

# Grails Services

Alexis Plantin

# Sommaire

- Pourquoi utiliser les services?
- Création d'un Service
- Squelette d'un service
- Injection de dépendances
- Définir un service
- Utiliser un service

# Pourquoi utiliser les services?

- Pour centraliser la logique métier
- Avoir une gestion des transactions
- Cas d'usage sur plusieurs domain classes
- Fournit une API pour des processus métiers

# Création d'un Service

```
grails create-service wrh.Util
```

- 2 fichiers créés:
  - Un Service dans `grails-app/service`
  - Un Unit Test dans `src/test`

# Squelette d'un service

- Une seule méthode
- Une propriété transactional pour nous assurer que notre code soit transactionnel

```
import grails.transaction.Transactional

@Transactional
class UtilService {

    def serviceMethod() {
    }
}
```

# Injection de dépendances

- Injecter un Service dans un Controller ou un autre Service est simple
  - On a un service UtilService dans `grails-app/service`
  - On déclare simplement un attribut dans l'artefact en respectant la convention de nommage

```
def utilService
```

# Definir un service

- Simplement définir les méthodes dans le service comme dans n'importe quelles autres classes

```
def sayHello(String name) {  
    return "Hello ${name}"  
}
```

# Utiliser un service

- Déclarer l'attribut par convention de nommage dans l'artefact
- L'utiliser pour appeler ses méthodes

```
def utilService

def index = {
    render utilService.sayHello("Paolo & Juergen")
}
```



# Questions?