

# WPF – MVVM – LINQ

- Considérations sur le design
- Considérations sur la séparation en couches du MVVM
- Considérations sur le modèle
- Datacontext hérité
- Démo

# Design

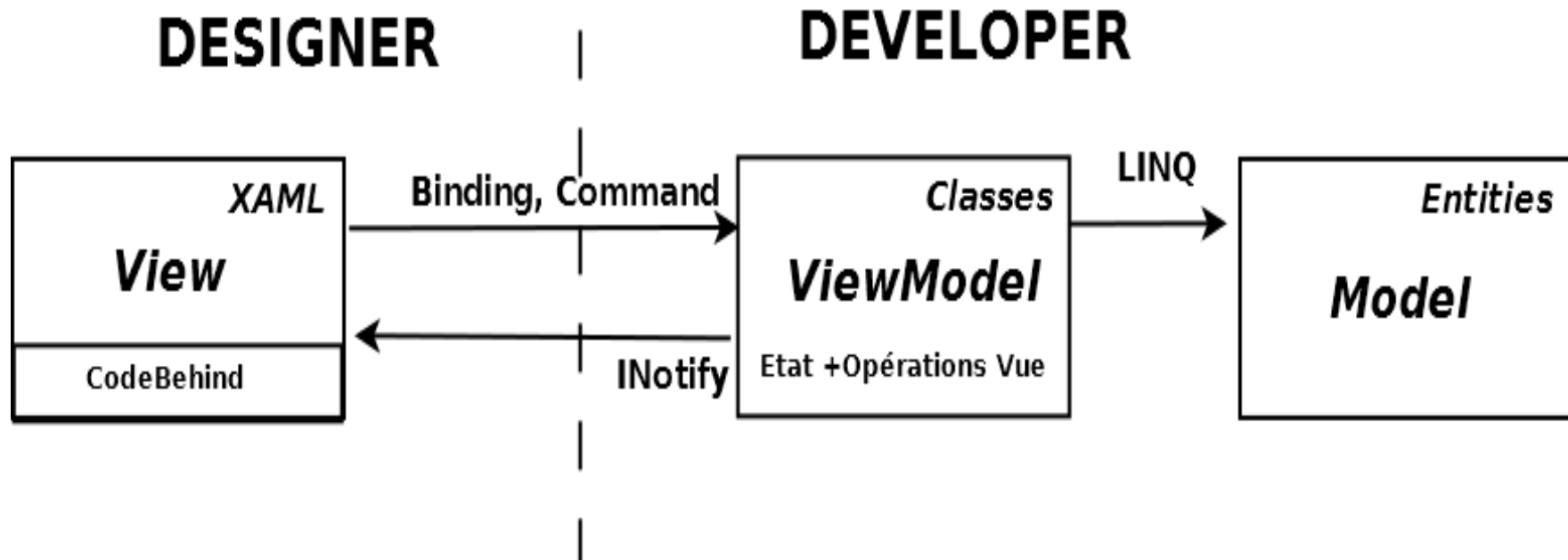
- Grid
  - Élément le plus employé
  - Les composants peuvent s'adapter (\*)
  - ShowGridLines = true

# Design



```
<Grid>
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="50"/>
    <ColumnDefinition Width="*" />
  </Grid.ColumnDefinitions>
  <Grid.RowDefinitions>
    <RowDefinition Height="50" />
    <RowDefinition Height="*" />
  </Grid.RowDefinitions>
  <Canvas Background="Red" />
  <Canvas Background="Blue" Grid.Column="1" />
  <Canvas Background="Yellow" Grid.Row="1" />
  <Canvas Background="Green" Grid.Column="1" Grid.Row="1" />
</Grid>
```

