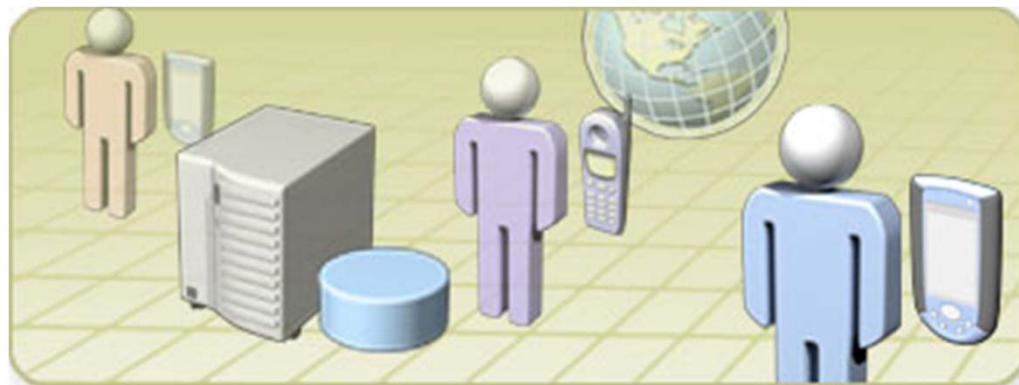


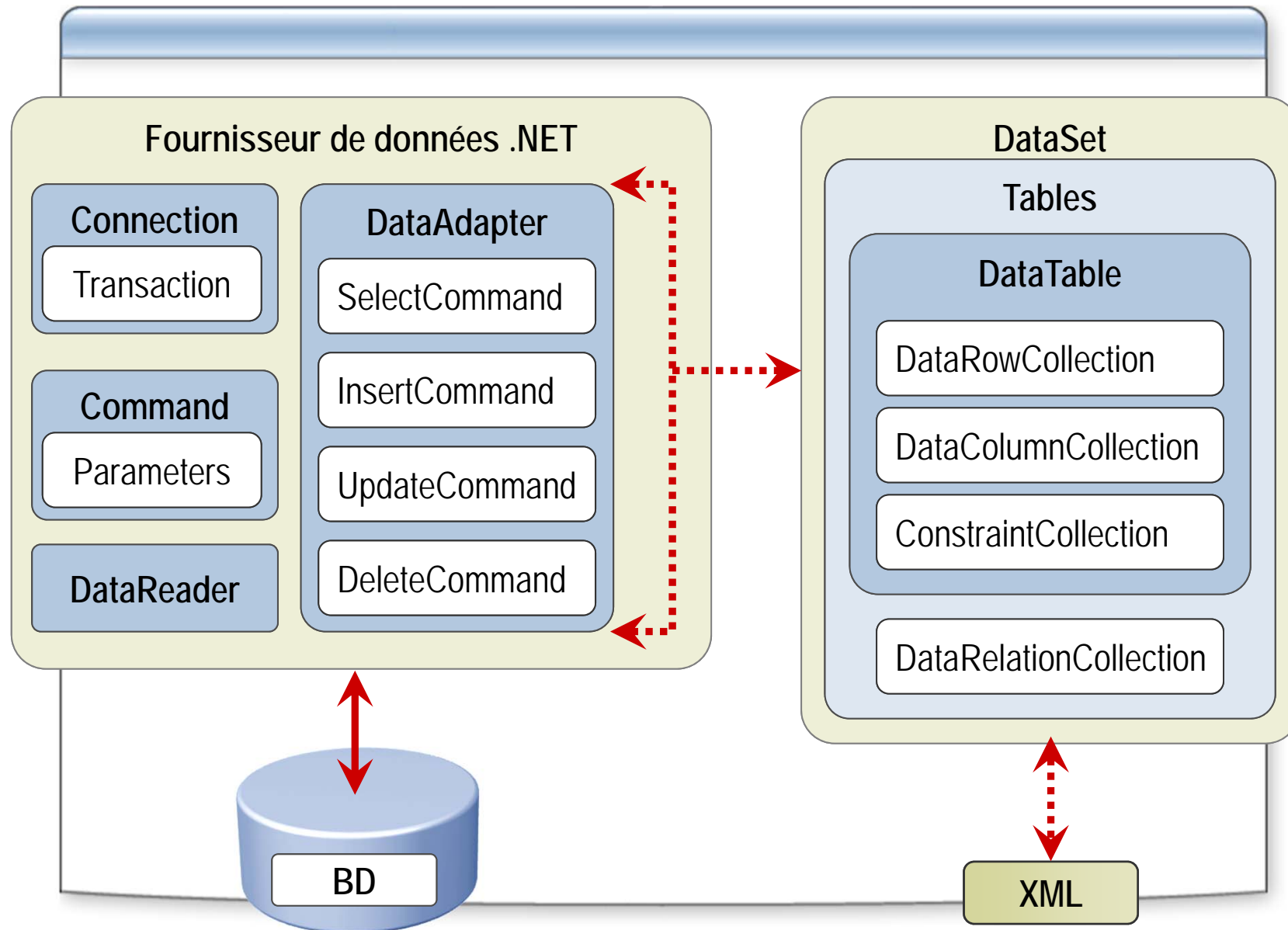
Datasets



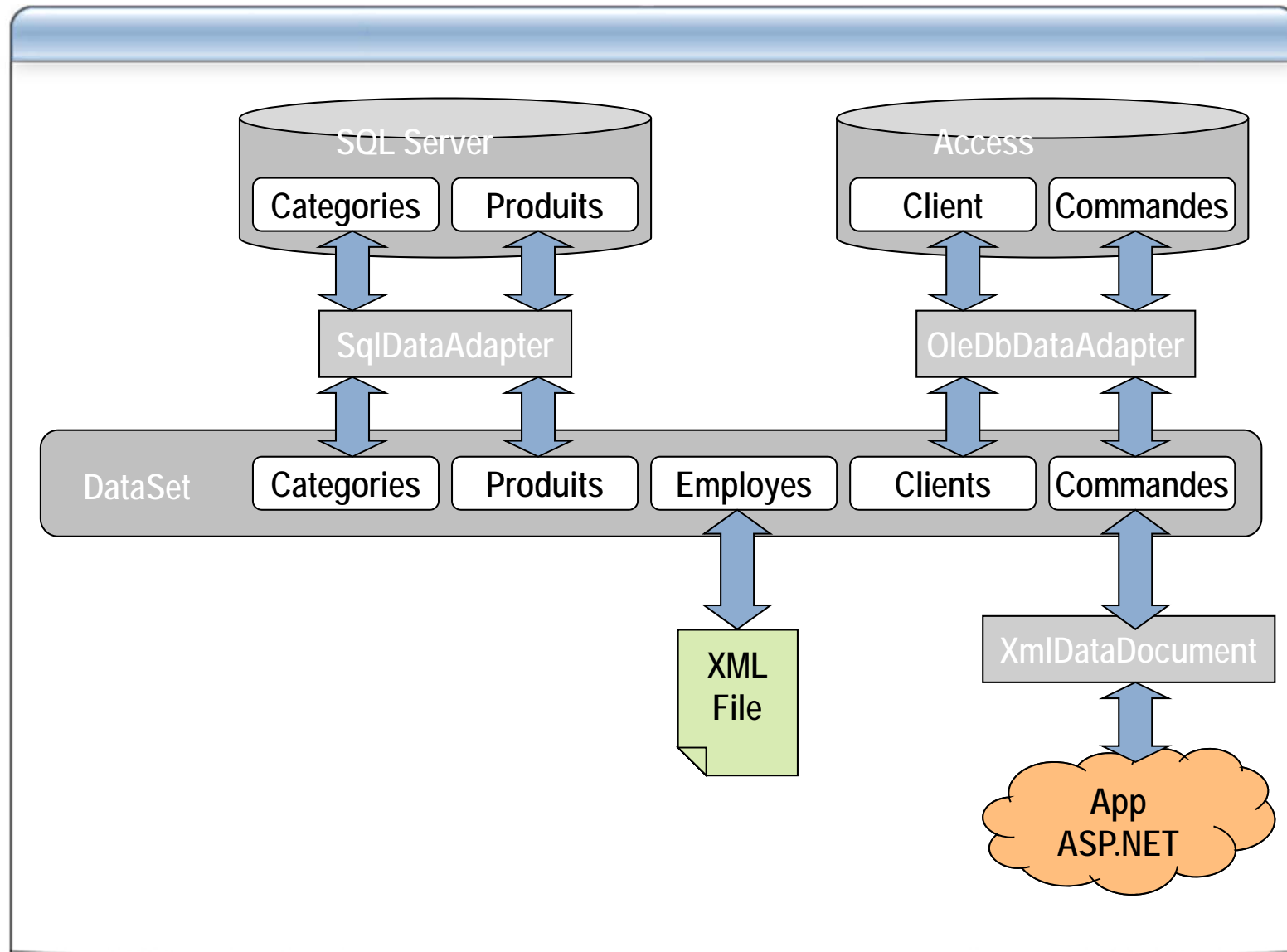
Utiliser les DataSets

- Le modèle ADO.NET
- Création d'un DataSet
- Remplissage du DataSet
- Liaisons des contrôles à un DataSet.

Le modèle ADO.NET



Architecture du mode déconnecté



Remplissage du DataSet

```
DataRow dr= ds.Tables["Tâches"].NewRow();  
dr["Nom"] = "IHM"; // ou bien dr[0]="IHM"  
dr["Début"] = "1/1/2008";  
dr["Durée"] = "1 jour";  
ds.Tables["Tâches"].Rows.Add(dr);)
```



| Nom | Début | Durée |
|------|----------|---------|
| IHM | 1/1/2008 | 1 jour |
| Code | 2/1/2008 | 2 jours |

