

FAQ Sharepoint

Date de publication : 12 juin 2007

Dernière mise à jour : 12 juin 2007

Cette FAQ principalement axée sur Sharepoint 2007 a été réalisée dans le but de vous apporter un éclairage sur la technologie Sharepoint. Si vous désirez contribuer à l'amélioration de cette FAQ veuillez contacter le Responsable

Ont contribué à cette FAQ :

Thomas Lebrun  - Stéphane Eyskens  

1. Généralités (6)	4
2. Architecture & objets (13)	6
3. Workflows (3)	9
4. Forms Services (Infopath) (4)	10
5. Outils et développement (14)	12
5.1. Requêtes CAML (3)	16
6. Business Data Catalog (3)	18
7. Sécurité (2)	20
8. Trucs et Astuces (1)	21

Sommaire > Généralités

De quel système d'exploitation ai-je besoin pour fonctionner avec Sharepoint?

Auteur : Stephane Eyskens

Vous devez disposer de Windows 2003 serveur tant pour le serveur lui-même qu'en tant que plateforme de développement.

Sharepoint, c'est quoi au juste?

Auteur : Stephane Eyskens

Sharepoint est un produit permettant de mettre au point des environnements intranet/extranet riches de manière efficace et rapide.

Il est encore principalement orienté gestion documentaire mais permet également de réaliser tous type d'application, surtout dans sa version 2007

Sur quelles technologies se base Sharepoint?

Auteur : Stephane Eyskens

Sharepoint repose sur une architecture web composée de IIS et ASP.NET 2.0. Il utilise aussi le CLR 3.0 dont WF est issu. Tout développement personnel consiste en général en un composant ASP.NET que l'on incorpore dans l'architecture Sharepoint.

Est-ce que WSS 3.0 est gratuit?

Auteur : Thomas Lebrun

Oui. WSS est un composant téléchargeable gratuitement sur le site de Microsoft. Cependant, étant donné que WSS ne s'installe que sur un serveur Windows 2K3, il vous faut disposer d'une CAL (Client Access Licence) pour ce serveur.

Principales différences entre WSS et MOSS (liste non exhaustive)?

Auteur : Stephane Eyskens

- a WSS est gratuit alors que MOSS est payant**
- b WSS n'englobe "que" les fonctionnalités de base de Sharepoint. MOSS est beaucoup plus riche en fonctionnalités "out of the box"**
- c MOSS intègre des services tels que Forms Services, Excel Services, le Business Data Catalog pas WSS**
- d MOSS dispose d'un moteur de recherche beaucoup plus poussé**

e **MOSS dispose d'une galerie de WebParts beaucoup plus complète que celle de WSS**

Quel type de projet doit-on s'attendre à effectuer avec Sharepoint?

Auteur : Stephane Eyskens

Tout projet nécessitant de la gestion documentaire sera grandement facilité avec Sharepoint. Tout projet nécessitant des processus d'approbation (workflow) également.

Sharepoint offre un cadre de sécurité (authorization) basé sur des rôles. Il est ensuite assez facile d'attribuer des rôles/permissions particulières sur les objets (listes, bdc...) contenus dans nos applications.

On peut techniquement réaliser tous type d'application Web car Sharepoint repose sur ASP.NET pour autant que l'on comprenne la philosophie du produit et que l'on essaye d'intégrer au mieux nos développements dans le canevas Sharepoint.

Il est évident qu'il faut essayer d'utiliser au maximum les fonctionnalités intégrées et ne pas vouloir réinventer la roue à chaque fois. Cependant, dans de nombreux cas, il est nécessaire de développer des composants/processus personnels et Sharepoint ouvre entièrement la porte aux développeurs.

Sommaire > Architecture & objets

Qu'est-ce qu'une application?

Auteur : Stephane Eyskens

L'application est l'objet le plus élevé dans la hiérarchie des objets Sharepoint. Une application contient une ou plusieurs site collections qui elles-mêmes contiennent un ou plusieurs sites. On peut décider d'attribuer une méthode d'authentification au niveau d'une application d'où l'intérêt de créer plusieurs applications (une intranet et une internet par exemple)

Qu'est-ce qu'une site collection?

