Reporting Services et SharePoint

Utiliser SharePoint comme source de données

Dans un précédent article, nous avons pu apprendre comment créer un rapport utilisant une source de données XML
La question qui revient très souvent est la création d'un rapport utilisant SharePoint comme source de données.
Nous verrons dans cet article comment créer un rapport selon ce besoin.

Introduction

Avec l'utilisation de SharePoint comme solution Intranet, on arrive rapidement à des solutions développées par les utilisateurs

Ces utilisateurs souhaitent rapidement obtenir du reporting basé sur ces données.

Présentation

Microsoft ne propose pas de source SharePoint dans les différents types de sources disponibles nativement dans Microsoft SQL Server Reporting Services.

Nous verrons dans cet article différentes solutions permettant de créer un rapport basé sur des données provenant d'une liste SharePoint.

Nous verrons dans un premier temps l'utilisation du Web service standard de SharePoint permettant d'obtenir les données stockées dans SharePoint sans passer par le site Web. Puis nous verrons deux solutions permettant de travailler directement avec des connecteurs SharePoint pour Reporting Services.

Utilisation du Web Service de SharePoint

On peut atteindre SharePoint de différentes manières. Nous connaissons tous l'interface Web qui nous permet de créer, visionner ou manipuler des données dans ses sites.

Il existe aussi plusieurs Web Services (existant pour SharePoint 2003 et SharePoint 2007) permettant d'accéder aux sites depuis des applications externes, que ce soit des WinForm .NET ou des applications développées dans d'autres langages.

La documentation est disponible sur ces Web Services sur le site MSDN :

- **Web Services SharePoint**

Le principe est donc d'utiliser un de ces Web Services comme source de données XML pour notre rapport SharePoint.

Nous devons utiliser le Web Service spécifique aux listes documentaires :

- **Lists Class (Lists) - List.asmx**

Ce Web Service possède une méthode **GetListItems** qui possède une liste de paramètres possibles :

- **listName :** Nom de la liste voulue
- **viewName :** GUID de la vue éventuelle
- **query :** Sélection souhaitée dans la liste (filtre ou classement) fournie en CAML
- **viewFields :** Liste des champs de la liste
- **rowLimit :** Nombre d'items maximum
- **queryOptions :** Options éventuelles sur la sélection
- **webID :** GUID du site web parent de la liste

Tous ces paramètres sont utilisables pour obtenir l'ensemble ou une partie des Items d'une liste, sauf le paramètre "**query**". Ce paramètre permet d'envoyer une requête de sélection en CAML. Ce format CAML n'est pas supporté par Reporting Services.

Si vous souhaitez limiter le trafic ou avoir un filtrage des données, il est dans ce cas conseillé de créer une vue dans SharePoint correspondant à votre besoin. Cette astuce est expliquée dans le message suivant :

- **Using SharePoint Web Service from SSRS**

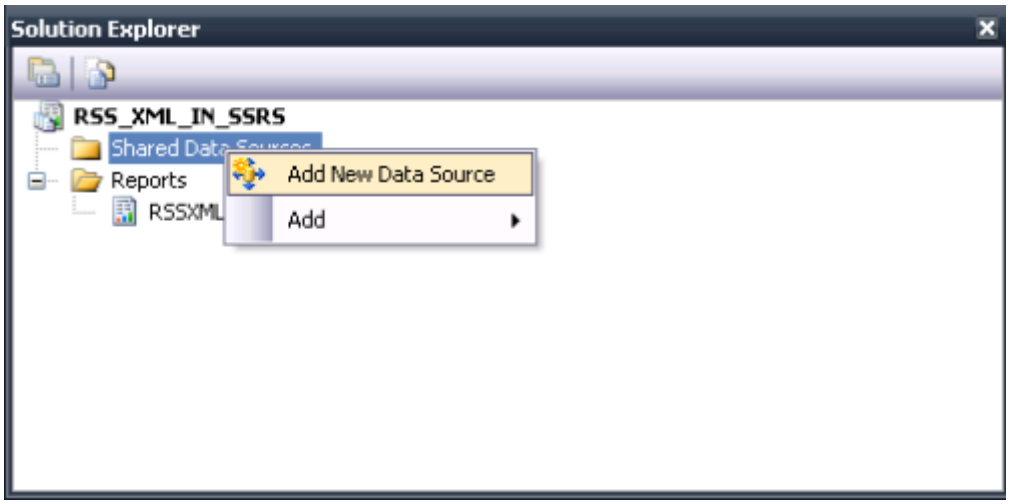
Voyons donc comment créer un rapport simple avec ce Web Service SharePoint.

Nous partirons d'une liste de document vidéo stocké dans un site SharePoint.

Création du rapport avec le Web Service

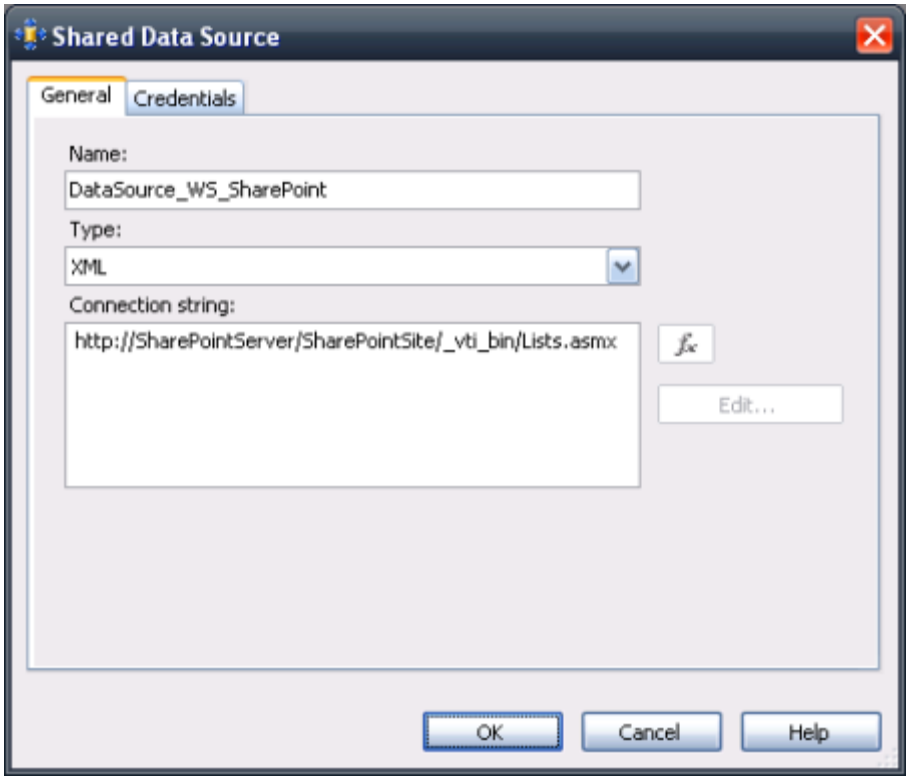
Création de chaine de connexion

Il faut créer une nouvelle source de données dans notre projet Reporting Services

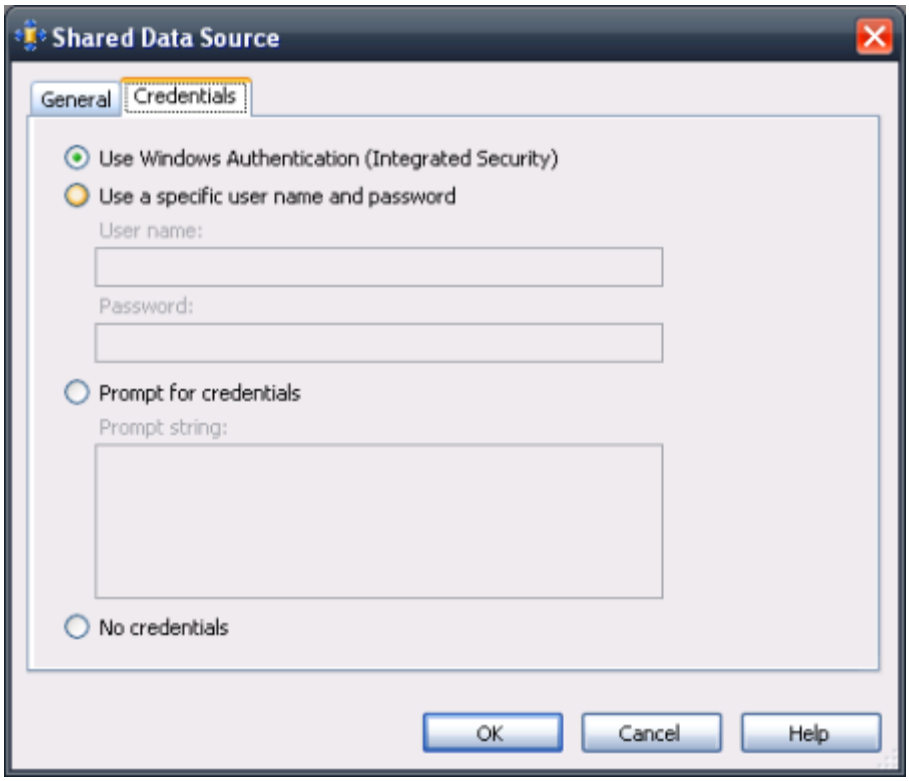


On définit le type XML et ajoutant l'URL vers le Web Service :

