

Création d'un gadget pour la Windows Sidebar

par MORAND Louis-Guillaume ([Page perso](#))

Date de publication : 11/09/06

Dernière mise à jour :

Article présentant les Gadgets de la Windows Sidebar de Windows Vista, ainsi que l'explication de leur développement.

- 1 - Introduction
- 2 - Principe du gadget
- 3 - Mon premier gadget
 - 3-1 - Explication
 - 3-2 - Développement
 - 3-3 - Installation
- 4 - Création avancée
 - 4.1 - Les options
 - 4.2 - Le design
 - 4.3 - Le code
 - 4.4 - Finition
- 5 - Création encore plus avancée
 - 5.1 - La localisation
 - 5.2 - L'effet dockable
- 6 - Conclusion
- 7 - Liens et téléchargements
- 8 - Remerciements

1 - Introduction

Depuis maintenant quelques années, et avouons-le copiés sur Mac OS, les compléments de bureau se développent dans tous les sens. On peut voir des docks, des menus complémentaires, ou encore des Widgets, sortes de petits éléments à fonctions bien précises que l'on place où l'on veut sur le bureau.

Force est de constater que ce genre de produits complémentaires sont utiles et très demandés par les utilisateurs, Microsoft a décidé d'intégrer les Widgets à son futur système d'exploitation: Windows Vista.

Microsoft a apporté sa petite touche à l'édifice en développant la Windows Sidebar, une barre se trouvant sur le côté de l'écran, accrochée au bureau. C'est sur cette Sidebar que peuvent déposés les Widgets nommés pour l'occasion: les Gadgets. Ces gadgets peuvent être placés aussi bien sur la Sidebar dans une forme réduite (nous l'appellerons *dockée*), que sur le bureau dans leur version complète (*non dockée*).

Voici ce à quoi peut ressembler votre Sidebar:



Cliquez pour voir en taille réelle

Bien que différents gadgets déjà tout faits puissent être trouvés sur Internet, nous allons nous intéresser à la création de notre propre gadget.

2 - Principe du gadget

Qu'est qu'un gadget? Quelle technologie cela utilise? Quels outils nous faut-il pour en développer un? Toutes ces questions qui, je suis sûr, trottent dans votre tête, et auxquelles je vais tâcher de répondre.

Tout d'abord, un Gadget est un fichier ayant pour extension `.gadget`. Il s'agit d'un fichier compressé ZIP dont l'extension a été changée. Il contient différents éléments qui, suivant des règles strictes, permettent de former un Gadget fonctionnel.

Niveau technologie, le Gadget utilise l'HTML, les css, le JavaScript mais peut également utiliser du Flash ou de l'ActiveX.

Quant aux outils, pour réaliser cet article, je vous demanderai de vous munir d'une souris, d'un clavier et de n'importe quel éditeur de texte: Notepad faisant très largement l'affaire. Oui, il ne faut rien de plus.

Passons à la structure du Gadget. Celui-ci est constitué d'un certain nombre d'éléments dont certains sont obligatoires et d'autres non. Voici un tableau détaillant rapidement les éléments possibles:

Nom	Nom du fichier	Description
Manifest	<code>gadget.xml</code>	Contient les informations sur le Gadget.(nom, author, copyright, chemin vers les icônes, etc.)
Fichier html	<code>XXX.html</code>	Contient le code principal du Gadget servant principalement à définir l'affichage du Gadget
Fichier de paramètres	<code>Settings.html</code>	Permet de contenir les paramètres de l'utilisateur (si votre gadget a des options)
Fichiers images	<code>XXX.png</code>	Icone représentant le Gadget dans la boîte à gadgets
Fichiers ressources	<code>*.html; *.js, etc.</code>	Différentes ressources pouvant être utilisées par le Gadget

3 - Mon premier gadget

3-1 - Explication

Avant de créer un vrai Gadget, nous allons créer un Gadget tout ce qu'il y a de plus simple et qui affichera un simple message. Lors du développement de ce Gadget, nous verrons un à un les différents éléments noyaux d'un Gadget. Dans un second chapitre, nous ferons évoluer ce Gadget pour le rendre plus interactif.

Passons aux choses sérieuses...

3-2 - Développement

Créez n'importe où un dossier que vous nommerez comme bon vous semble.