Auteur : Stephane Eyskens

Une site collection est un ensemble de site ayant un lien logique entre eux. On peut définir un ou plusieurs administrateurs de la collection qui pourront donc gérer celle-ci. L'intérêt de lier entre eux des sites d'une même nature ou destinés à un même public relève précisément de la sécurité. En effet, si l'on crée deux sites destinés à des publics complètement différents au sein d'une même site collection, on éprouvera plus de difficultés à attribuer des permissions aux utilisateurs par la suite. Il est donc préférable de songer à une architecture correcte avant tout développement.

Qu'est-ce qu'un Web?

Auteur : Stephane Eyskens

Un web est en fait un site qui lui-même peut contenir des objets divers tels que des listes de documents, des listes personnelles, des calendriers etc...

Qu'est-ce qu'une "Site Column" et à quoi ça sert?

Auteur : Stephane Eyskens

Une site column se définit au niveau d'un site ou d'une site collection et est utilisable par la suite par n'importe quelle liste et/ou par n'importe quel site content type. On peut comparer une site column à une colonne d'une table d'une base de donnée car leur but est identique : décrire la donnée. On appelle aussi les colonnes Sharepoint " Metadata ". L'avantage d'utiliser une Site Column est que l'on va pouvoir la définir une seule fois et l'utiliser partout, ce qui réduit fortement la maintenance. Imaginons que vous ayez une colonne représentant le nom d'une personne. Vous pourriez par exemple créer une colonne " NomPersonne " et utiliser celles-ci dans des listes " Fournisseurs " et " Clients ". L'avantage c'est que si par la suite, vous devez augmenter le nombre de caractères autorisés pour cette colonne, toutes les listes et site content types l'utilisant seront automatiquement mises à jour.

Qu'est-ce qu'un "Site Content Type" et à quoi ça sert?

Auteur : Stephane Eyskens

Un Site Content Type peut, au même titre qu'une Site Column avoir une portée qui va d'un site à une site collection. Un Content Type, littéralement " Type de Contenu " sert à définir la nature des informations/documents d'une ou plusieurs listes.

On peut définir la nature des informations en y associant du metadata (Site Column) et optionnellement un template de document. On peut également associer un workflow.

On peut par la suite associer ce type de contenu à une ou plusieurs listes ou même un autre type de contenu.

Il est en effet possible de faire hériter un type de contenu d'un autre. Prenons un exemple concret.

Vous créez un type de contenu nommé " BonDeLivraison ". Dans ce type de contenu vous créez les colonnes suivantes :

Exemple de metadata

- Numero
- Date
- NumeroCommande
-(toutes les colonnes génériques concernant les bons de livraisons)

Ensuite, vous vous rendez-compte que le type d'informations varie légèrement en fonction de la localisation du client. Par exemple, en fonction de celle-ci, vous émettez un document dans la langue du client.

Vous pourriez donc créer d'autres types de contenu qui s'appelleraient BonDeLivraisonFr, BonDeLivraisonEn etc... qui eux-mêmes hériteraient du type de contenu " BonDeLivraison " et qui spécifieraient leur propre template de bon et optionnellement du metadata qui leur est propre. Ainsi, lorsque vous les associez à la liste " BonDeLivraisons", vous cliquerez sur " Nouveau " " Le type de bon que vous désirez créer " et le système vous ouvrira automatiquement le bon template dans la bonne langue avec le metadata associé.

Comment associer un Content Type à une liste?

Auteur : Stephane Eyskens

Vous devez cliquer sur votre liste, aller dans les propriétés avancées et autoriser la manipulation du type de contenu. Ensuite, vous pourrez manuellement lier le type de contenu à la liste.

Comment associer un workflow à une liste?