- **http://SharePointServer/SharePointSite/_vti_bin/Lists.asmx**

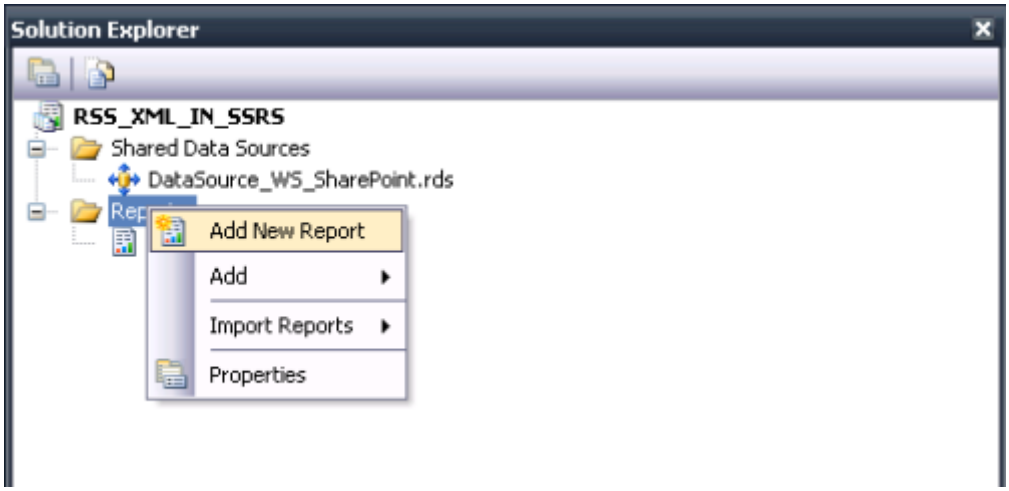


On sélectionne l'authentification intégrée (on peut aussi choisir la connexion anonyme).



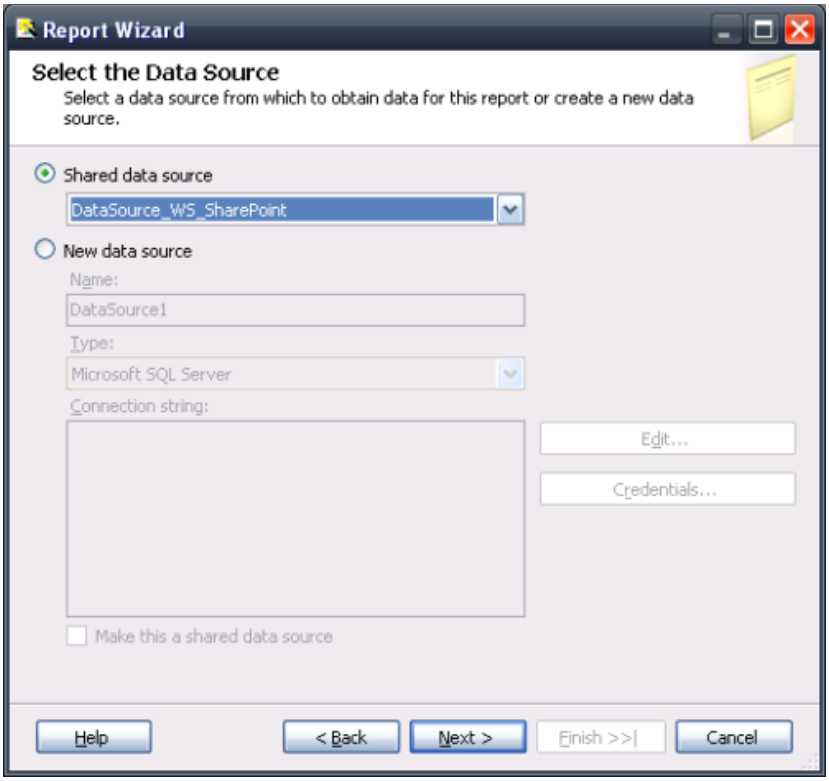
Création du rapport

On crée un nouveau rapport



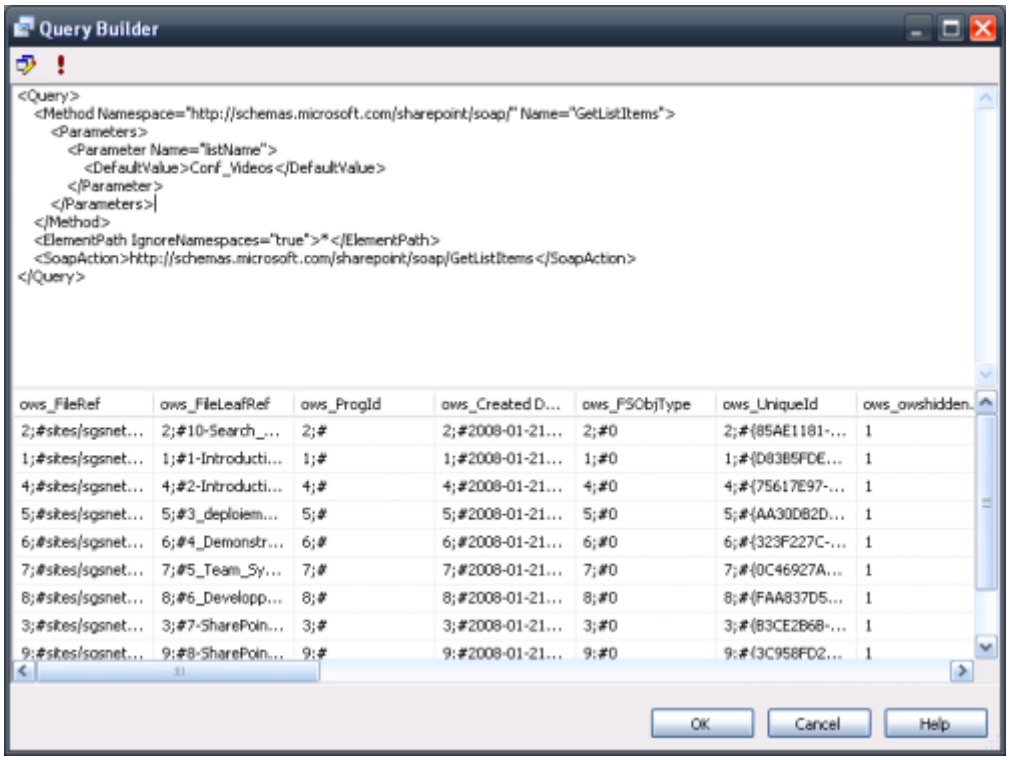
Un assistant se lance afin de nous aider dans la création de notre rapport

