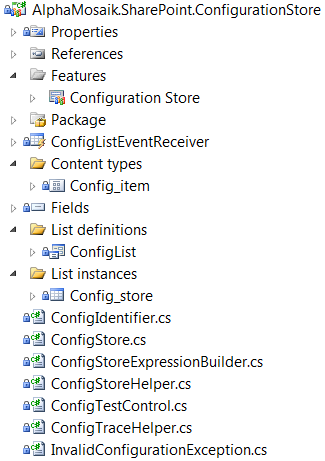
|  |
| --- |
| AlphaMosaik |
| Guide du développeur |
| AlphaMosaik.SharePoint.ConfigurationStore |
|  |
| **Simon Carpentier** |
| **10/12/2010** |

|  |
| --- |
| Ce document présente les informations nécessaires pour qu’un développeur puisse rapidement utiliser une liste pour emmagasiner et récupérer des éléments de configuration dans un projet SharePoint 2010 |

# Premiers pas

## Description

**Configuration Store** est d’abord un projet CodePlex[[1]](#footnote-1). Malheureusement, le projet est pour SharePoint 2007 et la dernière mise-à-jour date de septembre 2009.

En plus de proposer du code révisé et poli, cette version aux couleurs (et namespace) d’AlphaMosaik étend les fonctionnalités du projet d’origine en ajoutant des fonctionnalités et documentant tout le processus : de l’ajout de la DLL dans votre projet à l’utilisation des méthodes en passant par les meilleures pratiques.

## Anatomie du projet

Le projet contient :

* Un **content type** qui définit un élément de configuration
* Une **définition d’une liste** de configuration
* Une **instance de la liste** de configuration
* Quelques **classes** *helper* pour accéder facilement aux éléments de configuration

La solution « Configuration Store » est déployée au niveau du Site Collection.

## Ajouter dans le projet

Référencer AlphaMosaik.SharePoint.ConfigurationStore.dll dans votre projet.

|  |  |
| --- | --- |
| Caution | **ATTENTION**  *Afin de ne pas écraser les potentiels éléments de configuration déjà présents, vous devez déployer la fonctionnalité Configuration Store séparément de votre solution.* |

# Exemples de code

## Récupérer une information

Le processus de récupération d’une information se fait à partir de la classe statique *ConfigStore*, accessible une fois la DLL référencée.

var adminEmail = ConfigStore.GetValue("MyApplication", "AdminEmail");

## Récupérer plusieurs informations

var configIds = new List<ConfigIdentifier>()  
    {  
        new ConfigIdentifier("MyApplication", "AdminEmail"),  
        new ConfigIdentifier("MyApplication", "SendWorkflowEmails")  
    };  
   
Dictionary<ConfigIdentifier, string> configItems =

ConfigStore.GetMultipleValues(configIds);  
   
var adminEmail = configItems["AdminEmail"];  
var sendMails = configItems["SendWorkflowEmails"];

## Ajouter des éléments par code

ConfigStore.AddValue("Email", "ProjectSiteLifecycle", "Salut", "Template de l'email qui va se faire envoyer à la création d'un site", false);

|  |  |
| --- | --- |
|  | **PROTIP**  *Si votre projet s’attend à avoir certains éléments dans le config store, créez vos éléments dans un Feature Event Receiver.* |

1. <http://spconfigstore.codeplex.com/> [↑](#footnote-ref-1)