Entrez alors dans ce dossier et créez-y un nouveau fichier texte que vous nommerez gadget.xml, puis ouvrez ce fichier avec Notepad. Collez-y tel quel le code suivant

gadget.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<gadget>
  <name>Mon premier gadget</name>
  <namespace>Developpez</namespace>
  <version>1.0</version>
  <author name="pharaonix">
    <info url="www.developpez.com" />
    <logo src="logo.png" />
  </author>
  <copyright>2006</copyright>
  <description>C'est mon premier gadget qu'il est tout beau tout frais ;o</description>
  <icons>
    <icon height="48" width="48" src="pharaonix.png" />
  </icons>
  <hosts>
    <host name="sidebar">
      <base type="HTML" apiVersion="1.0.0" src="dvp.html" />
      <permissions>Full</permissions>
      <platform minPlatformVersion="0.3" />
    </host>
  </hosts>
  <version value="1.0.0.0" MinPlatformVersion="0.1" />
</gadget>
```

Expliquons-le maintenant. Nous remarquons tout d'abord la balise racine <gadget> dans laquelle se trouveront tous les éléments de notre Gadget.

Ainsi nous avons tout d'abord la balise <name> qui contient le nom de l'auteur

La balise <namespace> qui contient généralement le nom de l'entreprise. Balise utilisée dans le cas où vous créez plusieurs gadgets

La balise <version> qui contient le numéro de version du gadget

La balise <author> qui contient les informations sur l'auteur. Son nom (attribut name), son site web (balise <info>, attribut url) et son logo (balise <logo>, attribut src)

La balise <copyright> qui contient l'année du copyright

La balise <description> qui comme son nom l'indique contient une phrase décrivant les fonctions de votre gadget.

Nous avons ensuite la balise <icones> qui nous sert à définir l'icône qui représentera notre Gadget dans la boîte à gadgets.

La dernière partie est configurée pour signaler que ce sera un gadget Windows (et non Web). La seule chose à remarquer est l'attribut **src** de la balise base qui contient **dvp.html**

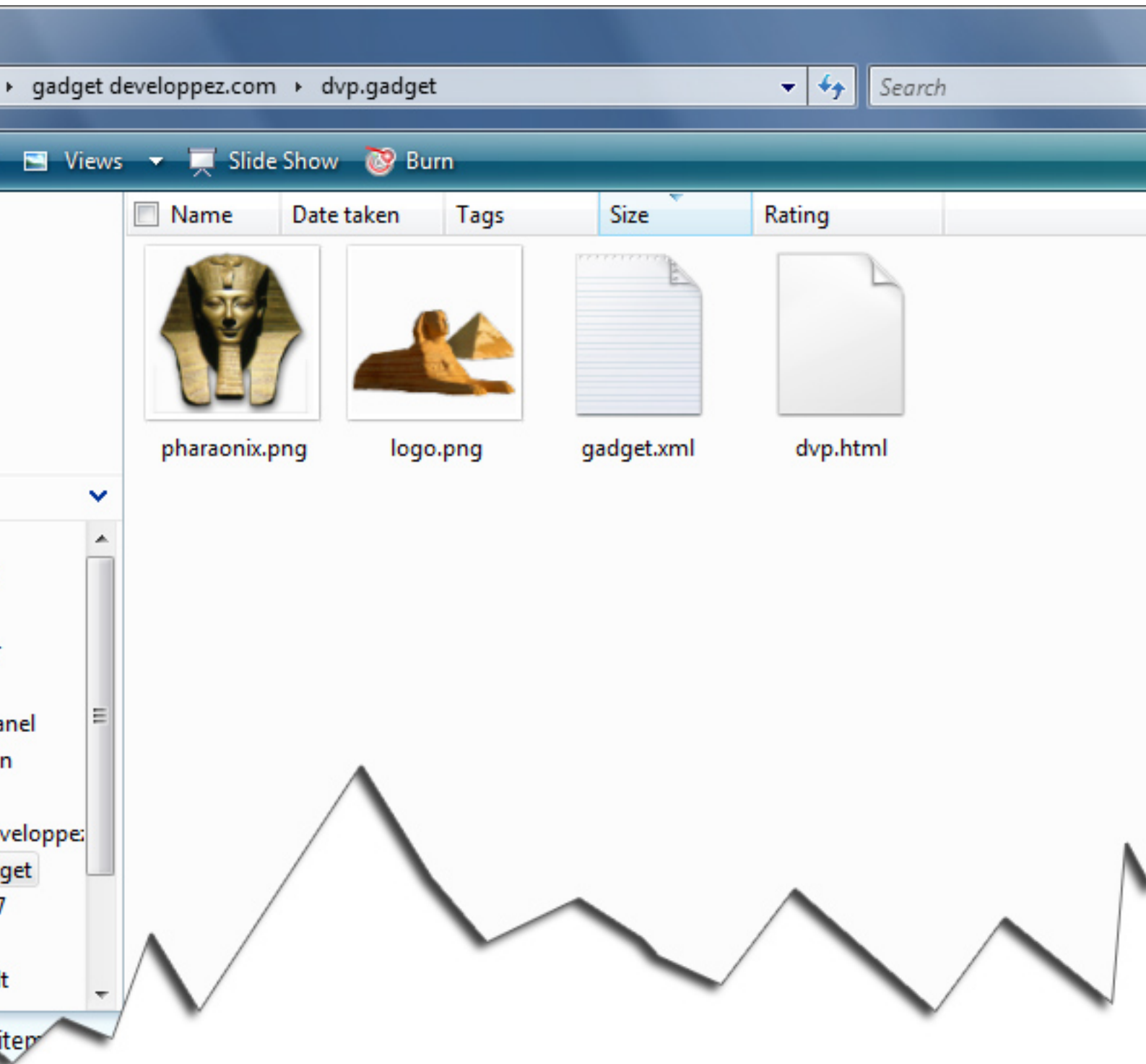
Toujours dans le dossier précédemment créé, créez maintenant un fichier que vous nommerez **dvp.html**. Ouvrez-le avec Notepad et collez-y le code suivant:

dvp.html

