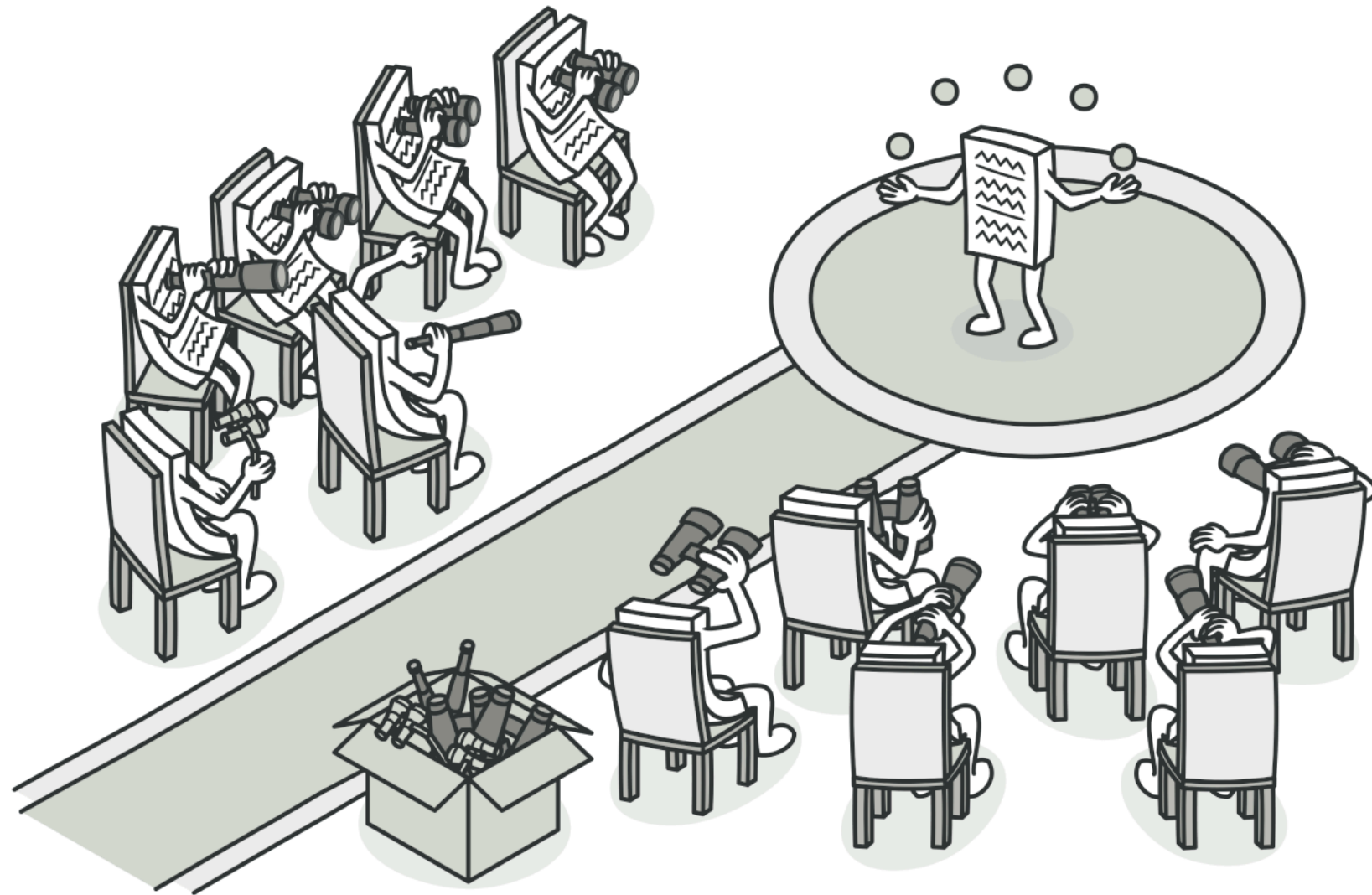


BACKEND FRAMEWORK (DJANGO)

