

# Entity-Component-System

## ECS由来

使用交通工具作为例子：

### 传统面向对象设计

	Bicycle	Car	Plane
属性	speed 速度 position 位置	speed 速度 position 位置 engine 引擎 frame 车架	speed 速度 position 位置 engine 引擎 wing 机翼
方法	Move() 移动 Turn() 转弯	Move() 移动 Turn() 转弯	Fly() 起飞 Turn() 转弯 Land() 降落 Move() 移动

```
public class Bicycle {
    private Vector3 _speed;
    private Vector3 _position;

    public void Move() {
        ...
    }
    public void Turn() {
        ...
    }
    private void Update(float dt) {
        _position += _speed * dt;
    }
}
```

```
public class Plane {
    private Vector3 _speed;
    private Vector3 _position;
    private Wing _wing;
    private Engine _engine;

    public void Move() {
        ...
    }
    public void Turn() {
        ...
    }
    public void Fly() {
        ...
    }
    public void Land() {
        ...
    }
}
```

```

    }
    private void Update(float dt) {
        _position += _speed * dt;
    }
}

```

缺点:

1. 属性/方法代码重复

## 组件模式

抽象出MovableComponent, 由各类型直接持有

```

public class MovableComponent {
    private Vector3 _speed;
    private Vector3 _position;

    public void Move() {
        ...
    }
    public void Turn() {
        ...
    }
    private void Update(float dt) {
        _position += _speed * dt;
    }
}

```

```

public class Plane {
    private MovableComponent _movable;
    private Wing _wing;
    private Engine _engine;

    public void Fly() {
        ...
    }
    public void Land() {
        ...
    }
}

```

极限设计:

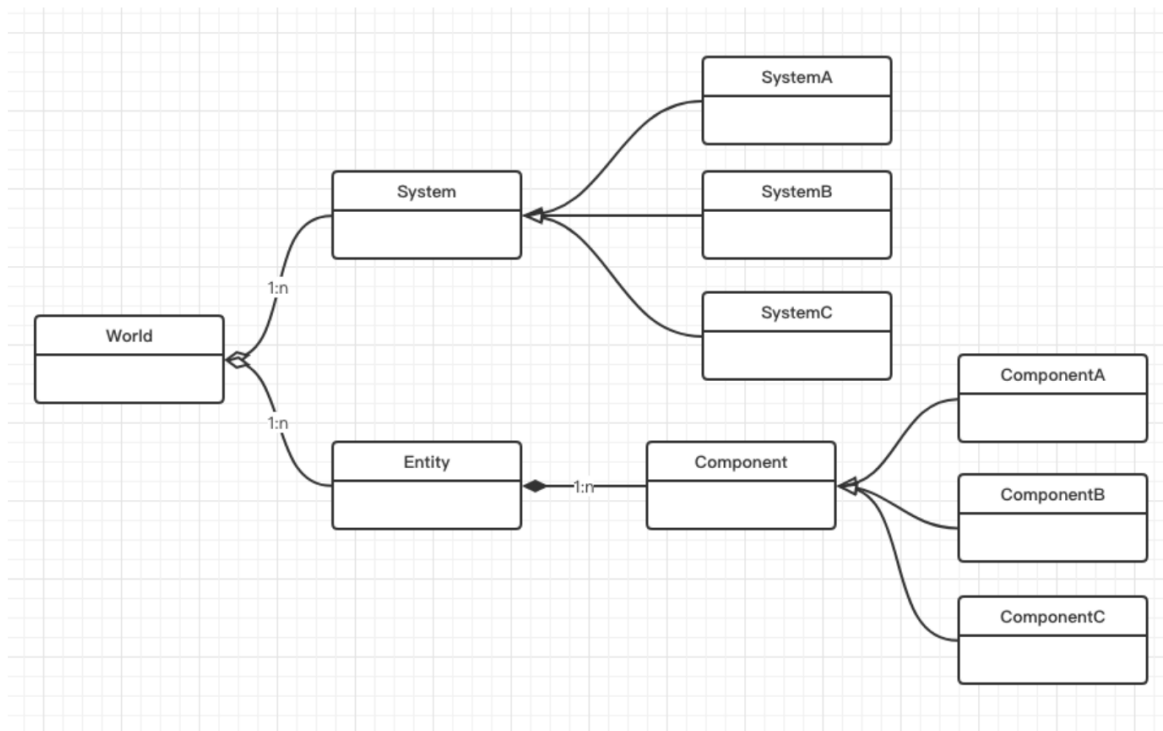
```

public class Plane {
    private MovableComponent _movable;
    private PowerSystem _powerSystem; // 动力系统
    private BrakingSystem _brakingSystem; // 制动系统
}

```

Plane成为彻底的容器, 增加TypeComponent后Plane类型也可舍弃

## ECS模式



- Entity 组件的集合
- Component 纯数据结构
- System 一个系列/一个组的数据结构的驱动逻辑

## 框架

<https://github.com/richardlord/Ash>

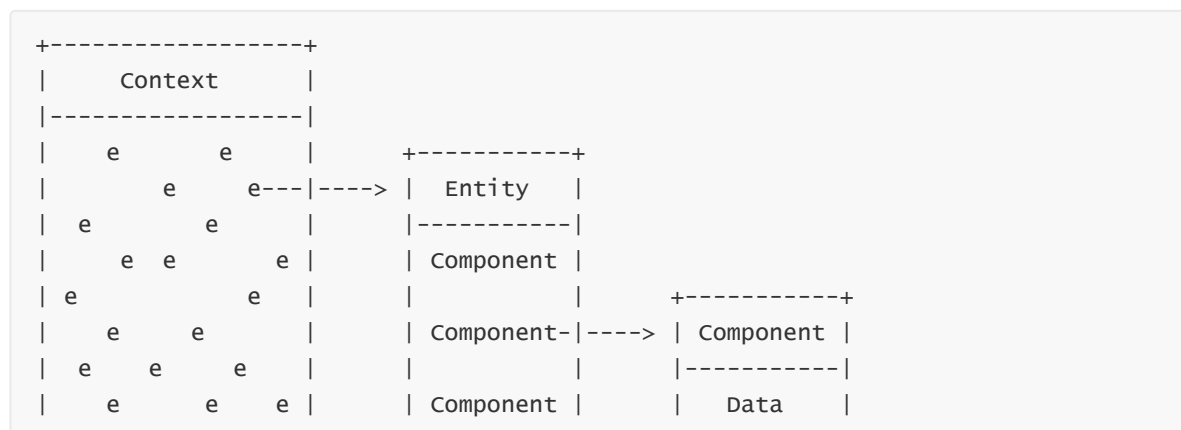
<https://github.com/sschmid/Entitas-CSharp>

## 文章推荐

[浅谈《守望先锋》中的 ECS 构架](#)

## Entitas

参考Wiki <https://github.com/sschmid/Entitas-CSharp/wiki/Introduction-to-Entitas>





3. Utils
4. Extension
5. 3rd