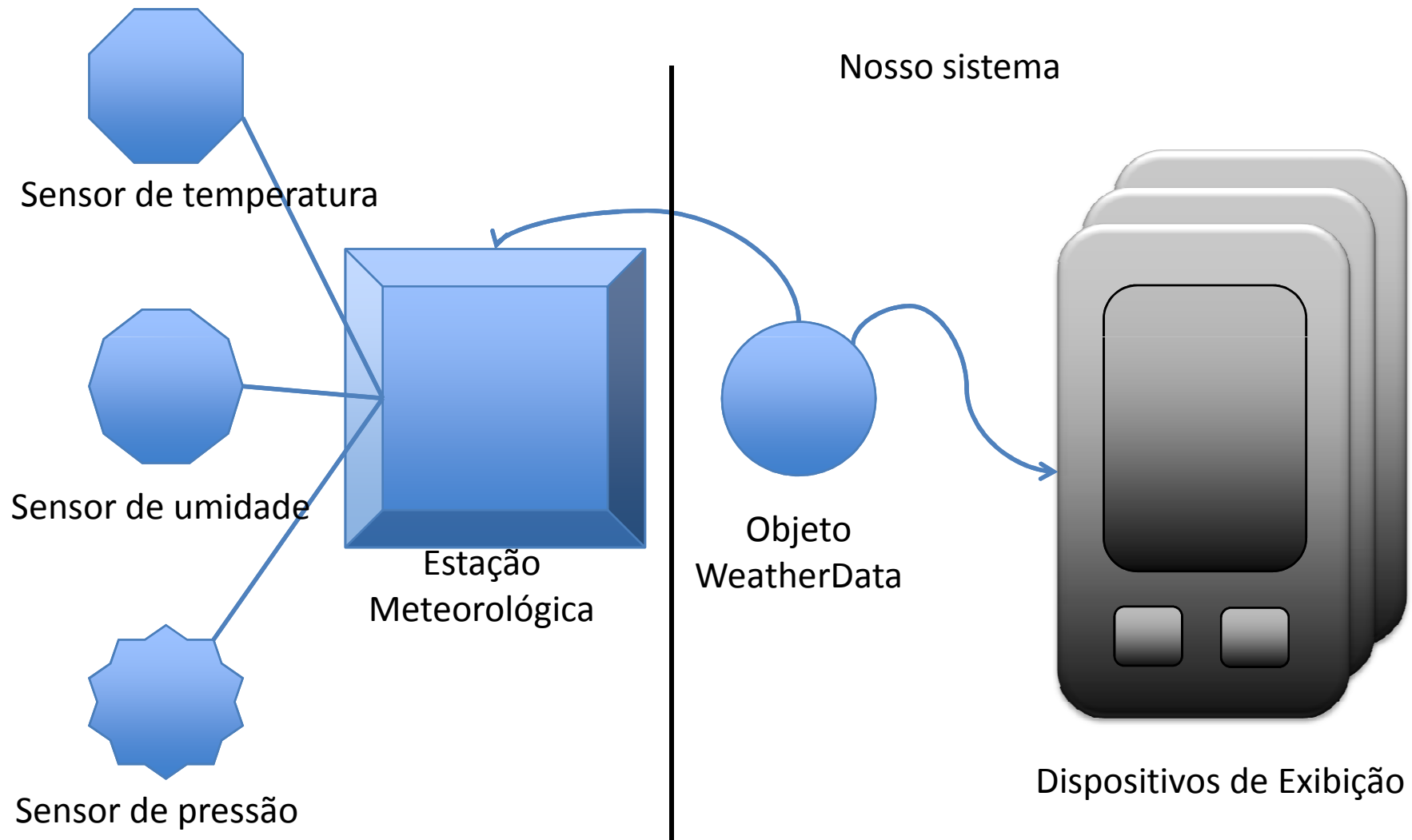
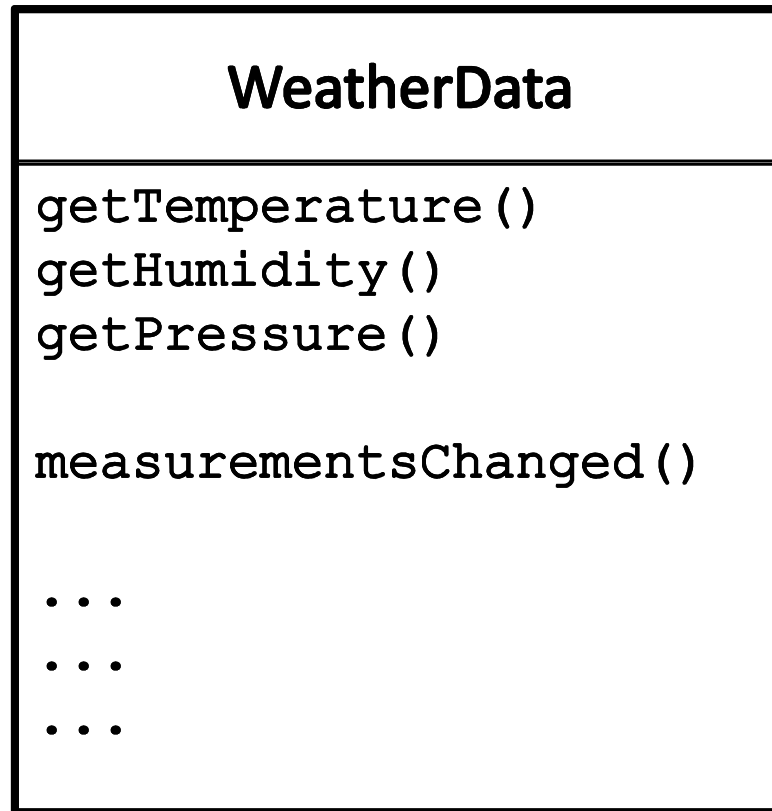


Padrão Observer

Estação Meteorológica



Objeto WetherData



Sobre o design

- O método `measurementsChanged()` é chamado sempre que novos valores dos dados meteorológicos estão disponíveis.
 - Não precisamos saber como.