La propriété RowState

- Lorsqu'on modifie une valeur dans un DataRow la valeur d'origine n'est pas immédiatement remplacée par la nouvelle valeur, les deux valeurs sont conservées.
- La propriété RowState détermine l'état d'une ligne, et peut prendre l'une des valeurs suivantes:
 - Added
 - Deleted
 - Detached
 - Modified
 - Unchanged

Démonstration de la propriété RowState

```
DataRow ligne;  
// Créer une nouvelle ligne  
ligne = dt.NewRow();  
// Detached  
MessageBox.Show(ligne.RowState);  
  
dt.Rows.Add(ligne);  
// Added  
MessageBox.Show(ligne.RowState);  
  
dt.AcceptChanges(); //les modifications peuvent annulées par appel de la méthode  
RejectChanges  
// Unchanged.  
MessageBox.Show(ligne.RowState);  
  
ligne["nom"] = "IHM";  
// Modified  
MessageBox.Show(ligne.RowState);  
  
ligne.Delete();  
// Deleted  
  
MessageBox.Show(ligne.RowState);
```

Lier un contrôle à un DataSet

- La propriété DataSource
 - Lie un contrôle à une source de données

```
DataTable dt = ds.Tables[" Tâches"];  
  
listBox1.DataSource = dt;  
//Colonnes à lier  
// Nom de la colonne à afficher dans la liste  
listBox1.DisplayMember = "Nom";  
// nom de la colonne qui fournira la valeur des  
éléments de la liste (propriété: SelectedValue)  
listBox1.ValueMember="ID" ;
```

