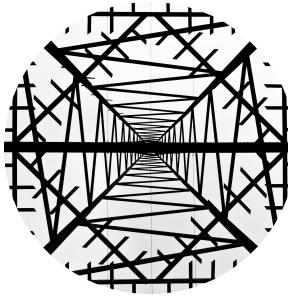


# Les événements – utilisation dans Symfony



Meetup symfony montpellier  
30 janvier 2017 - 1001 Pharmacies

# Bonjour et désolé!



## Julien Vinber

Développeur Application et Web  
pour l'éditeur de logiciel Yooda

# 1

## Programmation séquentielle

“

la programmation séquentielle est un paradigme de programmation dans laquelle le déroulement des instructions du programme est toujours le même (les instructions elles-mêmes peuvent être différentes en fonctions des embranchements,...).

Wikipedia

# Simple ?

$$1+1=3$$



## L'homme

Plus on est libre, plus les  
choix à faire sont complexes

Sauve-moi - Guillaume Musso

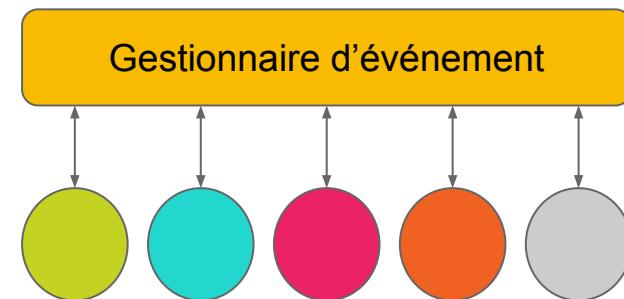
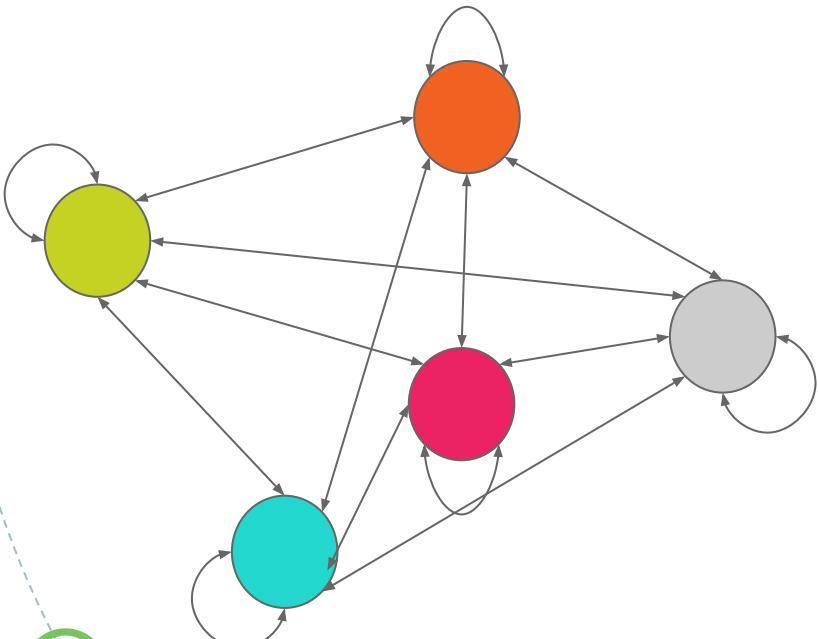
# Quelle solution ?

