

Comment intégrer les flux RSS au sein de votre application

par

Date de publication : 04/01/2008

Dernière mise à jour :

Cet article vous permettra d'intégrer l'annuaire RSS de Windows (Windows RSS Platform) au sein de vos applications .Net, ce qui vous permettra de rendre votre application "RSS-aware".

Introduction

1 - Présentation

1.1 - Flux RSS

1.2 - Subscription

1.3 - Common Feed List

2 - Utilisation

2.1 - Plateforme RSS Windows

2.2 - Intégration de la Common Feed List

2.2.1 - Listing des flux existants

2.2.2 - Récupération des billets

2.3 - Modifier la Common Feed List

2.3.1 - Ajout d'un dossier de rangement

2.3.2 - Ajout d'un flux

2.3.3 - Suppression


4 - Pour aller plus loin - les évènements

Conclusion

Liens

Remerciements

Introduction

Il a quelques semaines, Florian Casabianca publiait un excellent article sur la  **manipulation des flux RSS à l'aide du framework 3.5**. Il y explique notamment comment à l'aide de l'objet SyndicationFeed, il est très facile de lire et ou de créer des flux RSS aussi bien au format RSS 2.0 qu'ATOM 1.0 (les standards de RSS les plus utilisés à ce jour). Cet article m'a rappelé une plateforme au sein de Windows, appelée tout simplement "Windows RSS Platform" qui permet, surtout depuis l'arrivée d'Internet Explorer 7 d'aggréger vos flux RSS au sein d'une liste unique appelée Common Feed List. Cette liste est une sorte d'annuaire commun de flux RSS qu'il vous est alors permis de modifier ou d'intégrer directement au sein de votre application.

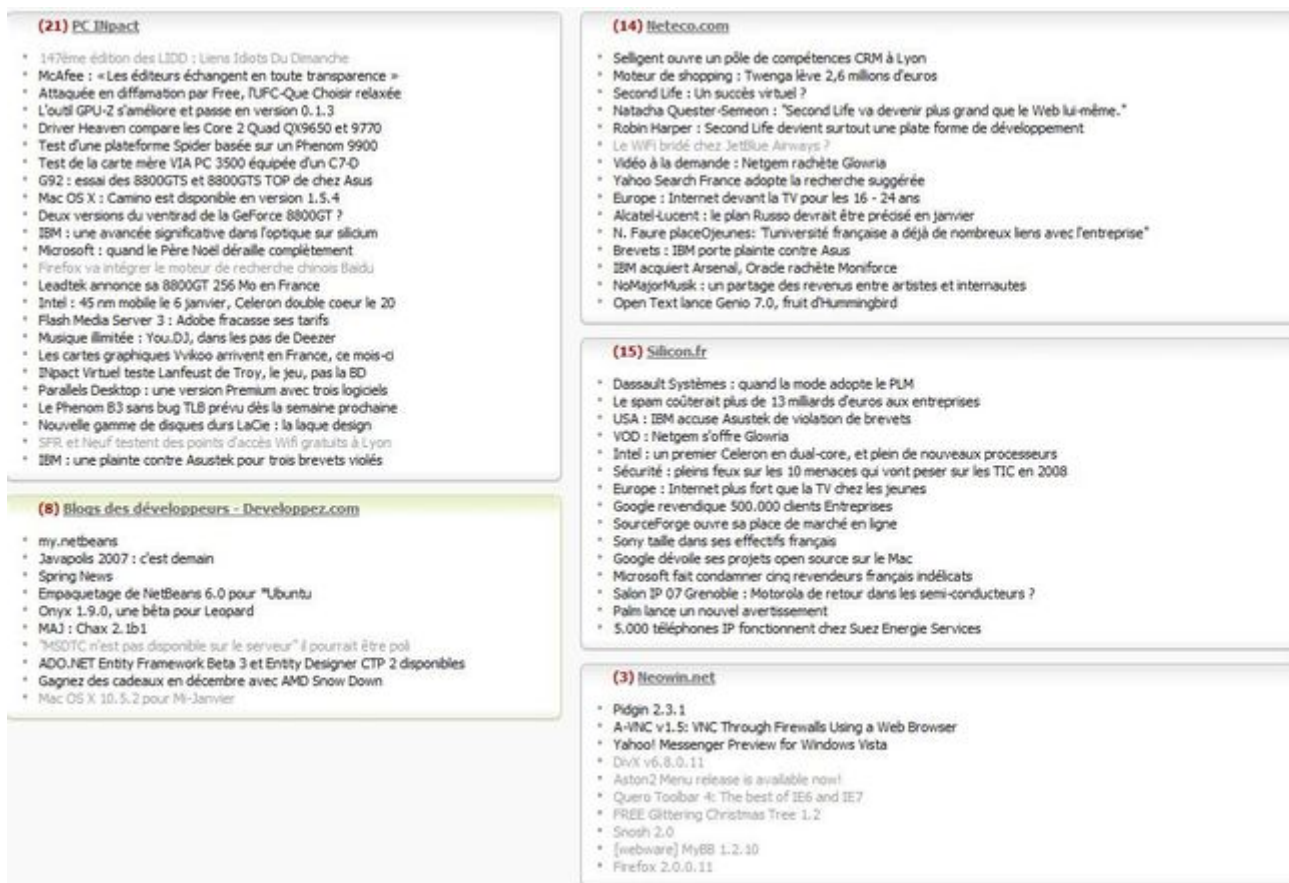
C'est sa manipulation que je vous propose de découvrir au sein de ce tutoriel.