Sobre o design

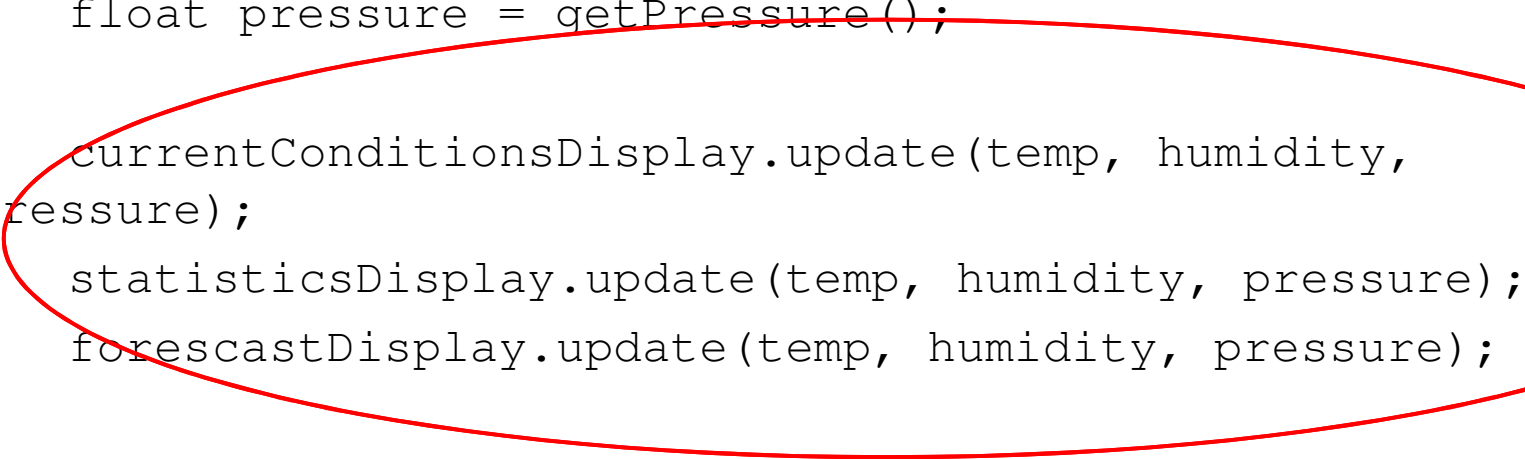
- Temos que implementar três dispositivos de exibição:
 - condições atuais (current conditions)
 - estatísticas (statistics)
 - previsão (forecast)
- O sistema deve ser expansível para novos dispositivos de exibição.

MeasurementsChanged() – V1

```
Public void measurementsChanged() {  
  
    float temp = getTemperature();  
    float humidity = getHumidity();  
    float pressure = getPressure();  
  
    currentConditionsDisplay.update(temp, humidity,  
pressure);  
    statisticsDisplay.update(temp, humidity, pressure);  
    forecastDisplay.update(temp, humidity, pressure);  
  
}
```

MeasurementsChanged() – V1

```
Public void measurementsChanged() {  
  
    float temp = getTemperature();  
    float humidity = getHumidity();  
    float pressure = getPressure();  
  
    currentConditionsDisplay.update(temp, humidity,  
pressure);  
    statisticsDisplay.update(temp, humidity, pressure);  
    forecastDisplay.update(temp, humidity, pressure);  
  
}
```



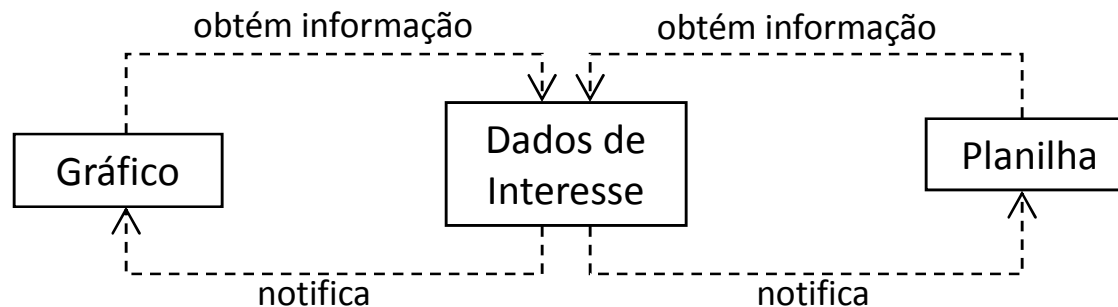
Qual o problema desse design?

