free», а затем привести меню «Bodies» в состояние изображенное на рисунке 12. Меню «Bodies» позволяет гибко настраивать поведение камеры.

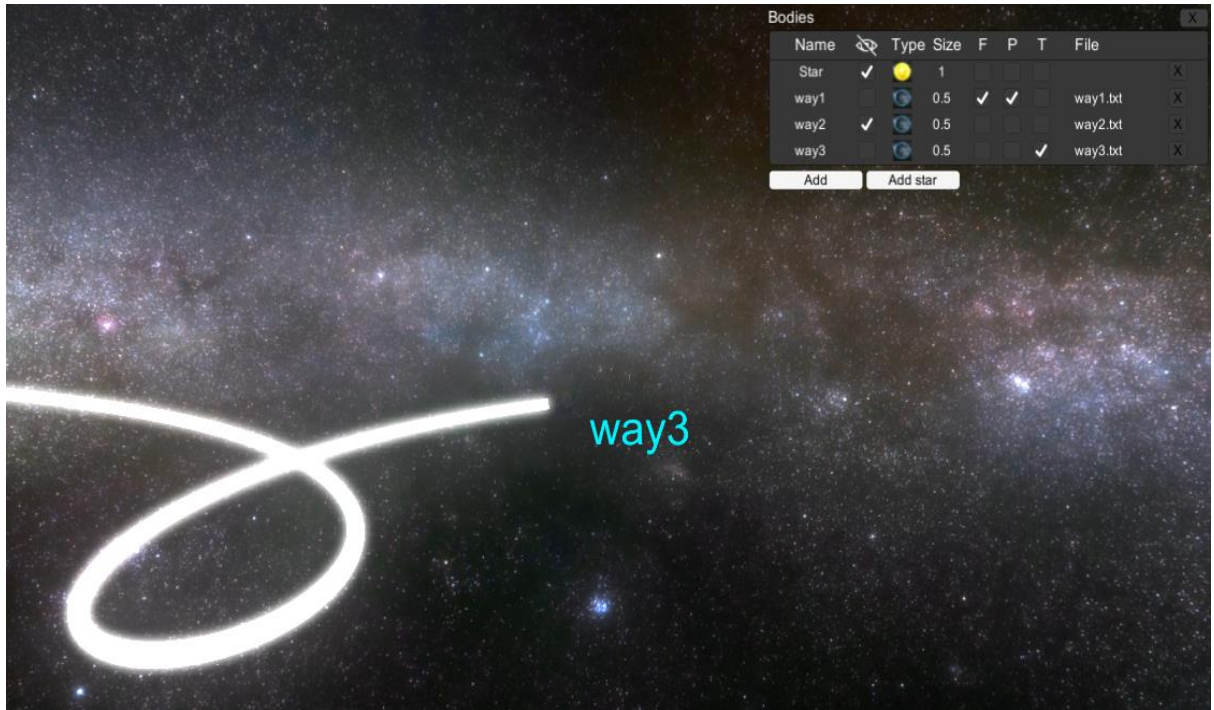


Рисунок 12 — Меню «Bodies», настроенное для наблюдения попятного движения тела «way»

Столбец «Name» — имя объекта. Для его изменения кликните левой мышкой по имени. Появится поле для ввода нового имени (Рисунок 13). Наберите новое имя на клавиатуре. По окончании ввода нажмите клавишу Enter или кликните левой кнопкой мыши в любое место экрана, свободное от каких-либо меню. Для отмены ввода и возвращения к старому имени нажмите клавишу Esc.