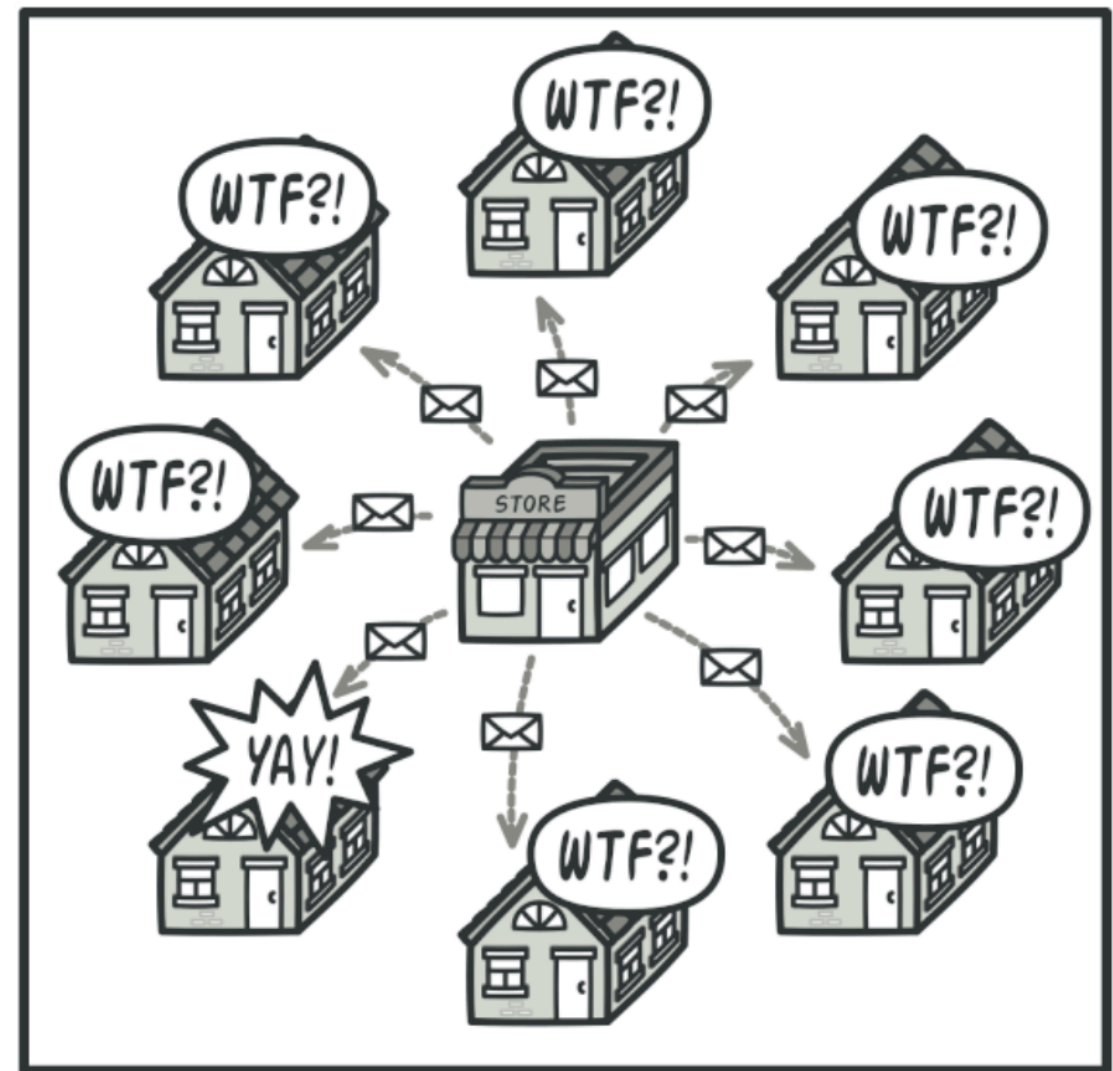