Auteur : Stephane Eyskens

Vous devez cliquer sur votre liste, aller dans les propriétés avancées et cliquer sur " workflow settings ". De là, vous pourrez associer un workflow à votre liste.

Que sont les Custom List?

Auteur : Stephane Eyskens

Ce sont des listes qui peuvent contenir tous types d'information. Elles ne sont pas dédiées à contenir un type prédéfini.

Que sont les Form library?

Auteur : Stephane Eyskens

Les Form Library(ies) sont destinées à contenir des formulaires infopath. On peut y associer un template par défaut qui s'ouvrira lorsque l'on créera un nouvel élément.

Que sont les "Document library"

Auteur : Stephane Eyskens

Ce sont des listes destinées à contenir tous types de documents (word, excel, powerpoint, page, etc...)

A quoi sert la liste Form Templates?

Auteur : Stephane Eyskens

La liste Form Templates contient tous les formulaires infopath destinés à être utilisés avec forms services. Avant d'aboutir dans cette liste, les documents ont été préalablement chargés dans Sharepoint par un administrateur (voir rubrique Forms Services pour plus d'info)

Qu'est ce que le MySite ?

Auteur : Thomas Lebrun

Le MySite est un espace personnel accessible à chaque utilisateur SharePoint. Il s'agit d'un site SharePoint sur lequel vous pouvez y déposer ce que vos souhaitez. Il y a 2 parties: une partie privée, visible uniquement par l'utilisateur en question et dans laquelle vous mettez "ce que vous désirez", et une partie publique, visible par tous les utilisateurs et sur laquelle ils peuvent voir votre place dans la hiérarchie, certaines de vos informations, etc....

Qu'est ce qu'une audience ?

Auteur : Thomas Lebrun

Une audience est une sorte de groupe, dont les membres sont définis en fonction d'une règle. Une audience permettra, par exemple, d'indiquer que tous les utilisateurs ayant pour valeur "Sales" la propriété "Department" de leur profil feront partie de l'audience "Sales"

Sommaire > Workflows

Qu'est-ce qu'un workflow?

Auteur : Stephane Eyskens

Un workflow est un ensemble d'actions que l'on va associer à un évènement se produisant sur une donnée. A la manière d'un déclencheur dans une base de données.

Le workflow sera composé d'un ou plusieurs " step " définissant les actions à faire.

On les utilise notamment pour envoyer des notifications lorsque tel ou tel évènement se produit, pour gérer un processus d'approbation, pour toutes sortes d'actions qui doivent s'exécuter lorsqu'un évènement se produit sur une donnée. Dans Sharepoint, on associe ces workflows à des listes et/ou content types qui sont eux-mêmes associés à des listes.

Les workflows sont développés grâce au moteur WF (windows foundation) sorti avec le framework 3.0

Peut-on développer des workflows dans WSS 3?

Auteur : Stephane Eyskens

Oui et c'est même plus fréquent qu'avec MOSS car WSS inclut beaucoup moins de workflow "built-in".

Comment développe-t-on un workflow?

Auteur : Stephane Eyskens

On peut développer un workflow en utilisant Sharepoint Designer car celui-ci fournit une interface graphique permettant de gérer des workflows.

Il permet de réaliser les tâches suivantes de manière aisée :

Sharepoint Designer

- Envoyer des emails de notifications
- Copier/Modifier/Supprimer/Ajouter des items à une liste d'un même site
- Construire une logique via des tests (if-elseif-else)
- Attendre dans le workflow qu'une donnée change
- Encore d'autres possibilités

Lorsque Sharepoint Designer ne suffit plus, il faut développer son workflow avec Visual Studio. Il faut au préalable installer l'extension pour VS (voir section outils)

Pour plus de détails sur le développement personnel de workflow, voir section développement.

Sommaire > Forms Services (Infopath)

Qu'est-ce qu'une "Administrator Form"?

Auteur : Stephane Eyskens

C'est un formulaire qui contient du code behind et/ou dont le niveau de sécurité est full trust. Lorsque l'on déploie une Administrator Form, on la lie à un Site Content Type qu'on lie ensuite à une liste de type " form ".