On sélectionne notre source de données préalablement créée

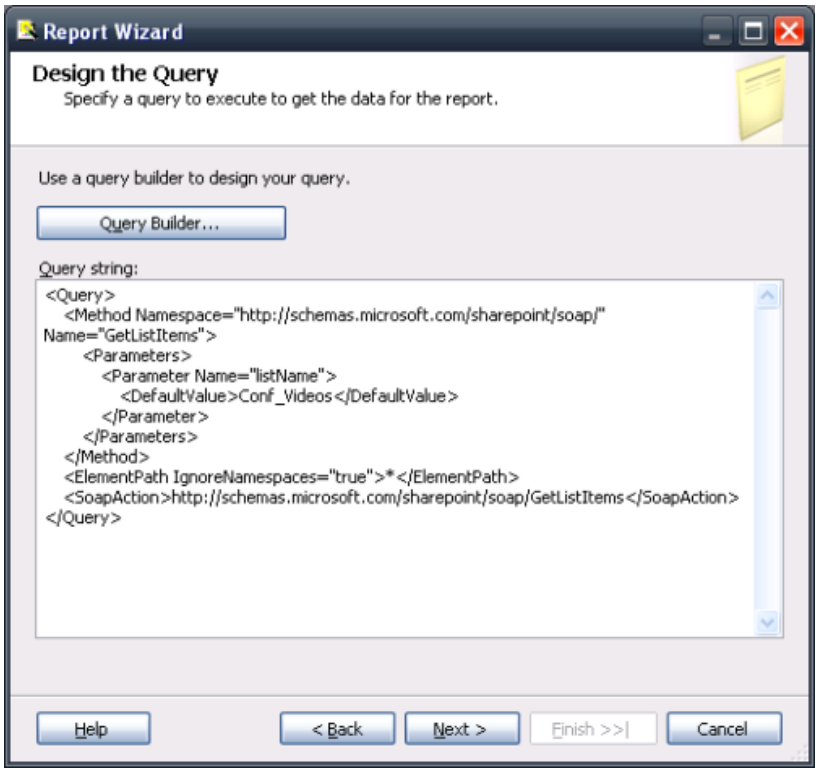


On peut cliquer sur le Query Builder pour nous aider dans la création de notre requête pour le Web Service. On ajoute le code suivant, fournissant le nom de notre liste. On pourrait utiliser les autres paramètres (Nom de la vue, Nom des champs, ...) pour affiner notre choix.

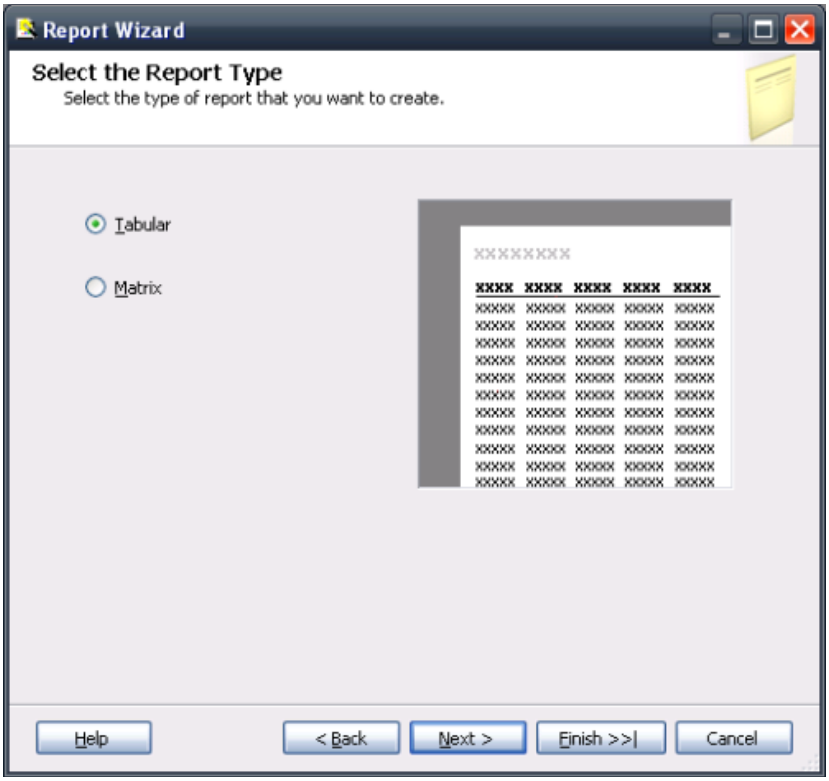
```
<Query>
   <Method Namespace="http://schemas.microsoft.com/sharepoint/soap/" Name="GetListItems">
      <Parameters>
         <Parameter Name="listName">
            <DefaultValue>Conf_Videos</DefaultValue>
         </Parameter>
      </Parameters>
   </Method>
   <ElementPath IgnoreNamespaces="true">*</ElementPath>
<SoapAction>http://schemas.microsoft.com/sharepoint/soap/GetListItems</SoapAction>
</Query>
```



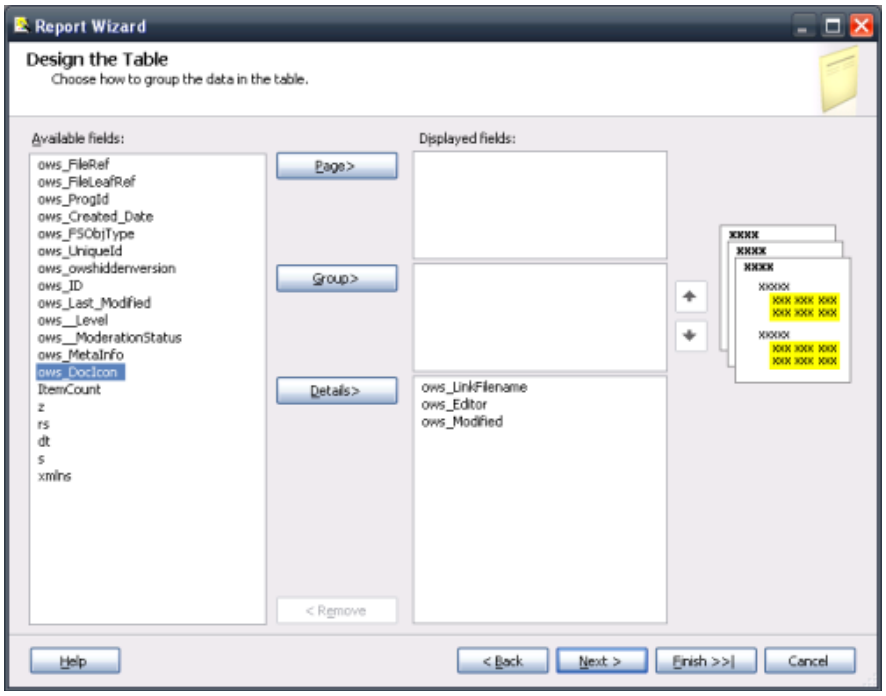
On peut donc cliquer sur OK et suivant



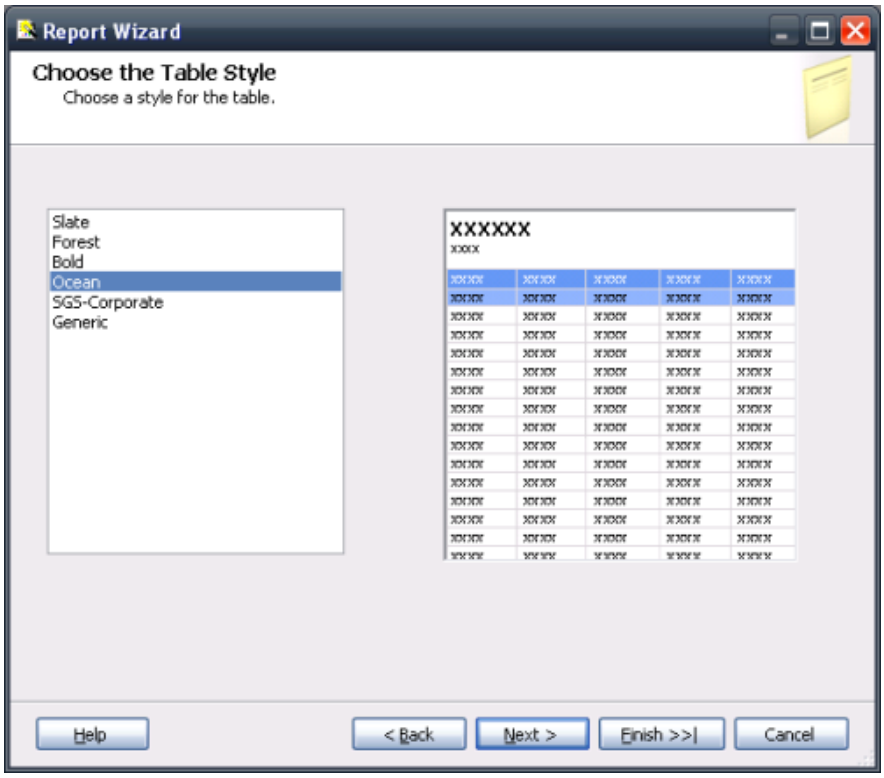
Nous choisirons un mode d'affichage Tabulaire (suffisant pour notre exemple)