1 - Présentation

1.1 - Flux RSS

RSS ou Really Simple Syndication est un "format de fichier XML dont le contenu est généré dynamiquement depuis une base de données. C'est un format de syndication de contenu. Le sigle peut également signifier : Rich Site Summary ou Real-time Simple Syndication." (Hugo Etiévant - **dico des développeurs**).

En quelques mots plus clairs, il s'agit la plupart du temps de résumés de news, permettant d'un seul coup d'oeil de connaître les "gros titres" ou nouvelles informations d'un site web. Bien souvent, ces flux RSS, présents sur les sites de tout genre, sont générés dynamiquement à partir des dernières nouvelles saisies, et se contentent de lister les titres ainsi que les premiers mots de ces news. Cela permet à l'utilisateur, d'avoir un agrégateur de flux RSS et en une seule page, d'avoir un résumé de plusieurs sites web comme sur la capture d'écran suivante:

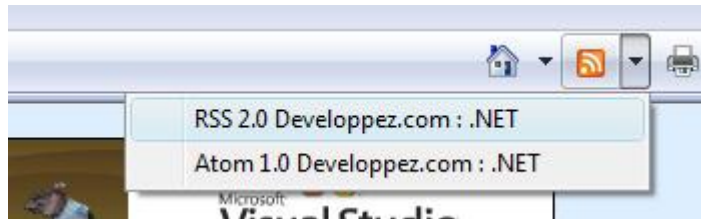


The screenshot displays a grid of RSS feed items. Each item is a small box with a title and a list of news snippets. The items are organized into three columns:

