# **Conception de l'architecture SOA**

L'architecture applicative MVC (Modele-Vue-Controller) retenue est la suivante :

* **Couche apptechnic** :

Contient tous les éléments techniques indispensables à toute l'application (chemin vers les properties, logos, images, chemins vers la base de données, ...).

* **Couche Modèle** :

Contient 3 sous-couches *bien découplées* les unes des autres :

*La couche Intégration (DAO)* :

1. Couche chargée de la **relation avec la base de données** (persistance aussi appelée intégration).
2. Cette couche DAO ne peut être *interrogée que par la couche SERVICES*.
3. Cette couche traite des objets de la couche métier en les mettant en base, les recherchant, les supprimant en base, ...(CRUD).

Par exemple, création d'un Elève en base, destruction d'un Elève en base, ...

*La couche métier* :

1. Couche contenant les **objets métier** de l'application (Compte en Banque, Elève, Transport exceptionnel, Section de trafic, ...).
2. Cette couche contient les "Java Beans" ou Pojo (**P**lain **O**ld **J**ava **O**bject), ou encore DTO (**D**ata **T**ransfer **O**bject) qui modélisent le métier de l'application.

Cette couche est en réalité *transverse à toute l'application*. N'importe quelle autre couche peut avoir besoin d'instancier un Java Bean pour qu'il *contienne* les données provenant d'un formulaire par exemple.

Mais les Java Beans doivent être **réutilisables** dans diverses applications *sans aucune modification*. Ils doivent donc jouer leur rôle de conteneur et d'entité persistante en cas de mise en base.

Ils ne doivent rien embarquer d'autre que leur propre logique (attributs, equals(), hashcode(), toString(), comparableTo(...), ...).

*La couche Services* :

1. Couche chargée de l'implémentation de tous les **cas d'utilisation (CU)** de l'application.
2. Cette couche SERVICES ne peut être interrogée que par la couche CONTROLLERS.
3. Cette couche appelle la couche métier pour instancier des Java Beans.
4. Cette couche appelle la couche DAO pour lui ordonner la mise en base, la destruction en base, ...

Cette couche fait le lien entre la couche Controllers et les couches métier/DAO.

La couche SERVICES :

* valide les données transmises par le CONTROLLER (données provenant elles-même d'une Vue),
* éventuellement les transforme,
* décide si il est possible de continuer ou pas en fonction des erreurs détectées lors de la validation/transformation,
* interroge la couche METIER.
* fournit à la couche CONTROLLER le verdict de sa validation/transformation ainsi que les éléments nécessaires pour la Vue à afficher.
* **Couche Vues** :

Contient les jsp en technologie web ou les vues Swing, Awt, JavaFx, ... en technologie "standalone" (aussi appelée Desktop).

La couche Vues ne connait que la couche CONTROLLERS.

* **Couche Controllers** :

AIGUILLE vers la VUE souhaitée en fonction de la réponse du SERVICE.

La couche CONTROLLERS :

* Contient les Servlets en technologies web ou les Listeners en technologie "standalone' (aussi appelée Desktop).
* Intercepte les données provenant de la couche Vues.
* Délègue à un SERVICE le soin d'interroger le métier pour savoir ce qu'il convient de transmettre à une Vue.
* Reçoit d'un SERVICE les éléments à afficher dans une VUE (ou les erreurs qui lui permettent de décider quel message envoyer à la VUE).



Diagramme N° 1 : Conception de l'architecture SOA