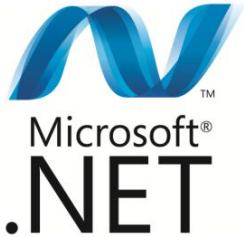
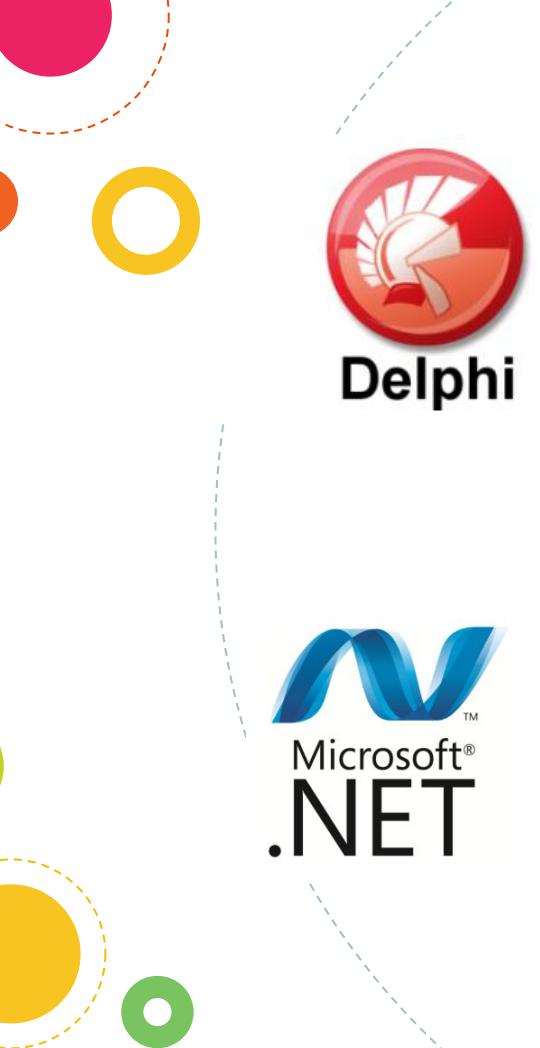


# 2

Programmation  
événementielle

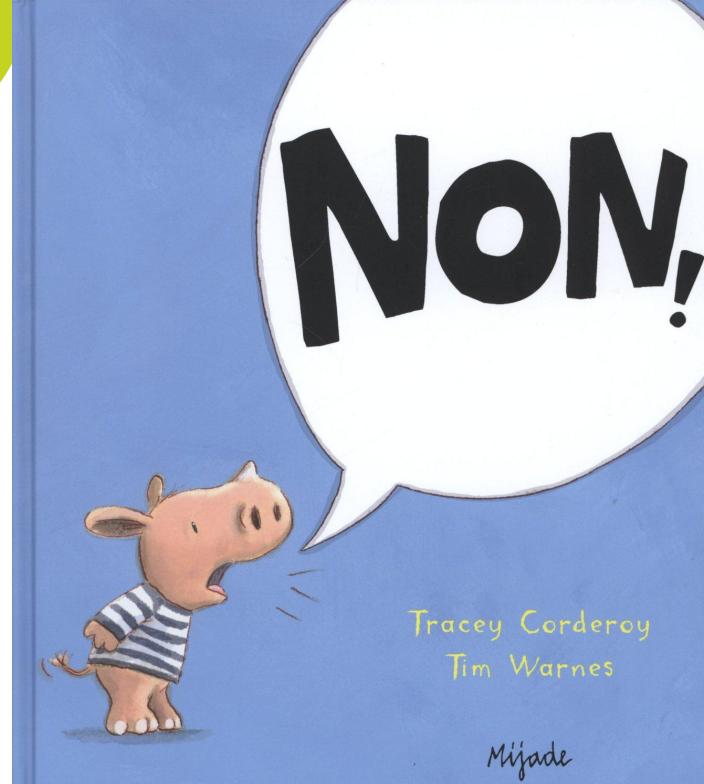
On ne cherche plus à complexifier du code pour gérer les interactions. Au contraire on le découpe en petite action simple et autonome.





# 3

PHP et les événements ?