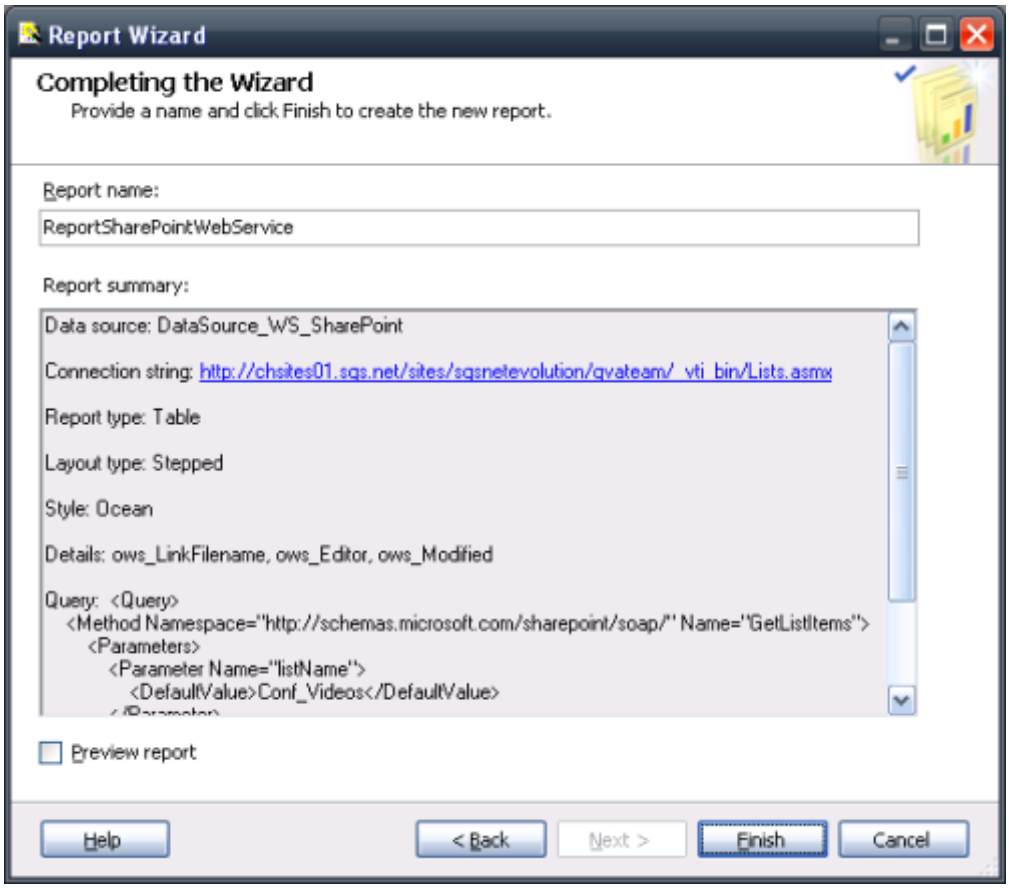
On sélectionne alors les champs que l'on souhaite voir affichés dans le rapport



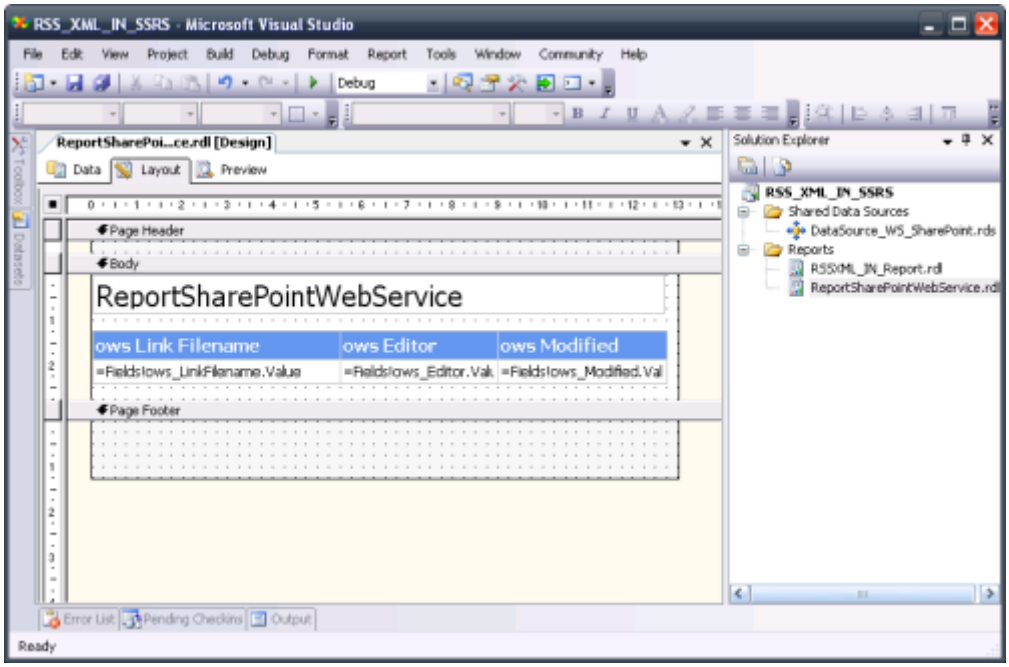
On choisit notre thème pour ce rapport (**Cf article précédent**)



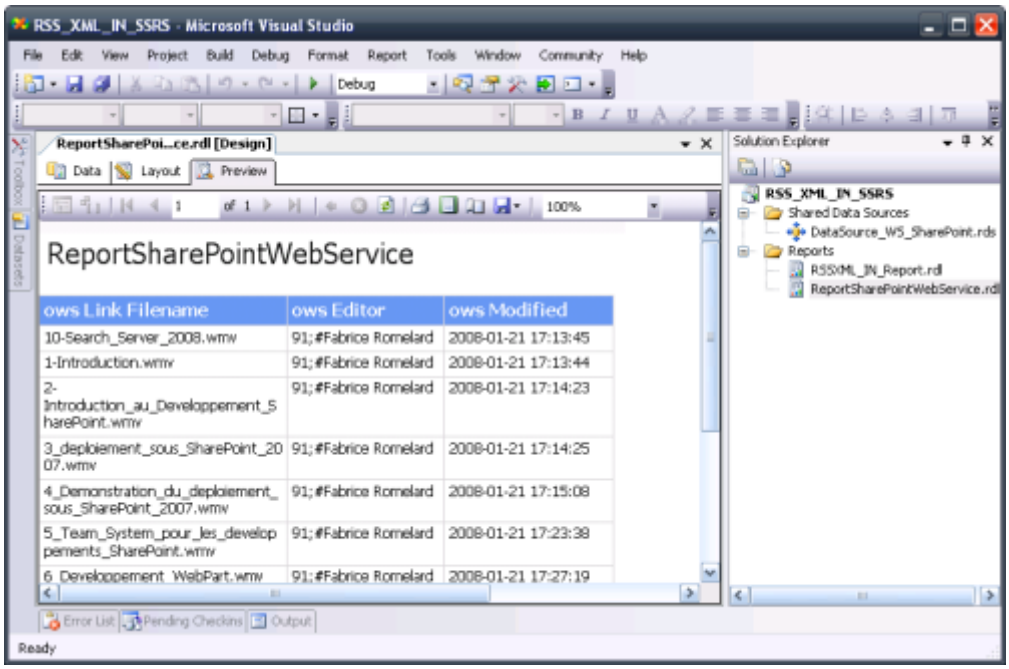
On fournit alors un nom à notre rapport



On obtient alors notre rapport dans Visual Studio .NET



Que l'on peut utiliser pour voir une prévisualisation de ce rapport



On peut ensuite affiner ce rapport en ajoutant des variables utilisant une seconde source de données, ou ajouter des filtres dans la sélection, mais ces options de filtre se déroulent après chargement des données.

Ainsi une liste SharePoint qui contient plusieurs milliers d'Items, enverra à Reporting Services tout le contenu de cette liste avant que celui-ci effectue un filtrage éventuel sur la date.

