

Vers un système d'information modulaire dédié à la sélection chez les plantes

Retour d'expérience sur 7 années d'évolutions en développement logiciel

Yannick De Oliveira
GQE-Le Moulon

Journées du PEPI IBIS 6 et 7 juin 2019

Contexte



Environ 70 permanents, 7 équipes de recherche et des structures transversales (Biologie Moléculaire, Informatique, Expérimentation)

Etudie la génétique des caractères à déterminisme complexe.

Deux modèles d'étude importants :



Atelier de Bioinformatique et Informatique

Analyse de données

Administration serveurs et poste de travail

Développement et déploiement de logiciels (DDL)

Un des objectifs actuels : développer un SI modulaire dédié à la sélection chez les plantes

2010

5 permanents
Dont 2 dédiés au DDL

2012

Première pierre de ce SI

2019

6,1 permanents
Dont 2 dédiés au DDL

Thématiques impliquant des besoins en DDL

Optimisation de la sélection assistée par marqueurs (SAM)

Identification de régions et de gènes impliqués dans des caractères d'intérêt (détection de QTLs)

Evaluation de variétés-populations en conditions décentralisées

Etude de la diversité cultivée dans les paysages Français et Européens

Gestion du matériel génétique, description (Variétés, Lots de graines)

Généalogie (Variété, Lots)
Phénotypage brut ou élaboré
Descripteurs (trait, environnement, pratiques culturelles)

Description des paysages

Structure des populations

Génotypage

Génotypage

Aide à la décision

Visualiser et explorer le génome aux positions des QTLs détectés

Evaluer au champ les variétés dans un réseau d'essai

Indicateurs de calcul de la diversité

Applications développées par ABI



Thaliadb 2004 - J. Joets, Y. De Oliveira, D. Steinbach

Une base de données dédiée à la génétique d'association pour les plantes.

matériel génétique

structure

généalogie

génotypage

phénotypage



SHiNeMaS 2011 - Y. De Oliveira

Une base de données dédiée à la traçabilité de lots de semences et l'évaluation au champ de variétés-populations dans un réseau de fermes

matériel génétique

généalogie

phénotypage



OptiMAS 2009 - F. Valente, F. Gauthier

Un logiciel d'aide à la décision dans des études de SAM

Aide à la décision



WheatBiodiversity 2008 - P. Montalent, M. Polart-Donat

Une base de données dédiée à l'étude de la diversité des paysages cultivés en France et en Europe

matériel génétique

génotypage

description des paysages

calcul de la diversité



BioMercator 2004 - J. Joets, O. Sosnowski, Y. De Oliveira

Un logiciel permettant l'identification et l'exploration de régions impliquées dans des caractères d'intérêts ainsi que la méta-analyse de QTLs détectés par linkage ou génétique d'association

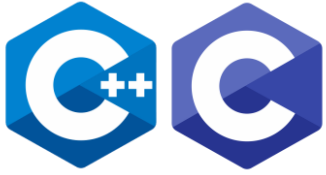
Visualisation/exploration



→ Un certain nombre d'applications gère des informations qui diffèrent légèrement mais malgré tout de nature proche

En 2012, de la diversité mais pas que génétique ...