Lier un contrôle à un DataSet

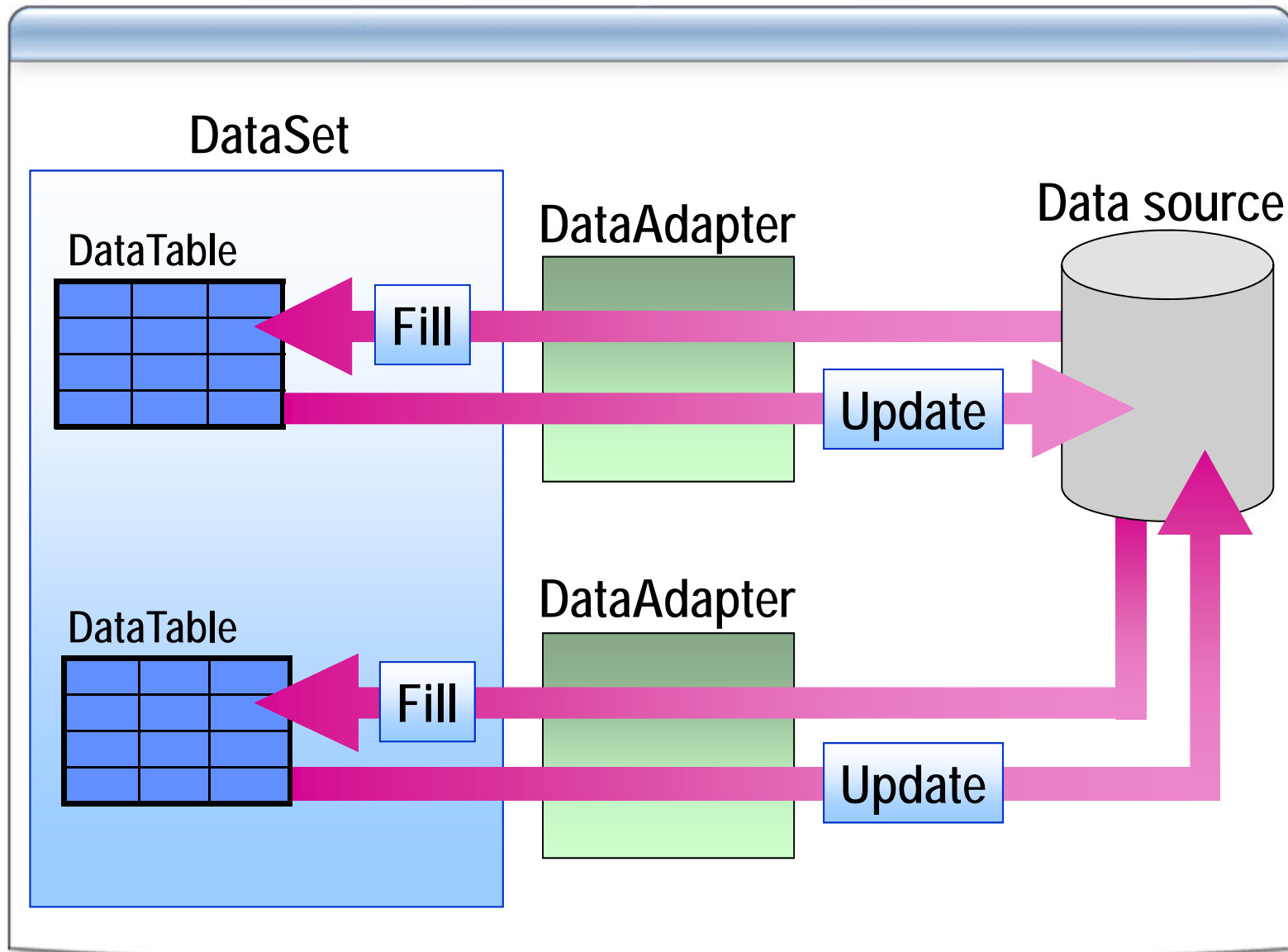
- La Collection DataBindings
 - Un DataBinding permet de lier une propriété d'un contrôle à un champ d'une table du DataSet

```
label1.DataBindings.Add("Text", ds.Tables["Tâches"],  
"Nom");
```

