



# Having fun with Rust and Bevy

Trondheim Rust Meetup  
2024-02-13

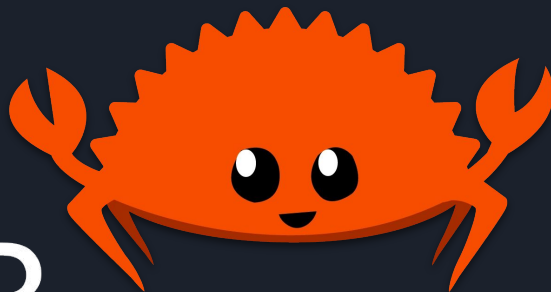


Me & Rust

Bekk

B k

B





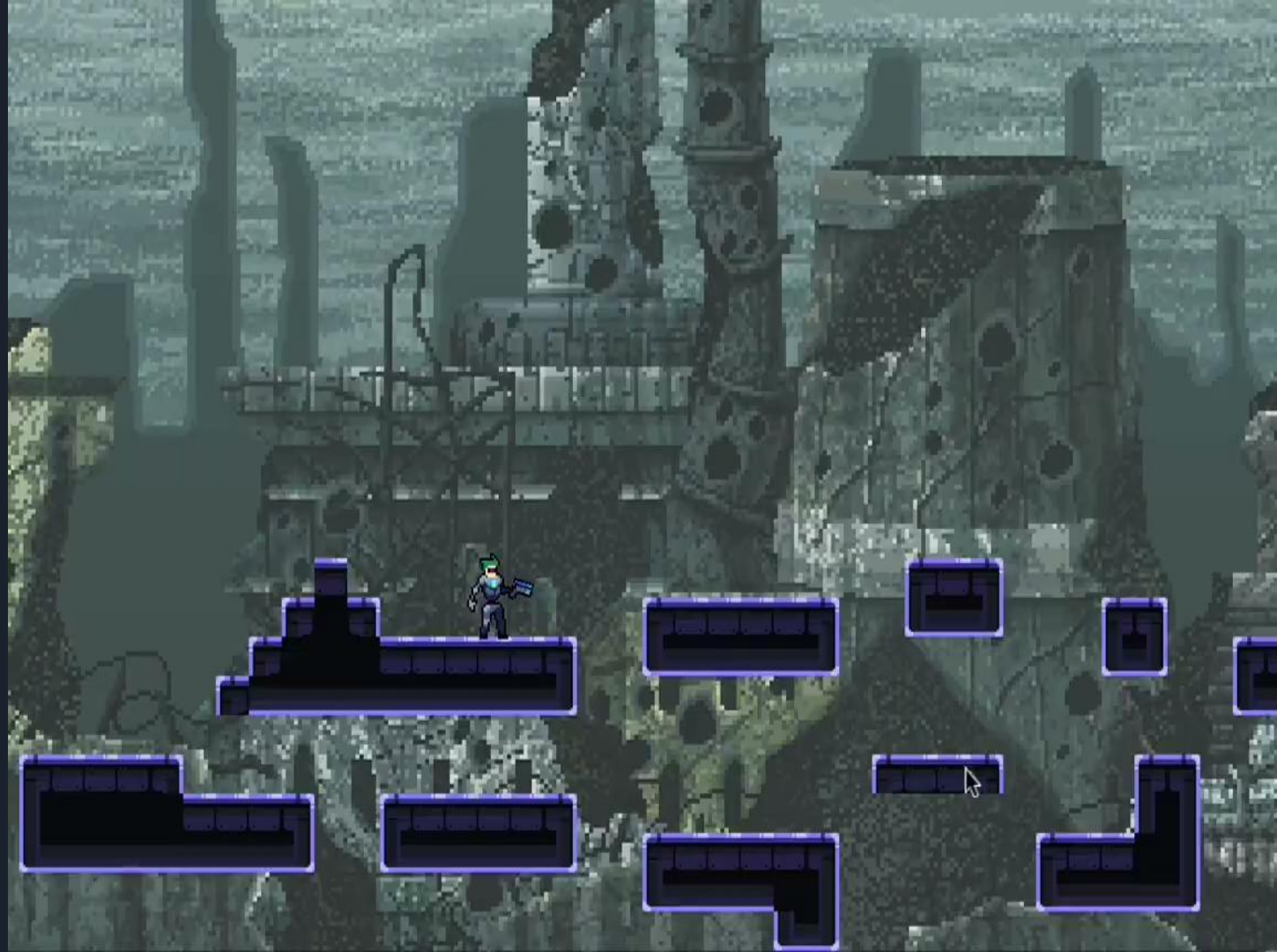
▼ Physics

Mode: Default

Box

Force Field







# Bevy



A refreshingly simple data-driven game engine built in Rust  
Free and Open Source Forever!