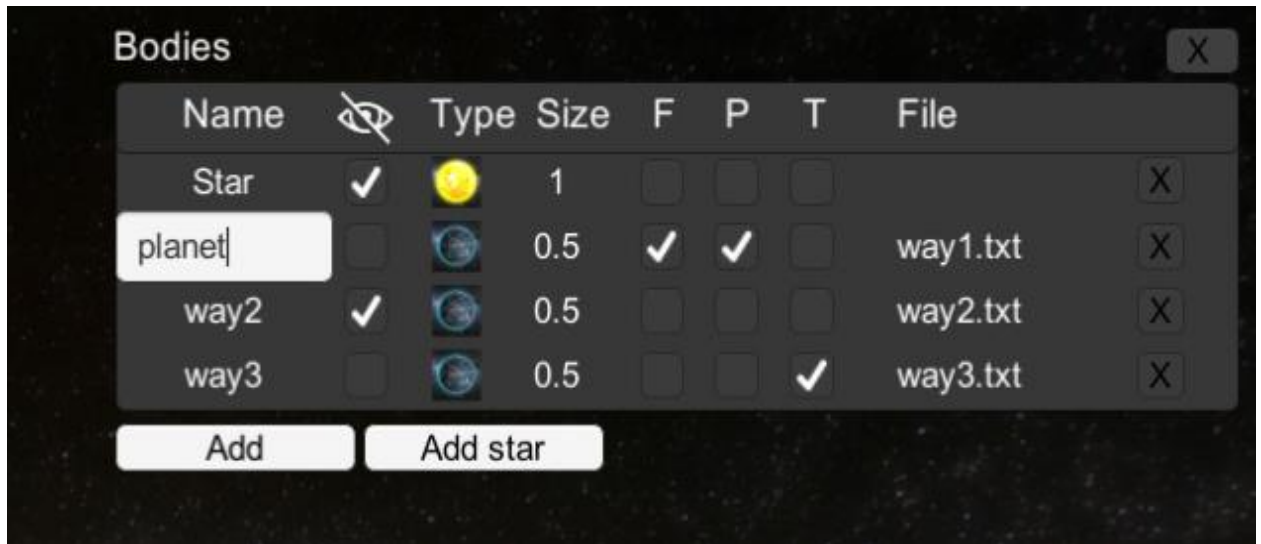


Рисунок 13 — Изменение имени объекта

Столбец «Type» — тип и внешний вид объекта. Для изменения нажмите по иконке в этом столбце. Откроется меню со списком доступных вариантов, в котором можно выбрать текстуру и 3D модель для объекта (Рисунок 14). Список можно прокрутить, используя полосу прокрутки. Доступные варианты разбиты на группы «Star», «Planet» и «Asteroid». Выберите нужный и нажмите по нему левой кнопкой мыши. Автоматически откроется меню «Bodies», а соответствующее тело изменит свой внешний вид.

Столбец «Size» — радиус космического тела в условных единицах. Доступные значения «1», «0.5», «0.1», «0.05», «0.01». Для задания телу нужного размера наведите курсор на число, зажмите левую кнопку мыши и, не отпуская её, двигайте курсор вверх (вниз), чтобы увеличить (уменьшить) размер объекта. Получив нужное значение размера, отпустите кнопку мыши. В столбце «File» указано имя файла с эфемеридой, сопоставленное телу. Чтобы изменить файл кликните по названию (Рисунок 15) левой кнопкой мыши. Подробнее об этих файлах можно почитать в следующем разделе. В последнем столбце находится кнопка крестик, позволяющая удалить тело со сцены.

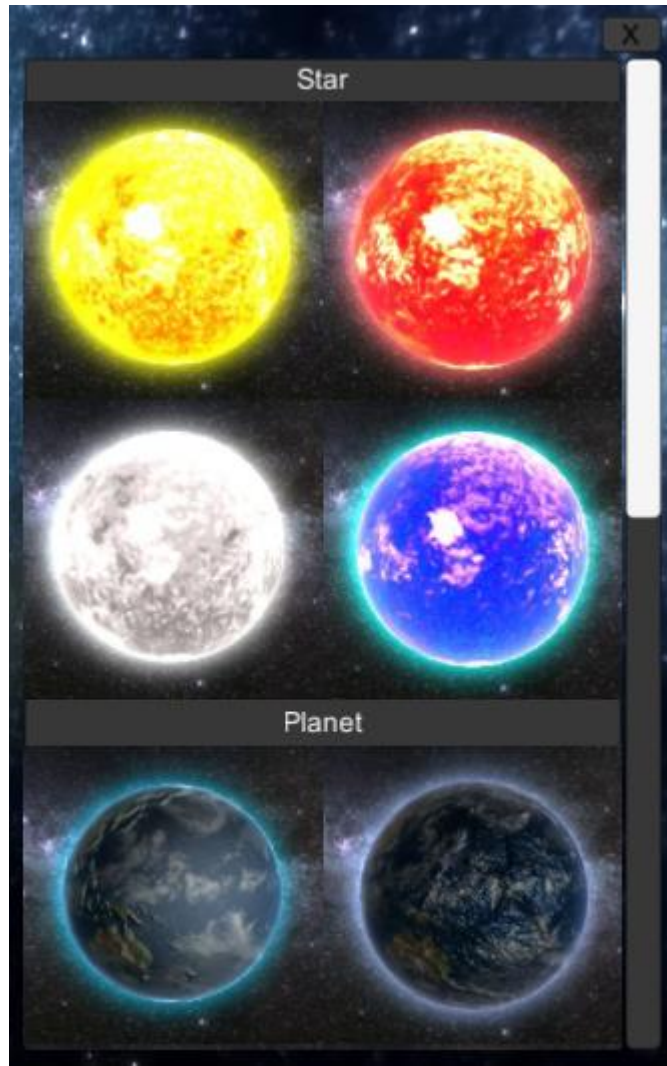


Рисунок 14 — Меню выбора внешнего вида объекта



Рисунок 15 — Файлы эфемерид. Красным квадратом выделена область, по которой необходимо кликнуть для вызова диалога Open File

