

ОСНОВНЫЕ ВОЗМОЖНОСТИ И ОБЛАСТИ ПРИМЕНЕНИЯ ИНСТРУМЕНТА ДЛЯ РАЗРАБОТКИ ДВУХМЕРНЫХ И ТРЁХМЕРНЫХ ПРИЛОЖЕНИЙ UNITY

Репецкая А. Я., Видман В. В., Иванов М.А.

Томский политехнический университет
aleksa_super_93@mail.ru

Введение

Unity это современная кроссплатформенная среда для создания игр и приложений, разработанная компанией Unity Technologies. Популярность применения Unity связана с возможностью использования разработанных приложений, как в настольных компьютерах, так и в мобильных устройствах, а также любых других устройствах, поддерживающих работу интернет-браузеров.

Приложение, разработанное в среде Unity, можно назвать уникальным, благодаря его функциональным возможностям и малому потреблению ресурсов. Ядро Unity является высокопроизводительным и способно предоставлять высокое быстродействие при довольно хорошем качестве графики. Рендеринг (термин в компьютерной графике, обозначающий процесс получения изображения по модели с помощью компьютерной программы) игры в Unity изначально ориентирован на быстроту и качество.

Проект в среде Unity состоит из сцен (уровней) - отдельных файлов, которые содержат игровые локации, с размещенными в них объектами, скрипты, шейдеры, настройки и другие ресурсы, используемые в проекте. Сцена состоит из объектов (моделей) и пустых игровых объектов, все они состоят из компонентов, с которыми взаимодействуют скрипты. У каждого игрового объекта есть название, может присутствовать тег или метка, а также слой, на котором он должен отображаться. У каждого объекта обязательно присутствует компонент Transform - это класс, в свойствах которого хранятся координаты местоположения, поворота и физических размеров игрового объекта в трехмерном пространстве (по осям x, y, z). При этом по умолчанию за 1 пункт измерения принят 1 метр в реальном пространстве, это необходимо учитывать при создании и последующем импорте моделей и локаций из среды поддерживаемых графических пакетов Maya, 3DS MAX и т.д. У предметов с видимой структурой присутствует еще и компонент Mesh Renderer с текстурой, картой нормалей или картой освещения, все это позволяет создавать эффекты шероховатости и рельефности поверхностей объектов игрового мира. Всем объектам можно назначить коллайдеры коллизий - границы допустимого пересечения с другими объектами игрового мира [1].

Достоинства и недостатки

Одним из самых больших плюсов Unity является то, что он поддерживает огромное количество различных форматов, которые можно импортировать из других приложений. Это позволяет создавать объекты и материалы в приложениях и в дальнейшем импортировать результаты в Unity [2].

Также достоинством Unity является интеграция игрового ядра в саму среду визуальной разработки приложений, что позволяет тестировать разрабатываемое приложение «на лету», что экономит время и позволяет сосредоточиться над другими важными аспектами разработки. Одним из таких аспектов является освещение. Возможно применение как статичного, так и динамичного освещения. Unity обладает такими возможностями, как Deferred Lighting (высокое качество освещения и теней), набор средств для создания специальных визуальных эффектов, SSAO (постэффект, который затеняет углы, впадины и складки, добавляя ощущение объема). Одной из возможных технологий освещения, которая может быть использована при создании проекта, является Lightmapping. Данная технология позволяет сохранить информацию о свете в самих текстурах, что сокращает количество затрачиваемых ресурсов.

Следует отметить, что Unity поддерживает такие известные языки программирования как C# и JavaScript. Также возможно применение языка Python's Boo. Написание кода осуществляется с помощью встроенного редактора либо Visual Studio.

Дополнительным преимуществом является наличие Asset Store, где имеется огромное количество готовых различных плагинов и ресурсов для создания игры. Данное хранилище снабжено удобным поиском и позволяет оперативно загрузить и интегрировать объекты в свое приложение.

К недостаткам Unity можно отнести закрытость кода, что не позволяет вносить изменения в само ядро [3].

Интерфейс программы

Интерфейс в Unity удобен и интуитивно понятен, начинающий пользователь способен изучить основной функционал среды и приступить к разработке за относительно небольшой временной интервал (рис.1).