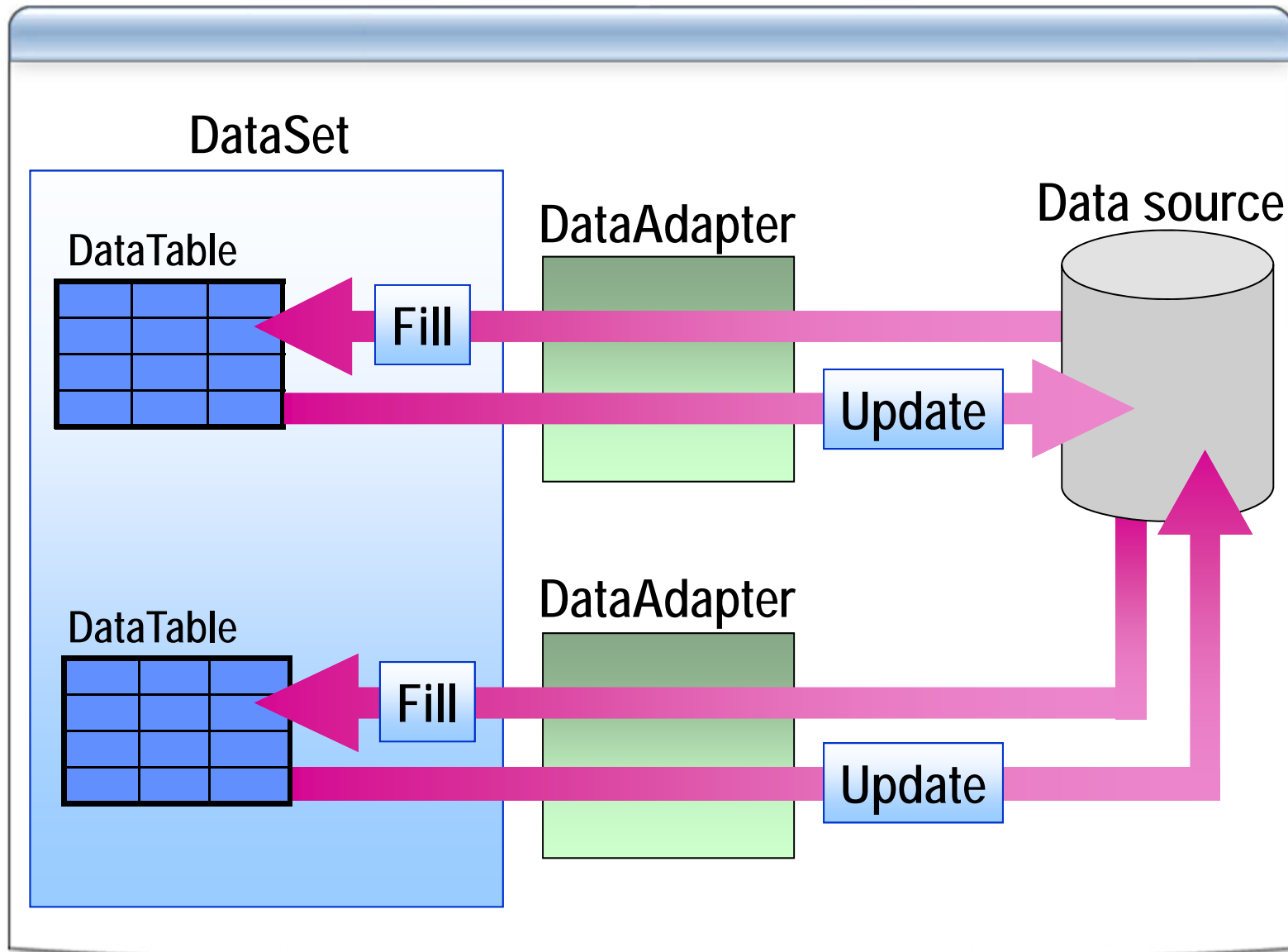
Créer un DataSet à partir d'une base de données

- Configuration d'un DataAdapter afin de récupérer des données
- Remplissage d'un DataSet à partir d'un DataAdapter
- Configurer le DataAdapter pour mettre à jour la source des données
- Sauvegarde des modifications effectuées dans le DataSet dans la source de données

