

Le modèle MVC



Le modèle MVC (1)

- Le modèle MVC permet de décomposer une application interactive en 3 types d'objets distincts:
 - le modèle (*model*)
 - la vue (*view*)
 - le contrôleur (*controller*)



Le modèle MVC (2)

- Le modèle :
 - Correspond aux données brutes de l'application (base de données, DataSet, fichier XML, fichier texte, collection de données, ...) et les opérations qui leur sont associées: lecture, suppression, ajout et mise à jour.
- La vue :
 - C'est ce que l'utilisateur voit (étiquettes, zones de texte, listes simples, listes combinées, DataGridView, barre de navigation, etc...).

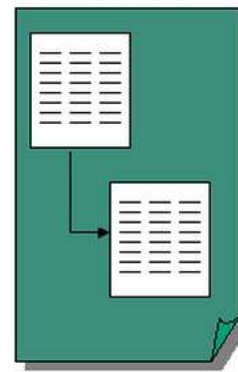
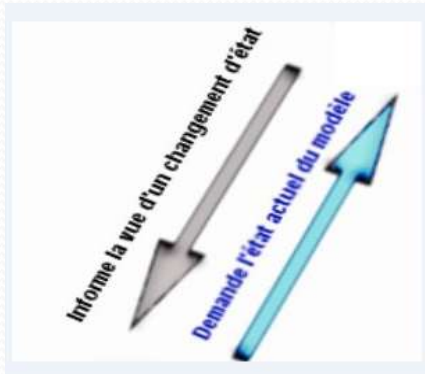
Le modèle MVC (3)

- Le contrôleur :
 - C'est le cœur de l'application.
 - C'est l'objet qui fait le lien entre le modèle et la vue.
 - Les données visibles dans la vue sont logiquement liées (par référence) aux données situées dans le modèle par le biais du contrôleur.

Architecture du modèle MVC

MODÈLE

DataSet



OrderID	CustomerID	OrderDate	EmployeeID	Freight	ShipPostalCode
10240	VINET	07.04.1996	Steven	3.0000	5111-0
10249	TOMSP	07.05.1996	Michael	11.0000	44087
10250	HANAR	08.07.1996	Margaret	65.8300	05454-076
10251	VICTE	08.07.1996	Janet	41.3400	69004
10252	SUPRD	09.07.1996	Margaret	51.3000	8-6000
10253	HANAR	10.07.1996	Janet	58.1700	05454-076
10254	CHOPS	11.07.1996	Steven	22.9800	3012
10255	RICSU	12.07.1996	Anne	148.3300	1204
10256	WELLI	15.07.1996	Janet	13.9700	08737-363
10257	HILAA	16.07.1996	Margaret	81.9100	5022
10258	ERNST	17.07.1996	Nancy	140.5100	8010
10259	CENTC	18.07.1996	Margaret	3.2500	05022
10260	OTTIK	19.07.1996	Margaret	55.0900	50738
10761	OLTFD	19.07.1996	Marcus	1.0900	01504-471

VUE

Action utilisateur : on contrôle !



La vue doit être mise à jour...



CONTRÔLEUR

Le modèle MVC (3)

1. L'utilisateur fait une modification au niveau d'une vue (par exemple, supprimer un employé dans un *DataGridView*).
2. Un événement se produit au niveau du contrôleur.
3. Le contrôleur appelle une méthode qui est située dans le modèle (par exemple la méthode qui supprime un enregistrement).
4. Le contrôleur avertit toutes les vues qu'il y a eu une modification dans le modèle.
5. Les vues s'actualisent en fonction du nouvel état du modèle.

Le modèle MVC (4)

- En **ADO.NET**, il existe un lien entre chaque donnée visible dans la vue et la donnée correspondante dans le modèle. C'est le contrôleur qui crée ces liens.
- Par conséquent, lorsque les données du modèle sont mises à jour, les données visibles dans la vue sont également mises à jour.



Le modèle MVC (5)

- Il peut arriver qu'une action de l'utilisateur ne nécessite aucune modification dans le modèle.
- Par exemple, l'utilisateur veut se positionner sur un enregistrement particulier. Dans ce cas, le contrôleur avertit toutes les vues qu'il y a eu un repositionnement.



Avantages du modèle MVC

- La vue ne connaît pas la nature du modèle: Base de données, DataSet, Collection, Tableau, Fichier texte, Fichier XML, etc.
- Le modèle ne connaît pas la nature de la vue: TextBox, ListBox, ComboBox, DataGridView, etc.
- Tout passe par le contrôleur car c'est lui qui fait la liaison entre les données situées dans le modèle et les données visibles dans la vue.
- Les méthodes du contrôleur sont génériques. Elles s'adaptent à tous les types de vues et à tous les types de modèles.



MVC: La liaison des données

- En **ADO.NET**, la liaison des données (*Data Binding*) implante le modèle MVC:
- Il décompose l'application en trois types d'objets :
 - Le modèle (*DataSet*) qui contient les données brutes
 - La vue qui contient la présentation: *DataGridView*, *TextBox*, *ComboBox*, *ListBox*, etc.
 - Le contrôleur (*BindingSource*) qui contient des traitements et qui établit la liaison des données. Il relie le modèle et la vue.