2.3. ДОБАВЛЕНИЕ НОВЫХ ОБЪЕКТОВ НА СЦЕНУ

Кнопка «Add» (Рисунок 17) добавляет объект, сопоставленный с текстовым файлом, содержащим эфемеридные данные, в соответствии с которыми объект будет двигаться во время визуализации. Сам файл состоит из строк. Каждая строка должна содержать четыре числа, разделенных запятой или пробелом. Первое число — момент времени t (не обязательно), второе, третье и четвертое — соответствующие ему прямоугольные координаты объекта x , y и z . Единицы измерения условные. Например, строка может выглядеть как последовательность чисел — 0.01, 0.50, 0, 16.99. Здесь $t = 0.01$, $x = 0.50$, $y = 0$, $z = 16.99$. Дробная часть числа должна быть отделена точкой или запятой. Пример текстового файла с эфемеридой на рисунке 16.

```

0 0 0 17
0.01 0.5099235 0 16.99235
0.02 1.019388 0 16.96941
0.03 1.527935 0 16.9312

```

Рисунок 16 — Пример текстового файла с эфемеридой.

Стоит отметить, что Unity3D представляет каждую координату в виде четырех байтового числа с плавающей точкой, поэтому хранить в файле числа с точностью более 3–6 знаков можно, но не имеет смысла, так как не помещающиеся в разрядную сетку цифры будут урезаны. Время t не должно уменьшаться от строки к строке.

При нажатии кнопки «Add» (Рисунок 17) откроется диалог «Open File», в котором можно выбрать нужный файл (Рисунок 18). Перемещаться по папкам на компьютере можно при помощи кнопок на верхней панели этого меню. Чтобы открыть папку или выбрать файл, однократно кликните по соответствующей строчке в списке.

Пусть в некотором каталоге есть заранее подготовленный файл с траекторией, например, файл с именем «way4.txt». Выберите этот файл левым кликом. В меню «Bodies» появится строка, соответствующая новому телу (Рисунок 19), которое будет двигаться по траектории, записанной в выбранном файле. Нажав кнопку «Add star», можно добавить на сцену звезду, которая будет находиться в начале координат.