- Column 1:**
 - (21) PC Impact**: 147ème édition des LIDD; McAfee; Attaquée en diffamation par Free; L'outil GPU-Z s'améliore; Driver Heaven compare les Core 2 Quad QX9650 et 9770; Test d'une plateforme Spider basée sur un Phenom 9900; Test de la carte mère VIA PC 3500 équipée d'un C7-D; G92: essai des 8800GTS et 8800GTS TOP de chez Asus; Mac OS X: Camino est disponible en version 1.5.4; Deux versions du ventrad de la GeForce 8800GT?; IBM: une avancée significative dans l'optique sur silicium; Microsoft: quand le Père Noël déraile complètement; Firefox va intégrer le moteur de recherche chinois Baidu; Leadtek annonce sa 8800GT 256 Mo en France; Intel: 45 nm mobile le 6 janvier, Celeron double coeur le 20; Flash Media Server 3: Adobe fracasse ses tarifs; Musique illimitée: You.DJ, dans les pas de Deezer; Les cartes graphiques Viviko arrivent en France, ce mois-ci; Impact Virtuel teste Lanfeust de Troy, le jeu, pas la BD; Parallels Desktop: une version Premium avec trois logiciels; Le Phenom B3 sans bug TLB prévu dès la semaine prochaine; Nouvelle gamme de disques durs LaCie: la laque design; SFR et Neuf testent des points d'accès WiFi gratuits à Lyon; IBM: une plainte contre Asustek pour trois brevets violés.
 - (8) Blogs des développeurs - developpez.com**: my.netbeans; Javapolis 2007: c'est demain; Spring News; Empaquetage de NetBeans 6.0 pour "Ubuntu"; Onyx 1.9.0, une bêta pour Leopard; MAJ: Chax 2.1b1; "MSDTC n'est pas disponible sur le serveur" il pourrait être poli; ADO.NET Entity Framework Beta 3 et Entity Designer CTP 2 disponibles; Gagnez des cadeaux en décembre avec AMD Snow Down; Mac OS X 10.5.2 pour Mi-Janvier.
- Column 2:**
 - (14) Neteco.com**: Selligent ouvre un pôle de compétences CRM à Lyon; Moteur de shopping: Twenga lève 2,6 millions d'euros; Second Life: Un succès virtuel?; Natacha Qvester-Semeon: "Second Life va devenir plus grand que le Web lui-même."; Robin Harper: Second Life devient surtout une plate forme de développement; Le WiFi bridé chez JetBlue Airways?; Vidéo à la demande: Netgem rachète Glowria; Yahoo Search France adopte la recherche suggérée; Europe: Internet devant la TV pour les 16 - 24 ans; Alcatel-Lucent: le plan Russo devrait être précisé en janvier; N. Faure placeOjeunes: l'université française a déjà de nombreux liens avec l'entreprise"; Brevets: IBM porte plainte contre Asus; IBM acquiert Arsenal, Oracle rachète Moniforce; NoMajorMusik: un partage des revenus entre artistes et internautes; Open Text lance Genio 7.0, fruit d'Hummingbird.
 - (15) Silicon.fr**: Dassault Systèmes: quand la mode adopte le PLM; Le spam coûterait plus de 13 milliards d'euros aux entreprises; USA: IBM accuse Asustek de violation de brevets; VOD: Netgem s'offre Glowria; Intel: un premier Celeron en dual-core, et plein de nouveaux processeurs; Sécurité: pleins feux sur les 10 menaces qui vont peser sur les TIC en 2008; Europe: Internet plus fort que la TV chez les jeunes; Google revendique 500.000 clients Entreprises; SourceForge ouvre sa place de marché en ligne; Sony taille dans ses effectifs français; Google dévoile ses projets open source sur le Mac; Microsoft fait condamner cinq revendeurs français indécents; Salon IP 07 Grenoble: Motorola de retour dans les semi-conducteurs?; Palm lance un nouvel avertissement; 5.000 téléphones IP fonctionnent chez Suez Energie Services.
- Column 3:**
 - (3) Icowin.net**: Pidgin 2.3.1; A-VNC v1.5: VNC Through Firewalls Using a Web Browser; Yahoo! Messenger Preview for Windows Vista; DivX v6.8.0.11; Aston2 Menu release is available now!; Quero Toolbar 4: The best of IE6 and IE7; FREE Glittering Christmas Tree 1.2; Snosh 2.0; [webware] MyBB 1.2.10; Firefox 2.0.0.11.

1.2 - Subscription

Ces flux RSS sont normalement disponibles au grand public et peuvent se trouver soit via des annuaires de RSS soit simplement en naviguant sur vos sites préférés. Aujourd'hui, la plupart des navigateurs vous indiquent lorsqu'un flux est disponible. Ainsi, tandis que Mozilla Firefox affichera le logo RSS à droite de la barre d'URL, Internet Explorer affichera un bouton listant les flux RSS disponibles.



En cliquant sur l'un des flux disponibles, vous pourrez alors vous inscrire (c'est gratuit et instantané) au flux RSS. Dans cette page qui s'est ouverte, vous est listé les dernières nouvelles du site en question. Notez que l'affichage que vous avez réalisé par votre navigateur web. Un flux RSS n'est en réalité qu'un simple fichier XML comme l'exemple suivant