Get Started

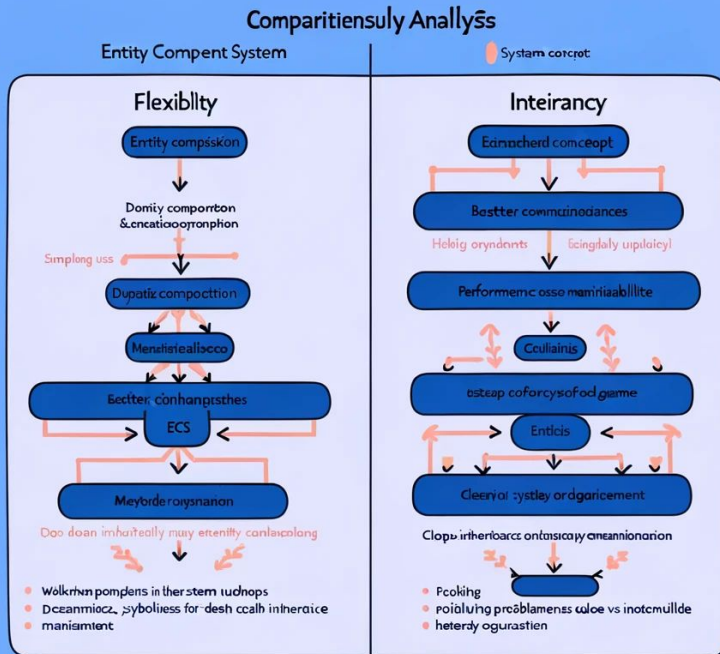
## Data Driven

All engine and game logic uses Bevy ECS, a custom Entity Component System

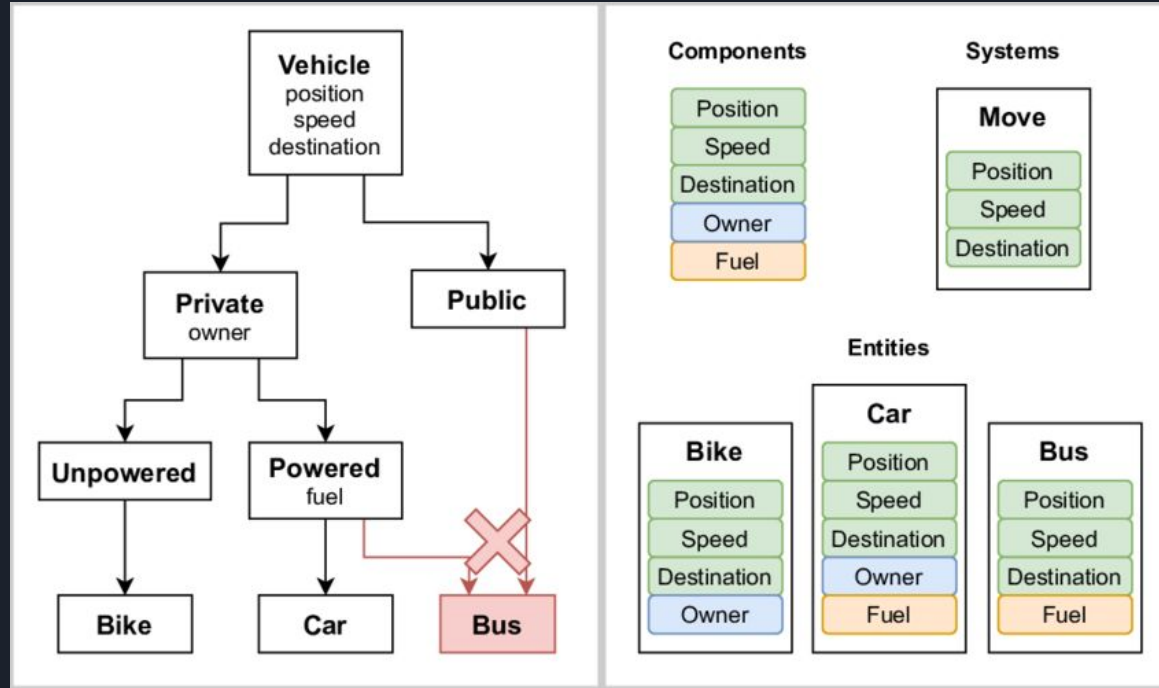
- **Fast:** Massively Parallel and Cache-Friendly. The fastest ECS according to some benchmarks
- **Simple:** Components are Rust structs, Systems are Rust functions
- **Capable:** Queries, Global Resources, Local Resources, Change Detection, Lock-Free Parallel Scheduler

```
#[derive(Component)]  
struct Player;  
  
fn system(  
    q: Query<Entity, &Player>  
) {  
}
```

# Entity Component System (ECS)



# Entity Component System (ECS)







# ECS in Bevy

- **Components:** Rust structs that implement the `Component` trait

```
#[derive(Component)]  
struct Position { x: f32, y: f32 }
```

- **Systems:** normal Rust functions

```
fn print_position_system(query: Query<&Position>) {  
    for position in &query {  
        println!("position: {} {}", position.x, position.y);  
    }  
}
```

- **Entities:** a simple type containing a unique integer

```
struct Entity(u64);
```



# ECS in Bevy - Plugins

```
pub struct HelloPlugin;

impl Plugin for HelloPlugin {
    fn build(&self, app: &mut App) {
        // add things to your app here
    }
}
```

Then register the plugin in your App like this:

```
fn main() {
    App::new()
        .add_plugins(DefaultPlugins, HelloPlugin)
        .add_systems(Startup, add_people)
        .add_systems(Update, (hello_world, greet_people))
        .run();
}
```

# ECS in Bevy - Resources

```
#[derive(Resource)]
struct GreetTimer(Timer);

fn greet_people(
    time: Res<Time>,
    mut timer: ResMut<GreetTimer>,
    query: Query<&Name, With<Person>>
) {
    // update our timer with the time elapsed since the last update
    // if that caused the timer to finish, we say hello to everyone
    if timer.0.tick(time.delta()).just_finished() {
        for name in &query {
            println!("hello {}!", name.0);
        }
    }
}
```

DEMO





# That's all, folks!

[https://github.com/geirsagberg/bevy\\_rust\\_meetup](https://github.com/geirsagberg/bevy_rust_meetup)