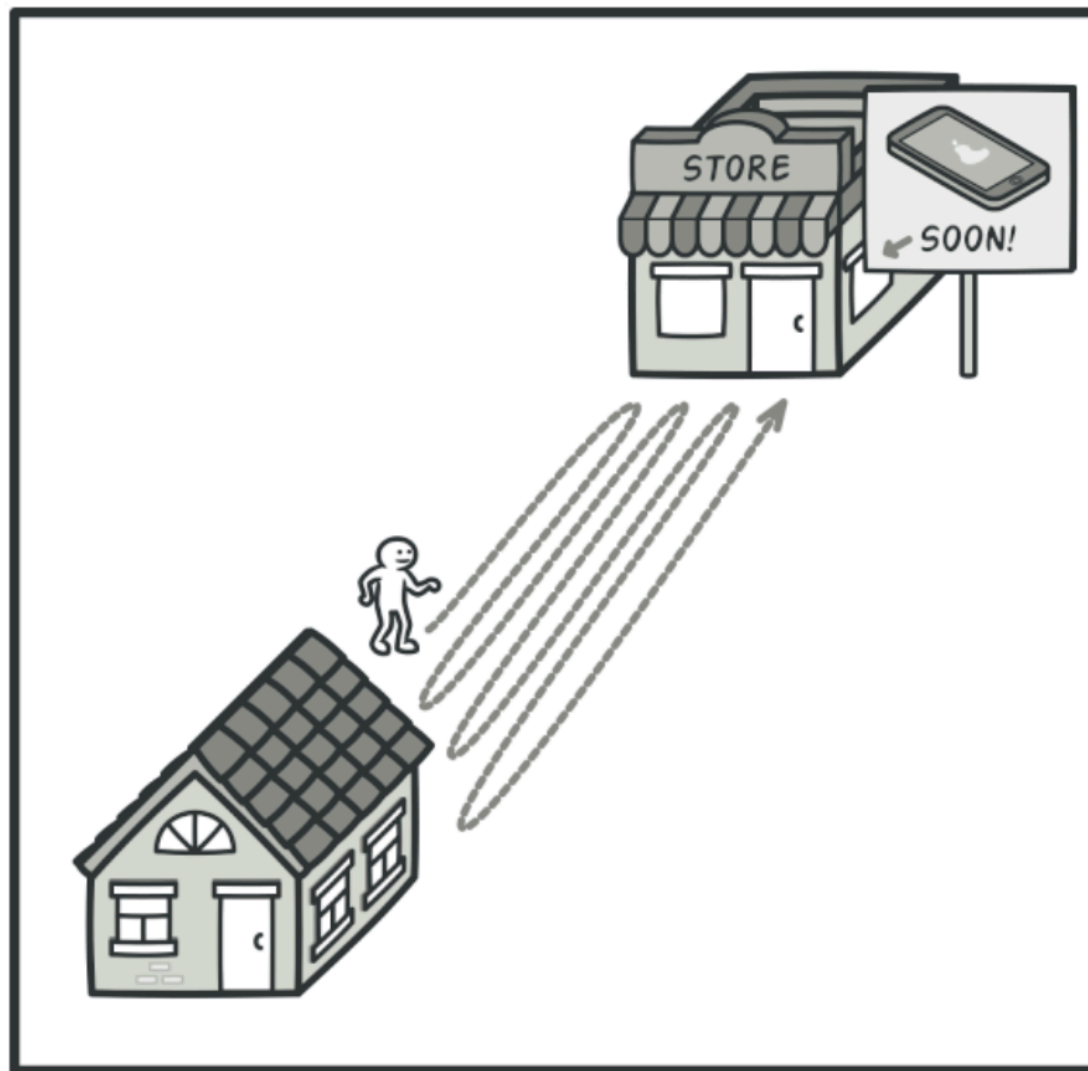
Lesson 10

