

АРХІТЕКТУРНИЙ ПАТЕРН ENTITY COMPONENT SYSTEM

Пасічко В. В.

Науковий керівник – проф. Хаханова І.В.

Харківський національний університет радіоелектроніки (61166, Харків,
пр. Науки, 14, каф. АПОТ, тел. (057) 702-13-26)

e-mail: volodymyr.pasichko@nure.ua

Entity–component–system (ECS) is an architectural pattern that is mostly used in game development.

Для впровадження нових механік в гру, зазвичай, потрібно змінювати код у багатьох місцях, редагувати ієрархію класів, проводити оптимізацію та тестування із-за зміненої логіки. Звичайно, архітектуру можна продумати з самого початку, але в практиці це недосяжна ціль в програмуванні ігор, тому що дизайн-документ досить часто змінюється, якісь частини викидаються, додаються абсолютно нові частини, які ніяк не пов'язані зі старою логікою поведінки. ECS – це рішення цієї проблеми, тому що додає додатку велику гнучкість, сильно спрощує впровадження редагувань та подальше розширення додатків новим функціоналом без кардинальних змін у поточному коді. Мета дослідження - підвищення швидкості роботи з оперативною пам'яттю, прискорення розробки ігор, легкий перенос коду між абсолютно різними іграми, можливість створення тестів, великий рівень абстракції, можливість легко та швидко впроваджувати нові механіки. Задачі дослідження - розробка фреймворку для середовища Unity 3D, яке дає можливість використовувати паттерн ECS. Фреймворк вміщує інтерфейси для основних частин ECS, генератор коду та бібліотеку на написання тестів.

Entity -Component-System – це шаблон проектування, який забезпечує велику гнучкість в проектуванні загальної архітектури програмного забезпечення. Такі великі компанії, як Unity, Epic або Crytek використовують цей шаблон в своїх фреймворках, щоб надати розробникам дуже багатий можливостями інструмент, за допомогою якого вони розробляють власне ПО.

Основна ціль ECS – розподіл різних проблем і завдань між сутностями (Entities), компонентами (Components) і системами (Systems). Це три основні поняття цього шаблону. Сутність це контейнер для компонентів. Компонент – це невеликий об'єкт, який не володіє ніякою логікою. Зазвичай компонент містить одне чи два поля. В ідеалі це об'єкти з простою структурою даних. Кожен тип компонента можна прикріпити до сутності, щоб дати їй щось на зразок властивості. Наприклад, до сутності можна прикріпити «Health-Component», що дозволить зробити її смертною, давши їй здоров'я, яке є звичайним цілочисельним або дробовим значенням в пам'яті. Системи містять в собі логіку, наприклад,

переміщення чи запуск анімації. Системи працюють з сутностями, які мають потрібний для неї набір компонентів.

Компонент є найпростішою частиною у ECS. Це просте представлення даних. Він може бути порожнім, мати одне або багато властивостей або бути позначеним як унікальний.

Сутність - це лише контейнер для компонентів. Сутність завжди повинна бути частиною контексту. Контекст - це керуюча структура даних, яка контролює життєвий цикл об'єктів.

Існує декілька видів систем. Update System - це система, яка повинна виконуватися кожен кадр. В цих системах зазвичай описується логіка, яка повинна відбуватися під час усієї ігрової сесії, наприклад рух чи програвання анімації. Start system - цей вид систем спрацьовує лише один раз на початку ігрового циклу. Ці системи зазвичай використовуються для створення ігрових об'єктів. End System - це система, яка спрацьовує в кінці ігрового циклу, використовуються, зазвичай, для знищення ігрових об'єктів.

ECS має багато переваг на звичайною системою компонентів в Unity. Поділ логіки і даних. Можливість змінювати логіку (мінати системи, видаляти / додавати компоненти), не ламаючи дані. Тобто можна будь-який момент відключити групу систем, що відповідають за певну функціональність, і все інше продовжить працювати(крім відключеного функціоналу) і це не торкнеться поточних даних. Ефективне використання пам'яті. Можна перевикористати створені об'єкти сутностей і компоненти, використовуючи пули; можна використовувати типи-значення для даних і зберігати їх в пам'яті поруч (Data locality).

Результатом дослідження є фреймворк, який дозволяє використовувати архітектурний патерн ECS у середовищі Unity 3D. Його практична значимість полягає у тому, що він є кращим рішенням для розробки ігор, оскільки має великий рівень абстракції та логіку дуже легко редагувати чи доповнювати.

Список джерел:

1. Understanding Component-Entity-Systems [Електронний ресурс]. – 2013. – Режим доступу до ресурсу: <https://www.gamedev.net/articles/programming/general-and-gameplay-programming/understanding-component-entity-systems-r3013>.
2. Nystrom R. Game programming patterns / Robert Nystrom., 2012. – 50 с.