Qu'est-ce qu'une Browser form?

Auteur : Stephane Eyskens

C'est un formulaire qui doit pouvoir être exécuté dans un navigateur uniquement en se passant du SmartClient d'infopath. Les browser forms vont certainement avoir un succès grandissant car elles ne nécessitent aucune installation particulière sur le poste client et permettent d'effectuer un grand nombre de tâches en bénéficiant de la puissance d'infopath.

Il y a certes pas mal de restrictions par rapport aux formulaires Infopath classiques mais elles ne sont néanmoins pas dénuées d'atouts.

Peut-on mettre du code behind dans une browser form?

Oui et on peut le faire de deux manières :

Soit en utilisant VSTA soit en utilisant Visual Studio. VSTA est une sorte de visual studio light dédié aux formulaires Infopath. Même si vous ne disposez pas de Visual Studio, vous aurez quand même accès à VSTA

Comment publier une form vers une liste sharepoint?

Auteur : Stephane Eyskens

Un formulaire normal

Lorsque votre formulaire n'a pas de code behind et lorsqu'il ne nécessite aucun niveau de sécurité particulier, vous pouvez directement publier votre formulaire via l'assistant de publication Infopath vers une liste Sharepoint. Infopath créera alors automatiquement cette liste pour vous.

Un form template

Lorsque vous avez du code behind et/ou un niveau de sécurité élevé, vous devrez publier votre formulaire en tant que " administrator approved form ". Dans ce cas, je vous conseille de d'abord générer votre metadata, ensuite effectuer une première publication vers un Site Content Type via l'assistant de publication d'Infopath.

Ajoutez ensuite votre code et republiez votre formulaire vers un fichier local sur votre disque.

Lorsque vous avez le fichier "nom".xsn, vous devrez le charger dans Sharepoint, vous pouvez soit utiliser la ligne de commande " stsadm -o " soit utiliser l'interface d'administration de Sharepoint. C'est cette méthode qui est décrite ci-dessous

Etapes pour publier un formulaire avec du code

- Ouvrez la centrale d'administration
- Allez dans Application Management
- Allez dans Manage Form Templates => Upload a form template
- Cliquez sur "Browse#" et précisez le fichier xsn que vous venez de publier
- Cliquez sur Upload (Verify est optionnel et permet de vérifier que votre formulaire est bien conforme aux normes)

- **Revenez à la page précédente et cliquez sur « Manage Form Template ». Vous devriez voir apparaître votre formulaire. Cliquez dessus et choisissez l'option « Activate to a site collection ».**
- **Choisissez la site collection cible.**

A partir de ce moment, votre formulaire devrait être disponible dans la liste " Form Templates ". Etant donné que vous l'aurez préalablement lié à un Site Content Type, vous pourrez ensuite lier ce Site Content Type à une liste de type " form " de votre choix pour commencer à créer des nouveaux documents.

Comment forcer une liste de type Form à afficher les formulaires infopath avec le navigateur plutôt qu'avec le SmartClient d'Infopath ?

Auteur : Stephane Eyskens

Par défaut, si le système détecte la présence du SmartClient sur la machine cliente, il ouvrira le formulaire Infopath avec sinon il l'ouvrira automatiquement avec le navigateur.

Si vous désirez malgré tout forcer l'ouverture du formulaire directement dans un navigateur, procédez comme suit :

- **Allez dans les Advanced Settings de votre liste**
- **Cliquez sur "Display as web page"**

Sommaire > Outils et développement

Où trouver tous les templates pour visual studio 2005?

Auteur : Stephane Eyskens

Template pour les workflow Sharepoint
Template pour les WebParts
Template de solution et de webparts

A quoi sert Sharepoint Designer?

Auteur : Stephane Eyskens

Sharepoint designer est un outil payant qui se veut être le successeur de FrontPage. Il sert à personnaliser les sites Sharepoint.