Attention aussi au nombre d'Items de la liste, le WebService envoi par défaut 1000 Items, ainsi dans le cas d'une liste dépassant cette valeur, il faut ajouter le paramètre :

```
<Parameter Name="rowLimit">
   <DefaultValue>10000</DefaultValue>
</Parameter>
```

Voyons maintenant un connecteur disponible rendant plus simple cette utilisation de SharePoint en source de données.

L'utilisation d'un projet communautaire

Ce projet en question est fourni par :

- **Teun Duynstee**

Il s'agit d'une extension pour Reporting Services (**dont le développement est expliqué ici**) qui utilise justement ce Web Service présenté juste avant :

- **Macaw.ReportingServices.SharepointListExtension**

Vous pouvez obtenir plus de détails, ainsi que les étapes d'installation de ce projet sur le site :

- **Using the SharePoint Lists extension on Reporting Services 2005**

Une fois que vous avez modifié les différents fichiers de configuration :

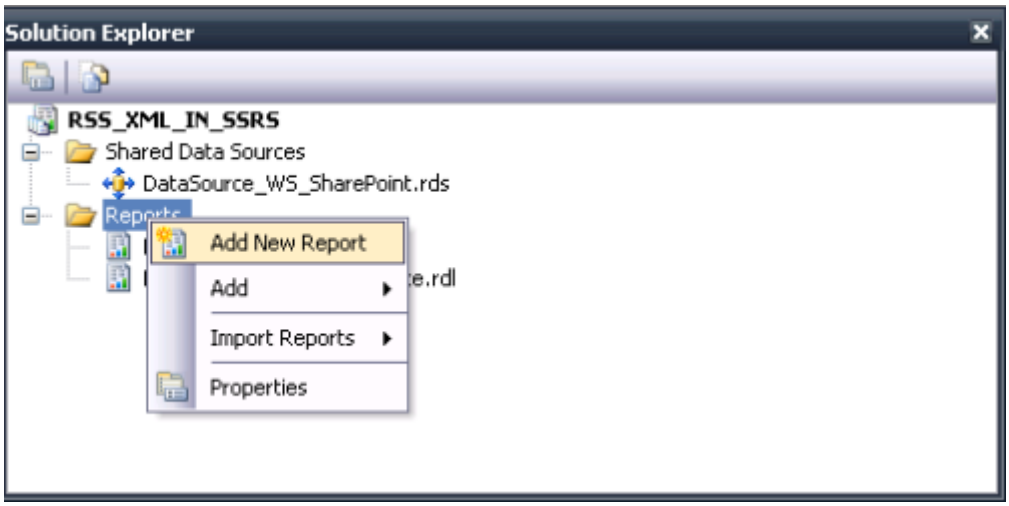
- **RSReportDesigner.config** pour le poste de développement
- **rsreportserver.config** pour le serveur

Vous pouvez créer un rapport très simplement (il n'est en revanche toujours pas possible d''envoyer de XAML au serveur SharePoint pour une requête personnalisée.

Création du rapport

Nous allons créer un rapport avec sa connexion intégrée, on pourrait faire une connexion partagée comme précédemment, mais cela permet de voir les différentes solutions. Nous travaillerons sur la même liste que précédemment.

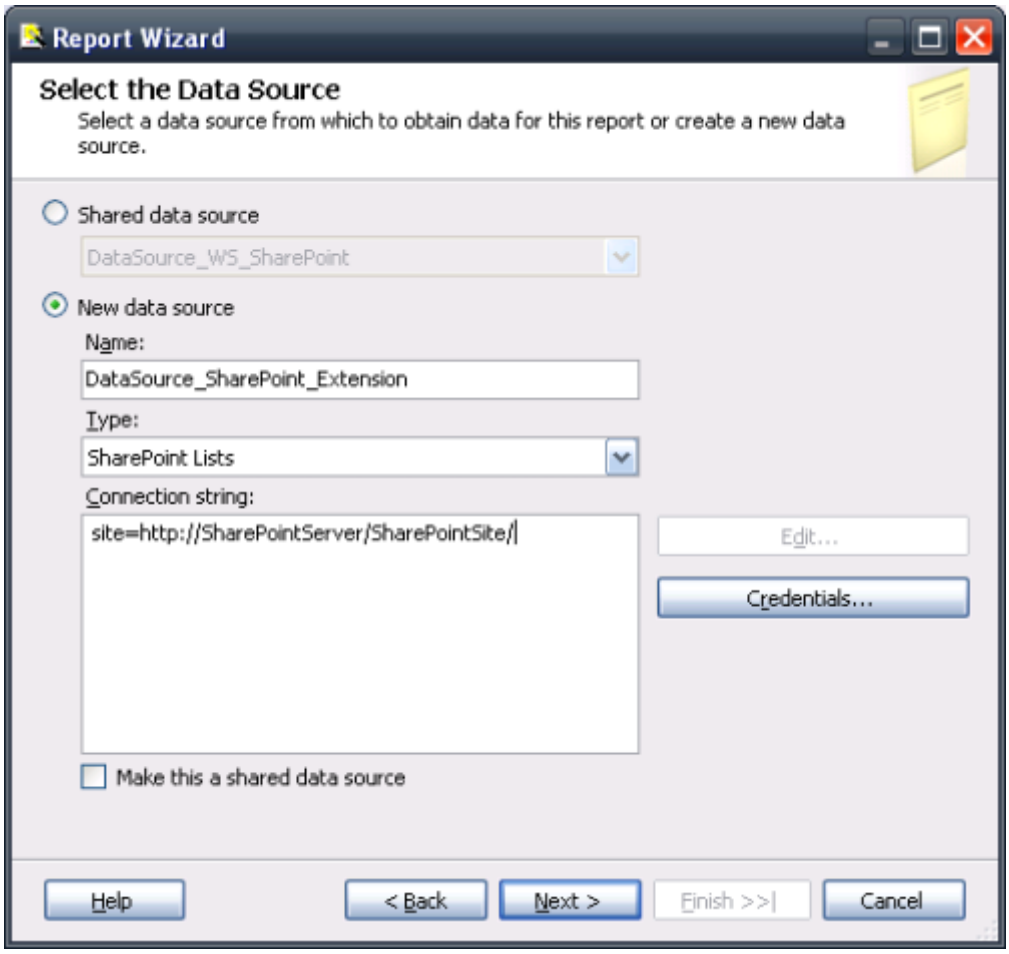
Nous devons créer un nouveau rapport



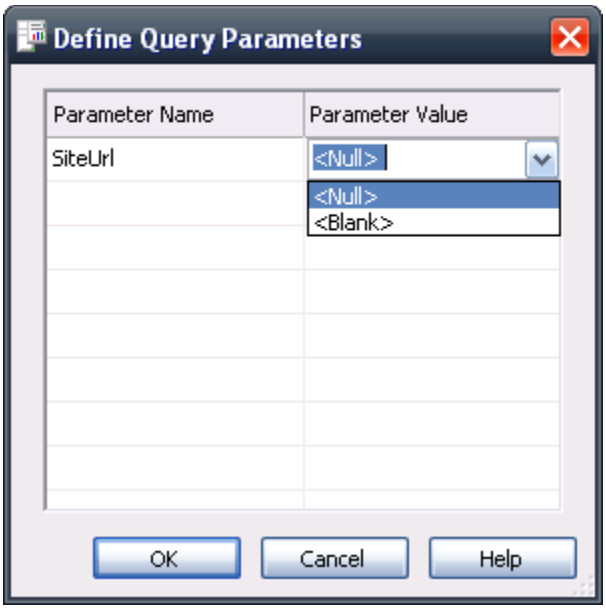
Le même assistant que précédemment se lance, nous cliquons sur Suivant. Nous définit une nouvelle source de données qui utilisera le type "SharePoint Lists", on ajoutera dans la chaine de connexion :