# Séparation en couches



# Modèle Vue vs Modèle

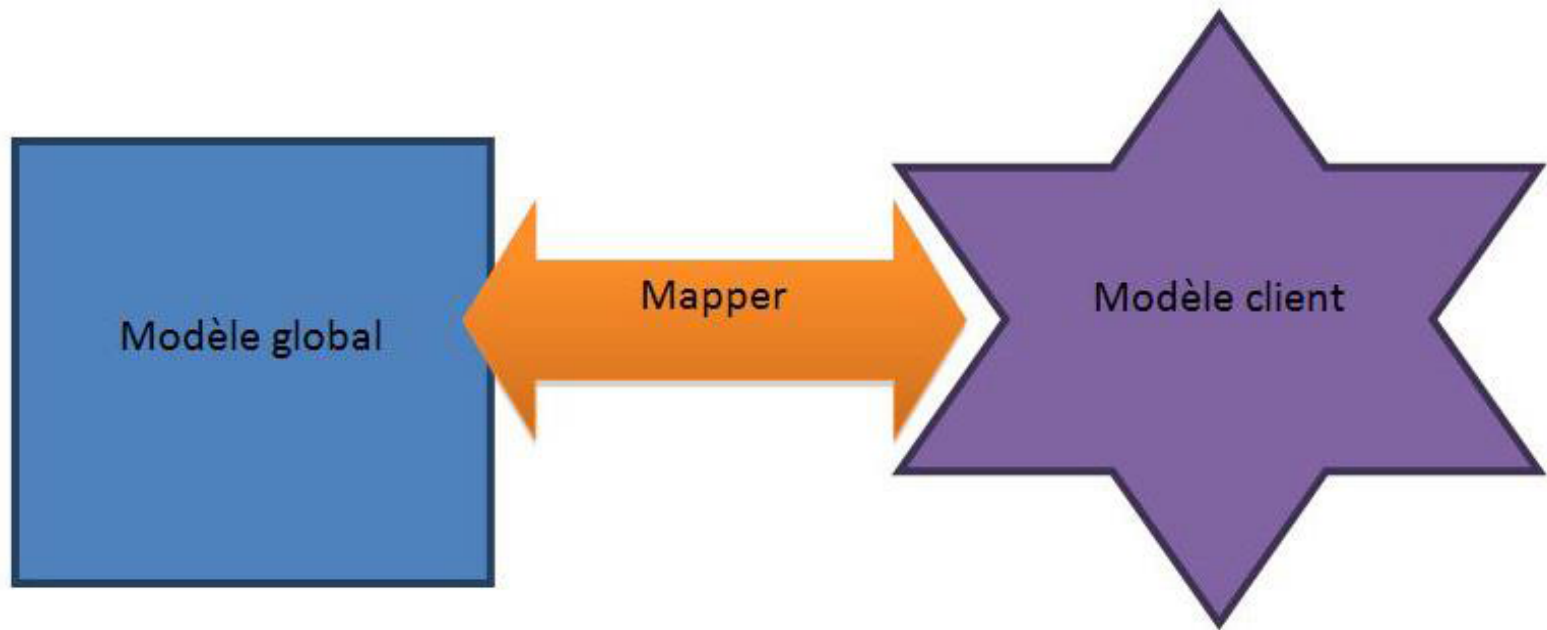
- Modèle global vs modèle client
- Modèle global indépendant de la partie client (svt Entities)
- Modèle client aussi appelé modèle de présentation
- Rôle modèle client : présenter les données , notifier, mettre en place des validations

# Le Modèle

- Le modèle global est souvent généré via LINQ to Entities
- Si le modèle change → quels impacts pour les ViewModels ?
- Changements liés à LINQ, changements métiers
- Mapping vs Wrapping