Par personnalisation, on entend :

- **Changer le style (look & feel) des pages**
- **Créer/Modifier de nouvelles pages ou en supprimer**
- **Créer des workflows**
-

Il propose une interface graphique permettant de visualiser les changements avant de les appliquer online. Il offre trois vues possibles :

- **Une vue graphique**
- **Une vue code source**
- **Une vue divisée entre graphique et code source**

Où puis-je trouver les principales DLL liées à Sharepoint

Auteur : Stephane Eyskens

Les DLL de Sharepoint sont principalement stockées dans c:\program files\common files\microsoft shared\web server extensions\12\isapi

L'assembly permettant de manipuler les objets Sharepoint se trouve dans Microsoft.Sharepoint.dll

Comment intégrer du code C# ou VB.NET dans une page ASP.NET déployée dans Sharepoint ?

Auteur : Stephane Eyskens

Vous avez créé une nouvelle page ou bien vous modifiez la page " Default.aspx " d'un site Sharepoint existant et vous n'arrivez pas à y mettre du code C# ou VB.NET car vous obtenez une erreur d'exécution ? C'est normal, la sécurité Sharepoint vous en empêche.

Vous devez donc explicitement autoriser le " inline code " dans vos pages. Pour cela, effectuer les étapes suivantes :

- **Editez le web.config de votre application Sharepoint**
- **Localisez/Ajoutez la section « PageParserPaths »**

Editez-là de manière à obtenir ceci

```
<PageParserPaths>
  <PageParserPath VirtualPath="/chemin/
page.aspx" CompilationMode="Always" AllowServerSideScript="true" />
</PageParserPaths>
```

Où " chemin/page.aspx " représente la page pour laquelle vous autorisez le code. Vous pouvez utiliser le wildcard " * " pour autoriser plusieurs pages (ex : *.aspx)

Comment déployer un WebPart dans Sharepoint ?

Auteur : [Stephane Eyskens](#)

Un WebPart est un contrôle serveur héritant de la classe WebPart. Le développeur doit créer son environnement graphique " manuellement " sans le support du designer (voir question suivante). Par déploiement manuel, on entend:

- Création d'un projet de type "project library" dont la classe principale hérite de WebPart
- Signature de votre assembly
- Déploiement de l'assembly dans la GAC
- Ajout de la signature de votre assembly obtenue avec Reflector dans la section <safecontrols> du fichier web.config de votre application Sharepoint
- Chargement de votre WebPart dans la galerie de WebParts

Existe-t-il un designer pour les WebParts que l'on crée dans Sharepoint ?

Auteur : [Stephane Eyskens](#)

Il n'existe pas à proprement parler de designer mais il existe le SmartPart. Le SmartPart permet d'intégrer un User Control ASP.NET au sein d'un WebPart. Vous pouvez ensuite déployer le SmartPart dans Sharepoint. Pour plus d'info, consultez le site de l'auteur [SmartPart](#)

Qu'est-ce qu'une feature?

Auteur : [Stephane Eyskens](#)

Une feature peut représenter n'importe quel type de développement. Une feature est une sorte de composant que l'on peut activer/désactiver au niveau d'un site ou au niveau d'une collection de sites en fonction du scope (portée) qui a été défini lors du déploiement de la feature.

Au sein de cette feature, on peut incorporer un webpart, des pages aspx, un workflow, ou encore modifier les menus intégrés de Sharepoint (comme le Site Actions par ex).

Qu'est-ce qu'un Event Handler?

Auteur : [Stephane Eyskens](#)

Un Event Handler est une DLL que l'on peut associer à une liste en vue d'intercepter un évènement se produisant sur celle-ci.

Ils permettent d'exécuter n'importe quel code à un moment bien précis comme par exemple à l'ajout d'un item dans une liste, la modification ou la suppression de celui-ci ou tout à la fois.

Ils pourraient par exemple aussi être utilisés pour empêcher toute personne de supprimer un item pour disposer d'une liste " indestructible " .

