Documentation technique ‘Modèle de classe et de base de données’

# Introduction

### Une approche ORM

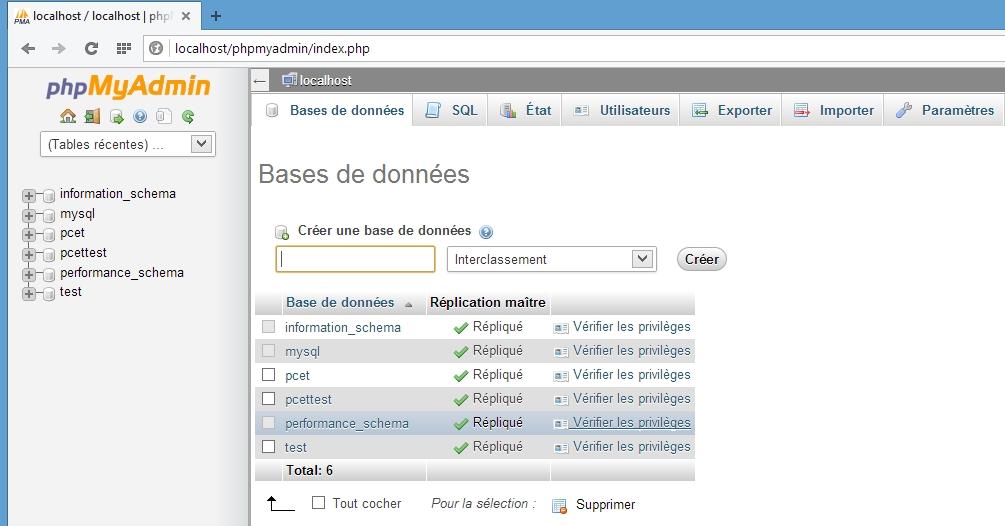
Le choix architectural, vis-à-vis de la gestion des données, a été une approche ORM (Object Relational Mapper).

L’outil utilisé est RedbeanPHP (<http://redbeanphp.com/>). Concrètement, ce choix offre une indépendance vis-à-vis d’un moteur de base de données et permet une approche orienté objet. Ainsi nous n’avons pas créer, à proprement parlé, de scripts de génération de base de données. Nous avons utilisé les objets Redbean pour créer les classes, avec leurs attributs et méthodes, et les relations entre elles.

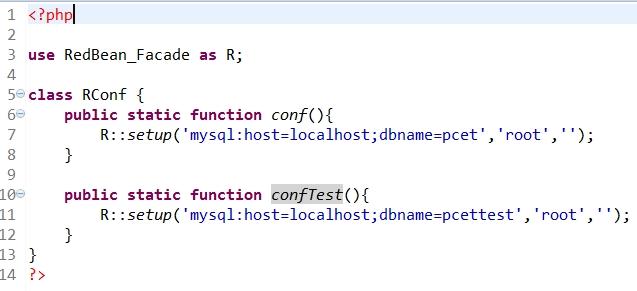
Au final, nous avons spécifié à RedbeanPHP, le support physique sur lequel nous souhaitons offrir la persistance des objets. Le choix s’est fait sur MySQL. Cette option peut être modifiée pour Oracle, SQL Server, des fichiers textes, etc. sans aucune modification du code.

### Installation et création de la base de données

Dans l’outil d'administration de PhpMyAdmin, disponible dans Wamp server, il faut créer les 2 bases de données (pcet et pcettest), au travers de l’interface suivante :



Ensuite, du côté PHP, il faut déclarer les chaînes de connexion à la base de données, en utilisant les librairies Redbean et la méthode ‘setup’ :