OptiMAS



WheatBiodiversity



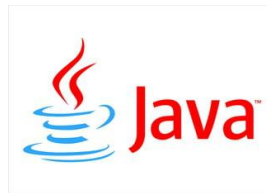
BioMercator



Thaliadb



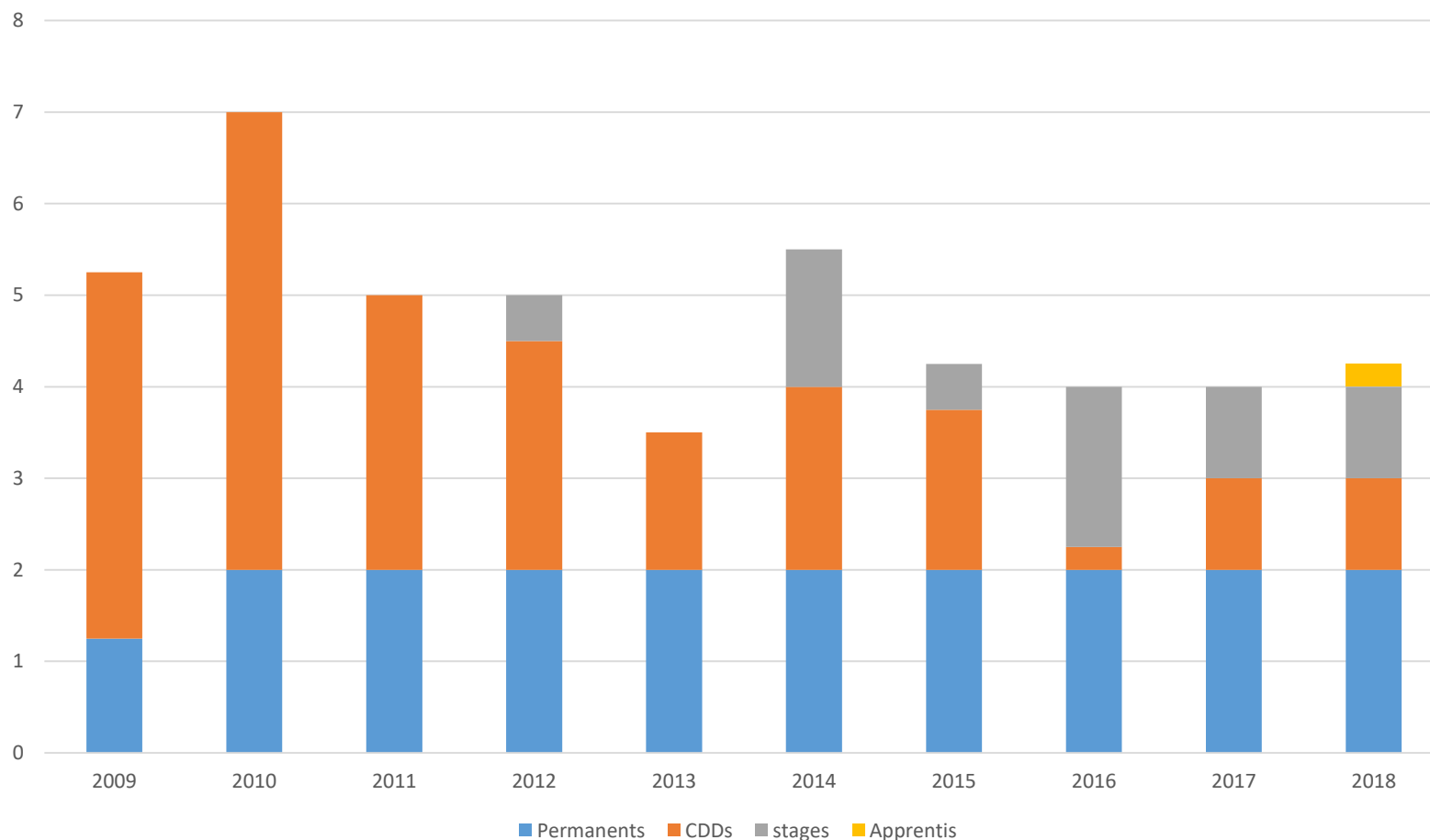
SHiNeMaS



- Beaucoup de connaissances (langages et frameworks) à maintenir dans un environnement comptant deux ingénieurs permanents en ingénierie logicielle
- Redondance partielle de code due à des projets différents dans des technologies différentes
- Moins de souplesse pour transférer un non-permanent d'un projet à un autre

Ressources humaines en DDL

Développement logiciel : répartition des RHs depuis 2009 par type de contrat



4/5 développeurs temps plein en moyenne

Mais ...

Beaucoup plus de temps de stage depuis 2014

- Baisse de la pérennité des compétences dans l'équipe voire perte (C/C++)
- Discontinuité dans la présence des non-permanents
- Travail de formation à refaire chaque année

Uniformiser les technologies utilisées

Migration de  **Thaliadb**

~~~~ ~~~~ ~~~~  

SI devenu difficilement maintenable
voire presque impossible à faire évoluer
(gestion des données à haut débit)







Développement de  **SHiNeMaS**

Suite logique de l'évolution des activités
de développement de l'équipe
Développements relancés sous
l'impulsion du projet européen H2020



Migration de  **WheatBiodiversity**

~~~~  

Pourquoi Python et Django ?



polyvalent



documentation



apprentissage



librairies



communauté



syntaxe simple et lisible



full-stack



validation



Stagiaire M2 rapidement opérationnel

DjangoCon Europe et US, DjangoGirls, Django-fr

1 framework pour développer

Django, comment ça marche ?