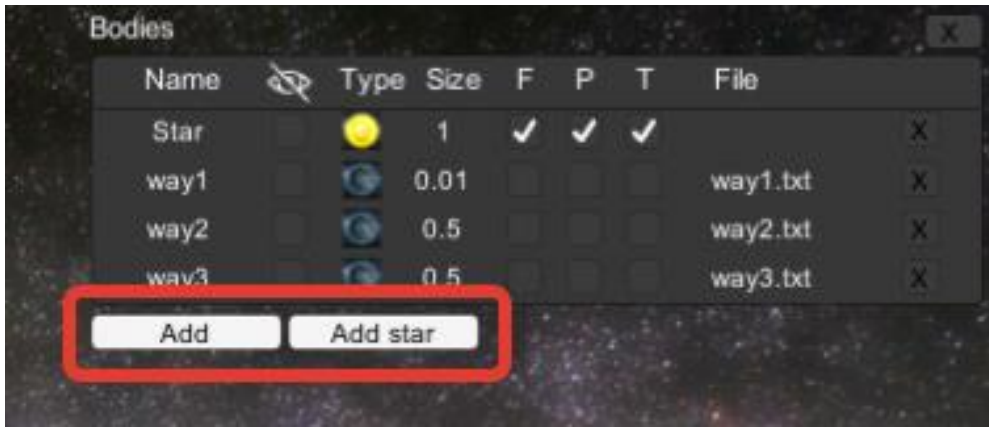


Рисунок 17 — Кнопки добавления новых тел

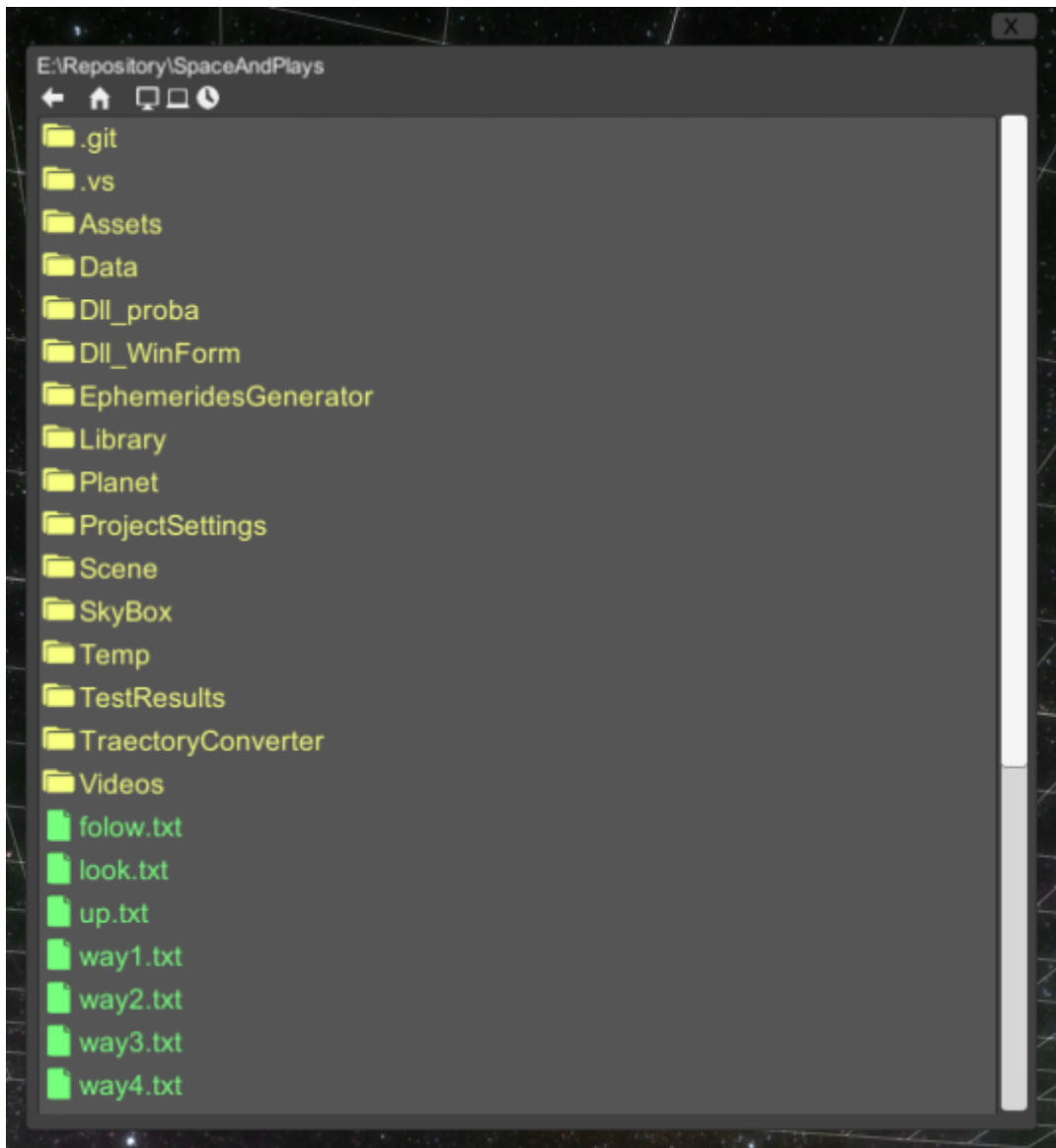


Рисунок 18 — Диалоговое окно Open File



Рисунок 19 — Новая строка в меню «Bodies»

2.4. РУЧНОЕ УПРАВЛЕНИЕ КАМЕРОЙ

Чтобы сменить ракурс, под которым камера смотрит на сцену, необходимо нажать правую кнопку мыши на любом свободном от каких-либо меню месте и, не отпуская ее, двигать мышью в вертикальном и горизонтальном направлении. Камера будет менять углы тангаж и рысканье, а также сразу перейдет в режим «Free». Чтобы приблизить или отдалить камеру от наблюдаемых тел, следует использовать колёсико мыши. В режиме «Free» на поведение камеры также влияют настройки меню «Bodies», описанные в разделе 2.2.

3. АРХИТЕКТУРА ПРИЛОЖЕНИЯ

3.1. СТРУКТУРА КЛАССОВ

Логика проекта на движке Unity3D описывается в так называемых скриптах. Они представляют собой части исходного кода. Как правило, в одном скрипте описывается один класс. Рассмотрим основные скрипты в приложении и их назначение (Рисунок 20).