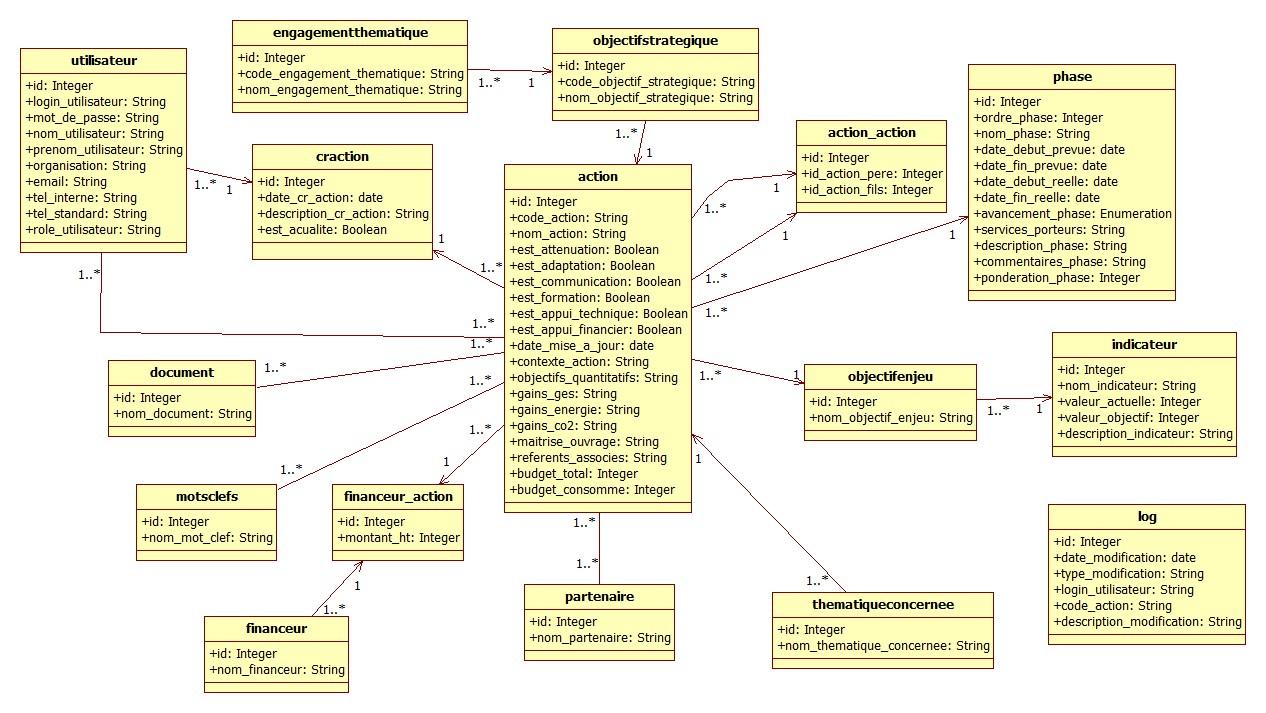


Finalement, la connexion à la base de données, est effectuée au travers de la méthode Conf() ou ConfTest() de la classe RConf.

Dans le chapitre consacré à Redbean, nous présenterons les fonctions utilisées dans le projet pour créer les objets et les relations.

# Description du modèle objet / base de données

## Schéma graphique des classes, des attributs et de leurs relations



Chacune de ces classes est définie au travers d’un fichier PHP. Les classes possèdent également des méthodes basics pour l’ajout, la suppression, la mise à jour et la recherche d’objets.

## Description détaillée des classes / tables

### Liste des classes / tables

|  |  |  |
| --- | --- | --- |
| Nom de la classe | Description | Commentaires |
| action | Les actions du PCAET |  |
| actionaction | Lien entre les actions | Interdépendance/Lien avec d’autres actions |
| craction | Compte rendu des actions | Cette table sert à gérer les commentaires / compte rendu des responsables actions sur les actions. |
| document | Liste des documents | Un lien entre les actions et les documents permet de lier des documents à une action |
| engagementthematique | Engagement thématique | Les engagements thématiques sont le plus haut niveau de catégorisation des actions. Ils englobent les objectifs stratégiques. |
| financeur | Financeur |  |
| financeur\_action | Montant versé par le financeur | Ce montant est par action et par financeur. |
| indicateur | Indicateurs | Les indicateurs sont regroupés par objectifs enjeux |
| log | Journal des modifications | Liste des toutes les modifications apportées au système |
| motsclefs | Mots clefs | Liste des mots clefs qui peuvent être affectés aux actions |
| objectifenjeu | Objectif enjeu | Liste des objectifs de suivi de la mise en place de l’action. Les objectifs sont divisés ensuite en indicateurs. |
| objectifstrategique | Objectifs stratégiques | Second niveau de catégorisation des actions après l’engagement thématique. Une action est définie selon un objectif stratégique, qui lui même est inclus dans un engagement thématique |
| partenaire | Partenaires | Liste des partenaires affectés à une action |
| phase | Phases de l’action | Liste des phases de l’action |
| thematiqueconcernee | Thématique | Liste des thèmes touchés par l’action |
| utilisateur | Utilisateurs | Ensemble des utilisateurs du système. La relation entre les utilisateurs et l’action permet de définir les responsables de l’action |