- Framework écrit en python pour le développement web (v0.9 en 2005)
- Architecture MVT (inspirée de MVC)

Model (*models.py*)

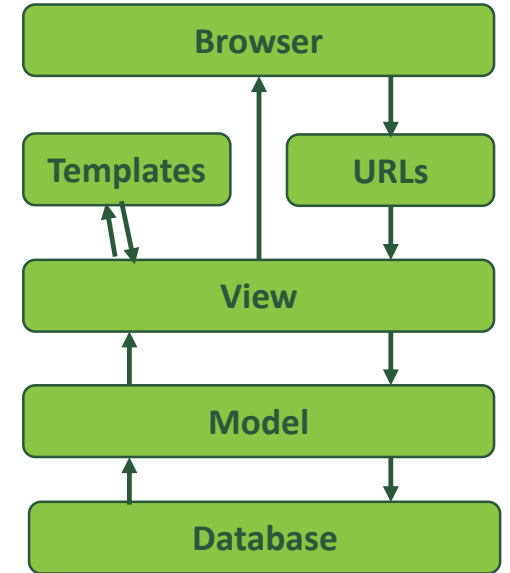
```
class SeedLot(Model) :  
    name = CharField(max_length=100)  
    date = IntegerField()  
    variety = ForeignKey(Variety)  
    location = ForeignKey(Location)  
    initial_quantity = IntegerField(null=True)
```

View (*views.py*)

```
def get_all_seedlots(request) :  
    seedlots_queryset = SeedLot.objects.all()  
    return render(request, 'mytemplate.html',  
                  {'allseedlots': seedlots_queryset})
```

Templates (*mytemplate.html*)

```
<html>  
{%for seedlot in allseedlots %}  
    {{seedlot.name}}  
{%endfor%}  
</html>
```



Un peu plus sur Django ...

Model



- Le relationnel est géré nativement par Django, mais quid du non relationnel ?
- 👎 : des projets existent mais non supportés officiellement
- 👍 : ces projets *paraissent* stables et sont souvent maintenus par des « core developpeurs » du framework



- Migration automatiques (depuis django 1.7)
- 👍 : Django créé un script python de mise à jour de la base de données
- 👎 : Nécessite d'être très rigoureux dans la gestion de ces fichiers en particulier avec Git/Svn

View

- Les vues peuvent être des classes ou des fonctions, Django fournit des vues prédéfinies (pour lister l'ensemble des objets d'un modèle par exemple)

... et pour en finir avec Django

Templates

- 👉 : le moteur par défaut de Django est peu performant
- 👉 : pas d'intégration AJAX, choix délibéré de la DSF
- 👉 : parfaitement intégré avec le framework, appel à l'API des modèles
- Jinja2 a été intégré depuis la version 1.7, plus performant
- Possible d'utiliser des moteurs non intégrés (cheetah), mais probablement perte de certaines fonctionnalités

Forms

- Bibliothèque de classes permettant de créer et traiter les formulaires de l'application
- 👉 : Django par son système de formulaire offre des systèmes de validation des données
- 👉 : Il est possible de personnaliser facilement la validation des données au niveau du formulaire ou du modèle

Admin site

- Django offre des bibliothèques permettant de créer une interface de gestion des données
- 👉 Simplissime (essentiellement de la configuration), beaucoup d'options qui peuvent être activées/désactivées

Quid des applications standalone ?

- Pas de nécessité de réécrire ces applications



BioMercator

- Nécessiterait un framework GUI pour python

tkinter

pyqt

kivy

...

→ Probablement aucun bénéfice à migrer

OptiMAS

- Perte de la compétence C/C++, développements en pause

→ Pérenniser cette compétence avec (on espère) un poste en mobilité

En 2019, Un peu moins de diversité!

OptiMAS



Thaliadb



SHiNeMaS

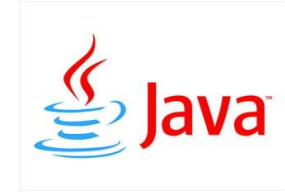


WheatBiodiversity



BioMercator

django



- 😊 Amélioration de la pérennité des compétences dans l'équipe
- 😊 Création d'applications Django modulaires et *transférables*
- 😊 Meilleur suivi des évolutions (langage/framework)

Cependant ...

- 😞 Transférable, mais n'empêche pas les divergences par la suite voire de nouvelles redondances
→ Comment éviter ces divergences/redondances ?
- 😞 La qualité du code source reste améliorable
→ Comment progresser dans la qualité du code ?

Poser les bases d'un SI modulaire

Objectif : créer un SI modulaire répondant aux besoins des différentes thématiques de recherche tout en évitant la redondance de code → Mutualiser Thaliadb, SHiNeMaS, WheatBioDiversity

“An app is a Web application that does something – e.g., a Weblog system, a database of public records or a simple poll app. A project is a collection of configuration and apps for a particular website. A project can contain multiple apps. An app can be in multiple projects.”