```
<html>
<style>
  body
  {
    width:130;
    height:50;
  }
</style>
<body>
  Vive Developpez.com !
</body>
</html>
```

Vous avez remarqué que dans le fichier XML, nous citons deux images. La première est obligatoire; je l'ai nommée **pharaonix.png**. Ajoutez donc un fichier png nommé ainsi. La deuxième image est facultative mais il peut parfois être intéressant d'y mettre son logo personnel ou le logo de son entreprise. Ajoutez aussi une image PNG de votre choix, que vous nommerez **logo.png**.

Votre dossier doit alors ressembler à quelque chose comme ceci:



Sélectionnez maintenant les fichiers et créez une archive ZIP que vous nommerez dvp.gadget (sachant que "dvp" peut être le nom que vous voulez). Vous obtenez votre premier gadget et c'est sous cette forme que vous devrez le distribuer si vous voulez en faire profiter d'autres personnes.

3-3 - Installation

Il faut maintenant tester notre gadget. Cette étape demande beaucoup de doigté, puisqu'il suffit de double-cliquer sur le gadget pour qu'une boîte de dialogue s'ouvre, et demande la confirmation d'installation du gadget.

Cliquez donc sur Installer.

Le fichier zip est alors dézippé dans un dossier qui sera appelé `VotreNamespace.Nomdugadget.gadget`. Dans mon cas: `developpez.dvp.gadget`.

Ce dossier se trouve ici: **H:\Users\%username%\Local Settings\Microsoft\Windows Sidebar\Gadgets**

C'est dans ce dossier que sont placés les gadgets installés et qui pourront être choisis dans la boîte à gadgets de la Windows Sidebar.

Ouvrez maintenant la Sidebar et ouvrez la boîte d'ajout de gadgets, vous devriez voir votre gadget dans la liste. Cliquez sur détails et vous retrouverez les informations sur le gadget et l'auteur que vous aviez saisi dans le fichier XML.

`<name>` 1

`<author name= >` 2

`<info url= >` 3

`<logo src= >` 4

`<description>` 5

`<copyright>` 6

`<icon src= >` 7

Search all gadgets

Contacts CPU Meter Currency Converter

Recycle Bin RSS Feeds Slide Show

pharaonix 2006 www.developpez.com

Get more ga...

Faites le glisser jusqu'à votre Sidebar et si tout va bien, votre premier gadget devrait apparaître tel quel:



Et voilà, vous venez de créer votre premier Gadget! Essayons maintenant d'aller plus loin.

4 - Création avancée

Nous allons ici créer un gadget qui ne vous sera pas forcément très utile, mais qui vous permettra de voir les différentes fonctionnalités que l'on peut donner à gadget. Ainsi, nous allons tenter de créer une barre de recherche pouvant lancer des recherches sur le site de son choix.

Cela aura pour effet de nous faire voir différents principes, comprenant l'interaction avec Internet, le design du Gadget et l'utilisation d'options sur un Gadget.

4.1 - Les options

Reprenez le code HTML et ajoutez-lui une balise <head> dans laquelle nous déclarerons un fichier CSS et un fichier JavaScript. Notez que nous avons retiré la balise <style> car nous allons la placer dans le fichier CSS.

dvp.html