Django Signals

Observer Design Pattern



Problem



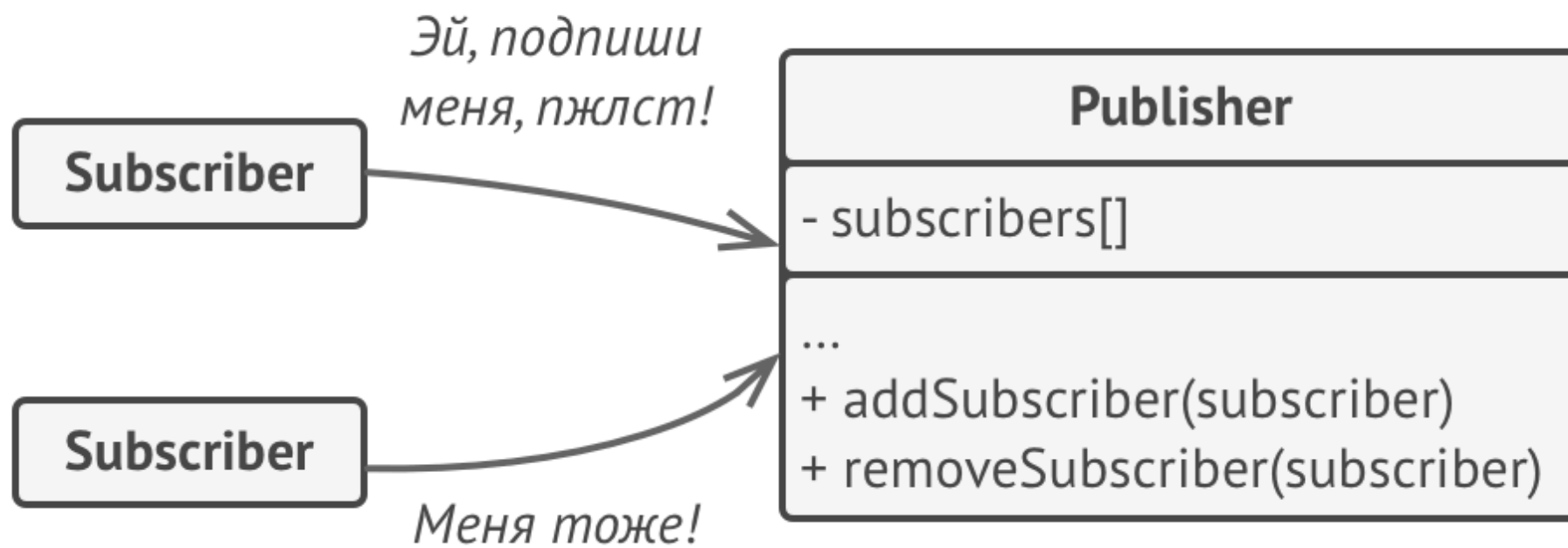
Solution

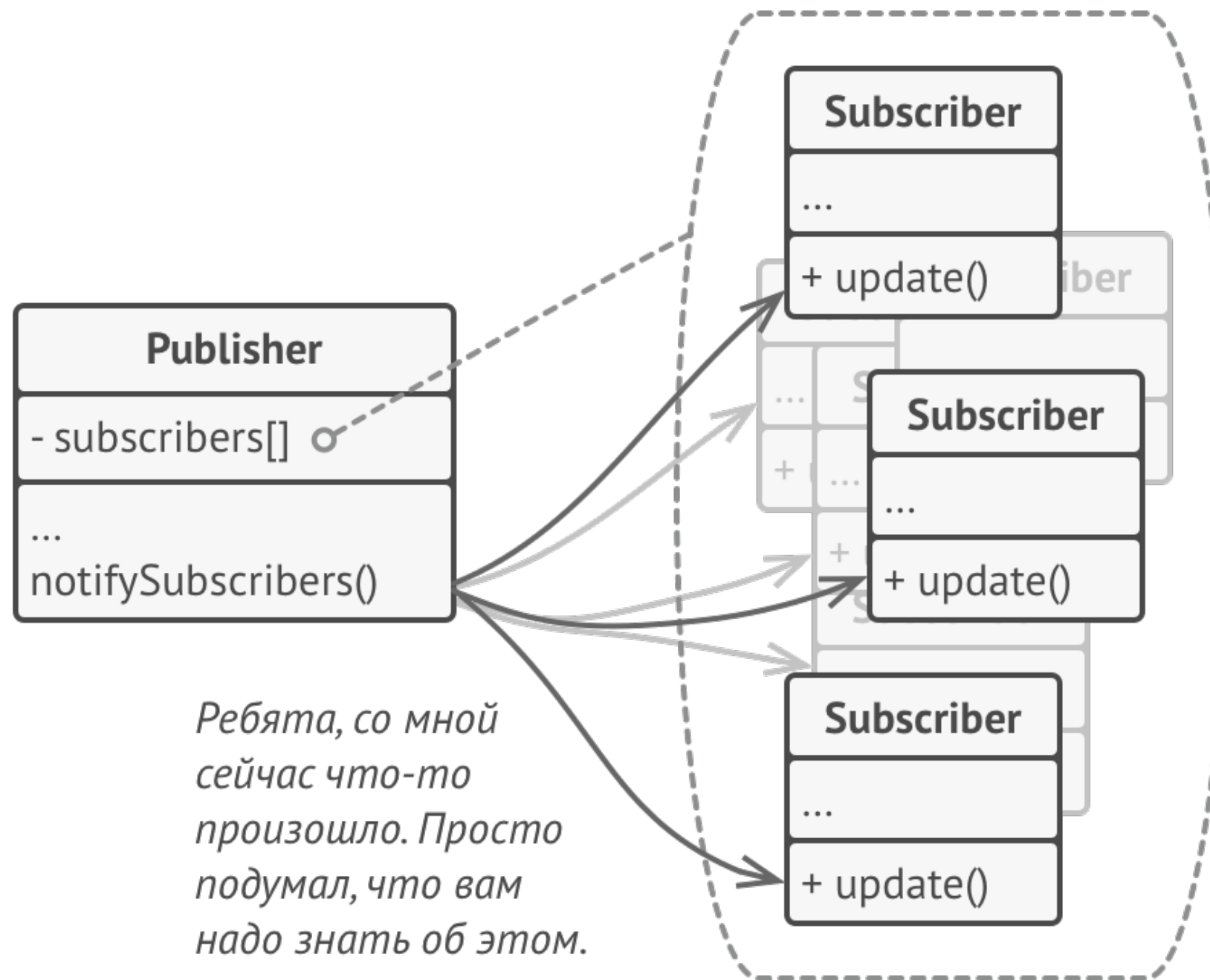