Documentation Django

« Django encourage multiple apps. Don't be afraid of multiple apps! »

Flávio Juvenal, DjangoCon Europe 2017

Apps

matériel génétique

structure

génotypage

phénotypage

description des paysages

généalogie

Projects



Thaliadb



SHiNeMaS

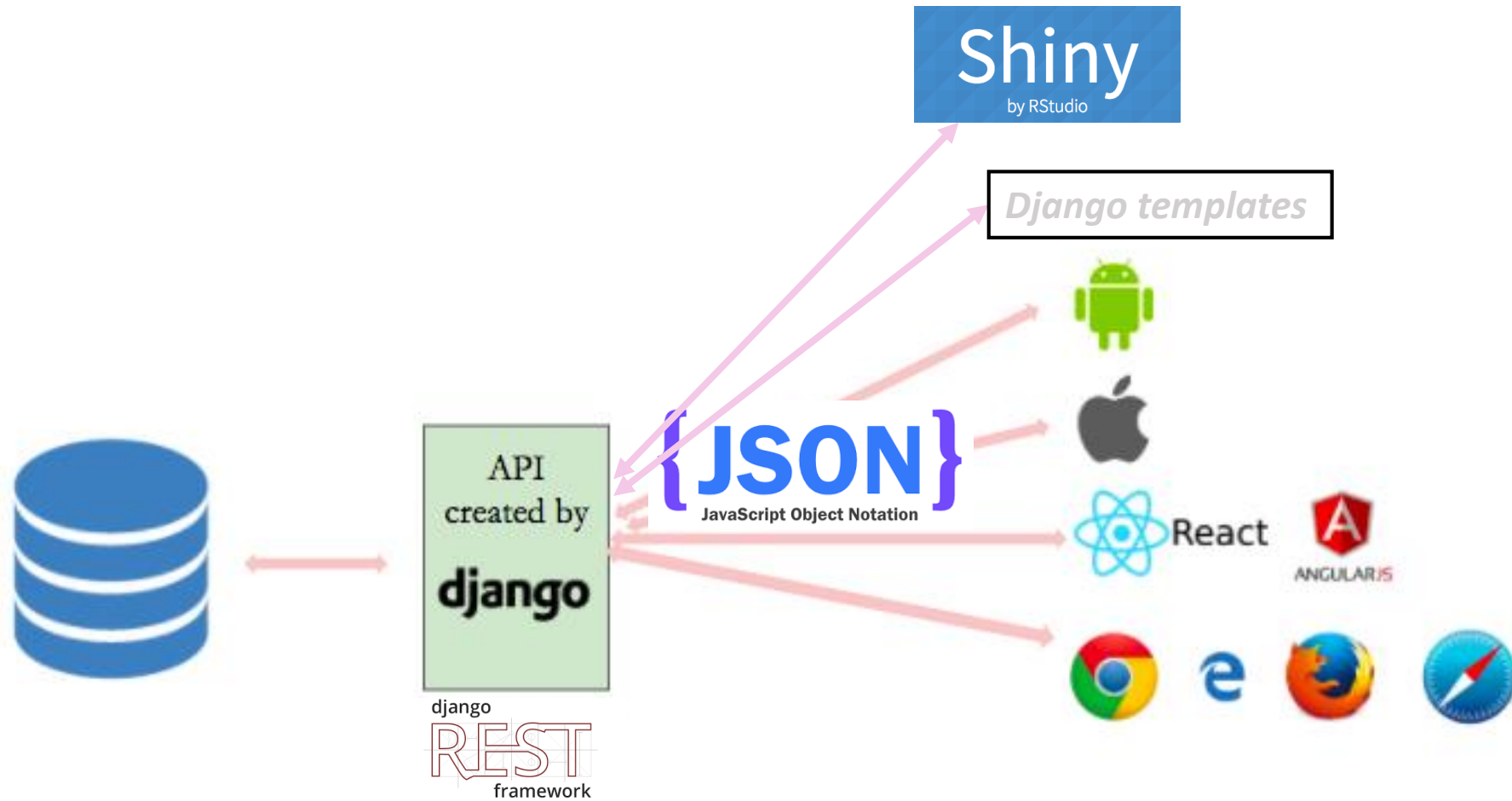


WheatBiodiversity

→ Effort important de conception sur le modèle de données et l'architecture logicielle

