ЗІНЧЕНКО С.Є. , ДЕМКІВСЬКА Т.І.

MESH DRAWING PIPELINE У ДВИГУНІ UNREAL ENGINE 4 ТА ЙОГО ВИКОРИСТАННЯ

ZINCHENKO S.E. , DEMKIVSKA T.I.

MESH DRAWING PIPELINE IN THE UNREAL ENGINE 4 ENGINE AND ITS USE

With the development of computing power of mobile devices, the relevance of the use of modern visualization technologies in the field of entertainment is increasing.

The purpose of this study is to develop a software product that will use modern technologies of three-dimensional visualization Physically-Based Rendering and technology for rendering three-dimensional grids Mesh Drawing Pipeline. The capabilities of these technologies will be demonstrated by the example of a game for the Android platform. The Unreal Engine 4 game engine will be used to develop the program, as well as the Microsoft Visual studio IDE for some internal game algorithms. Blender will be used to create three-dimensional models of game objects.

Вступ

Кожного року ринок смартфонів зростає, з'являються все більш потужні пристрої, з сучасними програмними та апаратними можливостями. З розвитком обчислювальної потужності, збільшується і попит на комп'ютерні розваги, такі як ігри та інші додатки. Перед розробниками постають більш комплексні і складні задачі по розробці великої кількості високоякісних продуктів.

Щоб задовольнити потреби невеликих компаній, які не мають можливості розробити свій ігровий двигун, у 2014 році компанією Epic Games було створено двигун Unreal Engine 4, високорівневе рішення, яке дозволяє автоматизувати та спростити рутинні задачі розробників.

В роботі проведено дослідження принципів роботи конвеєра відображення тривимірних сіток Mesh Drawing Pipeline та розглянуто сучасну технологію рендерінгу кадра Physically-Based Rendering.

Постановка завдання

Основною метою дослідницького проєкту є розробка мобільної гри для Android на базі ігрового рушія Unreal Engine 4, та дослідження принципів роботи конвеєра Mesh Drawing Pipeline.

Додаток має отримати повноцінну функціональність тривимірної гри:

- відображення повідомлень для гравця;
- архітектуру яка забезпечить повний ігровий цикл;
- відображення тривимірних об'єктів з урахуванням матеріалів та освітлення;
- програвання ігрових звуків;
- відображення спеціальних ефектів;

– використання технології доповненої реальності.

Основна частина

У двигуні Unreal Engine 4 Mesh Drawing Pipeline використовується конвеєр який відповідає за відображення тривимірних сіток. Технологія базується на концепції режиму збереження, де більшість процесів підготовки сцени проводяться заздалегідь, а не будуються кожен кадр. Конвеєр також проводить агресивне кешування та злиття викликів малювання для того, щоб використовувати властивості статичних мереж, які змінюються нечасто і результат операцій над ними може бути збережений і використаний повторно через велику кількість кадрів.

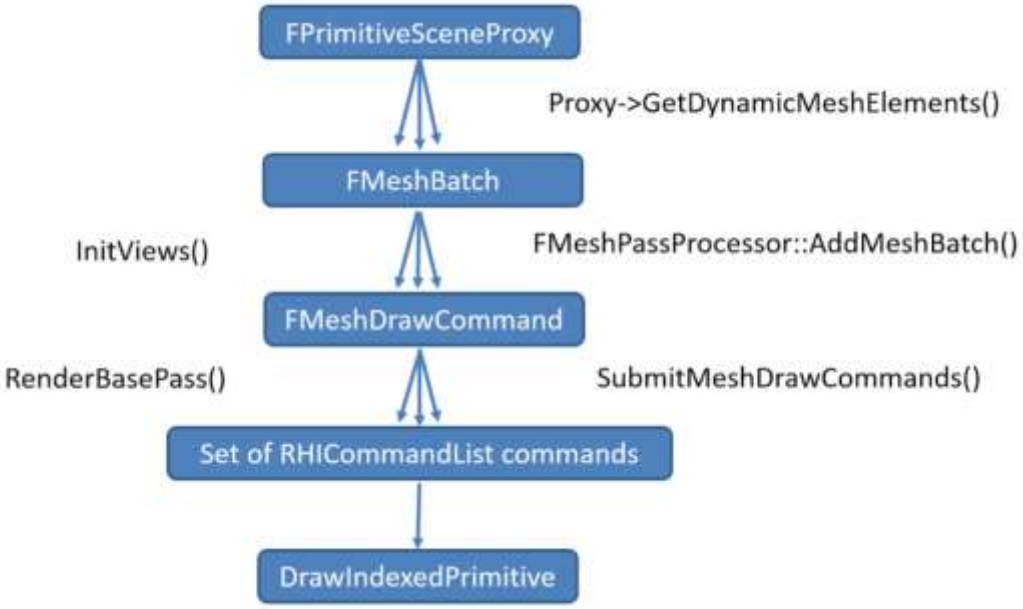

Схема Mesh Drawing Pipeline.