```
<html>
<head>
  <link href="dvp.css" rel="stylesheet" type="text/css" /> <!-- nom de notre css (que l'on créera
juste après -->
  <script src="dvp.js" language="JavaScript"></script> <!-- nom de notre fichier javascript (que
l'on créera juste après -->
</head>
<body>
  Vive Developpez.com !
</body>
</html>
```

Créez dans le répertoire un fichier dvp.css et éditez-le avec Notepad et collez-y le code suivant:

dvp.css

```
body
{
  width:130;
  height:50;
}
```

Nous compléterons la feuille CSS dans le chapitre Design. Créez maintenant un fichier texte vide que vous nommerez **settings.html**. Puis ouvrez le fichier **dvp.html** pour y coller dans la balise body le code suivant:

dvp.html

```
<script>
  System.Gadget.settingsUI = "settings.html";
</script>
```

Automatiquement le gadget détectera que des options sont possibles et le bouton option (la petite "clé à molette") s'activera.

Ouvrez le fichier settings.html précédemment créé et tapez-y ceci. Nous définissons l'affichage du panneau options, dans lequel nous plaçons une Textbox qui servira à stocker l'url du flux RSS. Nous nommerons cette Textbox **txtUrl**.

settings.html

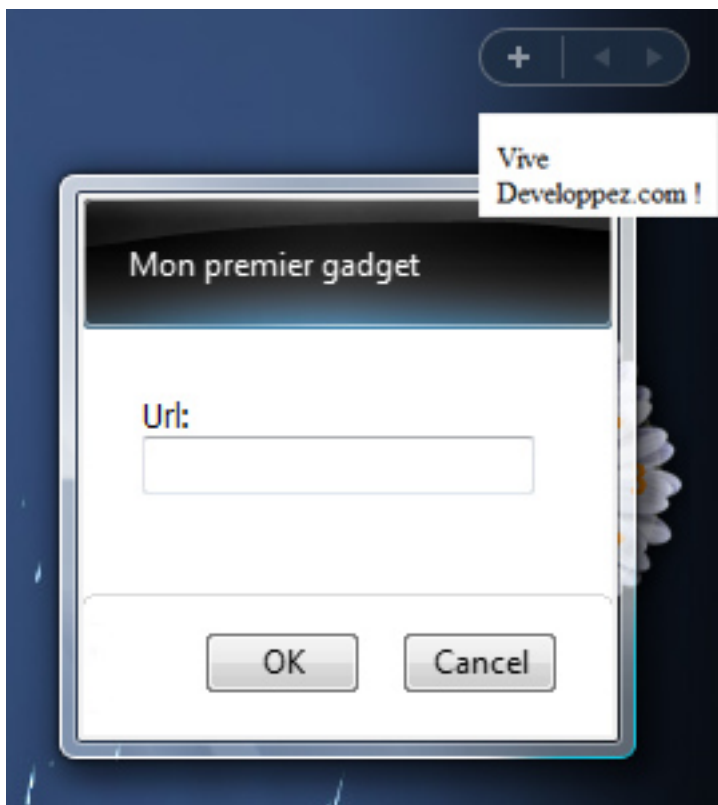
```

settings.html
<html>
<head>
  <style>
    body
    {
      width:160;
      height:50;
    }
  </style>
</head>
<body >
  Url:<br/>
  <input type="text" id="txtURL" length="100">

</body>
</html>

```

Dorénavant, lorsque vous cliquerez sur le bouton Options de votre Gadget, une fenêtre s'ouvrira, proposant de saisir l'url. Le code que l'on vient d'écrire nous donnera ceci:



Nous allons coder trois fonctions en JavaScript, qui nous permettront de charger l'URL actuellement utilisée et de sauvegarder l'URL saisie par l'utilisateur. Commençons par la méthode init():

```

settings.html

function init()
{
  var temp = System.Gadget.Settings.read("vURL");
  if (temp != "")
    txtURL.innerText = temp ;
  else txtURL.innerText = "http://www.developpez.com"
}

```

La méthode qui sauvegardera l'URL saisie

```
settings.html

function Save()
{
    System.Gadget.Settings.write("vURL", txtURL.value);
}
```

Et la méthode de fermeture qui sauvegardera lorsque la fenêtre des options se fermera.