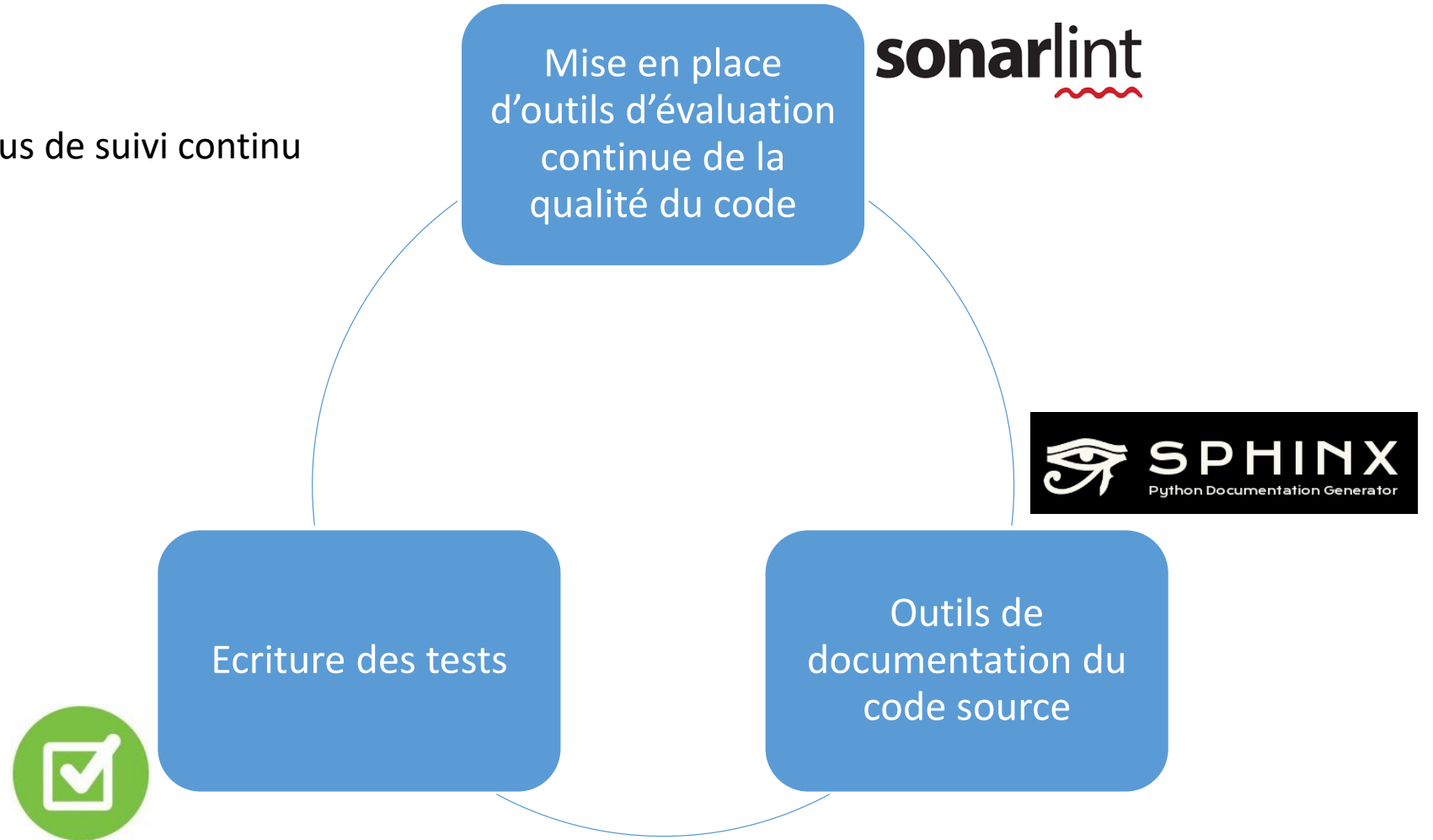
→ Contrat d'Apprentissage dédié 2018-2020

Une architecture en micro services



Amélioration de la Qualité

Objectif : Mettre en place un processus de suivi continu de qualité du code.



Conclusion

Uniformiser les technologies (Web) utilisées dans l'équipe

- ➔ Création d'applications Django modulaires et *transférables*
- ➔ Meilleur suivi des évolutions (langage/framework)
- ➔ Capitaliser sur le temps passé à former

Modularité du système d'information & amélioration du processus de suivi qualité

- ➔ Partager du code source entre différents projets
- ➔ Passer moins de temps à la reprise du code
- ➔ Gagner en flexibilité sur la couche client