### Table action

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| Integer | budget\_consomme | Budget du consommé, renseigné par le responsable action et permettant le suivi du budget |  |
| Integer | budget\_total |  | Mise en œuvre de l'action / Coût prévisionnel de l'action |
| String | code\_action | Code l'action tel que listé dans le PCAET |  |
| String | contexte\_action |  | Présentation de l'action / Contexte de l'action |
| DateTime | date\_mise\_a\_jour | Date de la dernière mise à jour de l'action | Entête de l'action |
| Boolean | est\_adaptation |  | Entête de l'action / Objectifs PCET |
| Boolean | est\_appui\_financier |  | Présentation de l'action / Type d'action |
| Boolean | est\_appui\_technique |  | Présentation de l'action / Type d'action |
| Boolean | est\_attenuation |  | Entête de l'action / Objectifs PCET |
| Boolean | est\_communication |  | Présentation de l'action / Type d'action |
| Boolean | est\_formation |  | Présentation de l'action / Type d'action |
| String | gains\_co2 |  | Présentation de l'action / Objectif quantitatif de l'action |
| String | gains\_energie |  | Présentation de l'action / Objectif quantitatif de l'action |
| String | gains\_ges |  | Présentation de l'action / Objectif quantitatif de l'action |
| String | maitrise\_ouvrage |  | Pilotage / Portage |
| String | nom\_action |  | Entête de l'action |
| String | objectifs\_quantitatifs |  | Présentation de l'action / Objectif quantitatif de l'action |
| String | referents\_associes |  | Pilotage / Directions et/ou agents référents associés |

**Table actionaction**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| Integer | id\_action\_fils | Identifiant de l'action : ce lien est bidirectionnel. | Présentation de l'action / Interdépendance |
| Integer | id\_action\_pere | Identifiant de l'action : ce lien est bidirectionnel. | Présentation de l'action / Interdépendance |

**Table crAction**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| DateTime | date\_cr\_action | Date de création du compte rendu de l'action |  |
| String | description\_cr\_action | Description du compte rendu |  |
| Boolean | est\_acualite | Booléen permettant d'inclure le compte rendu dans les actualités |  |

**Table engagementThematique**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| String | code\_engagement\_thematique | Code l'engagement tel que listé dans le PCAET |  |
| String | nom\_engagement\_thematique | Nom de l'engagement thématique |  |

**Table financeur**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| String | nom\_financeur |  | Mise en œuvre de l'action / Coût prévisionnel de l'action |

**Table financeuraction**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| Integer | montant\_ht |  | Mise en œuvre de l'action / Coût prévisionnel de l'action |

**Table indicateur**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| String | description\_indicateur |  | Suivi - évaluation de l'action / Objectifs et indicateurs de suivi-évaluation |
| String | nom\_indicateur |  | Suivi - évaluation de l'action / Objectifs et indicateurs de suivi-évaluation |
| Integer | valeur\_actuelle | Valueur renseignée par le responsable action |  |
| Integer | valeur\_objectif | Valueur renseignée par le responsable action |  |

**Table log**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| String | code\_action | Code alphanumérique de l'action |  |
| DateTime | date\_modification |  |  |
| String | description\_modification | Indication sur les valeurs ajoutées, supprimées, mises à jour |  |
| String | login\_utilisateur |  |  |
| String | type\_modification | Indication si il s'agit d'un ajout, suppression ou d'une mise à jour |  |

**Table motclef**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| String | nom\_mot\_clef | Valeur renseignée par le responsable action |  |

**Table objectifenjeu**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| String | nom\_objectif\_enjeu |  | Suivi - évaluation de l'action / Objectifs et indicateurs de suivi-évaluation |

**Table objectifstratégique**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| String | code\_objectif\_strategique | Code alphanumérique de l'objectif tel que listé dans le PCAET |  |
| String | nom\_objectif\_strategique |  | Entête / Objectif stratégique |

**Table partenaire**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| String | nom\_partenaire |  | Pilotage / Portage |