Un autre cas plus réaliste d'utilisation consisterait à définir une sécurité au niveau d'un item en fonction des données contenues par celui-ci, ce que la sécurité " statique " de Sharepoint ne nous permet pas de faire.

Comment permettre l'utilisation des Events Handlers ?

Auteur : **Stephane Eyskens**

Dans MOSS & WSS, l'utilisation des Events Handlers est désactivée par défaut. Pour les autoriser, vous devez procéder comme suit :

- Allez dans la centrale d'administration
- Allez dans Application Management
- Allez dans Advanced Settings
- Cliquez sur " Allow backward compatible event handlers"

Comment lier un Event Handler à une liste?

Auteur : **Stephane Eyskens**

Auparavant, Sharepoint permettait de lier un Event Handler à une liste directement via la centrale d'administration mais ce n'est plus le cas. Vous devrez donc le faire par le code directement.

Vous devez d'abord enregistrer votre DLL dans la GAC. Ensuite, voici un petit exemple (appli console) qui montre comment lier celle-ci sur l'évènement " ajout d'item " à une liste

```
Try
{
string AssemblyName = "nom assembly obtenu avec reflector"
string ClassName = " namespace.votreclasse "
SPSite Site = new SPSite(" url de votre serveur ");
SPWeb Web = Site.OpenWeb();
SPList List = Web.Lists["Votre Liste"];
List.EventReceivers.Add(SPEventReceiverType.ItemAdded, AssemblyName, ClassName);
List.Update();
Web.Close();
Site.Close();
}
Catch (SPException SpEx)
{
//Traitement...
}
```

Si votre DLL intercepte d'autres évènement (ItemAdding, ItemModified etc...), vous n'avez qu'à ajouter les EventReceivers les uns derrière les autres.

Il existe également un ou plusieurs outils tiers qui permettent de lier un Event Handler à une liste. Vous en trouverez notamment un [ici](#)

Comment créer un thème personnalisé?

Auteur : Stephane Eyskens

Créer un thème personnel constitue une approche intéressante pour personnaliser le look & feel des applications Sharepoint car un thème peut être activé/désactivé au niveau d'un Site indépendamment des autres sites.

Pour créer son propre thème, il faut au minimum procéder aux étapes suivantes :

Etapes de création d'un thème

- Localisez le thème built-in qui correspond le plus à celui que vous comptez créer
- Ouvrez un explorateur sur `c:\program files\common files\microsoft shared\web server extensions\12\layouts\themes`
- Copiez le répertoire du thème qui se rapproche le plus de ce que vous voulez faire et renommez-le `<votretheme>`
- Allez dans ce nouveau répertoire, ouvrez `<letheme.inf>` et remplacez les occurrences de `<lethme>` par `<votretheme>`. Remplacez ensuite le numéro dans le paragraphe version par un numéro unique du genre 22035. Fermez le fichier et renommez le en `<votretheme.inf>`
- Localisez `SPTHEMES.XML` (quelque part dans `web server extensions/12/...`) et ajoutez-y une entrée pour votre theme
- Allez dans Sharepoint, appliquez votre thème à un site
- Ouvrez Sharepoint designer, déployez le répertoire `_themes` et localisez votre thème. Vous y trouverez un fichier nommé `<votretheme0001.css>`, éditez-le et vos changements seront directement effectifs (rafraîchissez la page web du site qui utilise votre thème)

Notez que pour que les modifications s'appliquent entièrement au thème, vous devrez remodifier `theme.css` se trouvant dans `"_thmemes"` et `theme.css` se trouvant sur le disque. Modifiez aussi la version dans le fichier `<votretheme.inf>`.

Notez également qu'il n'est pas évident de personnaliser le thème copié car Sharepoint utilise des milliers (j'ai jamais compté) de classes CSS et il faut réussir à trouver à quels éléments elles se rapportent.

