

ECS in Game Development

Jorge Pinto Sousa (He/Him), Luiz Jardim

July 2021

Critical TechWorks

The Plan

What we'll talk about today

- The basics of ECS
- Some frameworks and engines that support it
- Refactoring code into ECS with the Bevy game engine

What we won't talk about today

- When you should use ECS
- How to implement ECS

Entity-component-system (ECS)

Entity-component-system (ECS)

- It addresses some of the problems with object orientation while promoting code reusability, extendability, maintainability and parallelizability (is this a word?).
- One of its greatest features is easily modifying behaviour at runtime.

ECS - Characteristics

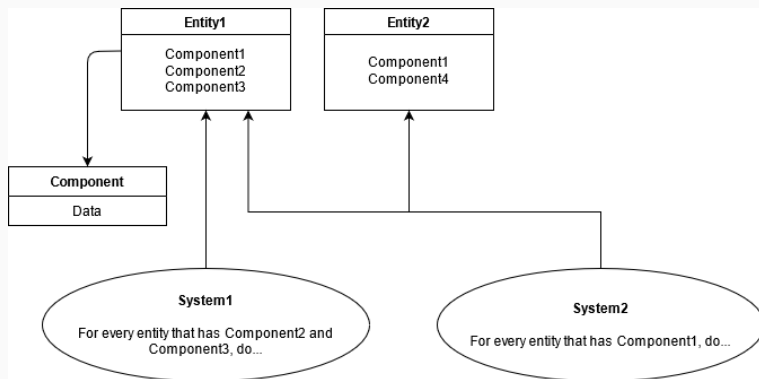
ECS has:

- **Entities** are unique “things” (identifiers).
- **Components** are just datatypes without **behaviour**
- **Systems** are functions that will act in **Entities** that have a certain set of **Components**.

Also:

- **Entities** can contain zero or more **components**.
- **Systems** can dynamically add or remove **components** from entities.
- Queries return a collection of **entities** that satisfy some **Component**-based predicate

ECS - Overview

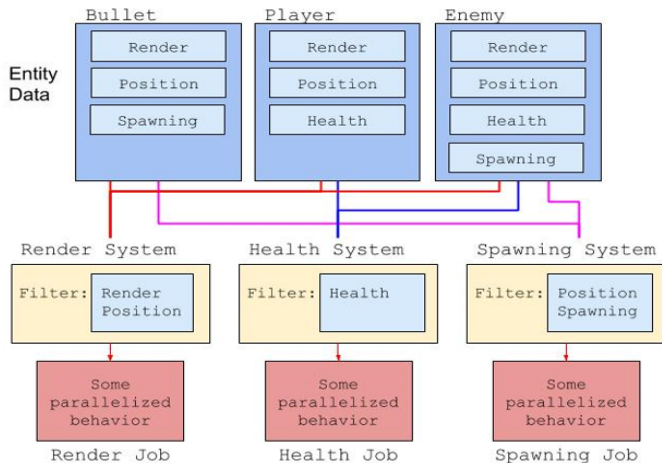


ECS vs OOP

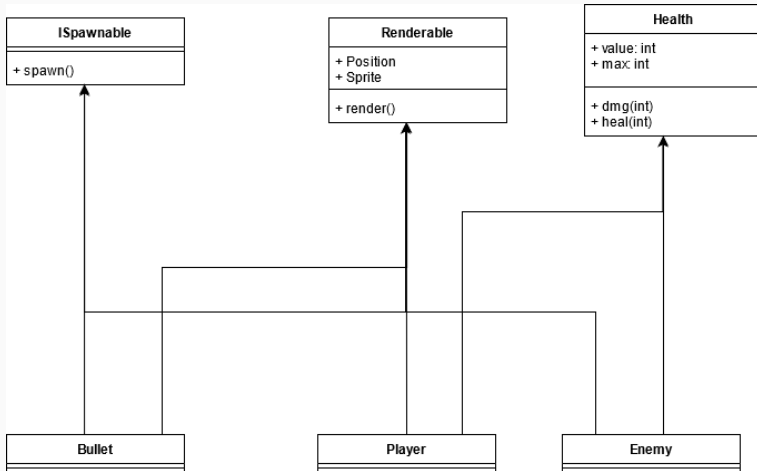
Is ECS too different from OOP?

- Composition only, no inheritance.
- Exposed data over encapsulation.
- Separate data and behaviour.
- OOP object instances are of a single non-changing type, while entities can have dynamically changing components.

ECS - An example



Without ECS



Engines, Frameworks and Libraries that support ECS

Engines and Frameworks that support ECS

- Unity (DOTS, Entitas, Svelto)
- EnTT
- Bevy
- Amethyst
- ECSY
- FLECS

References

- [Bevy](#)
- [ecs-faq](#)
- [Get Started with the Unity* Entity Component System \(ECS\)](#)
- [Bob Nystrom - Is There More to Game Architecture than ECS?](#)
- [Why isn't Godot ECS based?](#)
- [Vittorio Romeo - Implementation of a component-based entity system in modern C++](#)
- [Entity-Systems are the future of mmog development](#)