**Table phase**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| Enumeration | avancement\_phase | Réalisé, en cours, en projet, reporté/suspendu | Mise en œuvre de l'action / Phasage |
| String | commentaires\_phase |  | Mise en œuvre de l'action / Descriptif des phases de l'action |
| DateTime | date\_debut\_prevue |  | Mise en œuvre de l'action / Phasage |
| DateTime | date\_debut\_reelle | Valeur renseignée par le responsable action |  |
| DateTime | date\_fin\_prevue |  | Mise en œuvre de l'action / Phasage |
| DateTime | date\_fin\_reelle | Valeur renseignée par le responsable action |  |
| String | description\_phase |  | Mise en œuvre de l'action / Phasage |
| String | nom\_phase |  | Mise en œuvre de l'action / Phasage |
| Integer | ordre\_phase | Valeur calculée automatiquement selon l'ordre souhaité |  |
| Integer | ponderation\_phase | Valeur renseignée par le responsable action pour différencier les phases importantes pour le reporting. |  |
| String | services\_porteurs |  | Mise en œuvre de l'action / Phasage |

**Table thematiqueconcerne**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| String | nom\_thematique\_concernee |  | Présentation de l'action / Type action |

**Table utilisateur**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| String | email |  |  |
| String | login\_utilisateur |  |  |
| String | mot\_de\_passe |  |  |
| String | nom\_utilisateur |  |  |
| String | organisation |  |  |
| String | prenom\_utilisateur |  |  |
| String | role\_utilisateur |  |  |
| String | tel\_interne |  |  |
| String | tel\_standard |  |  |

**Table document**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Nom | Commentaire | Lien avec la fiche action (section / catégorie) |
| Integer | id | Identifiant technique unique |  |
| String | nom\_document | Chemin complet du document |  |

# Utilisation de Redbean

## Introduction

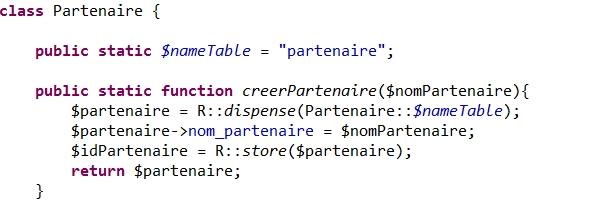
Redbean ne nécessite pas de configuration particulière, il est basé sur des stricts conventions de nommage. C’est pourquoi, il est essentiel de respecter les règles de nommage qui sont explicitées dans les prochains chapitres.

## Méthode de base

### Création et ajout/modification avec les méthodes ‘Dispense’ et ‘Store’

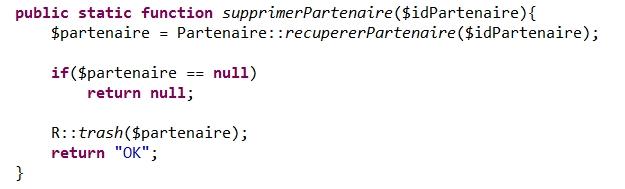
La méthode ‘Dispense’ permet d’instancier un nouvel objet Redbean. A cet objet, il faudra lui spécifier les attributs utilisés. Finalement, la méthode ‘Store’ permet de stocker l’enregistrement dans le base de données, si la table n’existe pas, elle sera alors créée. Cette méthode sert également à l’ajout et la mise à jour d’information.

Voici un exemple de création d’un objet ‘Partenaire’, de l’assignation d’une variable à l’attribut ‘nom\_partenaire’, et de la sauvegarde de l’enregistrement :



### Suppression

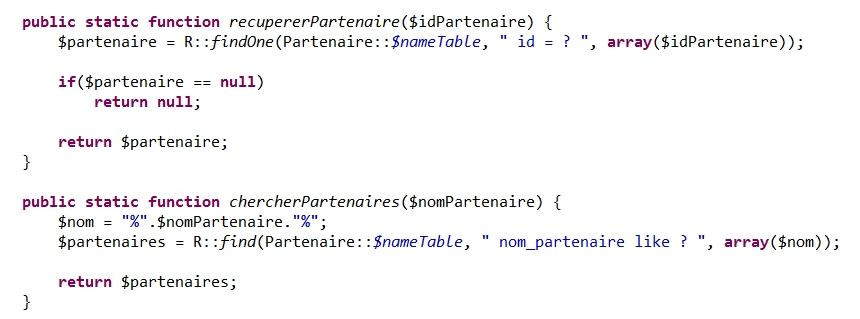
Pour supprimer une enregistrement, il faut utiliser la méthode ‘Trash’, avec l’objet à supprimer :



### Recherche d’enregistrement avec les méthodes ‘FindOne’, ‘Find’ et ‘FindAll’

Pour rechercher des enregistrements, il faut utiliser une des méthodes suivantes : ‘FindOne’, ‘Find’ ou ‘FindAll’ de l’objet Redbean.

