

ФРЕЙМВОРК ДЛЯ РЕНДЕРІНГУ 3D СЦЕН НА ПЛАТФОРМАХ, ЩО ПІДТРИМУЮТЬ METAL API

Афанасьєва І. В., Перов О. С.

Харківський національний університет радіоелектроніки, Харків, Україна

З розвитком технологій апаратного забезпечення персональних комп'ютерів та мобільних пристроїв збільшуються попит на програмне забезпечення, яке могло б ефективно використовувати потужності сучасних графічних адаптерів. Програмування графіки залишається відносно невідомою формою програмування, про яку існує мало літератури, хоча ми і користуємося можливостями графічних адаптерів кожен день. Дуже мало серед існуючої літератури тієї, що розрахована саме на людей без досвіду програмування графічних застосунків [1]. Однією з найменш задокументованих і розповсюджених технологій для створення графічних програмних застосунків є Metal, створений Apple у 2014 році в якості аналога DirectX, Vulkan і OpenGL. Пропріетарність цієї технології ускладнює її поширення серед розробників, проте дозволяє розробляти більш оптимізоване та ефективне програмне забезпечення, що використовує усі апаратні можливості як десктопних, так і мобільних платформ, які її підтримують.

Метою доповіді є побудова архітектури фреймворка для рендерінга 3D сцен на платформах, що підтримують Metal, з використанням сучасних підходів.

В доповіді наводяться результати порівнянь архітектурних рішень розроблюваної архітектури та існуючих аналогів. Наведені дані показують, що більшість фреймворків використовують OpenGL, що програє Metal у можливості створення архітектури з мінімальними накладними витратами, або взагалі не мають версії для платформ Apple. Також з тих, що мають відкритий код, жоден фреймворк не використовує такі сучасні підходи, як GPU-driven rendering, що дозволяє значно знизити навантаження на процесор і одночасно знизити залежність графічного адаптера від нього у процесі рендерінга кадру [2]. В зв'язку з цим чинності набуває потреба у створенні такого фреймворку, що використовував би сучасні підходи до створення архітектури для графічних програмних застосунків, використовував сучасний графічний API та підтримував як мобільні, так і десктопні платформи.

Список літератури

1. Mikkonen A. Graphics Programming Then and Now : How the ways of showing pixels on screen have changed. *Theseus*. 2021. URN: <https://urn.fi/URN:NBN:fi:amk-2021060213531>.
2. Peng C. High-performance computer graphics technologies in engineering applications. *World Journal of Engineering*. 2019. Т. 16, №2. С. 304–308. DOI: <https://doi.org/10.1108/WJE-05-2018-0158>.