MeasurementsChanged() – V1

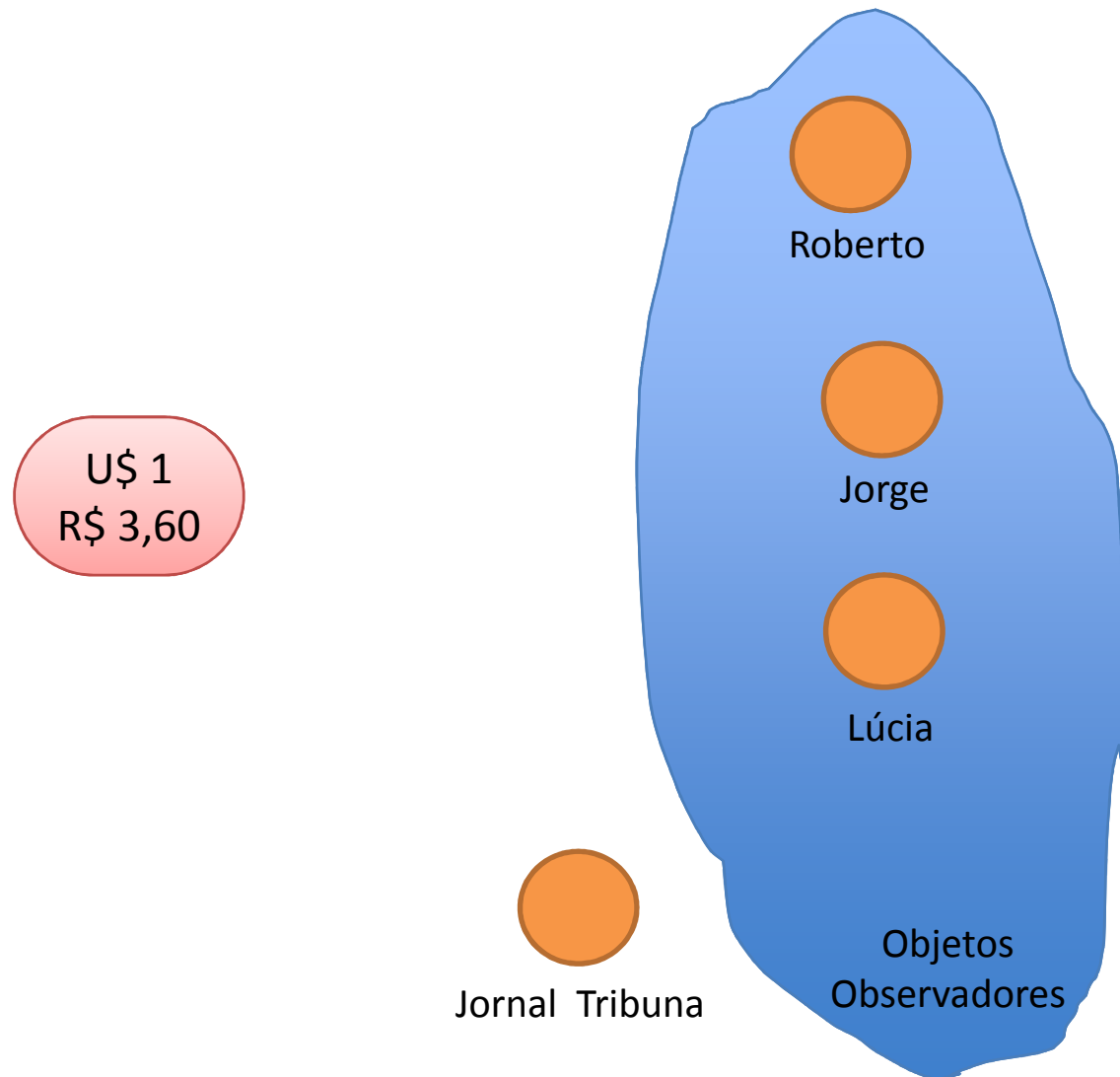
- Para adicionarmos ou removermos dispositivos de exibição, temos que alterar WeatherData.

Motivação

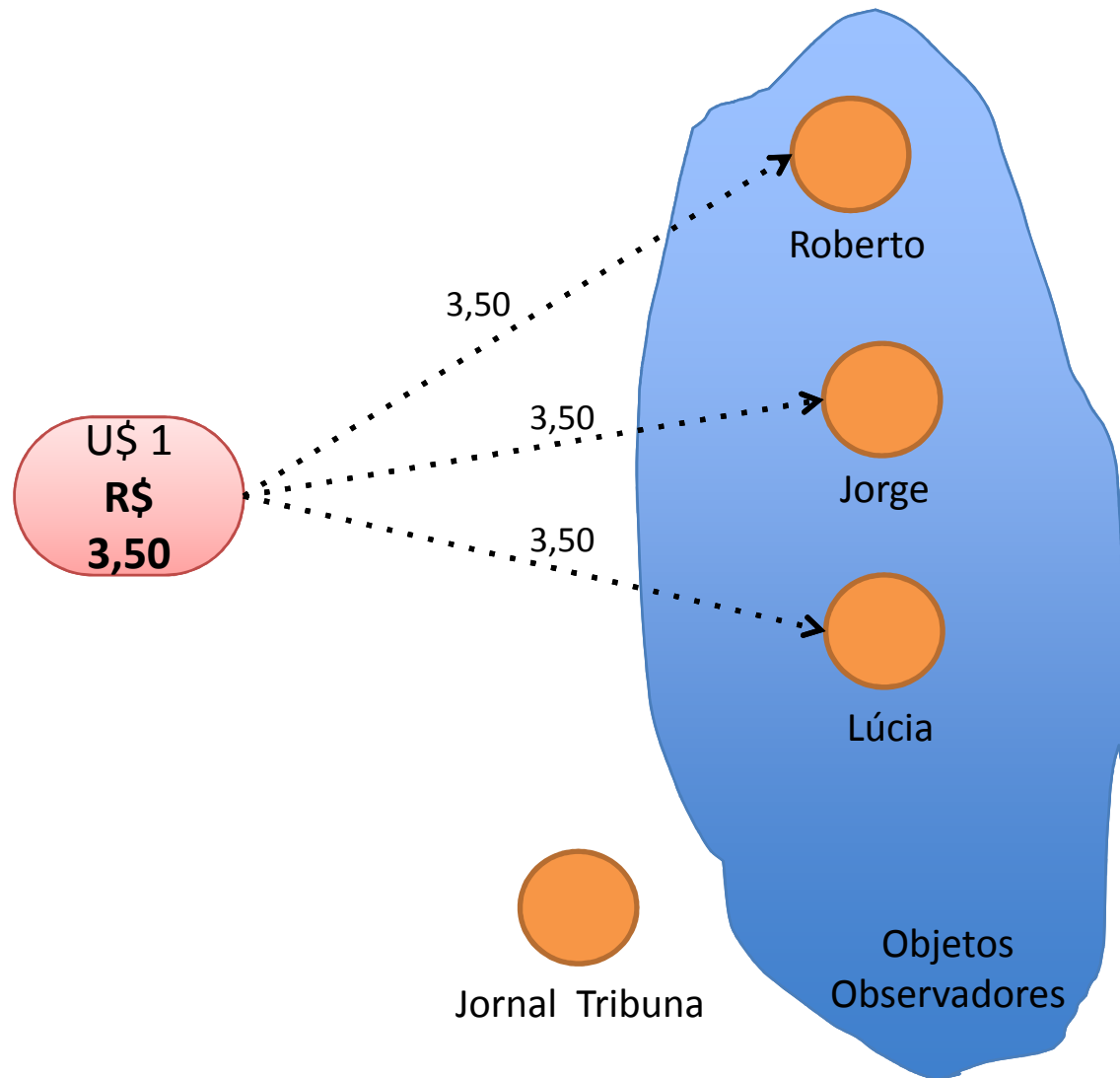
- Como garantir que objetos que dependem de outro objeto fiquem em dia com mudanças naquele objeto?
 - Como fazer com que os observadores tomem conhecimento do objeto de interesse?
 - Como fazer com que o objeto de interesse atualize os observadores quando seu estado mudar?
- Possíveis riscos
 - Relacionamento (bidirecional) implica alto acoplamento.