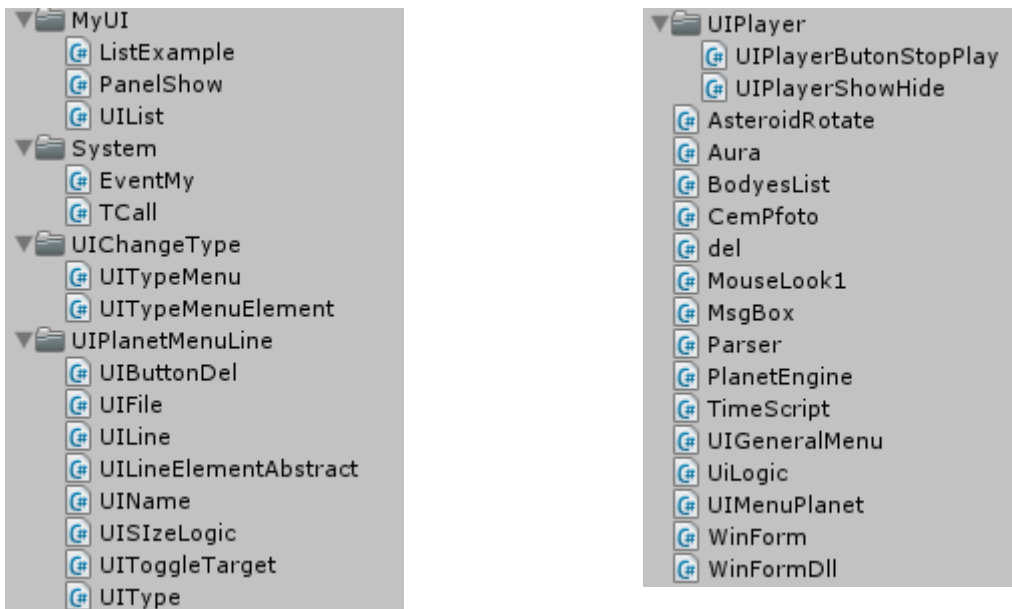


Рисунок 20 — Список скриптов приложения разбитый по каталогам

PlanetEngine реализует линейную интерполяцию положения объекта между точками, записанными в файле траектории. Содержит логику создания экземпляра объекта, по данным из файла `_General.dat` и наоборот сохранение настроек в этот файл.

TimeScript предоставляет другим скриптам информацию о текущем времени в приложении. Минимальное и максимальное время рассчитывается при загрузке новых файлов с траекторией. Обратное течение времени не реализовано.

Parser предоставляет интерфейс для чтения данных из текстовых файлов, при чтении отсеивает комментарии.

BodiesList инкапсулирует структуру данных, реализующую интерфейс `IList` и содержащую все объекты на сцене.

AsteroidRotate организует анимацию вращения астероидов. Во вращении всегда есть случайная составляющая.

MouseLook1 контролирует движением камеры с использованием различных интерполяций. Реализует эффекты плавных перехода камеры и пово-

ротов с использованием кватернионов. Кватернионы получили большую популярность в трех мерной графике для описания поворотов и операция над ними, так как в сравнении с матрицами они обладают большей вычислительной устойчивостью и могут быть более эффективными.

WinForm — надстройка над WinAPI операционной системы Windows, предоставляющая интерфейс для работы с диалоговым окном Open File.

UiLogic содержит основную логику загрузки и сохранения сцены.

MsgBox силами Unity3D реализует функционал аналогичный функции MessageBox WinAPI операционной системы Windows.

UIPlayerShowHide и UIPlayerButonStopPlay — логика проигрывателя и его интерактивных графических элементов.

Группа скриптов UIPlanetMenuLine — содержит логику строчек меню «Bodies». В приложении была реализована каскадная модель обновления интерфейса, которая зарекомендовала себя как просто реализуемая, удобная в использовании и модернизации. Каскад обновлений строчек меню «Bodies» представлен на рисунке 21.

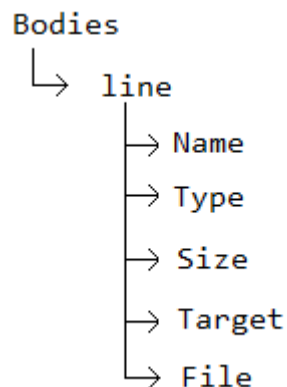


Рисунок 21 — Каскад обновлений меню «Bodies»

Классы UILineName, UILineType, UILineSize, UILineTarget, UILineFile являются наследниками абстрактного класса UILineElementAbstract и переопределяют процедуру обновления элемента. Таким образом, можно легко реализовать обновление всех строк, обновление определенных строк, добавить в архитектуру новый элемент меню, реализовать любую логику элемента не переделывая уже написанный код.

Группа скриптов UIChangeType содержит логику меню выбора внешнего вида космического тела.

EventMy — класс Singleton. Реализует паттерн наблюдатель (Observer).

TCall — класс Singleton. Организует отложенный вызов методов. Дан-

ный класс очень удобен в использовании и значительно облегчает реализацию любых задач растянутых во времени. Например, исчезновение меню через промежуток времени, изменение текста на короткий промежуток времени и многое другое. Класс имеет несколько публичных процедур, выполняющих отложенный вызов различных по сигнатуре методов (Рисунок 22). Аргумент `deltaTime` — время, через которое должен произойти вызов целевого метода, `action` — целевой метод.

```
public static void CallVoid(float deltaTime, System.Action action) ...
public static void Call(float deltaTime, Action0 action) ...
public static void Call(float deltaTime, Action1 action, float argument) ...
public static void Call(float deltaTime, ActionGo action, GameObject argument) ...
```

Рисунок 22 — Интерфейс класса TCall

UIList реализует логику прокрутки списка в меню «Bodies», автоматически рассылает сообщения о необходимости обновления списка.