Publisher

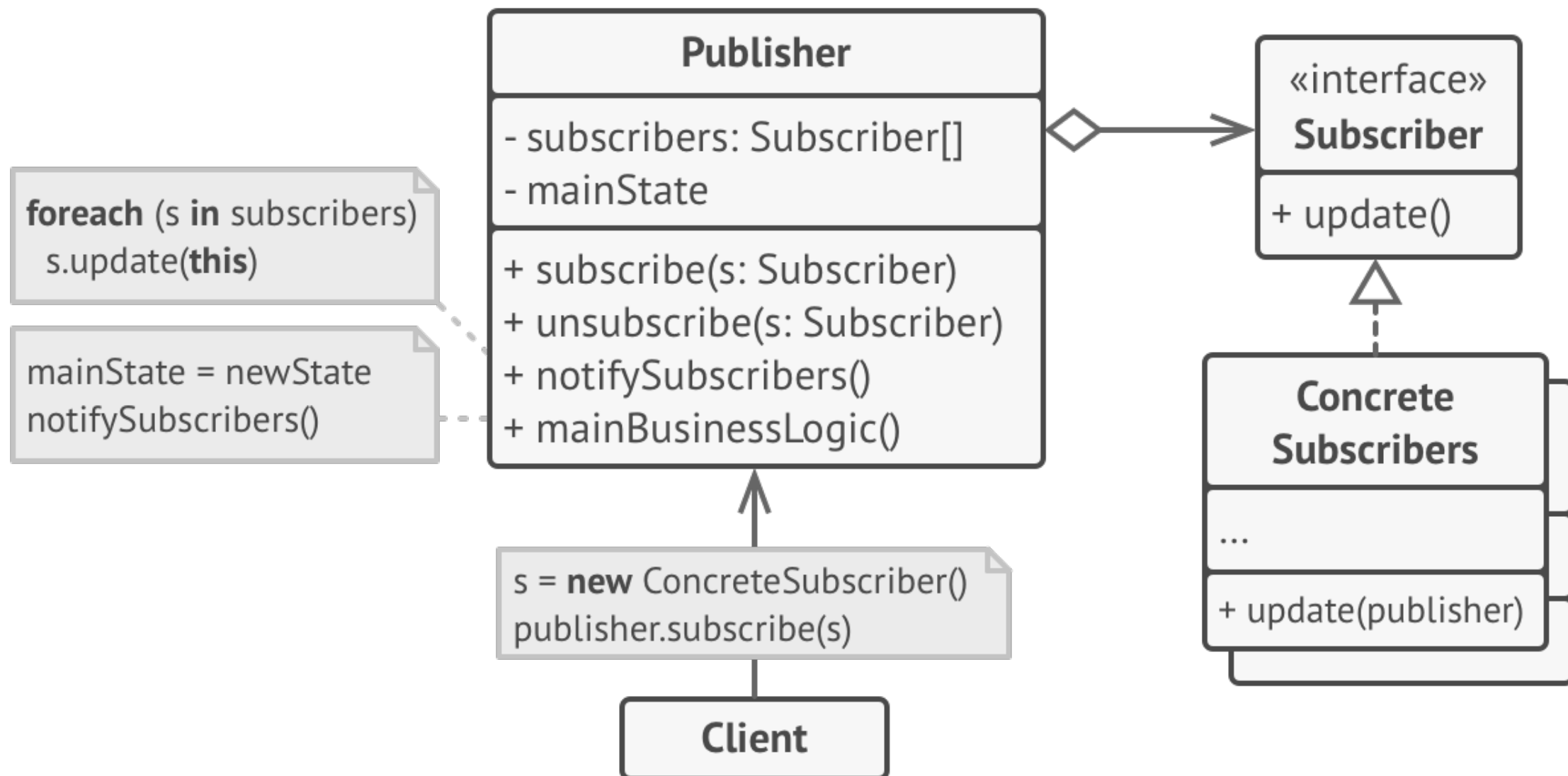
objects that contain an important or interesting data for others