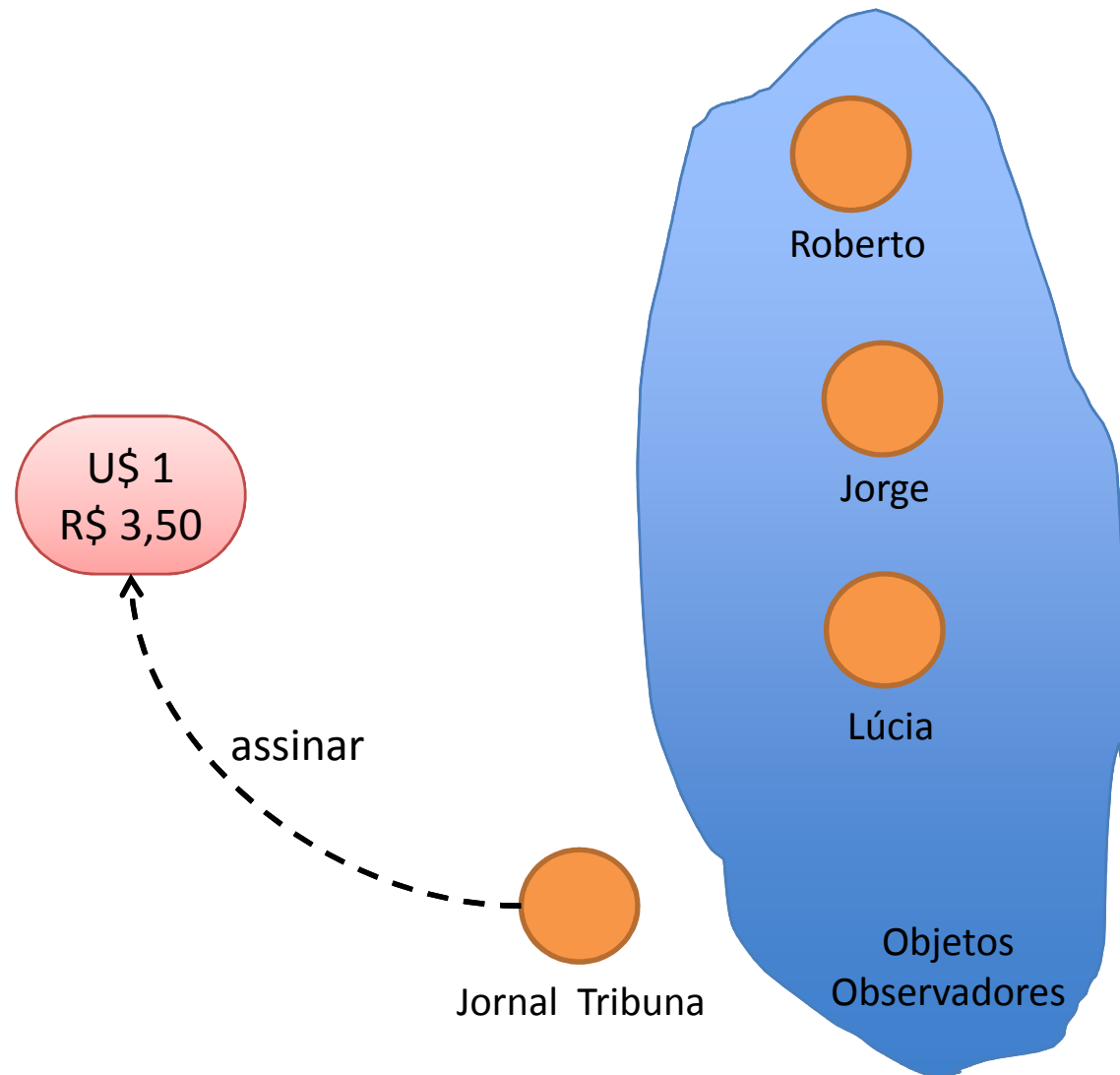
Idéia do Padrão Observer



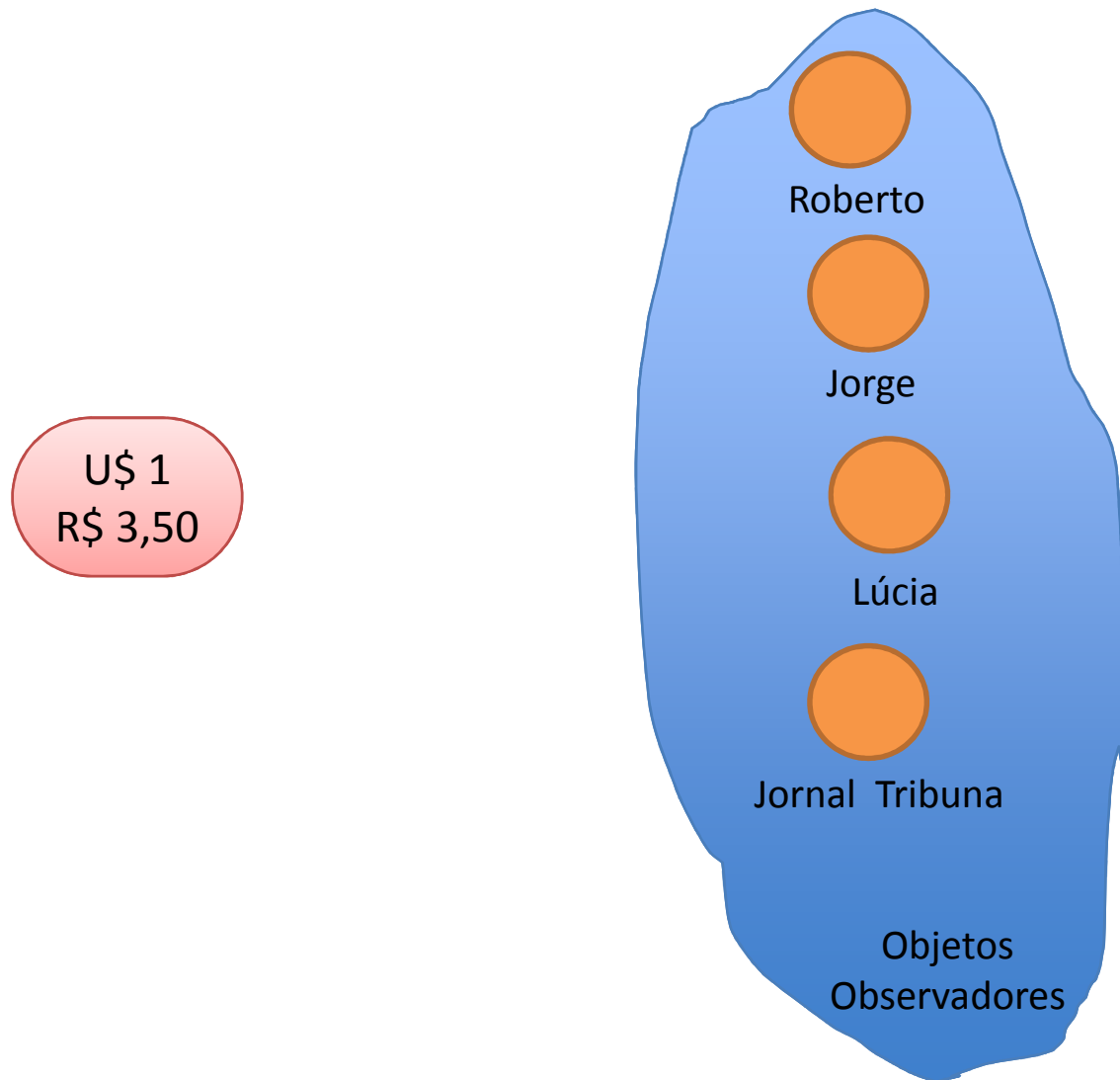
Idéia do Padrão Observer