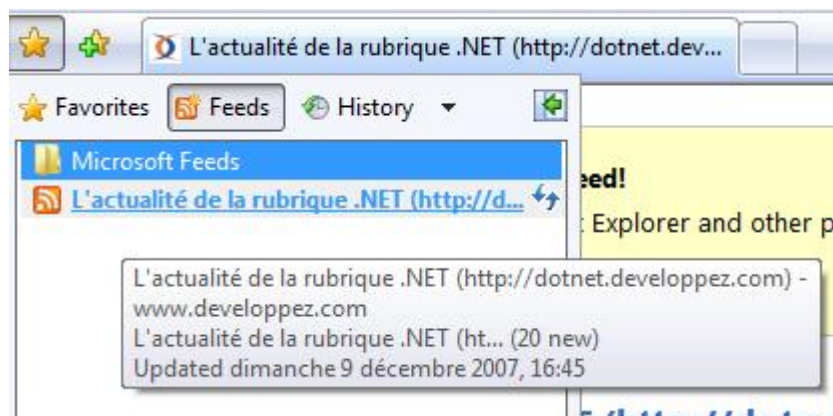
Exemple de flux RSS (format RSS 2.0)

```
<rss version="2.0">
  <channel >
    <title>Titre du flux</title>
    <link>http://monflux.com</link>
    <description>Ceci est un exemple de flux</description>
    <lastBuildDate>Sat, 13 Oct 2007 17:29:38 Z</lastBuildDate>
    <category domain="CategoryScheme">Catégorie du flux</category>
    <item>
      <guid isPermaLink="false">ItemID</guid>
      <link>http://monsie/items</link>
      <title>Titre de l'item</title>
      <description>Contenu de l'item</description>
    </item>
  </channel>
</rss>
```

1.3 - Common Feed List

Où sont donc enregistrés ces flux RSS auxquels vous vous inscrivez? Cela dépend. Certains navigateurs web comme Mozilla Firefox utilisent leur propre moteur interne afin de les afficher comme des favoris dynamiques, tandis qu'Internet Explorer les enregistre dans un annuaire local : la Common Feed List.

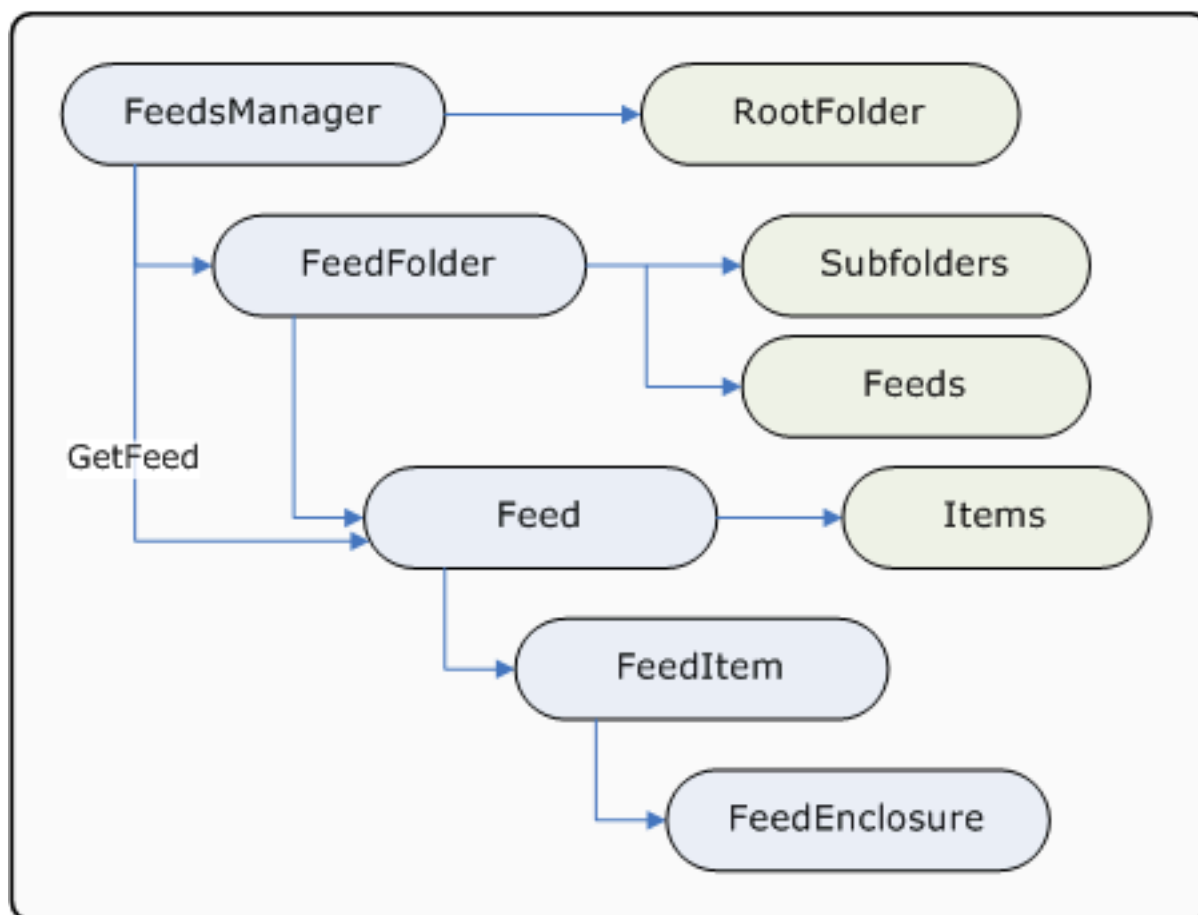
Cette dernière est visible à l'intérieur même d'IE7 (Internet Explorer 7) dans un panel latéral mais peut, et c'est là que réside tout son intérêt, être utilisé à l'intérieur d'autres applications.