Процес рендерінгу сіток починається з об'єкта FPrimitiveSceneProxy, який є репрезентацією об'єкту UPrimitiveComponent в гільці рендерінгу. FPrimitiveSceneProxy і відповідальний за доставлення груп тривимірних сіток FMeshBatch в рендерер за допомогою зворотніх викликів таких як GetDynamicMeshElements та DrawStaticElements. FMeshBatch відокремлює імплементацію користувацького коду в FPrimitiveSceneProxy від потоків сіток, і включає в себе всі данні що потік повинен знати, наприклад остаточні шейдерні прив’язки та стани рендеру, таким чином що FPrimitiveSceneProxy ніколи не знає які потоки оброблюються.

Наступний крок перетворення FMeshBatch в специфічний для потоку сіток команди FMeshDrawCommand. FMeshDrawCommand - це інтерфейс який використовується між FMeshBatch та RHI. Це повністю безстановий опис малювання, який зберігає все що потрібно знати RHI про виклик відображення:

– які шейдери використовувати;

– прив’язки ресурсів для шейдерів;

– параметри викликів малювання.

Це дозволяє кешування та злиття викликів малювання перед рівнем RHI, тим самим зменшуючи час обробки набору команд. FMeshDrawCommand створюється з FMeshBatch специфічним для потоку сіток процесором FMeshPassProcessor.

Наступним кроком є виклик SubmitMeshDrawCommands, який використовується для перетворення FMeshDrawCommand у серію RHI команд. В результаті формується список команд RHICommandList, використовуючи цей список, GPU малює усі примітиви які знаходяться на тривимірній сцені в полі зору камери.

На відміну від технології минулого покоління Forward Rendering, двигун Unreal Engine 4 використовує Dithered Rendering сучасний фізично коректний підхід до рендерінгу зображення Physically-Based Rendering, який передбачає створення окремих проходів, кожен з яких відображає інформацію про певну властивість матеріалу чи освітлення. Наприклад:

– diffusion - колір поверхні;

– roughness - нерівності поверхні;

– metallicity - металічність поверхні;

– translucency and transparency (використовується для напівпрозорих поверхонь).

Після завершення рендерінгу останнього проходу GPU компонує данні проходів, отримуючи зображення яке відображає усі об'єкти на сцені з потрібними властивостями матеріалів та правильним освітленням.

Останнім етапом є Physically-Based Post Process - обробка кадру. Post Process включає в себе спеціальні ефекти такі як Bloom, Depth of Field, Vignette або Screen Space Reflections які покращують загальний вигляд зображення або імітують недосконалість лінз камери в реальному світі.

Висновки

Проведено дослідження принципів роботи конвеєра відображення тривимірних сіток Mesh Drawing Pipeline у двигуні Unreal Engine 4. Ця технологія дозволила використовувати сучасний підхід для створення реалістичної Physically-Based Rendering графіки на мобільних пристроях, а також дозволила розробникам створювати більш гнучкі та комплексні матеріали для ігрових об’єктів без значної втрати продуктивності в процесі створення кадру.

Розроблено програмний продукт, що відповідає всім стандартним вимогам тривимірної комп'ютерної гри. Додаток має зрозумілий інтерфейс, звукове супроводження ігрових подій, тривимірну графічну складову з ефектами та повноцінний ігровий цикл.

МІРОШНИЧЕНКО Д.В., ДЕМКІВСЬКА Т.І.

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ СТВОРЕННЯ СИСТЕМИ КОНВЕРТАЦІЇ ФОРМАТІВ .XLS, XLS X, HTML, PDF НА БАЗІ TELEGRAM BOT API

MIROSHNYCHENKO D.V., DEMKIVSKA T.I.

DEVELOPENT OF SOFTWARE FOR CREATING THE FORMAT XLS, XLS X, HTML, PDF CONVERT SYSTEM BASED ON TELEGRAM BOT API

The article touches the subject of to use the telegram platform and its telegram api bot to implement an automated bot that will convert files of different formats.

It also explains the principles of developing an automated bot that are embedded in the design structure, which allows you to build the work process as quickly and efficiently as possible. Which in turn allows you to reduce the load on the server and minimize its work.

Вступ

У сучасному світі ключовим фактором розробки програмного забезпечення є якість і швидкість розробки. Великі компанії розуміючи це, створюють великі платформи, які дозволяють самостійно реалізовувати будь-які програмні рішення.

Платформа мессенджера “Telegram” дозволяє на своїй базі програмно створити бота, який може реалізовувати різні програмні рішення. Це дає змогу заощадити час та кошти.

Постановка завдання

Головною метою даного дослідження є використання платформи мессенджера “Telegram” в створенні бота, який буде конвертувати файли з одного формату в інший швидко, просто і максимально ефективно для користувача, а також розробка архітектури проекту та створення дружнього інтерфейсу взаємодії з користувачем. Бот має бути виконаний за правилами патернів Strategy та State. В залежності від вибору користувача формату для конвертації, реалізація конвертування має бути представлена по правилам паттерну Strategy, де в залежності від контексту ми можемо змінювати реалізацію виконання. Завданням роботи є демонстрація того, як телеграмм-бот конвертує файли з одного формату в інший де результатом виконання конвертації будет файл с потрібним для вас форматом сформований ботом в процесі конвертації .

Основна частина

Розроблено бот, який наглядно демонструє використання патерну Strategy для більш ефективної реалізації алгоритму швидкого конвертування з одного формату в інший.