- site=http://SharePointServer/SharePointSite/

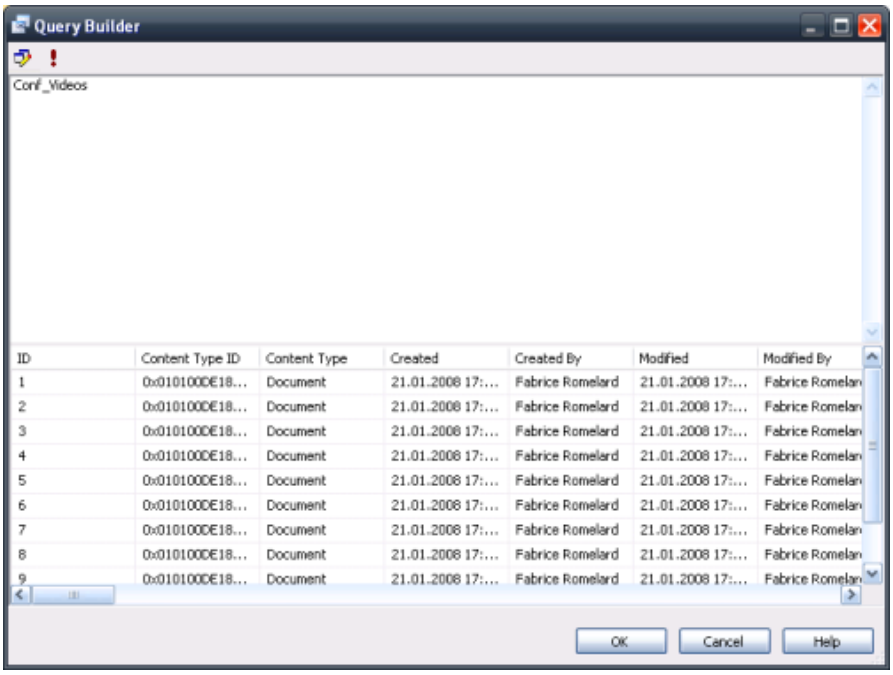
L'authentification est paramétrée comme pour le Web Service (Authentification intégrée ou Anonyme).



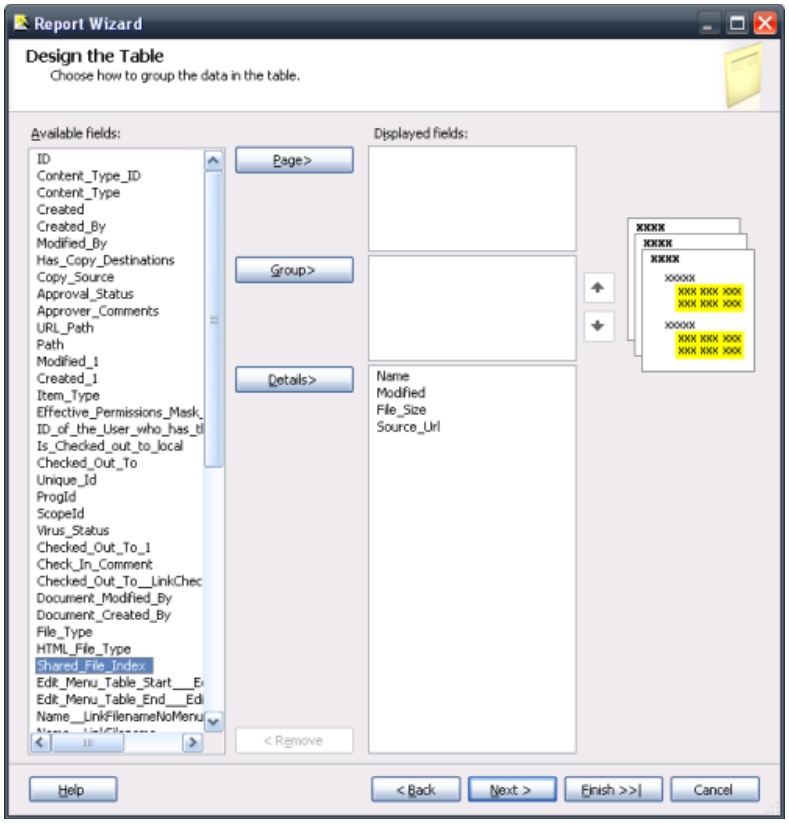
Le Query Builder nous demande simplement le nom de la liste, on peut fournir au paramètre "SiteURL" qui s'affiche, ""



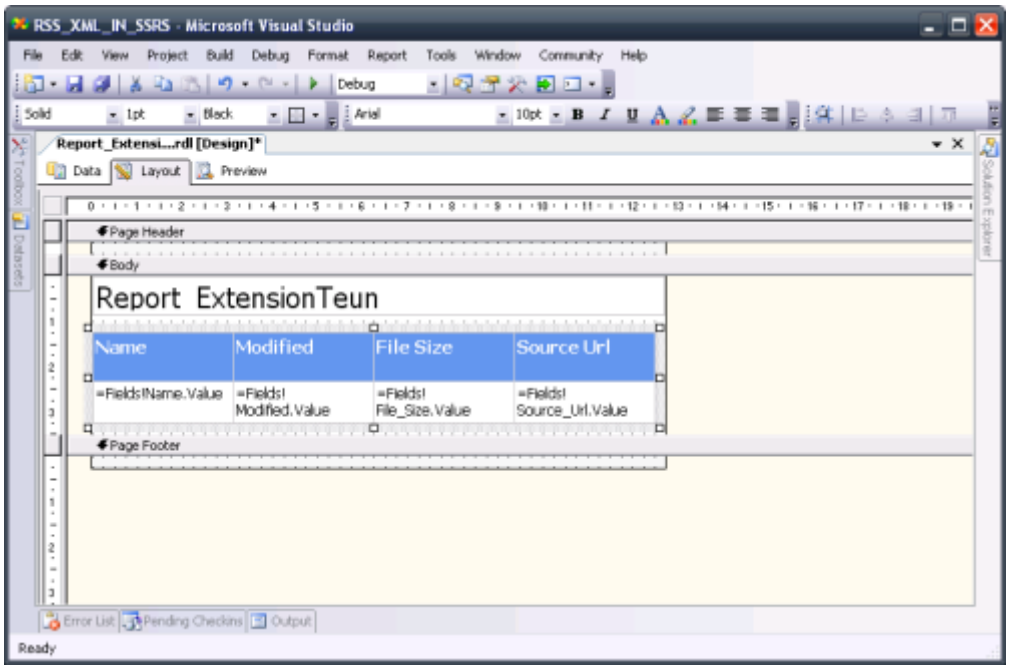
On obtient alors les informations de notre liste



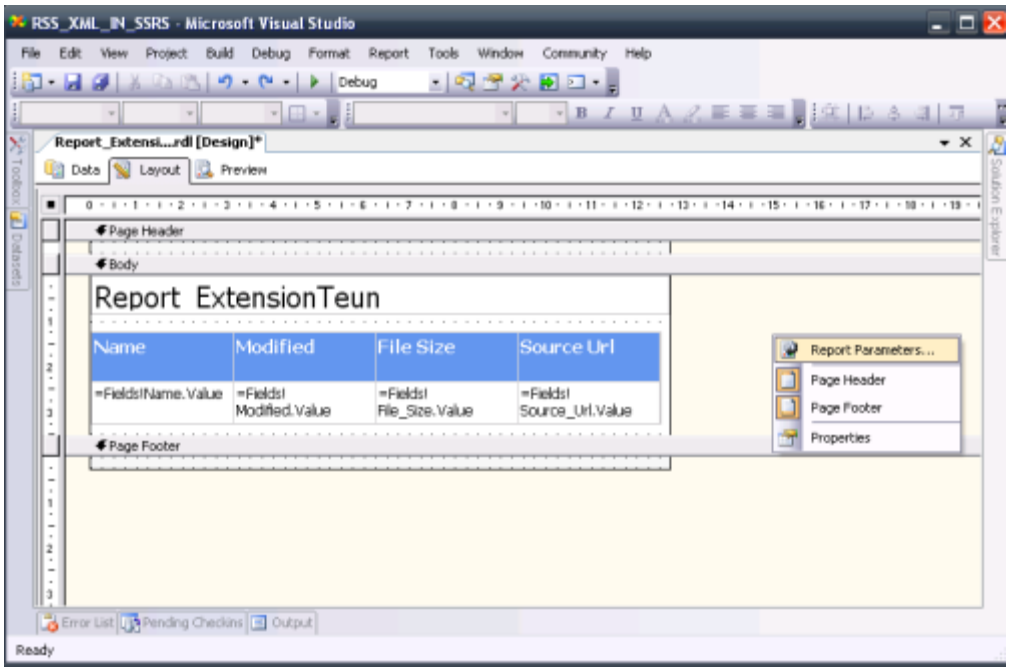
Nous choisirons un rapport tabulaire et les informations à afficher dans notre rapport, on peut d'ailleurs noter que ces informations sont plus précises que lors de l'utilisation du Web Service.