# Le Modèle

- Mapping



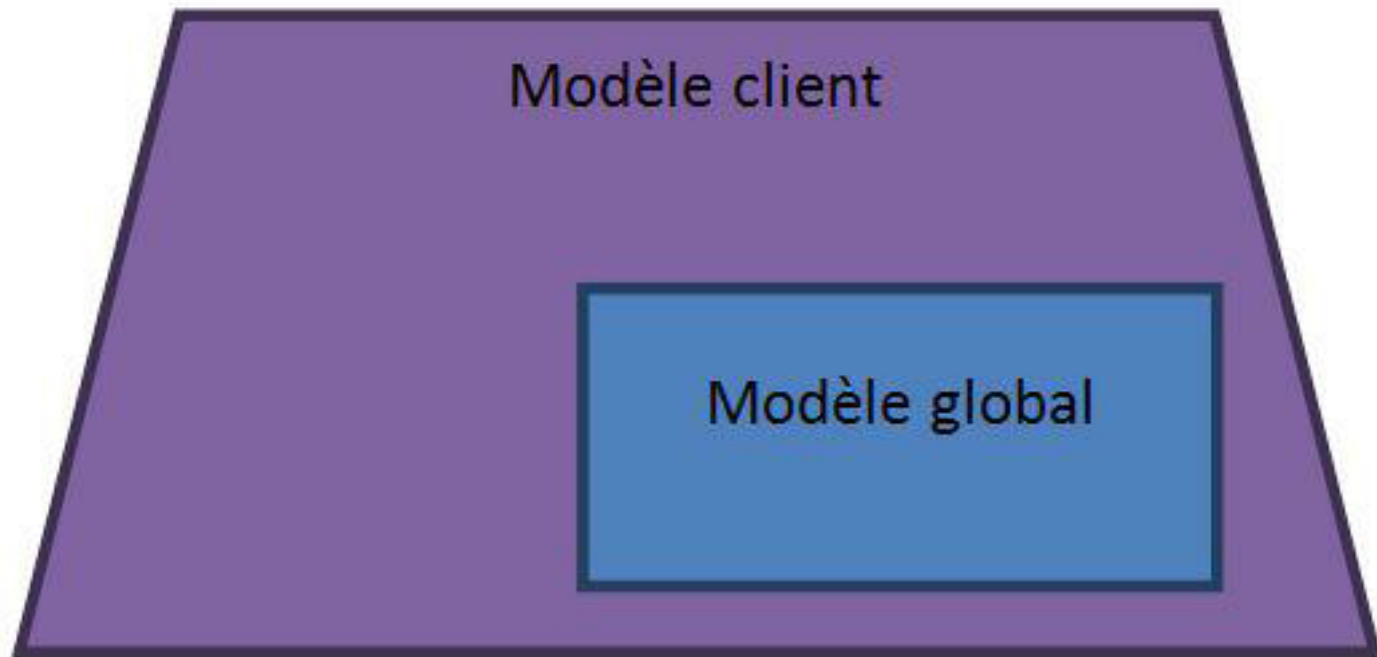
# Le Modèle

- Mapping avantages
  - Indépendance forte modèle client / modèle global
  - Maintenance facilitée
- Mapping inconvénients
  - Coût d'écriture initial
  - Puissance de calcul supplémentaire



# Le Modèle

- Wrapping (encapsulation)



# Le Modèle

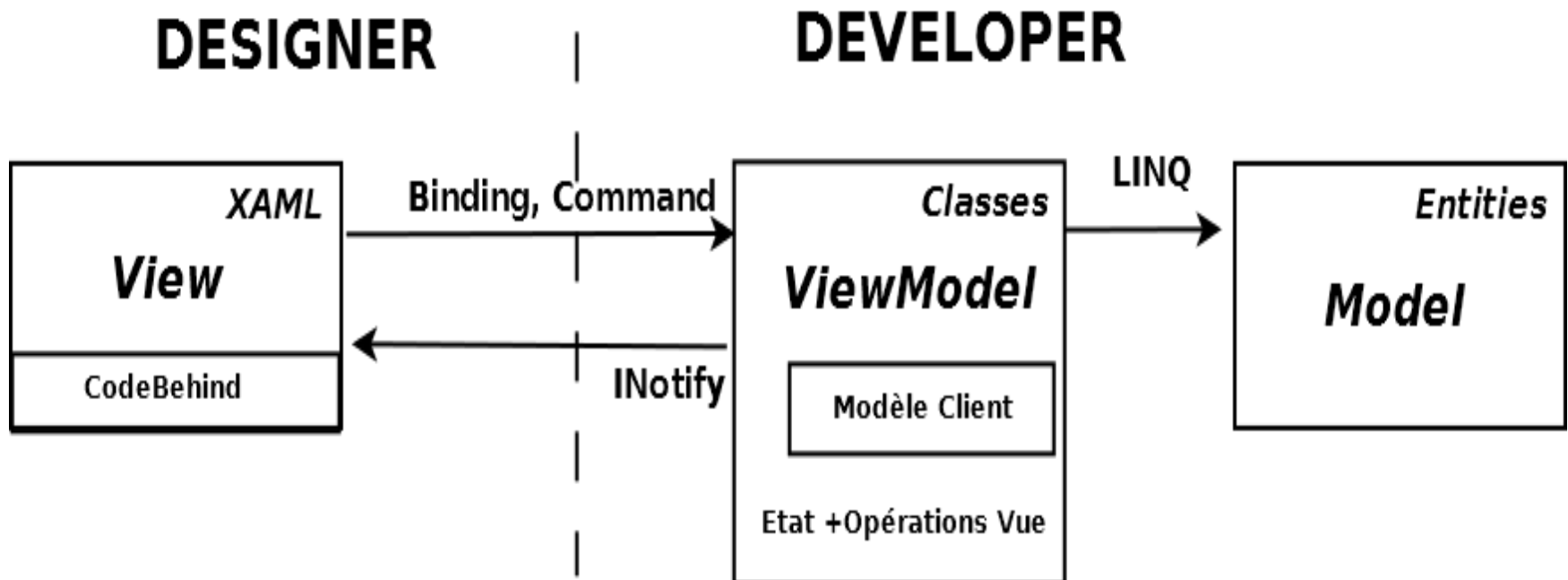
- Wrapping avantages
  - Peu de puissance de calcul nécessaire
  - Ecriture facilitée
- Mapping inconvenients
  - Forte Dépendance au modèle global
  - Nécessite connaissances et un contrôle du code du modèle global

# Le Modèle

- Wrapping

```
public class ContactModel : ModelBase
{
    private readonly Contact _Contact;
```

# Séparation en couches



# Datacontext hérité

- Le datacontext est hérité
- Exemple :
  - fenêtre → liste d'employés comme datacontext
  - Grid du formulaire de la fenêtre → un employé
  - Les champs du formulaire ont par défaut le datacontext du grid → un employé
  - Et si je veux celui de la fenêtre ?
  - Demo Legumes (!!!)