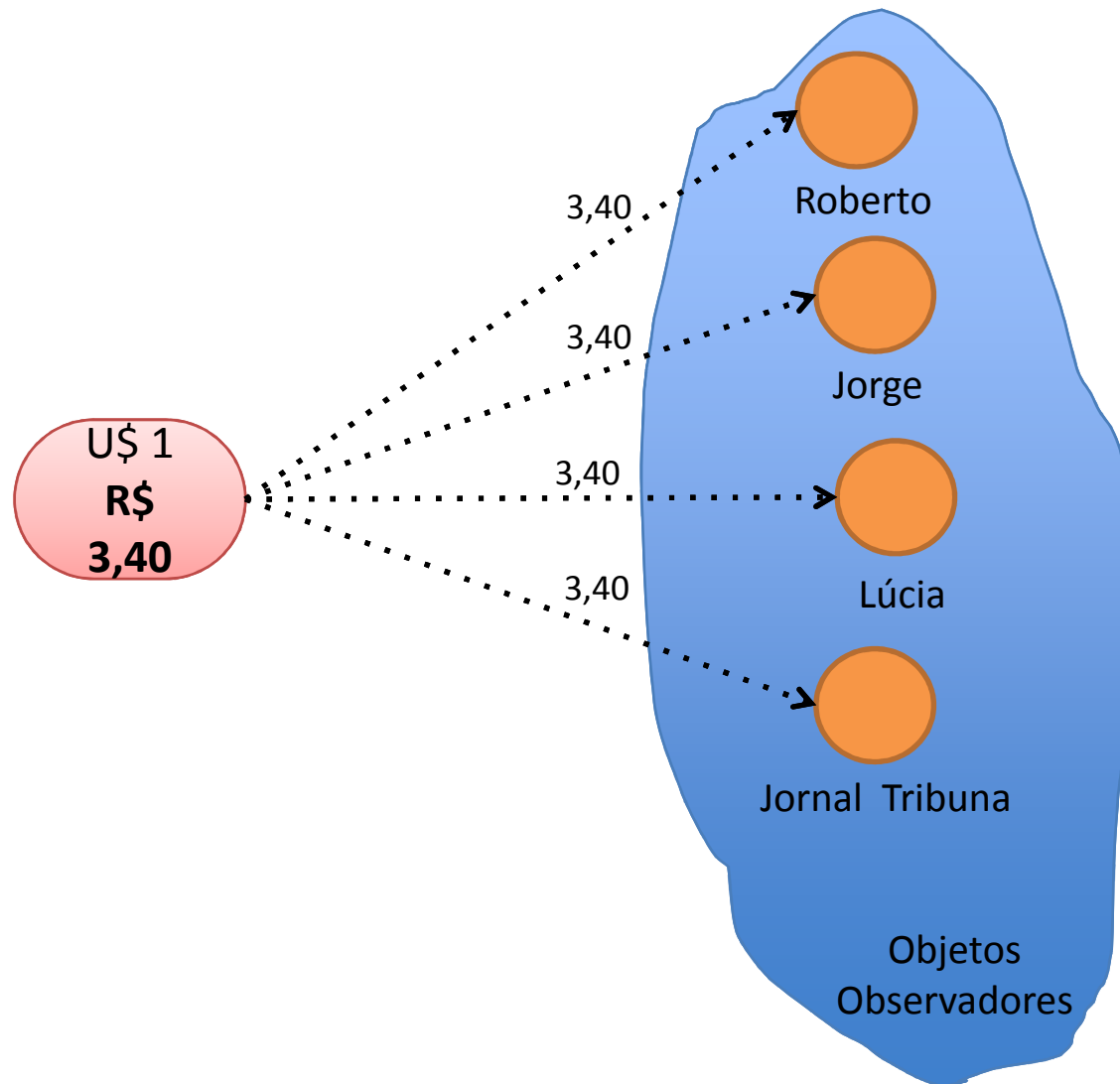
Idéia do Padrão Observer



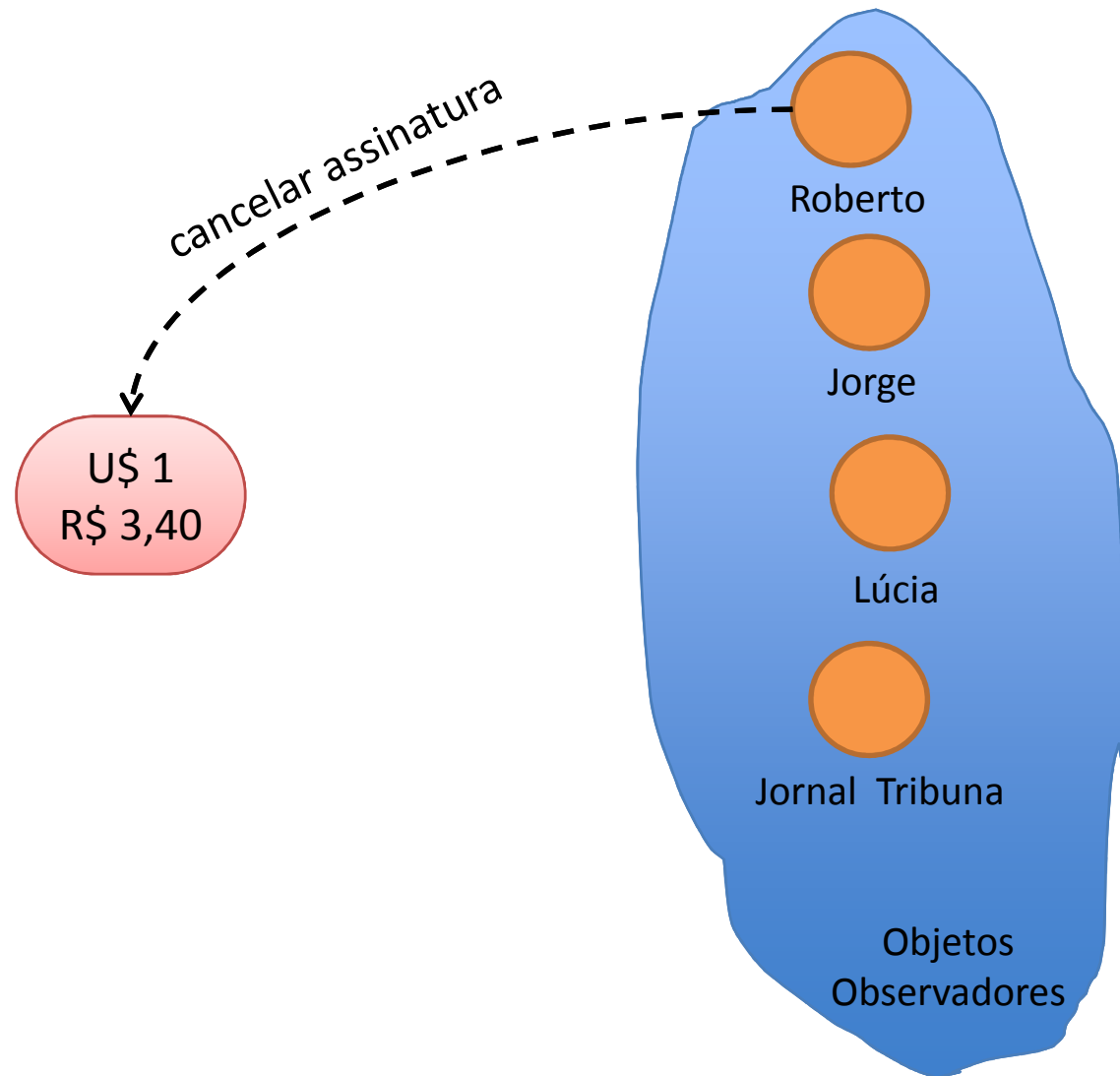
Idéia do Padrão Observer