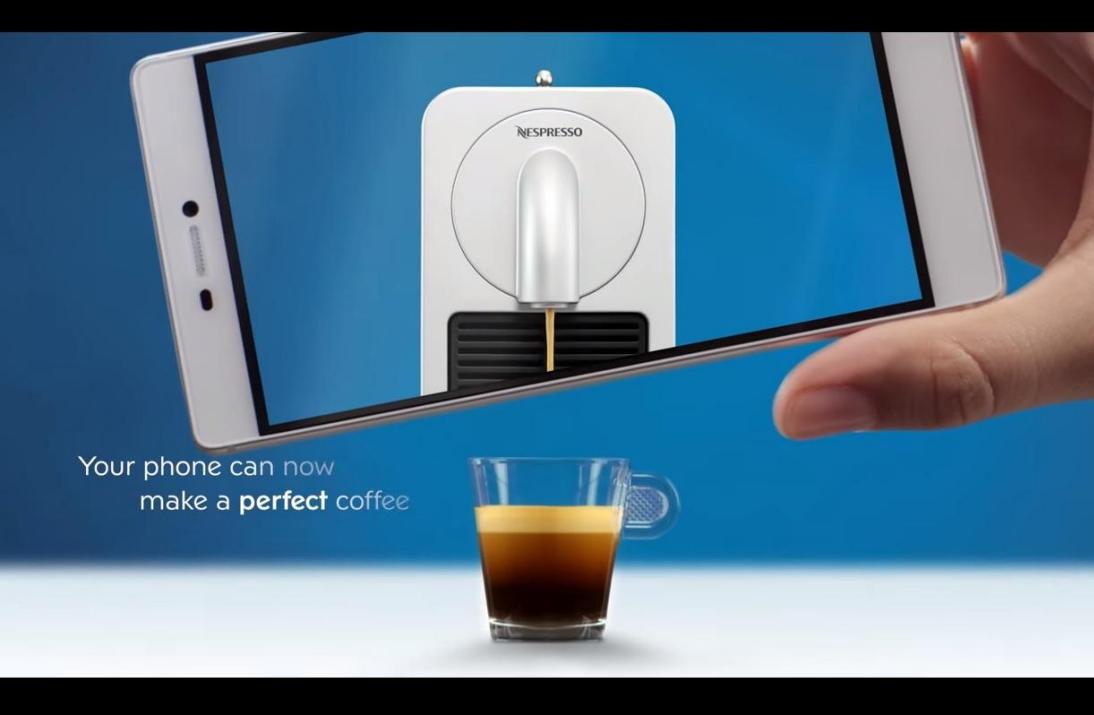


Oui, mais voilà, les applications se complexifient

**Par exemple lors de la création d'un utilisateur on peut vouloir :**

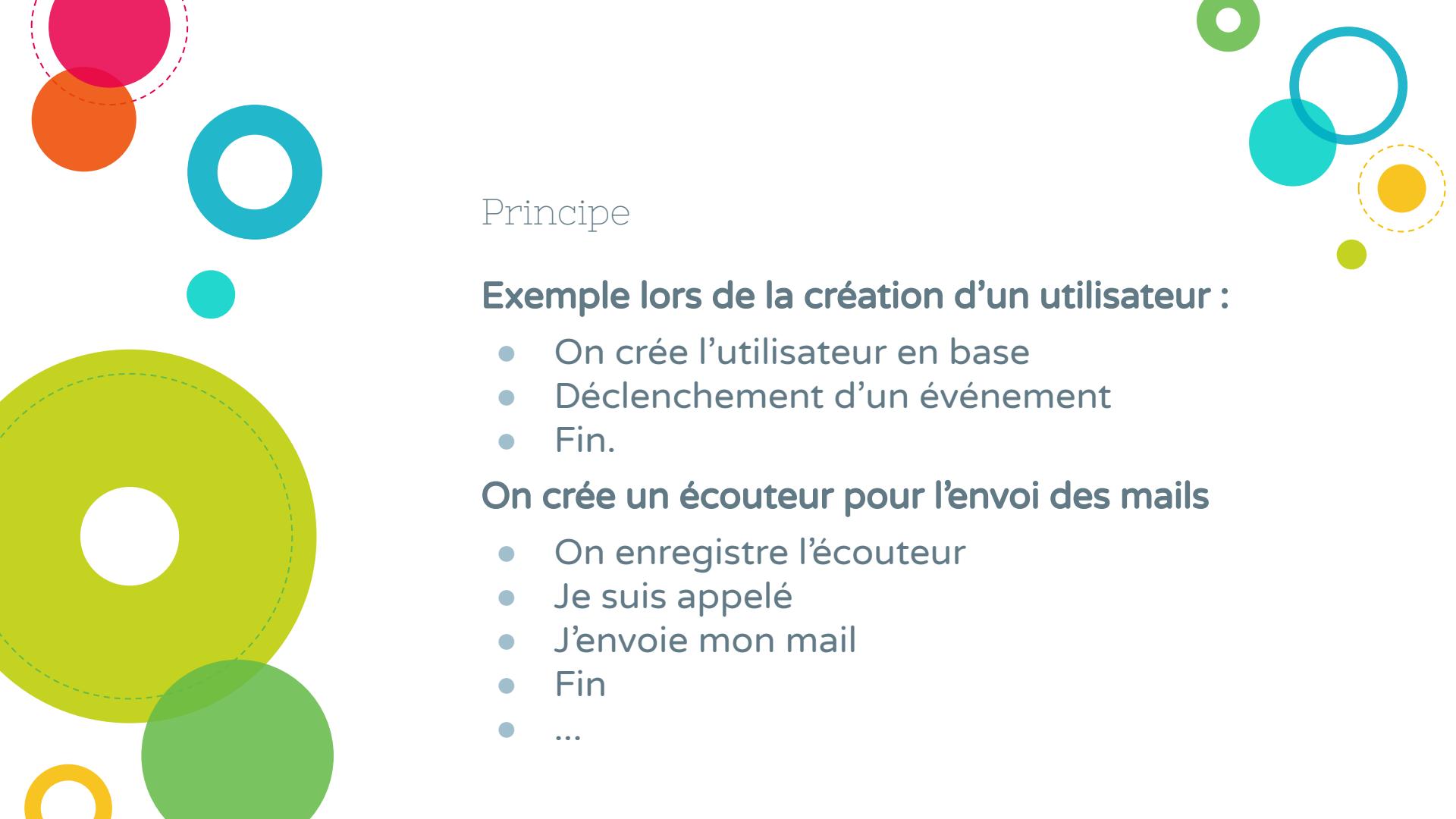
- Envoyer un mail de confirmation
- Créer un utilisateur dans la gestion Co.
- Envoyer une alerte aux commerciales pour contacter le client.
- Lui envoyer dans 7 jours une offre promo.
- Valider son mail.
- Vérifier s'il y a un parrain et lui envoyer une récompense.
- ...

Résultat, un code qui fait tout et n'importe quoi.



# 4

Les événements avec  
Symfony.



## Principe

### Exemple lors de la création d'un utilisateur :

- On crée l'utilisateur en base
- Déclenchement d'un événement
- Fin.

### On crée un écouteur pour l'envoi des mails

- On enregistre l'écouteur
- Je suis appelé
- J'envoie mon mail
- Fin
- ...

### Avantage

- Enlève les dépendances
- Découpage en action simple.
- Test simplifier.
- On revient sur un découpage métier.

### Inconvénient

- Plus de code dans l'absolu.
- Perte de linéarité du code.
- Plus difficile à debugger

### Particularité

N'étant pas géré en natif. Le code reste mono thread et synchrone.

# 5

Exemple de code :

# 5.1

Créer un événement

## Créer l'objet événement

```
class UserEvent extends Event
{
    const EVENT_USERCREATE = 'user.create';
    protected $user;
    public function __construct (User $user)
    {
        $this->user = $user;
    }
    public function getUser ()
    {
        return $this->user;
    }
}
```

## Déclenchement

```
$user = new User();  
$user->setEmail('julien@vinber.fr');  
  
$event = new UserEvent($user);  
  
$this->get('event_dispatcher')->dispatch(UserEvent::EVENT_USERCREATE, $event);
```

# 5.2

Répondre à un événement.



