**MVC – REFACTORING MEMO**

« Garder propre chez soi est une rigueur de tous les instants » (*Fabien Potencier*)

**M**odel

1. Doctrine

* Ne pas toucher aux classes doctrine/base
* Implémenter dans modelTable ou dans model(Record) selon le besoin
* => si ambiguïté, plusieurs table en relation :

=>Mettre dans la table parent/pivot

* Méthode statiques, constantes de classe, test createQuery

1. Web services (Lavoisier, GGUS, …)

* Classe & fichier ont le même nom
* Préfixé par id du web service et suffixé éventuellement par View

Ex :

* + ‘lv\_nagiosNotification’ (une NN issue de Lavoisier)
  + ‘gg\_ticketView’ (Une collection de tickets GGUS)
* Stocker dans /lib/model/nomWebService

**View (templates, partials)**

* + Minimiser au maximum la présence de code PHP
  + Préférer

<?php echo ‘<p>’.$title.’</p>’ ?>

à

<p><?php echo $title ?></p>

**C**ontroller (Action) :

- Met en relation le modèle et la vue

- Ne connaît pas les règles métiers

- N’est pas intelligent

- 15 lignes maximum

Nomenclature :

* Tout en camelCase excepté :
  + les index de tableau associatif : xxx\_xxx
  + les constantes de class MY\_CONST
  + les noms des widget et id : xxx\_xxx
  + les colonnes des tables my sql : xxx\_xxx