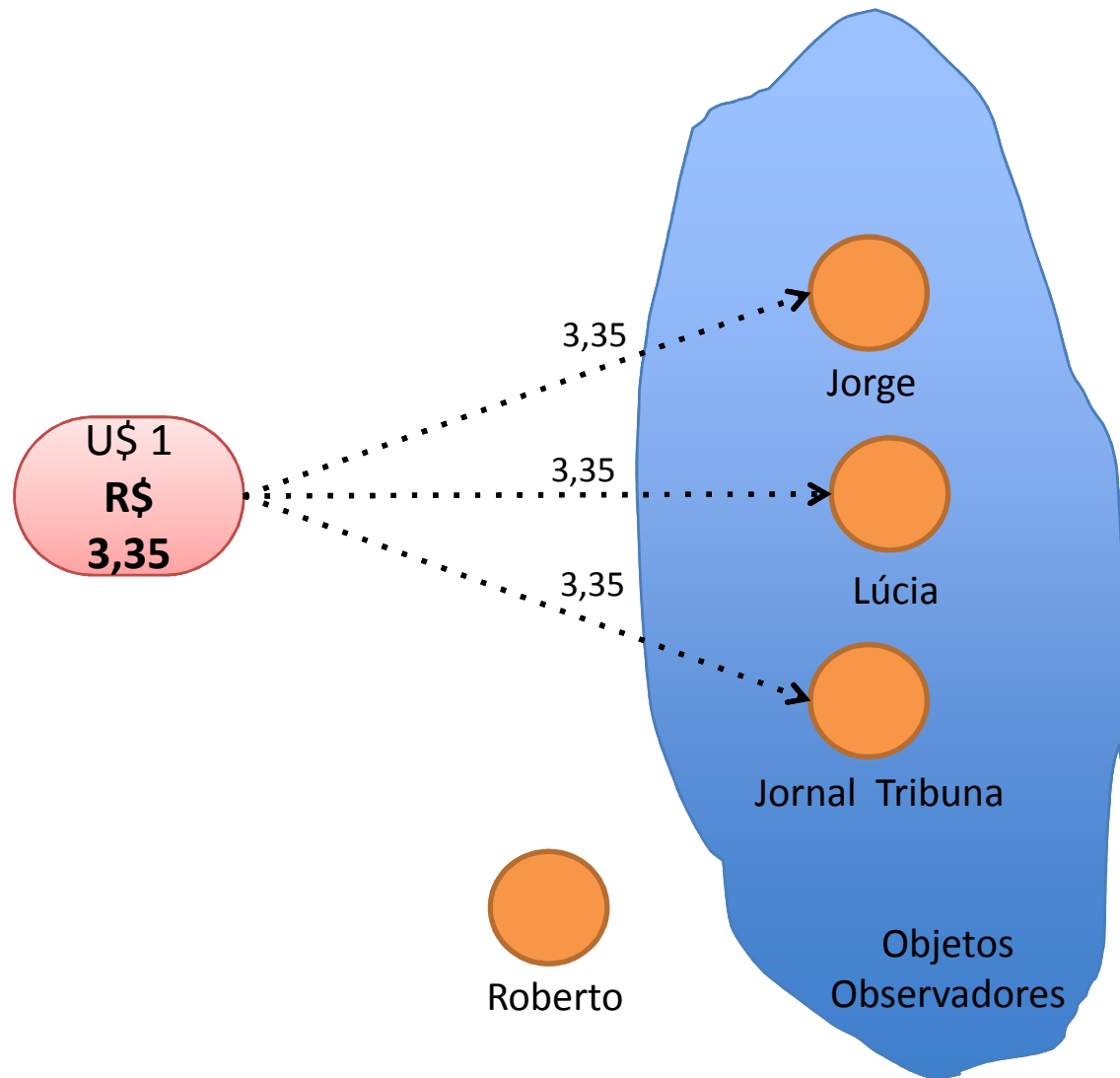
Idéia do Padrão Observer



Idéia do Padrão Observer



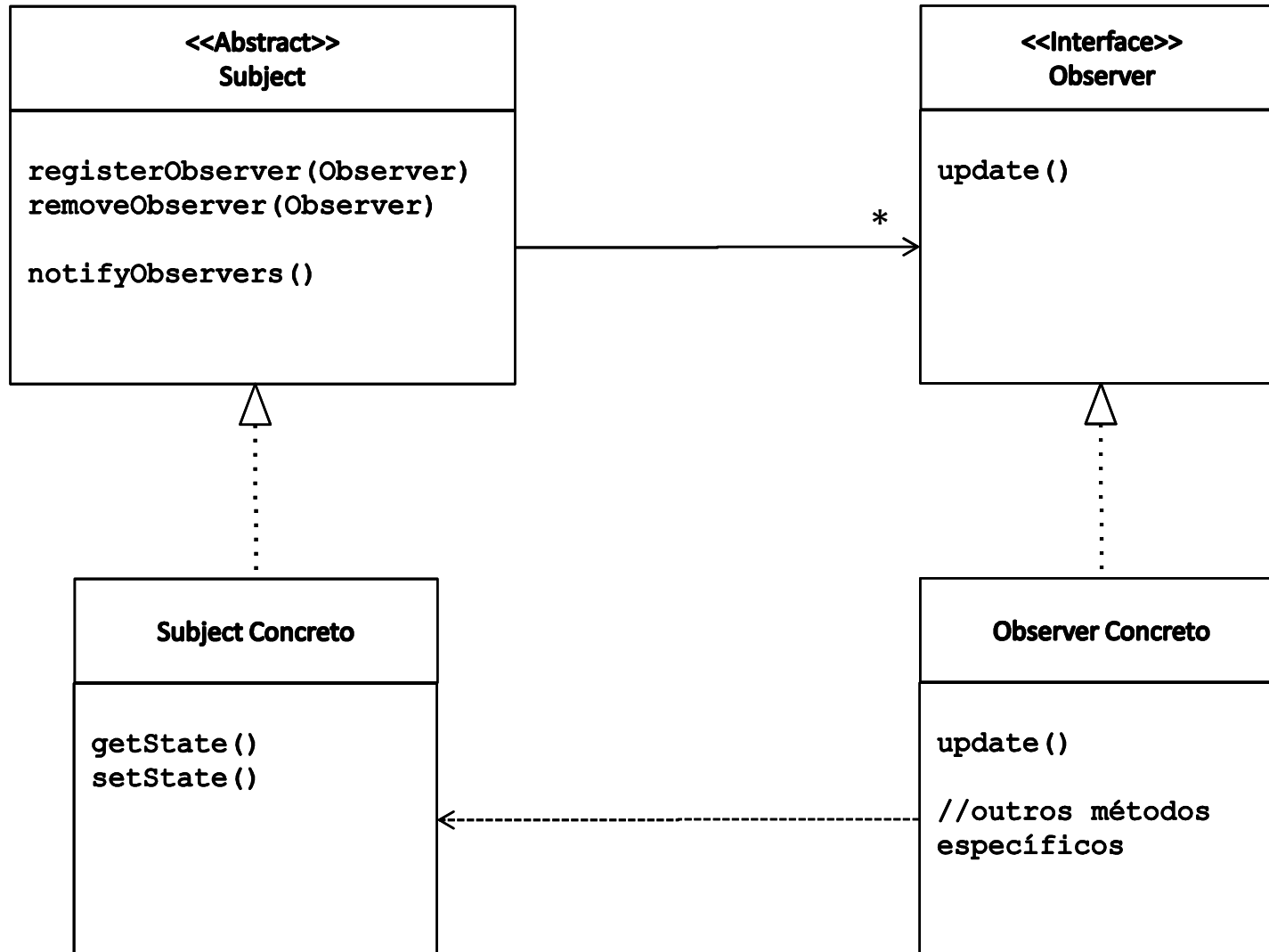
Idéia do Padrão Observer