2 - Utilisation

2.1 - Plateforme RSS Windows

Commençons par voir quelle est l'architecture de cette "plateforme RSS" et quelles interfaces nous sont proposées pour l'interroger ou la piloter.



Notre objet principal sera donc le **FeedsManager** qui servira de grand conteneur dans lequel tous les autres éléments se trouveront. Dans ce conteneur peuvent déposer des dossiers (**FeedFolder**) ou directement des flux RSS (**Feed**). Normalement, les flux RSS sont censés être rangés dans des dossiers mais si par hasard vous les placez à même le conteneur (**FeedManager**), celui-ci est alors considéré comme le **RootFolder** et donc propose les mêmes méthodes qu'un objet **FeedFolder**.

Enfin, un flux RSS est composé d'un nombre dynamique de **FeedItems** qui correspondent à chaque billet publié dans le fichier RSS distant.

2.2 - Intégration de la Common Feed List

2.2.1 - Listing des flux existants

Nous venons de voir qu'Internet Explorer était l'exemple même de l'intégration de cet annuaire de flux au sein d'une application. Il faut savoir que cette intégration peut se faire sous des dizaines de formes possibles puisque c'est

vous qui choisissez les informations que vous affichez et surtout, comment vous les affichez. Pour ma part, j'ai eu envie d'avoir un panneau latéral listant les flux et les rangeant dans leurs dossiers respectifs, ces dossiers étant représentés par des groupes dans une listView.

L'intégration se fait donc en deux étapes. Tout d'abord, je vais créer les groupes dans ma listView puis je fais lister les flux en les plaçant dans le bon groupe.

Je commence par récupérer une instance de mon FeedManager.