[Sommaire](#) > [Outils et développement](#) > [Requêtes CAML](#)

Comment effectuer une requête CAML dans une liste Sharepoint

Auteur : [Stephane Eyskens](#)

Vous pouvez utiliser la classe SPQuery. Voici, un exemple.

```
try
{
    SPSite site = new SPSite("serveur_url");
    SPWeb web = new site.OpenWeb();
    SPList list = web.Lists["votre_liste"];
    SPQuery query = new SPQuery(list.DefaultView);
    query.Query = "<Where><Eq><FieldRef Name='VotreChamp' /><Value Type='Text'>Une Valeur</Value></Eq></Where>";
    SPListItemCollection items = list.GetItems(query);

    foreach (SPListItem item in items)
    {
        Console.WriteLine(item["Title"]);
    }
}
catch (SPException SpEx)
{
    //Traitement d'erreur
}
```

Ici, nous utilisons l'opérateur "<Eq>", nous pourrions par exemple utiliser <Contains pour une recherche correspondant à un LIKE SQL.

Vous devez bien spécifier le nom interne du champ de liste et non son titre "Title".

Il existe également des outils graphiques permettant de générer des requêtes CAML. Une petite recherche sur CAML Query Builder sur google vous guidera vers ce type d'outil.

Comment spécifier les champs à inclure dans une requête CAML?

Auteur : [Stephane Eyskens](#)

```
try
{
    SPSite site = new SPSite("serveur_url");
    SPWeb web = new site.OpenWeb();
    SPList list = web.Lists["votre_liste"];
    SPQuery query = new SPQuery(list.DefaultView);
    query.Query = "<Where><Eq><FieldRef Name='VotreChamp' /><Value Type='Text'>Une Valeur</Value></Eq></Where>";
    SPListItemCollection items = list.GetItems(query);

    foreach (SPListItem item in items)
    {
        Console.WriteLine(item["Title"]);
    }
}
catch (SPException SpEx)
{
    //Traitement d'erreur
}
```


Dans l'exemple précédent, nous spécifions la vue "DefaultView". Nous aurions pu utiliser une vue définie au niveau de la liste. Il est aussi possible de spécifier les champs à inclure comme ceci:

```
query.ViewFields = "<FieldRef Name='Champ1'/><FieldRef Name='Champ2'/>...";
```

Comment connaître le nom interne d'une colonne de liste Sharepoint à utiliser dans une requête CAML?

Auteur : Stephane Eyskens

Lorsque vous effectuez une requête CAML sur une liste Sharepoint via SPQuery, il faut veiller à bien spécifier le nom interne des colonnes dans la clause **Where**.

La propriété **InternalName** de la classe **SPField** vous renseignera.

```
try
{
    SPSite site = new SPSite("url du serveur");
    SPWeb web = site.OpenWeb();
    SPList list = web.Lists["votre liste"];
    foreach(SPField field in list.Fields)
    {
        Console.WriteLine("Nom usuel : {0} Nom Interne: {1}", field.Title, field.InternalName);
    }
}
catch (SPException SpEx) { //traitement }
```

Sommaire > Business Data Catalog

Qu'est-ce que le business data catalog?

Auteur : Stephane Eyskens

Le business data catalog est un composant inclus dans MOSS 2007 (pas dans WSS) permettant d'interfacer Sharepoint avec des bases de données et des web services. Il permet de centraliser l'acquisition de données au sein de Sharepoint et de gérer la sécurité liée à cette acquisition également dans Sharepoint.

Le BDC peut aussi être considéré comme une couche d'abstraction car il peut s'interfacer avec moteurs de bases de données différents mais la manière de travailler avec le BDC ne changera pas d'un iota.

Enfin, outre l'acquisition de données proprement dites, le BDC est livré avec une panoplie de WebParts disponibles dans la " Webpart Gallery " et regroupé dans la catégorie " Business Data ". Ces WebParts permettent de lister ces données sous toutes les formes puisqu'on peut utiliser la puissance des vues/filtres comme avec les listes normales.

On peut également utiliser le BDC pour lier des listes Sharepoint à des données externes. Il suffit de créer des colonnes de type " business data ".