Subscriber

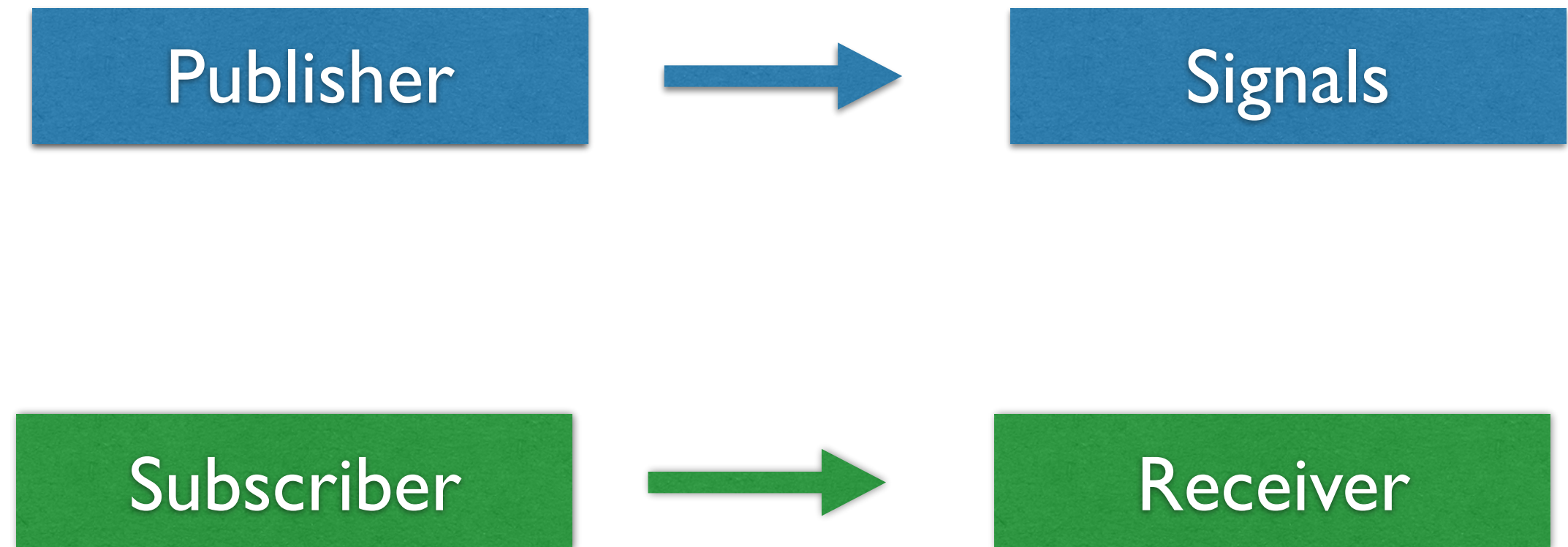
who want to track changes of interesting data







Django Signals



django.db.models.signals

- **pre_save | post_save**
 - Sent before or after a model's **save()** method is called.
- **pre_delete | post_delete**
 - Sent before or after a model's **delete()** method
- **m2m_changed**
 - Sent when a ManyToManyField on a model is changed
- **request_started | request_finished**
 - Sent when Django starts or finishes an HTTP request

<https://docs.djangoproject.com/en/2.2/topics/signals/>

User profile creation

using **Django signals**

```
from django.db.models.signals import post_save
from django.dispatch import receiver
```

```
@receiver(post_save, sender=User)
def user_created(sender, instance, created, **kwargs):
    if created:
        Profile.objects.create(user=instance)
```

Tell Django where the receivers are

- **__init__.py** of application

```
default_app_config = 'users.apps.UserConfig'
```

- **apps.py** of application

```
class UsersConfig(AppConfig):
```

```
    name = 'users'
```

```
    def ready(self):
```

```
        import users.signals
```

User profile creation

overriding **save()** method of User model

```
def save(self, *args, **kwargs):  
    super(User, self).save(*args, **kwargs)  
    Profile.objects.create(user=self)
```

User profile creation

overriding **save()** method of User model

```
def save(self, *args, **kwargs):  
    super(User, self).save(*args, **kwargs)  
    Profile.objects.create(user=self)
```

what is the problem in above code?

Questions?