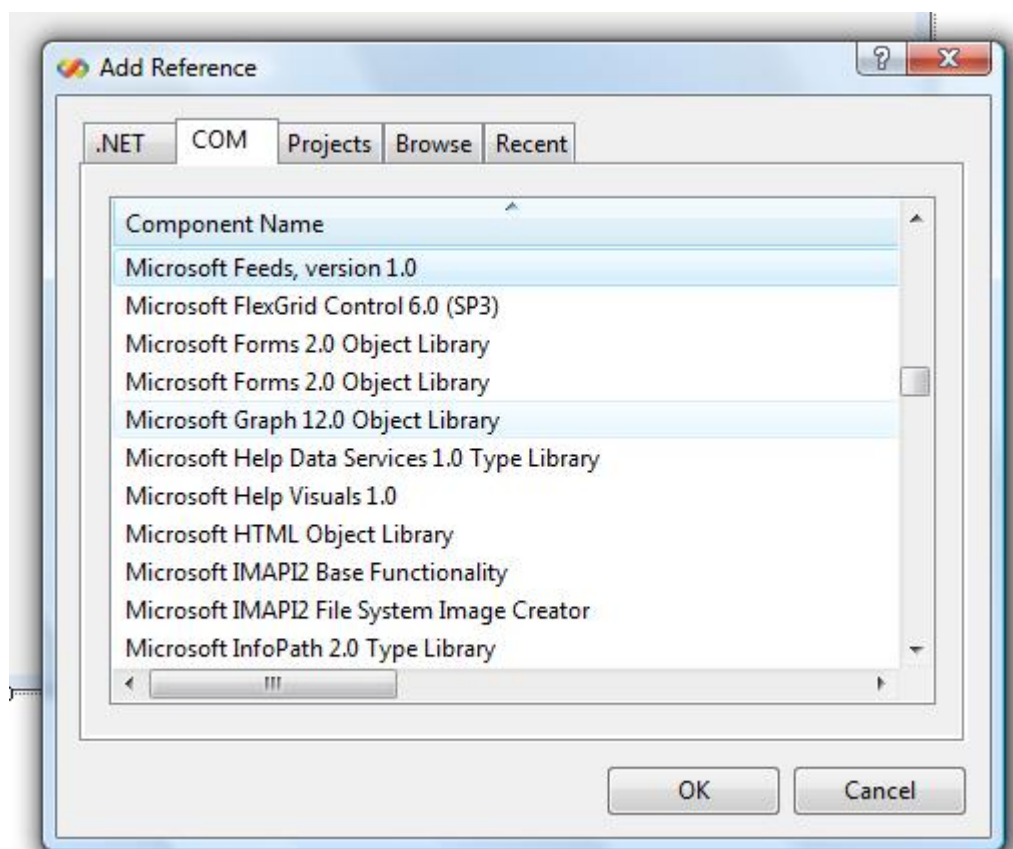
Récupération de la Common Feed List

```
FeedsManager myFMng = new FeedsManager();
```

Petit problème, cet objet FeedManager n'est pas connu par Visual Studio. Il manque donc une référence.

Ajoutez la librairie **Microsoft Feeds** à votre projet puis importez la avec :

```
using Microsoft.Feeds.Interop;
```



2.2.2 - Récupération des billets

Avoir une liste de flux est intéressant mais il peut être plus intéressant d'afficher leur contenu. Ces éléments, des news ou bien des billets de blog, sont constitués de différents éléments en fonction du format du flux RSS. Les plus

communs sont RSS 2.0 et ATOM 1.0 qui, pour chaque élément, spécifient son titre, son auteur et sa description (souvent un résumé). Heureusement pour nous, l'API Microsoft nous fournit un objet `IFeedItem` qui représente chaque élément et qui fournit plusieurs propriétés permettant d'obtenir le nom de l'auteur, le lien vers le billet ou la description d'une news.

Dans notre cas, nous allons, lorsqu'un flux est sélectionné, lister tous ses éléments dans une `listView` puis, lorsqu'un de ces éléments sera sélectionné, son contenu sera affiché dans un `webBrowser` intégré.

```
private void listView1_SelectedIndexChanged(object sender, EventArgs e)
{
    if (listView1.SelectedItems.Count > 0)
    {
        updateFeed(listView1.SelectedItems[0].Text); // methode qui va mettre à jour la liste des
        billets
    }
    else clearFields();
}
```

Et voici la méthode `updateFeed()`. Dans mon exemple, je traite différemment les flux qui sont placés dans un sous-dossier et ceux qui sont à la racine. Selon votre configuration, il sera sûrement plus intéressant de travailler avec le "Path" d'un flux, ce qui correspond à son "dossier/son nom".

Ainsi, après avoir récupéré le dossier dans lequel le flux se trouve, je récupère une référence dans le `FeedManager` puis je vais énumérer ses `Items` en récupérant les informations qui m'intéressent pour construire ma `listView`.

```
/// <summary>
/// R  cup  re la liste des billets
/// </summary>
/// <param name="feed"></param>
private void updateFeed(String feed)
{
    clearFields();
    IFeed myF = null;

    if (listView1.SelectedItems[0].Group.Name != "Autres")
    {
        IFeedFolder curFol =
        (IFeedFolder)myRootFld.GetSubFolder(listView1.SelectedItems[0].Group.Name);
        myF = (IFeed)curFol.GetFeed(feed);
    }
    else if (myRootFld.ExistsFeed(listView1.SelectedItems[0].Text))
    {
        myF = (IFeed)myRootFld.GetFeed(feed);
    }

    if (myF != null)
    {
        foreach (IFeedItem fit in (IFeedsEnum)myF.Items)
        {
            ListViewItem lvi = new ListViewItem(fit.Title);
            lvi.SubItems.Add(fit.Author);
            lvi.SubItems.Add(fit.PubDate.ToShortDateString());
            listView2.Items.Add(lvi);
            listView2.Items[listView2.Items.Count - 1].BackColor =
            (listView2.Items.Count % 2 == 0) ? Color.Gainsboro : Color.White;
        }
    }
}
```