Le BDC n'a par contre pas été conçu pour mettre à jour les données externes car Microsoft est parti du principe qu'offrir une interface pour intégrer " en lecture " des systèmes tels que SAP, CRM etc... suffisaient amplement puisque ces outils ont déjà leurs propres couches applicatives permettant d'encoder/mettre à jour les données.

Donc, le BDC relève bien de l'acquisition de données uniquement.

Comment créer son propre BDC au sein de Sharepoint

Auteur : Stephane Eyskens

Pour l'instant, Microsoft n'a pas développé d'outil permettant de créer un business data catalog de manière automatisée. Ils ont préféré développer des APIs permettant aux tiers de concevoir des outils.

Actuellement, il faut créer un document XML qui contient toute la structure du BDC et il faut le faire... manuellement.

Il existe bien l'un ou l'autre outils payants mais ne fournissant pas de version d'évaluation digne de ce nom.

On peut par contre se baser sur les exemples fournis dans le starter kit et sur les différentes screencasts traitant ce sujet (voir forum Sharepoint) pour créer son propre BDC.

Quelles sont les principales API du bdc

Auteur : Stephane Eyskens

L'api "runtime" (Microsoft.Office.Server.ApplicationRegistry.Runtime) permet d'accéder au BDC par le code que ce soit pour les besoins d'un contrôle ASP.NET ou d'une application Windows/Console classique.

L'api "metadata" (Microsoft.Office.Server.ApplicationRegistry.MetadataModel) permet de générer du Metadata pour produire le fichier XML qui sert à injecter le BDC dans Sharepoint.

Malheureusement, ces APIs étant très peu documentées car très récentes, elles ne sont pas évidentes à manipuler.



[Sommaire > Sécurité](#)**Comment accorder les droits d'exécution d'une entité du business data catalog?****Auteur : Stephane Eyskens**

Vous avez créé des entités dans le BDC mais lorsqu'un utilisateur se connecte à une page contenant un objet issu du BDC, vous recevez une erreur vous indiquant que le système n'a pas pu recevoir les données?

C'est normal, il faut au minimum accorder les droits d'exécution de l'entité/des entités pour l'utilisateur en question.

Pour cela, vous devez ouvrir la centrale d'administration, aller dans la partie Shared Services Provider (SSP), localisez la zone Business Data Catalog, cliquez sur View Entities -> Permissions et ajoutez le/les utilisateur(s) pour lequel(s) vous désirez accorder les droits d'exécution.

Quel utilisateur est utilisé pour se connecter à la base de contenu de Sharepoint?**Auteur : Stephane Eyskens**

Si vous avez opté pour les options par défaut lors de l'installation de Sharepoint (WSS et MOSS), l'utilisateur par défaut qui se connectera à la base de données de contenu de votre application Sharepoint sera

l'utilisateur défini au niveau de l'application pool utilisé par celle-ci dans IIS.

Ceci revient à dire que si vous développez une application web ou un web service hébergé par une autre application Web, vous devez spécifier un utilisateur qui a le droit d'accéder à la base de données de contenu comme utilisateur de l'application pool de votre application web.

[Sommaire > Trucs et Astuces](#)

Comment éviter un IISReset à chaque fois que l'on déploie une feature, un webpart ou un workflow?

Auteur : Stephane Eyskens

Il est de coutume de redémarrer IIS lorsque l'on déploie une feature ou un composant personnel dans Sharepoint.

Lorsque l'on travaille sur une machine de développement qui nous est entièrement dédiée, ça ne pose pas de souci mais lorsque d'autres développeurs utilisent aussi le serveur, cela peut devenir problématique.

Il est donc possible d'éviter l'éternel iisreset en ne faisant qu'un recyclage de l'application pool qui héberge notre/nos application(s).

Pour cela, vous pouvez utiliser la commande suivante:

```
cscommand.exe c:\windows\system32\iisapp.vbs /a "application pool cible" /r
```

Cela va beaucoup plus vite, ne dérange pas les autres utilisateurs/développeurs et est suffisant pour le déploiement de composants.