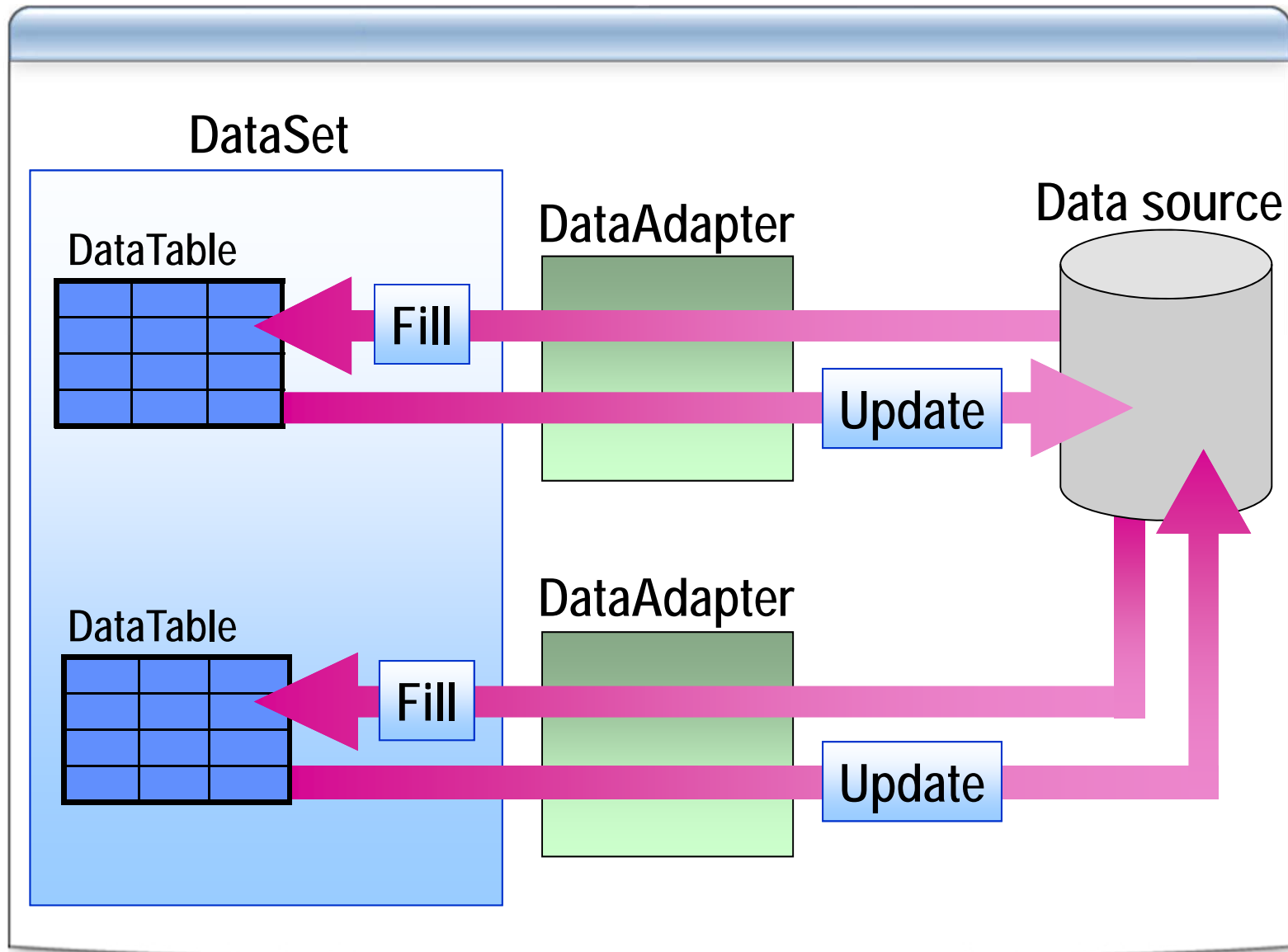
DataAdapter



DataAdapter



DataAdapter



Création et configuration du DataAdapter

```
SqlDataAdapter daProduits = new SqlDataAdapter();  
SqlConnection cnx= new SqlConnection( "data  
source=(PC\SQLEXPRESS);initial catalog=base;" + "integrated  
security=true");  
SqlCommand cmdSelect = new SqlCommand( "SELECT * FROM  
Produits", cnx);  
daProduits.SelectCommand = cmdSelect;
```

- Remplissage du DataSet à partir du DataAdapter

```
DataSet ds= new DataSet();  
ds.Tables.Add(new DataTable("Produits"));  
daProduits.Fill(ds, "Produits");
```

Mise à jour de la source de données

- Définir les propriétés InsertCommand, UpdateCommand et DeleteCommand du DataAdapter

```
daClients.Update(ds, "Clients");
```