2.3 - Modifier la Common Feed List

2.3.1 - Ajout d'un dossier de rangement

Lorsque vous ajoutez un flux, vous pouvez soit l'ajouter à la racine du FeedManager soit le ranger dans un sous dossier, fonctionnalité très intéressante lorsque vous avez un grand nombre de flux RSS.

Supposons que nous souhaitons ajouter un sous-dossier nommé "test", il nous suffit d'avoir une référence vers le dossier racine de notre FeedManager puis d'appeler la méthode CreateSubFolder :

Création d'un sous-dossier

```
myRootFld.CreateSubfolder("test");
```

2.3.2 - Ajout d'un flux


Nous avons vu comment intégrer la Common Feed List au sein d'une application mais nous allons maintenant chercher à la modifier. Il faut savoir que si vous la modifiez dans votre application, elle sera également modifiée dans toute application qui l'utilise (Internet Explorer 7 par exemple).

Comme première modification, nous allons donc ajouter notre propre flux RSS. Ici, je souhaiterais ajouter le flux RSS de la rubrique Dotnet dont l'URL est <http://dotnet.developpez.com/rss.php>. Je vais appeler mon flux "rubrique dotnet" afin de le reconnaître facilement.

Pour cela, j'utilise la méthode CreateFeed();

Ajout d'un flux

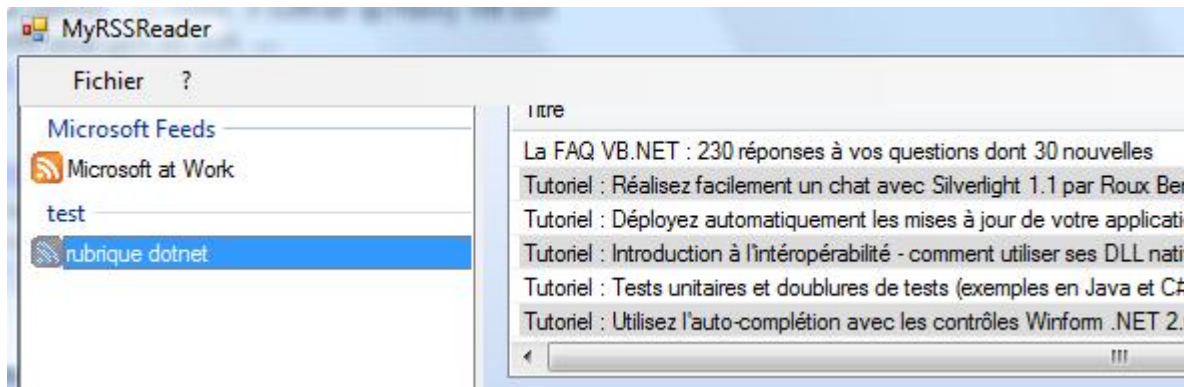
```
if (!myFMng.IsSubscribed("http://dotnet.developpez.com/rss.php"))  
{  
    (myRootFld.CreateFeed("rubrique Dotnet", "http://dotnet.developpez.com/rss.php"));  
}
```

 *je vérifie si je ne suis pas déjà inscrit sur cette adresse afin de ne pas avoir de flux en double.*

Dans le cas précédent, nous avons ajouté un flux à la racine de notre FeedManager mais maintenant que nous savons comment créer des sous-dossiers, autant y placer directement notre flux. Il nous suffit simplement de récupérer une référence vers le sous-dossier désiré plutôt que le dossier racine

```
if (!myFMng.IsSubscribed(txtUrl.Text))  
{  
    ((IFeedFolder)myRootFld.GetSubfolder("test")).CreateFeed("rubrique Dotnet",  
        "http://dotnet.developpez.com/rss.php");  
}
```

Il ne nous reste plus qu'à vérifier que notre flux se trouve effectivement dans notre sous-dossier.