```
settings.html

System.Gadget.onSettingsClosing = SettingsClosing;
function SettingsClosing(event)
{
    if (event.closeAction == event.Action.commit) {
        Save();
    }
    event.cancel = false;
}
```

Revoyons le code que nous avons tapé. La méthode *init()* tente de récupérer la valeur de la variable *vURL* qui est sauvegardée par le gadget. Ne l'ayant pas encore défini, elle est vide et cela remplira avec la valeur par défaut "http://www.developpez.com".

Nous créons alors la méthode *SettingsClosing()* que nous "accrochons" à l'événement de fermeture de la fenêtre. Ainsi, cela appellera cette méthode uniquement pendant la fermeture. Cette méthode appelle alors la méthode *save()* qui sauvegarde le contenu de la Textbox dans la variable *vURL*, stockée par le gadget. Cette variable sera stockée par la Windows Sidebar dans le fichier *Settings.ini* qui se trouve dans le dossier **Local Settings\Microsoft\Windows Sidebar**. Ainsi, au prochain lancement, cette variable sera automatiquement chargée avec la dernière valeur enregistrée.

4.2 - Le design

Nous allons remodifier le code du fichier **dvp.html** pour avoir un Gadget un peu plus esthétique. Récupérons d'abord sur Internet, une quelconque image de loupe, puis enregistrons-la dans le dossier de notre gadget. Nous allons alors ajouter l'image au code HTML, puis une Textbox et enfin un bouton:

```
<html>
<head>
    <link href="css/dvp.css" rel="stylesheet" type="text/css" /> <!-- nom de notre css (que l'on
    créera juste après -->
    <script src="js/dvp.js" language="JavaScript"></script> <!-- nom de notre fichier javascript
    (que l'on créera juste après -->
</head>
<script>
    System.Gadget.settingsUI = "settings.html";
</script>
<body>
<span style="color: #6389D8;font-weight: bold;font-size: 10pt;">
    Rechercher :</span>
<input id="txtSearch" type="text" size="14" maxlength="30" />
<div align="right"><input type="submit" value="OK" id="search" name="search" onclick="search()"
    /></div>
</body>
</html>
```

Notre gadget ressemblera dorénavant à ceci:



4.3 - Le code

Niveau code, nous en avons déjà écrit pour les options mais il faut maintenant en écrire pour l'interface principale du gadget. Mais comme vous allez le voir, le plus dur a été fait. Nous allons simplement créer une méthode qui ouvre une fenêtre avec pour adresse, l'url des options concaténée avec la valeur de la Textbox de recherche, comme suit (placez le dans un fichier dvp.js):

dvp.js

