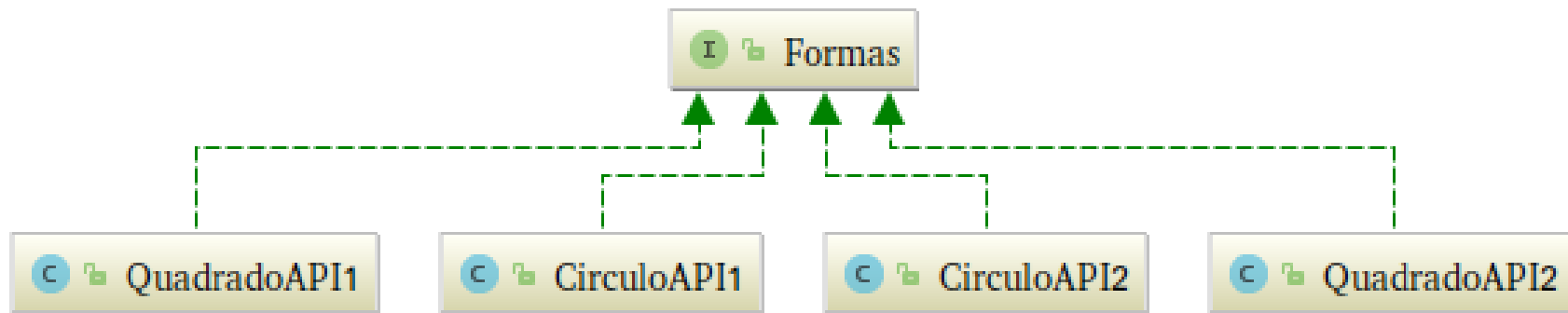




# Bridge

Eric dos Santos Coelho

# Problema

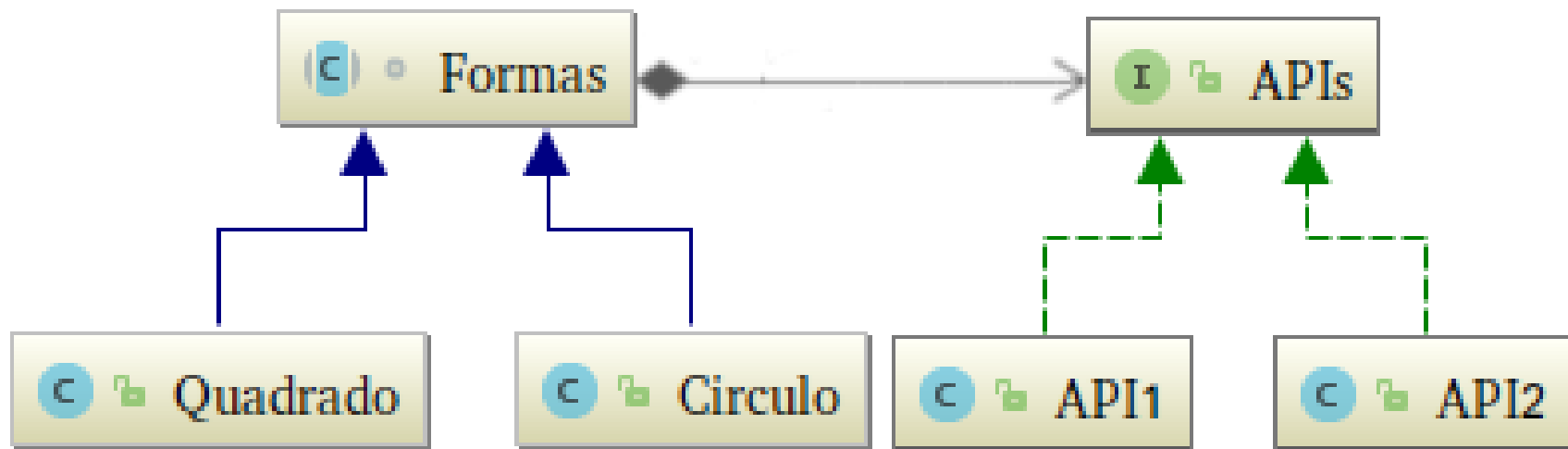




# Motivação

- Desacoplar uma abstração da sua implementação, de modo que as duas possam variar independentemente
- Abstrações e implementações podem ser estendidas independentemente
- Evitar proliferação de classes

# Solução



```

abstract class Formas {
    protected APIs APIdesenho;

    protected Formas(final APIs APIdesenho){
        this.APIdesenho = APIdesenho;
    }

    public abstract void desenhar();
}

```

```

public class Circulo extends Formas{

    protected Circulo(APIs APIdesenho) {
        super(APIdesenho);
    }

    @Override
    public void desenhar() {
        APIdesenho.desenharCircunferencia(raio)
    }
}

```

```

public interface APIs {
    void desenharLinha(double x, double y);
    void desenharCircunferencia(double raio);
}

```

```

public class APIx implements APIs {
    @Override
    public void desenharLinha(double x, double y) {
        /////
    }
    public void desenharCircunferencia(double raio){
        /////
    }
}

```



# Consequências

- A implementação pode ser configurada em tempo de execução
- Hierarquias da abstração e implementação podem ser estendidas independentemente