2.3.3 - Suppression

Il est bien sûr possible de supprimer un flux. Pour cela, il suffit d'obtenir l'objet IFeed souhaité et d'appeler la méthode Delete().

Dans l'exemple suivant, je récupère l'objet grâce à la méthode GetFeed() et le nom du feed qui est affiché dans ma listView puis je le supprime.

```
((IFeed)myRootFld.GetFeed(listView1.SelectedItems[0].Text)).Delete();
```

Pour faire plus propre, et éviter d'avoir une listView désynchronisée du vrai FeedManager, il aurait été mieux de créer une classe personnelle ayant une propriété Nom (type string) et une autre de type IFeed et nous aurions binder cette liste directement sur une combobox ou listbox.

Cela permet ainsi de récupérer directement l'objet IFeed désiré. Je n'ai pas fait cela car il m'aurait fallu créer un contrôle dérivant de ListView pour gérer son DataBinding. Avec une combobox, cela ressemble à ceci :

```
List<maClass> maList = new List<maClass>();
foreach(IFeed feed in root.Feeds)
{
    maList.Add(new maClass(feed.Name, feed));
}
maCombobox.DataSource = maList;
maCombobox.DisplayMember = "Nom";
maCombobox.ValueMember = "Feed";
maCombobox.DataBind();
...
public class maClass()
{
    // les getter sont enlevés pour une meilleure lisibilité
    public string Nom;
    public IFeed Feed;
    public maClass(string nom, string feed)
    {
        this.Nom = nom;
        this.Feed = feed;
    }
}
```

4 - Pour aller plus loin - les évènements

Il existe des évènements que l'on peut gérer directement dans l'application. Quels sont ces évènements allez vous me dire, et bien il s'agit d'informations sur les flux, comme quand par exemple, un flux est mis à jour. Ainsi, dans l'application de demo, lorsque vous cliquez sur un flux, ses données sont listées mais si le flux a été supprimé en dehors de votre application, vous allez avoir une belle exception. La solution est alors de détecter quand celui-ci est supprimé et de le supprimer de votre côté (en rafraichissant la liste dans mon cas).

Reprenons un bout de code pour avoir une instance de notre FeedManager :

```
FeedsManager manager = new FeedsManager();  
IFeedFolder root = (IFeedFolder)manager.RootFolder;
```

Puis nous créons notre watcher qui va "écouter" les évènements. Une fois de plus un cast explicite est nécessaire :


```
IFeedFolderEvents_Event watcher;  
watcher = (IFeedFolderEvents_Event) root.GetWatcher(FEEDS_EVENTS_SCOPE.FES_ALL,  
FEEDS_EVENTS_MASK.FEM_FOLDEREVENTS);
```

Il ne nous reste plus qu'à mettre l'évènement que l'on souhaite.

```
watcher.FeedDeleted += watcher_FeedDeleted;
```

Ainsi que la méthode appelée qui recharge notre liste quand un flux est supprimé :

```
public void watcher_FeedDeleted(string path)  
{  
    LoadExistingFeeds();//recharge la liste des feeds et les dossiers  
}
```

Les évènements sont nombreux, il y en a pour les flux mais également pour les dossiers. Que cela soit du renommage, du téléchargement, de la mise à jour, suppression, ajout ou autre, vous pouvez être tenu au courant de quasiment tout ce qui est possible. Vous trouverez la liste des évènements possibles à  [cette URL](#)

Conclusion

Nous avons découvert tout au long de cet article qu'il existait un annuaire commun de flux RSS sur Windows et qu'il était possible de l'intégrer ou de le manipuler au sein d'une application .Net.

Les utilisations possibles sont nombreuses, allant du simple agrégateur RSS à la fonctionnalité qui surveille un flux bien précis pour être au courant d'une mise à jour logiciel et pourquoi pas télécharger automatiquement la nouvelle version. Il existe encore plein d'autres possibilités et je vous laisse donc faire bonne usage de cette technologie.

Liens

 **Manipuler les flux RSS 2.0 et Atom 1.0 avec le Framework 3.5** par Florian Casabianca

 **Windows RSS Platform (MSDN)**

sources du projet de démonstration utilisé au sein de cet article

Remerciements

Je tiens à remercier **Fleur-Anne Blain** pour ses conseils pour l'amélioration de cet article.