PanelShow — класс, выполняющий анимацию выпадения элементов интерфейса.

Во время реализации приложения возникли разнообразные сложности. Unity3D — это кроссплатформенный движок, а значит платформозависимые компоненты удалены из библиотеки классов .NET Framework. Это приводит к возникновению непредвиденных сложностей в самых неожиданных местах. Например, открытие диалога Open File пришлось реализовывать самостоятельно вместо использования стандартного диалога.

3.2. ИНТЕРПОЛЯЦИЯ ДВИЖЕНИЯ КАМЕРЫ И ТЕЛ

Движение камеры и тел на сцене сглаживается при помощи доработанного алгоритма интерполяции Безье. Полученный метод имеет два параметра. Первый отвечает за плавность и при значении ноль вырождается в линейную интерполяцию. Второй определяет угол касательной к сглаженной траектории в узлах интерполяции. На рисунках 23, 24 и 25 графически показана идея алгоритма при различных значениях параметров. Зеленая линия — траектория, представленная кусочно-линейной интерполяцией, белые точки — сглаженная траектория модифицированным методом Безье. Реализация метода на языке программирования C# находится в методе «Vector3 GetPos (floattime)» файла «Way.cs»

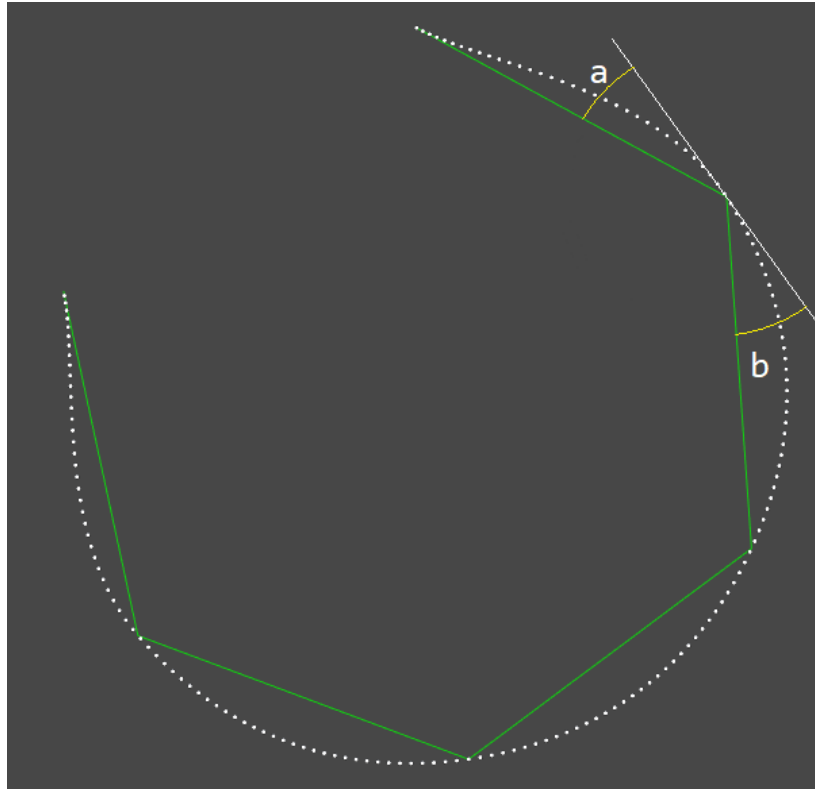


Рисунок 23 — Демонстрация полученного метода интерполяции.
Угол «а» равен углу «b».

