



Dotnet France  
Technologies Sharepoint, SQL Server & .NET

Association Dotnet France

# ASP .NET Dynamic Data : introduction

*Version 1.1*

***www.Mcours.com***

Site N°1 des Cours et Exercices Email: [contact@mcours.com](mailto:contact@mcours.com)



James RAVAILLE

<http://blogs.dotnet-france.com/jamesr>

# Sommaire

---

1	Introduction.....	3
1.1	Présentation .....	3
1.2	Pré-requis .....	3
2	ASP .NET Dynamic Data dans Visual Studio 2008.....	4
2.1	Les modèles de projet .....	4
2.2	Les éléments des projets.....	4
2.3	Les contrôles ASP .NET .....	4
3	Bases fondamentales pour la création d'applications ASP .NET Dynamic Data .....	6
3.1	Technologies utilisées.....	6
3.2	Le modèle de données .....	6
3.3	Les modèles de page .....	6
3.4	Les modèles de contrôles.....	6
3.5	La génération automatique du modèle de données.....	6
4	Conclusion .....	8



## 1 Introduction

### 1.1 Présentation

ASP.NET Dynamic Data est une infrastructure permettant de créer facilement des applications Web, permettant de naviguer automatiquement au travers d'un ensemble de données, contenues dans une source de données, pouvant être une grappe d'objets ou une base de données. Elle est capable de parcourir l'ensemble des informations structurelles sur cette source de données, afin de pouvoir consulter, ajouter, modifier et supprimer les données qu'elle contient. Autrement dit, ASP.NET Dynamic Data permet de créer très rapidement, avec un faible effort de développement une application de gestion de données.

ASP.NET Dynamic Data est une nouveauté apportée par Visual Studio 2008 SP1 et le Framework .NET 3.5 SP1. Il est nécessaire de les installer sur votre poste de développement avant de pouvoir l'utiliser.

Ce cours est le premier chapitre d'une série de trois chapitres sur ASP .NET Dynamic Data. Il permet d'introduire ce sujet, en le présentant de manière globale, tout en présentant les bases fondamentales.

### 1.2 Pré-requis

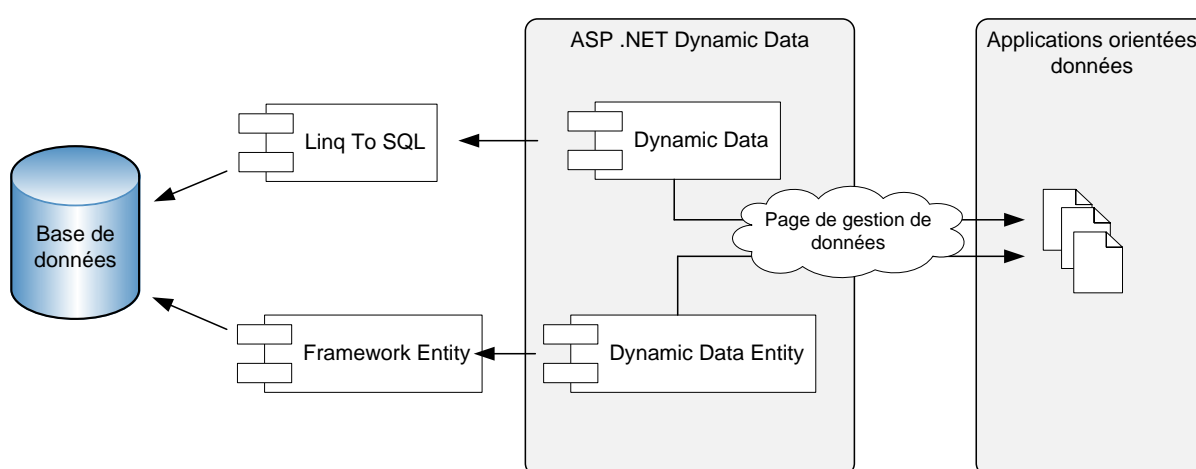
Pour comprendre le contenu de ce cours et des cours suivant sur ASP .NET Dynamic Data, il est nécessaire d'avoir des connaissances dans la création d'applications ASP .NET, de connaître un langage .NET et de savoir créer un modèle d'entités avec le Framework Entity. Sur Dotnet-France, des cours sont à votre disposition sur ces différents sujets.

## 2 ASP .NET Dynamic Data dans Visual Studio 2008

### 2.1 Les modèles de projet

Visual Studio 2008 permet de créer deux types de projet :

- Applications Web Dynamic Data : ce type de projet permet de créer une application ASP .NET Dynamic Data, qui s'appuiera sur le modèle de données de type Linq To SQL (à savoir qu'il nécessite l'utilisation du composant Classes Linq To SQL).
- Applications Web d'entités Dynamic Data : ce type de projet permet de créer une application ASP .NET Dynamic Data, qui s'appuiera sur le modèle de données de type Framework Entity (à savoir qu'il nécessite l'utilisation du composant Entity Data Model).