Le code qui est appelé.

```
class EnvoyerMailConfirmation
{
    public function onUserCreate (UserEvent $event)
    {
        $user = $event->getUser ();
        dump ($user);
    }
}
```

## Enregistrement

```
app.envoyermail.user.create:  
    class: App\AppBundle\Event\EnvoyerMailConfirmation  
    tags:  
        - { name: kernel.event_listener, event: user.create }
```

# 6

Événement existant

 **Symfony Profiler**

http://obim.lc/app\_dev.php/  
Method: GET HTTP Status: 200 IP: 127.0.0.1 Profiled on: Mon, 30 Jan 2017 13:11:42 +0100 Token: 00345e

search on symfony.com Search

Last 10 Latest Search

**Event Dispatcher**

Called Listeners 32 Not Called Listeners 28

Priority Listener

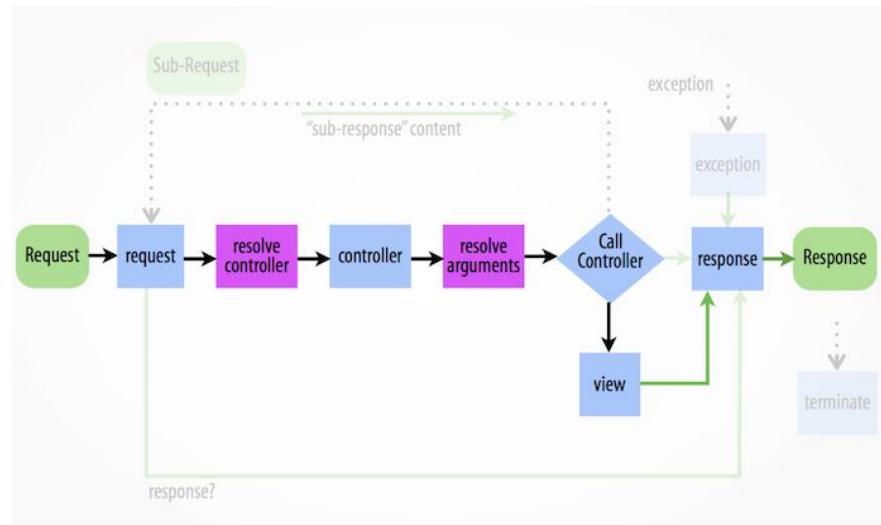
**kernel.request**

2048	"Symfony\Component\HttpKernel\EventListener\DebugHandlersListener::configure()"
256	"Symfony\Component\HttpKernel\EventListener\ValidateRequestListener::onKernelRequest()"
128	"Symfony\Bundle\FrameworkBundle\EventListener\SessionListener::onKernelRequest()"
48	"Symfony\Component\HttpKernel\EventListener\FragmentListener::onKernelRequest()"
32	"Symfony\Component\HttpKernel\EventListener\RouterListener::onKernelRequest()"
16	"Symfony\Component\HttpKernel\EventListener\LocaleListener::onKernelRequest()"
10	"Symfony\Component\HttpKernel\EventListener\TranslatorListener::onKernelRequest()"
8	"Symfony\Component\Security\Http\Firewall::onKernelRequest()"
0	"Knp\Bundle\MenuBundle\EventListener\VoterInitializerListener::onKernelRequest()"
0	"Knp\Bundle\PaginatorBundle\Subscriber\SlidingPaginationSubscriber::onKernelRequest()"
0	"Symfony\Bundle\AsseticBundle\EventListener\RequestListener::onKernelRequest()"

**kernel.controller**

0	"Symfony\Bundle\FrameworkBundle\DataCollector\RouterDataCollector::onKernelController()"
0	"Symfony\Bundle\FrameworkBundle\DataCollector\RequestDataCollector::onKernelController()"

Kernel.request  
Kernel.controller  
Kernel.view  
Kernel.response  
Kernel.terminate  
Kernel.exception



Merci!



Des questions?

julien@vinber.fr