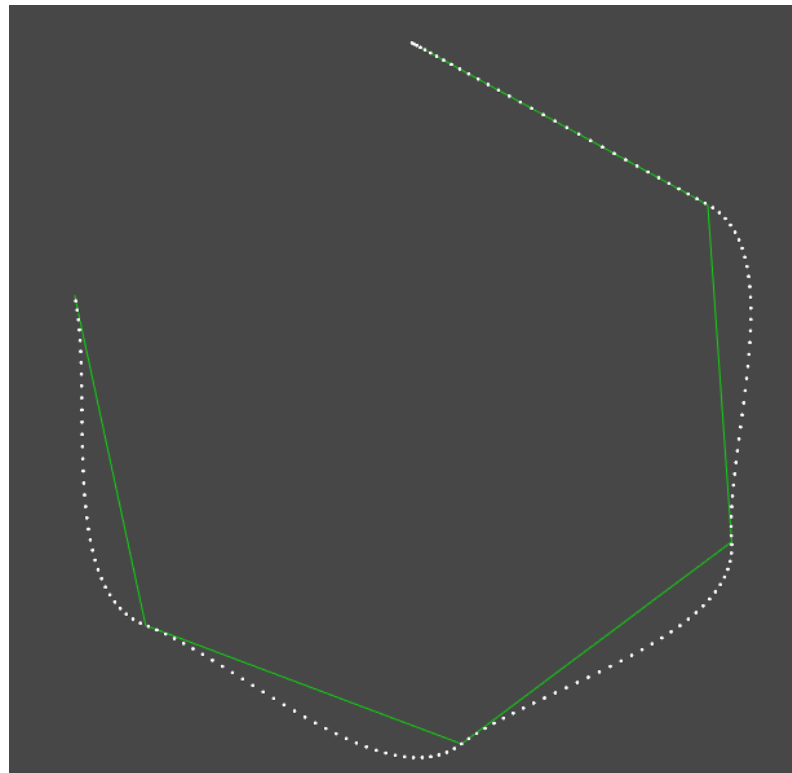


Рисунок 24 — Демонстрация полученного метода интерполяции.
Второй параметр интерполяции равен нулю.

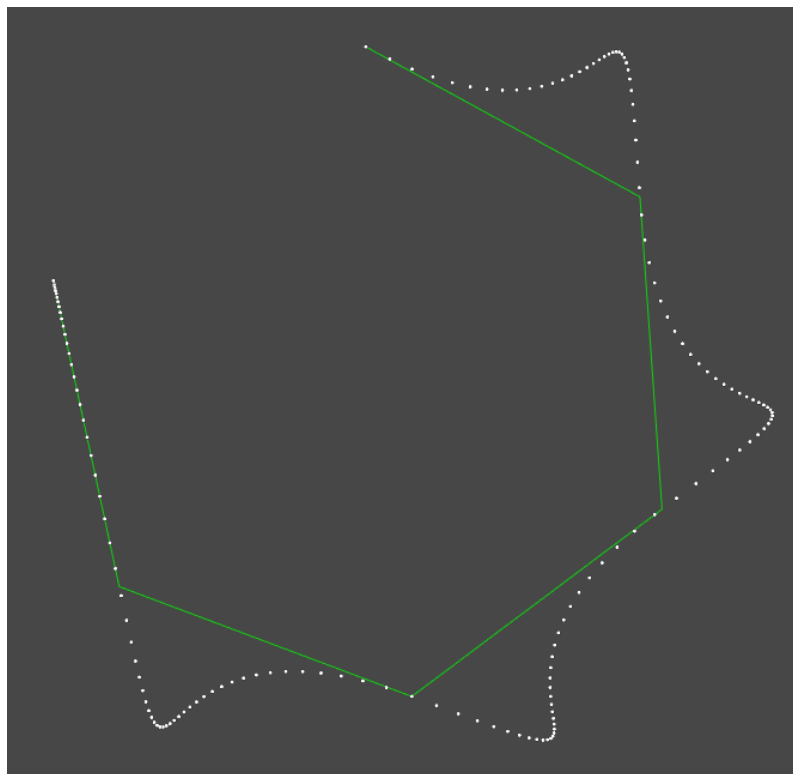


Рисунок 25 — Демонстрация полученного метода интерполяции. Второй параметр интерполяции равен единице, а первый сильно завышен.

3.3. АЛГОРИТМЫ АВТОМАТИЗАЦИИ ДВИЖЕНИЯ КАМЕРЫ

В основе режимов «Around» и «Tracking» в меню «Camera» лежит метод нахождения такого положения виртуальной камеры, при котором будет получен хороший обзор сцены, на картинке созданной этой камерой. Под хорошим понимается такой обзор, при котором бы тела на картинке были, как можно более равномерно распределены в кадре, а не сосредоточены в углу и без перекрытий друг друга. Для получения такого положения используется следующий алгоритм:

- 1) Находим эллипсоид, описывающий распределение объектов в пространстве относительно центра масс системы. Эллипсоид определяется матрицей разброса данных A :

$$a_{ij} = \sum_{k=1}^K \sum_{n=1}^N (x_{ni}^k - x_{Bi})(x_{nj}^k - x_{Bj}) \quad (i, j = 1, 2, 3).$$

Здесь x_{ni}^k — компонента i -го вектора положения тела номер k в кадре n ; K, N — суммарное количество тел и ключевых кадров в сцене соответственно; x_B — центр масс всей системы. Если бы положения небес-

ных тел были случайными, то матрица A имела смысл ковариационной матрицы.

- 2) Степенным методом последовательно находим собственные вектора матрицы A .
- 3) Последний из найденных векторов, соответствующий наименьшему собственному числу, интерпретируем как нормаль к плоскости событий. Сумма вектора положения центра масс системы и нормали, умноженной на коэффициент, пропорциональный наибольшему собственному числу и обратно пропорциональному тангенсу угла обзора камеры, количеству тел и количеству ключевых кадров, даст вектор положения камеры.