Voici 2 exemples, la méthode ‘FindOne’ renvoie un enregistrement et la méthode ‘Find’ peut en renvoyer plusieurs :



La méthode ‘FindAll’ renvoie tous les enregistrements d’une table.

### Règles de nommage

Afin d’éviter tout problème, il est important :

* de nommer les tables en minuscule, sans espaces, sans underscore :
  + par exemple : ‘*engagementthematique’*
* de nommer les champs en minuscule et de les séparer avec un underscore :
  + par exemple : ‘*nom\_partenaire’*

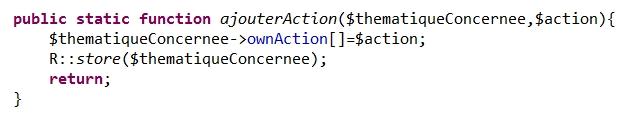
## Lien entre les classes

Pour réaliser des liens entre les classes, il faut utiliser des listes. Ces listes permettent, notamment, de modéliser des liens “1 à plusieurs” et “plusieurs à plusieurs”.

### Mise en place des relations “1 à plusieurs”

Les relations “1 à plusieurs” sont exprimées avec le mot clef “**own***Nomobjet*”. Cette relation est bidrectionnelle. C’est à dire qu’en spécifiant la relation d’un objet A vers un objet B, le lien inverse de l’objet B vers A est automatiquement spécifé.

Voici un exemple avec le lien entre la ‘thématique concernée’ et ’action’. Dans notre cas, chaque action est reliée à une thématique concernée. Nous avons donc créé dans la classe ‘thématique concernée’ une liste servant à contenir toutes les actions attachées :



La mise en place de cette relation permet depuis la classe action, de pointer sur l’objet thématique concernée associé. Dans l’exemple précédent, depuis un objet ‘action’, la thématique concernée est accessible de cette manière : *$action->thematiqueconcernee*.

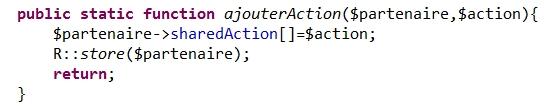
A nouveau, les règles de nommage doivent être respectées :

* la relation ‘own’ doit être suivie par le nom de l’objet avec la première lettre en majuscule et le reste en minuscule : monpremier*objet.ownSecondobjet[]*
* de même, l’objet relié est atteint en spécifiant le nom de l’autre objet en minuscule : *nomsecondobjet->monpremieobjet*

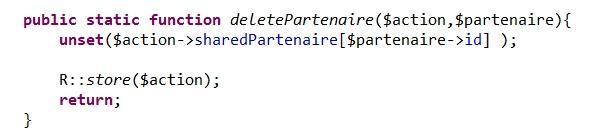
### Mise en place de relations “plusieurs à plusieurs”

Les relations “plusieurs à plusieurs” son exprimées avec le mot clef “**shared***Nomobjet*”. Cette relation est bidrectionnelle. C’est à dire qu’en spécifiant la relation d’un objetA vers un objet B, le lien inverse de l’objet B vers A est spécifié.

Voici un exemple entre la classe ‘partenaire’ et la classe ‘action’. Une action comportent plusieurs partenaires, et un partenaire est impliqué dans plusieurs actions.



De la même manière que pour la relation ‘own’, la liste partagée est aussi créée et accessible depuis ‘action’. Voici, la manière de supprimer la relation entre un ‘partenaire’ et une ‘action’ en utilisation la méthode ‘unset’ :



Les règles de nommage précédemment citée pour le relation ‘own’, s’applique également pour la relation ‘shared’.

### Cas spéciaux

Entre certaines classes, des classes associatives ont été mises en place. C’est le cas entre une action et un financeur. Cela s’explique par le fait que le montant d’un financeur est différent par action qu’il finance.

Finalement, une classe associative a été mise en place sur la classe action, pour lier les actions entre elles. C’est la classe ‘actionaction’. Ni la méthode ‘own’, ni la méthode ‘shared’ n’ont permis d’exprimer cette relation réflexive de façon satisfaisante. C’est pourquoi une classe dédiée et “détachée” (sans lien) a été mise en place. Les méthodes d’ajout, de suppression et de recherche ont dû être adaptées pour gérer cette relation. Cette relation se veut bidirectionnelle, ainsi lorsque qu’une action père est liée à une action fils, cette relation est vue à la fois sur la fiche action père et sur la fiche action fils.