ALAN BERBEROV

How do the Object-Oriented approach and DOTS impact the performance and efficiency of games made in Unity Engine on mobile devices?

Introduction

Each year, the game industry experiences growth alongside the increasing quality of games. While a significant number of people worldwide lack access to high-performance hardware for gaming, developers optimize games to ensure an enjoyable experience across different devices. Furthermore, in countries with low GDP per capita, mobile gaming has gained popularity, providing access to gaming for a wider population. Considering these factors, this Bachelor's thesis aims to explore the potential of different programming paradigms in improving the gaming experience.

Abstract

The thesis paper explores the analysis of multiple programming paradigms and their effect on performance in Unity Engine for mobile devices, emphasizing the most commonly used Object-Oriented approach and Data-Oriented Technology Stack (DOTS), which empowers game developers to efficiently expand processing capabilities while maintaining high performance. The paper investigates these two methods by defining their core principles, advantages, and drawbacks while evaluating how they can theoretically optimize mobile games made in Unity Engine. Additionally, it explores existing projects that have employed these paradigms, considering their success in improving mobile game performance to identify best practices.

During the Unity project phase, experimental projects will be created to measure each paradigm's (Object-Oriented approach and Data-Oriented Technology Stack) effectiveness through multiple tests, such as elementary physics simulations and rendering scenarios. Four of the most popular critical parameters: FPS (frames per second), CPU usage, GPU usage, and battery consumption, will be utilized as benchmarks to determine each method's impact on overall performance. The results will be reported in the documentation by various graphs, also presented as builds during the final presentation.

Personal Connection

This topic is relevant to me because optimization is one of the most fascinating and difficult topics in game programming. Writing high-performance code is a challenge for almost any level of programming. Since Unity recently released DOTS v1.0 in the LTS 2022, I decided to cover how it works and compare it to a classic Object-Oriented approach. However, benchmarking on a PC requires access to hardware I do not have. Meanwhile, I own various mobile devices that I am going to use as testing devices.

Methodology

Hardware:

- iPhone X
- iPhone 13 Pro
- iPhone 8
- Samsung Galaxy 21 Ultra
- Xiaomi Mi Mix 2

Software:

- Unity Engine
- Git
- Unity Remote 5
- JetBrains Rider

Timeline

Thesis:

Week 1 - Reading the sources and Introduction

Week 2 - Overview and Subsections 3.1 - 3.2 and 4.1 - 4.2

Week 3 - Subsection 3.3 - 3.4

Week 4 - Subsection 4.3 and revision

Week 5 - Conclusion

Week 6 - Abstract and polishing

Project:

Week 1 - Setting up projects and Hardware

Week 2 - Coding custom benchmarking tools

Week 3 - Object-Oriented based scene rendering

Week 4 - Object-Oriented based scene rendering

Week 5 - DOTS based scene rendering

Week 6 - DOTS based scene rendering

Week 7 - Object-Oriented based physics simulation

Week 8 - Object-Oriented based physics simulation

Week 9 - DOTS based physics simulation

Week 10 - DOTS based physics simulation

Week 11 - Polishing / Buffer

Week 12 - Polishing / Buffer

Bibliography

Books:

Fernando, R. (2007). *GPU gems : programming techniques, tips, and tricks for real-time graphics*. Boston: Addison-Wesley.

Nguyen, H. and Nvidia Corporation (2008). *GPU gems 3*. Upper Saddle River: Addison-Wesley.

Pharr, M. (2006). *GPU gems 2 : programming techniques for high-performance graphics and general-purpose computation*. Upper Saddle River: Addison-Wesley.

Websites:

DB, G.I.G. (n.d.). *Genshin Impact Game DB*. [online] Genshin Impact Game DB. Available at: https://www.gensh.in/genshin-impact/mobile-performance.

docs.unity3d.com. (n.d.). *Entities overview | Entities | 1.0.10*. [online] Available at: https://docs.unity3d.com/Packages/com.unity.entities@1.0/manual/index.html.

Fernández-Villaverde, J. and Guerrón, P. (2021). *Programming Paradigms (Lectures on High-performance Computing for Economists VII)*. [online] Available at: https://www.sas.upenn.edu/~jesusfv/Lecture_HPC_7_Programming_Paradigms.pdf.

Samsung Developers. (n.d.). *Adaptive Performance in Call of Duty Mobile*. [online] Available at: https://developer.samsung.com/galaxy-gamedev/gamedev-blog/cod.html.

Technologies, U. (n.d.). *Tic Toc - DOTS speeds up mobile game performance | Unity Case Study*. [online] unity.com. Available at: https://unity.com/case-study/tic-toc-games.

Unity Learn. (n.d.). DOTS Best Practices. [online] Available at: https://learn.unity.com/course/dots-best-practices.