В режиме камеры «Tracking» описанным алгоритмом находится несколько векторов положения для разных промежутков времени сцены. Затем камера двигается от одного положения к другому с использованием интерполяции описанной в пункте 3.2. В режиме «Around» алгоритм применяется один раз для всей сцены, а затем полученный вектор просто вращается вокруг второго собственного вектора.

ЗАКЛЮЧЕНИЕ

Таким образом, реализованная на движке Unity 3D программа Space & Plays решает поставленные задачи. Она позволяет в удобной для просмотра форме представить динамику движения космических тел. Программа имеет удобный интерфейс и позволяет быстро создать сцену из нескольких тел. Стандартные шейдеры Unity 3D обеспечивают достаточно красивую картинку.

Однако, как показала практика использования приложения в реальных условиях, оно имеет ряд значительных недостатков. Добавление на сцену большого количества космических тел крайне неудобно. Решить данную проблему возможно путем введения специального меню, содержащего необходимый функционал для массового добавления тел. Разумеется, удобнее группировать объекты в меню «Bodies».

Самым амбициозным планом является реализация функции подключения расширений к приложению. Множество программ стали всемирно известны именно благодаря огромным возможностям в расширении. Ярким примером является игра Warcraft III. Её разработчики реализовали мощнейший редактор, с помощью которого энтузиасты по всему миру уже 12 лет занимаются написанием карт для этой игры. Беспрецедентную популярность получила карта под названием «DotA». На данный момент она имеет сотни версий, а также стала толчком для создания самостоятельной игры «DotA 2».

Говоря о недостатках приложения Space & Plays, отметим его ограничения, налагаемые на формат хранения данных. Очевидно, он не сможет удовлетворить всех пользователей. Вероятно, кому-то потребуется загружать данные из базы данных или получать их от удаленного сервера по какому-нибудь сетевому протоколу. Предусмотреть все невозможно, но пользователь самостоятельно сможет создать плагин, позволяющий получать данные из любого источника любым способом. Реализовать расширяемость будет полезной, хотя и сложной, но в то же время интересной задачей. Планируется выложить программу и её исходные коды в открытый доступ в известном репозитории GitHub [8]. Этот шаг также позволит сторонним разработчикам принять участие в доработке проекта.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Solar System Scope — <http://www.solarsystemscope.com>
2. Eyes on the Solar System — <https://eyes.nasa.gov>
3. UniverseSandbox — <http://universesandbox.com>
4. Solar Suit Display — <http://solarsuit.net/downloads.html>
5. Unity3D (Википедия) —
[https://ru.wikipedia.org/wiki/Unity_\(игровой_движок\)](https://ru.wikipedia.org/wiki/Unity_(игровой_движок))
6. Unity3D (официальный сайт) — <https://unity3d.com/ru>
7. Unity 3D (официальная документация) —
<http://docs.unity3d.com/ru/current/Manual/UnityManualRestructured.html>
8. GitHub — <https://github.com>

Уважаемый пользователь! Обращаем ваше внимание, что система «Антиплагиат» отвечает на вопрос, является ли тот или иной фрагмент текста заимствованным или нет. Ответ на вопрос, является ли заимствованный фрагмент именно плагиатом, а не законной цитатой, система оставляет на ваше усмотрение.

Отчет о проверке № 1

ФИО: Тырышкин Дмитрий
дата выгрузки: 03.06.2017 18:07:38
пользователь: yabloko9392@yandex.ru / ID: 4638677
 отчет предоставлен сервисом «Антиплагиат»
 на сайте <http://www.antiplagiat.ru>

Информация о документе

№ документа: 6
Имя исходного файла: Диплом 2017 Готов.pdf
Размер текста: 1188 кБ
Тип документа: Не указано
Символов в тексте: 28614
Слов в тексте: 3651
Число предложений: 218

Информация об отчете

Дата: Отчет от 03.06.2017 18:07:39 - Последний готовый отчет
Комментарии: не указано
Оценка оригинальности: 93.91%
Заимствования: 6.09%
Цитирование: 0%



Оригинальность: 93.91%
 Заимствования: 6.09%
 Цитирование: 0%

Источники

Доля в тексте	Источник	Ссылка	Дата	Найдено в
4.55%	[1] Unity (игровой движок)	http://ru.wikipedia.org	17.11.2016	Модуль поиска Интернет
4.31%	[2] Unity (игровой движок)	http://ru.wikipedia.org	30.11.2014	Модуль поиска Интернет
1.06%	[3] Шаблоны титульных листов ВКР	http://tsu.ru	22.09.2016	Модуль поиска Интернет