Les éléments décrits ci-dessous sont utilisés lors de la création d'un projet Dynamic Data, quelque soit son type. Vous pourrez les rencontrer en observant le code des pages ASP .NET générées.

### 2.2 Les éléments des projets

Dans tout projet ASP .NET, il est possible d'ajouter un composant de type « Champ Dynamic Data ». Via les mécanismes ASP .NET Dynamic Data, il permet de créer des modèles de champs personnalisés, dans le but d'afficher dans des pages, des informations provenant d'un modèle de données, dans le but de pouvoir les consulter et/ou les modifier.

### 2.3 Les contrôles ASP .NET

Lorsque nous créons ou ouvrons un projet ASP .NET, nous trouvons les contrôles utilisateurs suivant :

- *DynamicDataManager* : contrôle permettant de mettre en œuvre les mécanismes ASP .NET Dynamic Data dans les pages. Il doit être présent dans toutes les pages les utilisant. Il ne peut

être présent dans une page. Dans le cas où une Master Page est utilisée, il peut uniquement être positionné dans la Master Page, les pages de contenu en bénéficiant.

- *DynamicControl* : ce contrôle permet d'afficher le contenu d'une propriété d'une entité d'un modèle de données.
- *DynamicValidator* : permet d'intercepter les erreurs de validation des données, et d'afficher un message à l'utilisateur.



## 3 Bases fondamentales pour la création d'applications ASP .NET Dynamic Data

### 3.1 Technologies utilisées

Les applications créées avec ASP .NET Dynamic Data sont des applications ASP .NET utilisant les extensions Ajax du Framework .NET, afin d'améliorer les performances de l'application, et les interactions entre l'utilisateur et l'application : mise à jour partielle page...

Pour en savoir plus sur ces extensions, nous vous invitons à consulter le cours concernant les bases fondamentales sur ASP .NET Ajax, publié sur Dotnet-France.

### 3.2 Le modèle de données

ASP .NET Dynamic Data permet de créer des applications de gestion de données, reposant sur un modèle de données, appelé *méta-modèle*. Ce modèle de données peut avoir été créé avec Linq To SQL, ou le Framework Entity. Il peut avoir été créé dans le projet ASP .NET Dynamic Data lui-même, ou dans un projet distinct.

### 3.3 Les modèles de page

ASP .NET Dynamic Data propose des modèles de page, permettant gérer les données des entités du méta-modèle en mode CRUD (lecture, ajout, modification et suppression). Ces pages sont modifiables, afin de les personnaliser et leur permettre de répondre à notre besoin.

### 3.4 Les modèles de contrôles

Pour afficher les données, et pouvoir les modifier, ASP .NET Dynamic Data propose des modèles de contrôles sous de contrôles utilisateurs Web (ascx). Il est possible de les personnaliser. Par exemple, une donnée booléenne doit être affichée dans un contrôle une case à cocher, ou deux radios boutons respectivement libellés « oui » et « non » ? Par défaut, une case à cocher est utilisée. Nous aurons l'occasion dans les futurs chapitres de voir comment utilisé deux boutons radios.

### 3.5 La génération automatique du modèle de données

ASP .NET Dynamic Data permet de créer des applications Web orientées données. Pour naviguer au travers des données, ASP .NET Dynamic Data propose un mécanisme automatisé appelé *génération automatique*. Il s'agit d'un mécanisme d'affichage et de parcours des données, permettant d'afficher dynamiquement des pages de données en s'appuyant sur un modèle de données. Ces pages sont créées à partir des modèles de pages.

Les avantages liés à l'utilisation de ce mécanisme sont les suivants :

- Il permet de générer une application de gestion de données avec un minimum effort de conception et de développement.

- Grâce aux modèles de pages, les pages affichées dans le navigateur permettent de réaliser toutes les opérations de gestion de données en mode CRUD (affichage, insertion, modification et suppression), incluant aussi des possibilités de pagination et de tri.
- La validation des données fournies par l'utilisateur lors d'un ajout ou d'une modification repose par défaut sur le modèle de données (qui elles-mêmes peuvent être issues du schéma de la base de données).
- Des filtres générés automatiquement sont créés pour chaque clé étrangère ou champ booléen.



## 4 Conclusion

Ce cours vous a permis de découvrir ASP .NET Dynamic Data, son principe de fonctionnement et ses bases fondamentales.

Le prochain cours sur ce sujet vous permettra de le mettre en œuvre au travers d'un cas pratique.