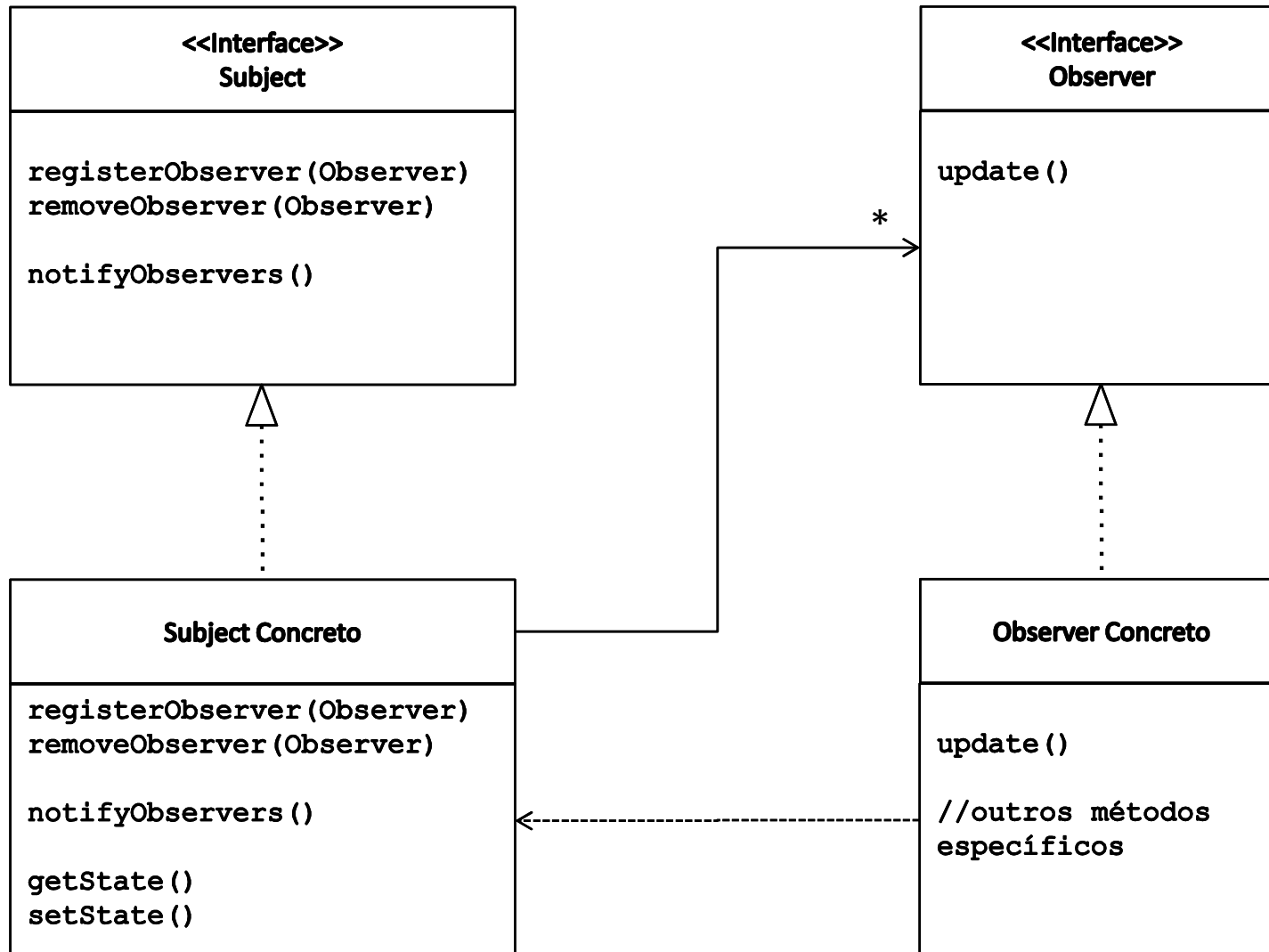
Padrão Observer

- Define a dependência entre objetos para que quando um mude de estado todos os seus dependentes sejam notificados.
- Só que ele faz isso por meio de dependências “leves”.

Padrão Observer



Padrão Observer



Padrão Observer

