

C.2 Déterminer une architecture logicielle

Objectif

Mettre en oeuvre une architecture logicielle en couches.

On fait le choix d'une architecture très classique en 3 couches (pattern Architectural Layer).

Préambule

L'architecture logicielle décrit d'une manière symbolique et schématique les différents éléments d'un ou de plusieurs systèmes informatiques, leurs relations et leurs interactions.

Styles d'architecture

L'architecture logicielle peut se catégoriser en styles. En effet, malgré les millions de systèmes informatiques construits de par le monde au cours des cinquante dernières années, tous se classent parmi un nombre extrêmement restreint de styles architecturaux. De plus, un système informatique peut utiliser plusieurs styles.

Quelques architectures « célèbres » :

- Architecture en couches
- Architecture centrée sur les données
- Architecture orientée objets

Architecture trois tiers

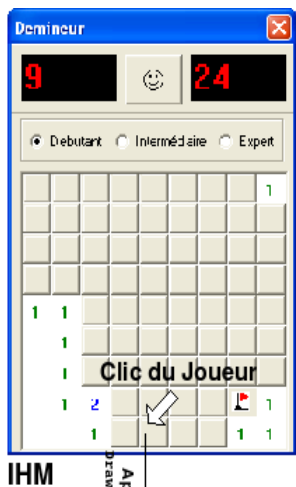
L'architecture trois tiers (« 3-Tier » anglais tier signifiant étage ou niveau) ou architecture à trois niveaux est un modèle logique d'architecture applicative qui vise à séparer très nettement trois couches logicielles au sein d'une même application ou système, à modéliser et présenter cette application comme un empilement de trois couches dont le rôle est clairement défini :

- **la présentation des données (couche présentation)** : correspondant à l'affichage, la restitution sur le poste de travail, le dialogue avec l'utilisateur (IHM) ;
- **le traitement métier des données (couche métier ou application)** : correspond à la partie fonctionnelle de l'application, celle qui implémente la « logique » applicative ;
- **l'accès aux données persistantes (couche accès aux données ou persistance)** : correspondant aux données qui sont destinées à être conservées sur la durée, voire de manière définitive.

Voir aussi : le Modèle-Vue-Contrôleur MVC

Il existe aussi le Modèle-Vue-Contrôleur (MVC, Model-View-Controller) qui est une architecture et une méthode de conception qui organise l'interface Homme-Machine (IHM) d'une application logicielle. Il divise l'ihm en un modèle (modèle de données), une vue (présentation, interface utilisateur) et un contrôleur (logique de contrôle, gestion des évènements, synchronisation), chacun ayant un rôle précis dans l'interface.

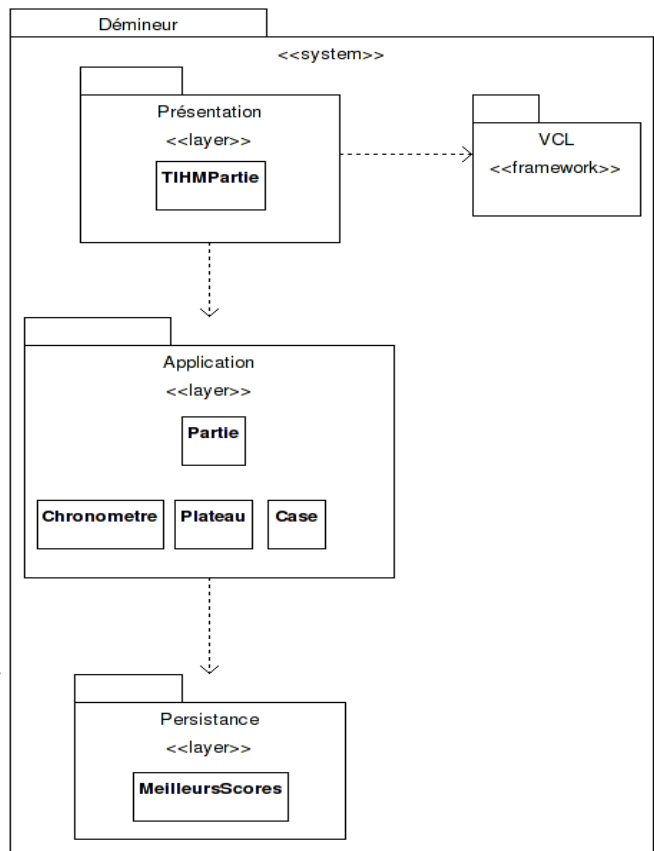
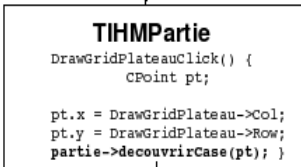
On représente l'architecture de l'application en utilisant un **diagramme de packages** d'architecture



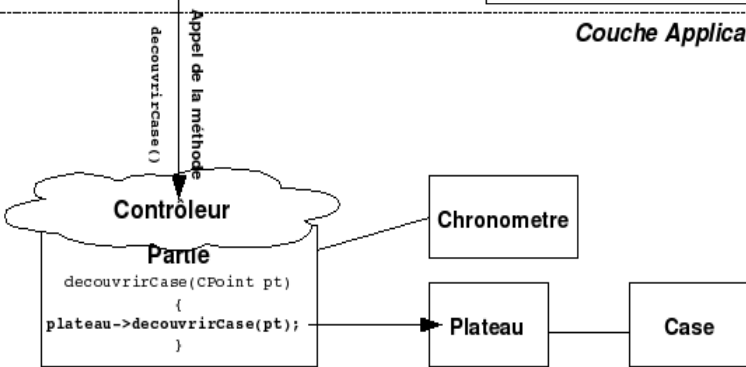
IHM

Message WINDOWS
Appel de la méthode DrawGridPlateauClick()

Couche Présentation



Couche Application



Couche Persistence



Remarques

Le principal objectif de cette séparation en trois couches est d'isoler la logique de l'application de(s) classe(s) de présentation (IHM) ainsi que d'interdire un accès direct aux données stockées par ces classes. L'intérêt est de pouvoir modifier l'interface de l'application sans devoir modifier la couche Application et de pouvoir changer de stockage sans avoir à retoucher l'interface et la couche Application.

Pour éviter que les objets de l'IHM ne soient obligés de connaître les détails de la couche Application, on introduit un objet, souvent appelé **contrôleur**, qui joue le rôle de façade vis-à-vis de la couche Présentation (voir pattern GRASP « contrôleur »).