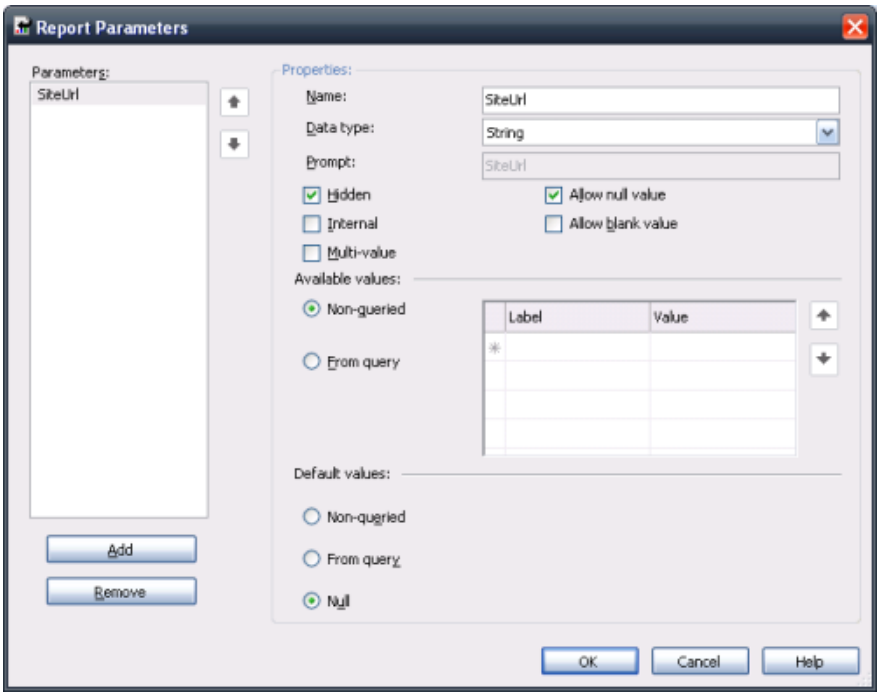
On choisit alors le thème et le nom de notre rapport pour enfin voir notre rapport.



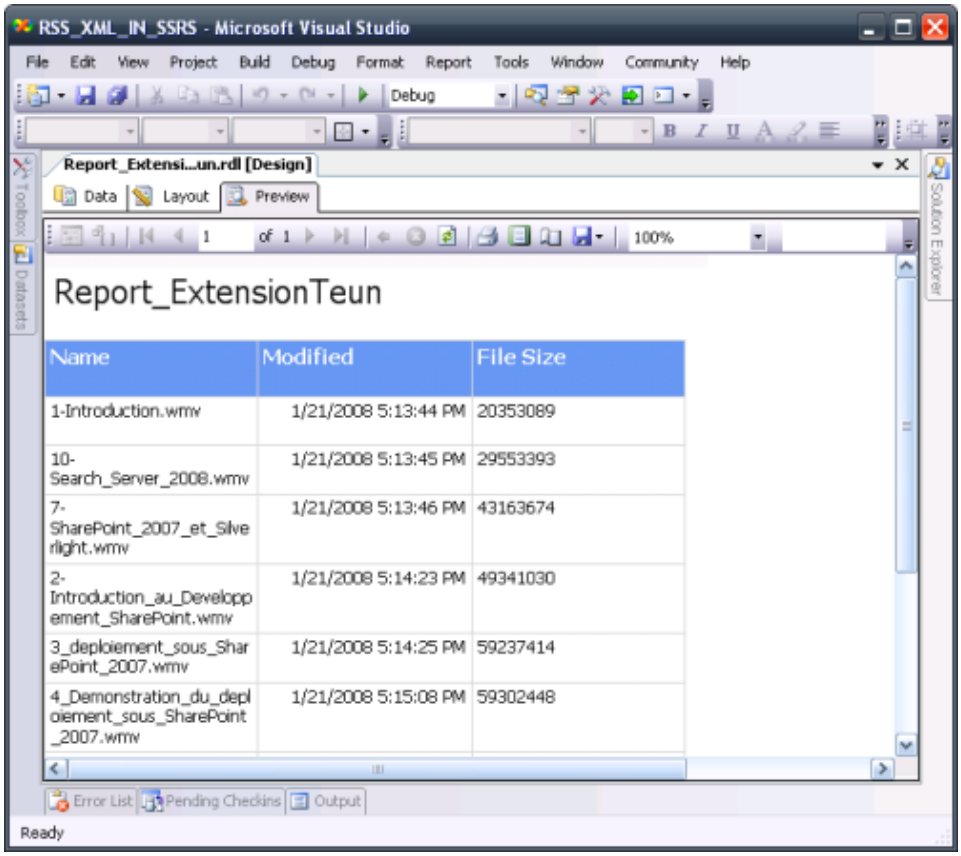
Il nous faut changer le paramètre SiteURL pour le mettre non visible et avec la valeur NULL par défaut. Pour ceci on clique avec le bouton droit sur le rapport et choisit "Report Parameters"



On coche alors les deux premières cases (Hidden et Allow Null value) et on clique sur OK



On peut alors voir la prévisualisation du rapport



Ce connecteur simplifie donc la création de rapport utilisant des données issues de SharePoint, mais elle possède les mêmes inconvénients que le Web Service.

Il faut aussi noter que le nombre d'Items demandé au Web Service est laissé à 1000 (valeur par défaut), si vous souhaitez en recevoir plus, il faut modifier le code de l'extension afin de mettre celle souhaitée.

Voyons maintenant une autre extension disponible.

Enesys RS Data Extension

Enesys propose un connecteur SharePoint pour Reporting Services, avec plusieurs versions possibles :

- **Enesys RS Data Extension**

Vous pouvez télécharger les différentes versions (gratuites ou payantes pour essai) :

- **Enesys RS Data Extension - Downloads**

Le gros avantage de ce connecteur est de supporter les requêtes CAML, comme l'explique la page :

- **Using report parameters with SharePoint lists**

Nous utiliserons la version **Community Edition Version - V1.4.3** pour notre exemple.

Une fois qu'on a téléchargé le fichier ZIP et décompressé celui-ci, on obtient une DLL pour SQL Server Reporting Services 2000 et une autre pour SQL Server Reporting Services 2005. Un fichier PDF et des exemples accompagnent ces fichiers DLL.

Le fichier PDF explique parfaitement comment installer l'extension sur le serveur et sur le poste de développement (exactement la même procédure que pour la version présentée juste avant).

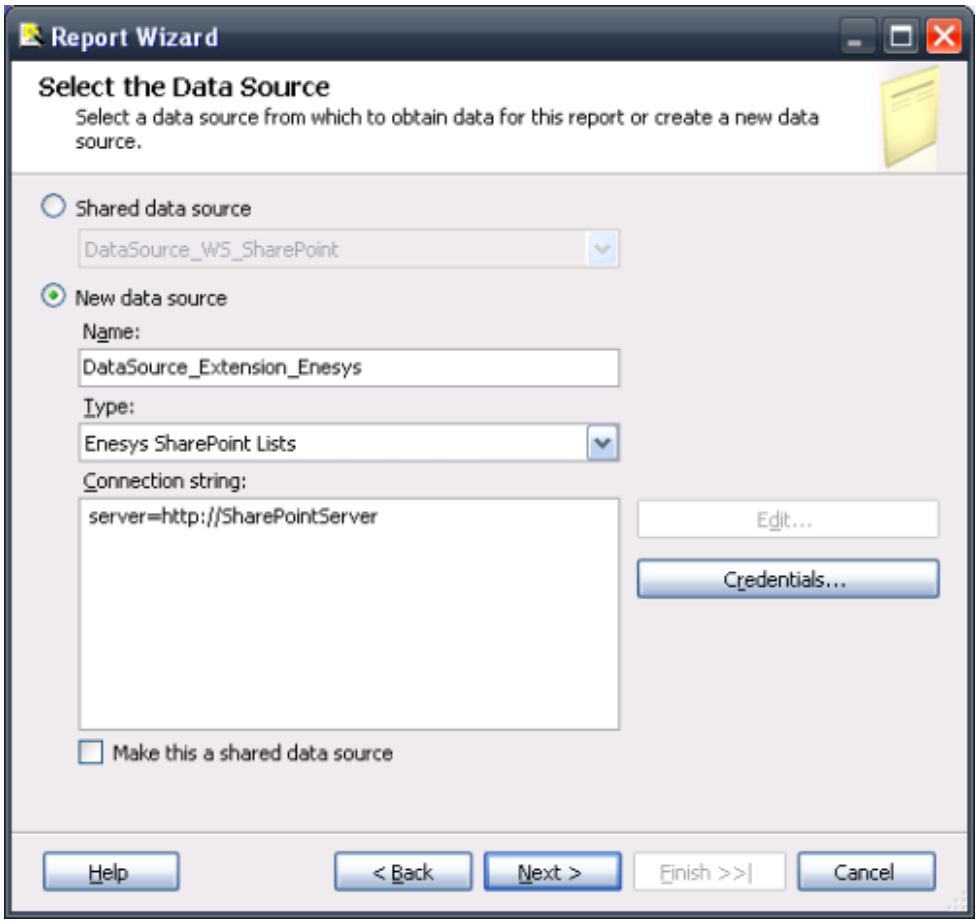
Une fois cette installation effectuée, nous pouvons créer un rapport d'exemple utilisant une requête CAML.