Главное окно редактора состоит из нескольких вкладок, называемых Видами (Views). В Unity

есть несколько типов видов - все они предназначены для конкретных целей.

Окно Scene используется для выделения и позиционирования элементов сцен, камеры и прочих объектов. Консоль обеспечивает вывод сообщений, предупреждений, ошибок.

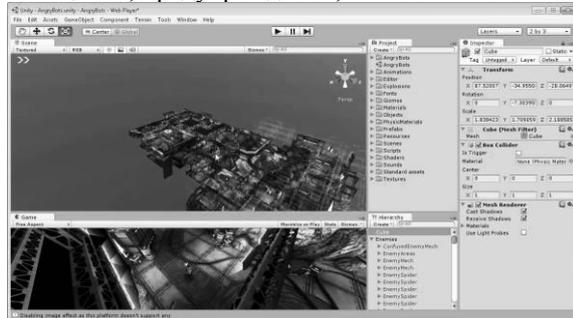


Рис. 1. Интерфейс среды Unity

Окно Animation используется для анимации объектов в сцене.

Профайлер используется для обзора производительности и поиска «узких» мест приложения.

Окно Asset Server, в случае его использования, дает возможность контролировать версии проекта.

Окно Lightmapping, используя встроенные в Unity инструменты, позволяет настраивать карты освещения.

Окно Occlusion Culling используется для настройки параметров производительности приложения [3].

Как правило, разработанные в Unity игры состоят из множества игровых объектов, которые содержат полигональную сетку (совокупность вершин, ребер и граней, которые определяют форму многогранного объекта в трехмерной компьютерной графике и объемном моделировании), скрипты, звуки или другие графические элементы, вроде источников света. Инспектор отображает детальную информацию о текущем выбранном объекте, включая все прикрепленные компоненты и их свойства [3].

Примеры использования Unity

Unity можно использовать для реализации как крупных AAA проектов [2] (высококачественные проекты с высоким бюджетом и соответствующим уровнем качества), так и для небольших 2D игр под мобильные платформы, при этом поддержка 2D в Unity появилась относительно недавно.

В настоящий момент на базе Unity активно создаются браузерные игры. Ярким примером может служить игра отечественных разработчиков - Джаггернаут (рис.2).

Большинство разработчиков используют ядро Unity для моделирования виртуальных объектов, но применение Unity на этом не ограничивается [4]. Так, например, данное ядро отлично себя зарекомендовала для решения задач визуализации объектов реального мира.



Рис.2. Интерфейс игры Джаггернаут, созданной с помощью движка Unity

К подобным сферам применения можно отнести виртуальные туры по музеям, памятникам и другим культурным объектам.

Визуализация объектов строительства дает возможность предварительно понять и оценить объект с различных точек зрения, застройщику и потенциальному покупателю [2].

Создание различных тренажеров, на которых можно было бы осуществлять тренировку в виртуальном пространстве, также является актуальной задачей. Это связано с тем, что большинство тренажеров дорого стоят и их покупка обходится существенно дороже разработки виртуального тренажера. В таких случаях использование виртуального тренажера очень удобно и является оптимальным.

Заключение

Несмотря на кажущуюся простоту в освоении и применении Unity следует учитывать тот факт, что для разработки полноценных и успешных проектов в этой среде следует либо обладать компетенциями дизайнера, программиста и сценариста, либо иметь соответствующую команду профессионалов. В любом случае Unity – это всего лишь удобный инструмент, а качество результата напрямую зависит от опыта и квалификации того, кто его применяет.

Литература

1. Unity3D - видеоуроки, уроки, статьи и документация к игровому движку по-русски [Электронный ресурс]. – URL: <http://unity3dforge.com/>
2. 4 преимущества Unity3D, а также полезный инструмент для ускорения процесса разработки приложений / Блог компании XIM & XIMAD / Хабрахабр [Электронный ресурс]. – URL: <http://habrahabr.ru/company/ximad/blog/252525/>
3. Unity - Руководство: Руководство Unity [Электронный ресурс]. – URL: <http://docs.unity3d.com/ru/current/Manual/UnityManualRestructured.html>
4. Unity3D для реальной реальности [Электронный ресурс]. – URL: <http://www.pvsm.ru/unity3d/30451>