Création du rapport

Nous créons un nouveau rapport comme précédemment avec la connexion intégrée au rapport et pointant toujours sur la même liste SharePoint.

Cette fois on sélectionne le Type "Enesys SharePoint Lists" et on fournit juste l'URL vers notre serveur SharePoint

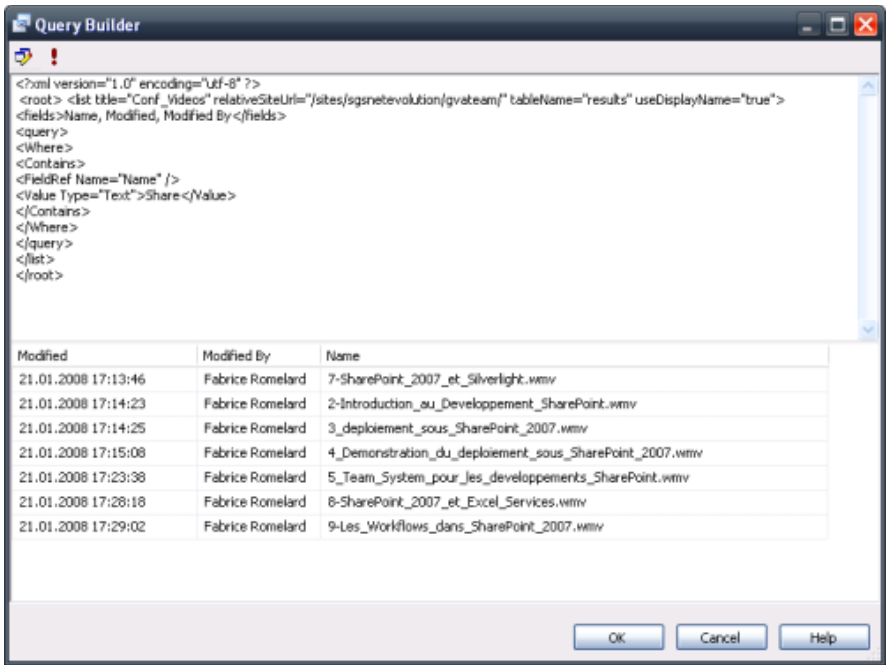
Cette extension supporte tous les types d'authentification (contrairement aux précédents), mais nous conserverons celle intégrée à Windows.



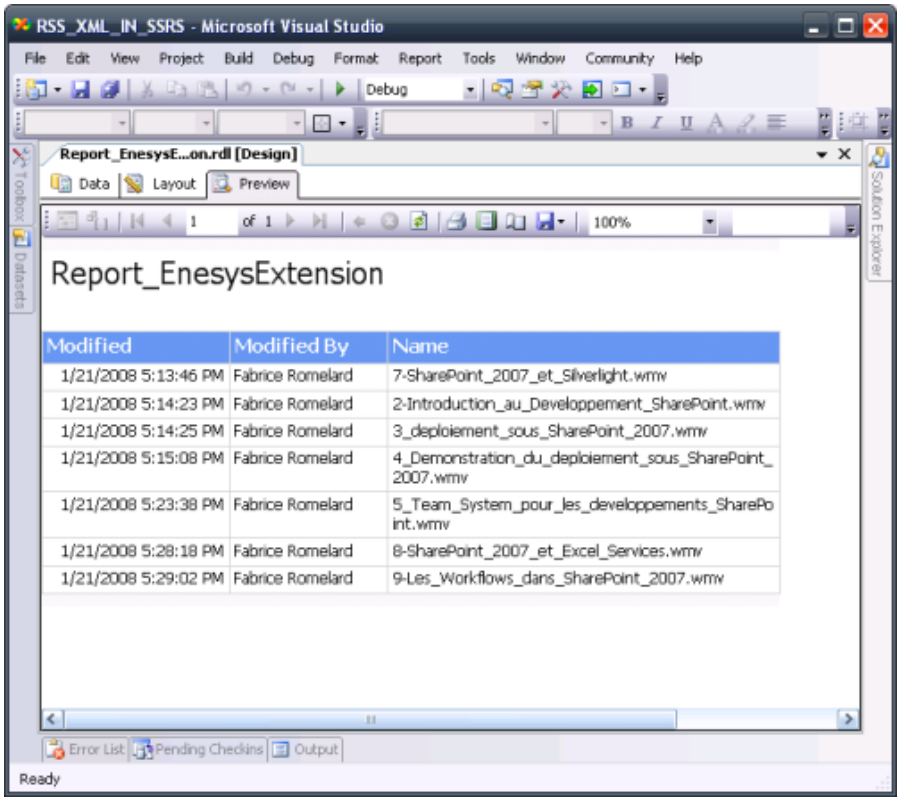
Le Query Builder nous permet de créer une sélection plus précise directement depuis le serveur (afin d'éviter de récupérer des champs inutiles), en fournissant la liste des champs à lister et la requête CAML éventuelle.

Ceci peut donner l'exemple ci-dessous :

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <list title="Conf_Videos" relativeSiteUrl="/SharePointSite/" tableName="results" useDisplayName="true">
    <fields>Name, Modified, Modified By</fields>
    <query>
      <Where>
        <Contains>
          <FieldRef Name="Name" />
          <Value Type="Text">Share</Value>
        </Contains>
      </Where>
    </query>
  </list>
</root>
```



Ce qui nous donne pour la visualisation



Cette extension est très complète (même dans sa version gratuite). Il vous permet aussi de faire des requêtes multi-listes ou des regroupements de listes dans le même rapport.

Je vous invite à consulter la documentation présente sur le site de l'éditeur afin d'in savoir plus.

Conclusion

Nous avons vu au travers de ces différents exemples que SharePoint devient une source de données comme les autres pour Reporting Services.

Nous pouvons dès lors imaginer des solutions qui utilise SharePoint pour le stockage et l'utilisation des données (comme un HelpDesk) et Reporting Services (intégré dans SharePoint) comme outil d'analyse et de visualisation de la performance de notre plateau de support.

Il ne faut malgré tout pas prendre SharePoint pour un DWH, car le système n'est pas du tout conçu pour cela.

Voici quelques liens utiles si cet article vous a intéressé :

- **Microsoft SQL Server : Reporting Services**
- **Connecting SQL Reporting Services to a SharePoint List**
- **Accessing SharePoint List Items with SQL Server 2005 Reporting Services**
- **Didacticiels de Reporting Services**
- **Débuter avec les Reporting Services (FR)**
- **SharePoint 2007 et Reporting Services 2005 : un workaround pour intégrer les données d'une liste dans un rapport (FR)**
- **Reporting Services: Interrogation de listes Sharepoint (FR)**
- **Using SharePoint Web Service from SSRS (US)**
- **Connecting SQL Reporting Services to a SharePoint List (US)**
- **Reporting over SharePoint lists with MS Reporting Services (US)**

En vous souhaitant de bons projets de développement.

Romelard Fabrice (alias F___)
Intranet/Extranet CTO - **SGS**