```
function search()
{
    var temp = System.Gadget.Settings.read("vURL");
    window.open(temp+txtSearch.value, 'window');
}
```

Il nous reste plus qu'à appeler cette méthode. Pour cela, ajoutez la propriété onclick au bouton OK de dvp.html

dvp.html

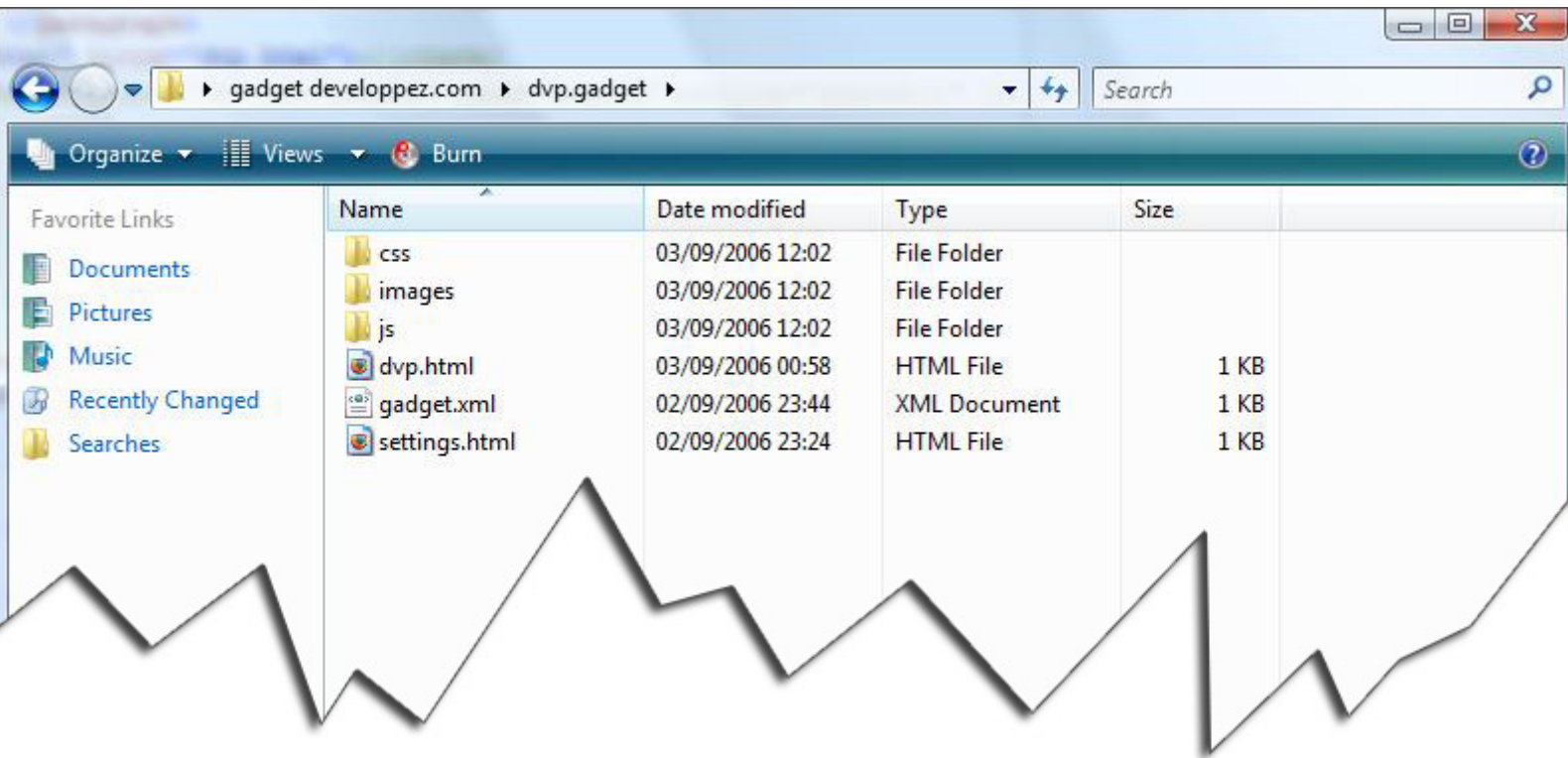
```
<input type="submit" value="OK" id="search" name="search" onclick="search()" />
```

A ce moment précis, votre gadget est fonctionnel et ira lancer une recherche sur le site web que vous avez passé en paramètre. Il faut bien entendu une URL qui accepte des mots en paramètre dans l'url (paramètres GET).

Il nous reste à fignoler notre gadget pour qu'il soit le plus réussi possible.

4.4 - Finition

Voilà, nous avons un gadget qui marche mais avouons-le, il est pas très bien rangé. Nous (vous :)) allons donc créer un dossier css dans lequel nous déplacerons le fichier dvp.css, un dossier JavaScript où déplacer le fichier dvp.js, et un dossier images où placer toutes nos images. Une fois le rangement fait, il faudra éditer les chemins des fichiers, dans chaque fichier (principalement la css et dvp.html).



Sélectionnez tous les fichiers créés et comme précédemment, créez une archive ZIP que vous nommerez search.gadget. Double-cliquez dessus pour l'installer.

Maintenant, allez dans les options de votre gadget. Comme aucune URL n'a été précisée, la case se remplit automatiquement avec l'URL du dictionnaire de Developpez.com. Fermez alors les options et tapez un mot à rechercher. Une fenêtre du dictionnaire devrait s'ouvrir, lançant automatiquement la recherche avec le mot saisi. Et voilà, c'est fini :)

5 - Création encore plus avancée

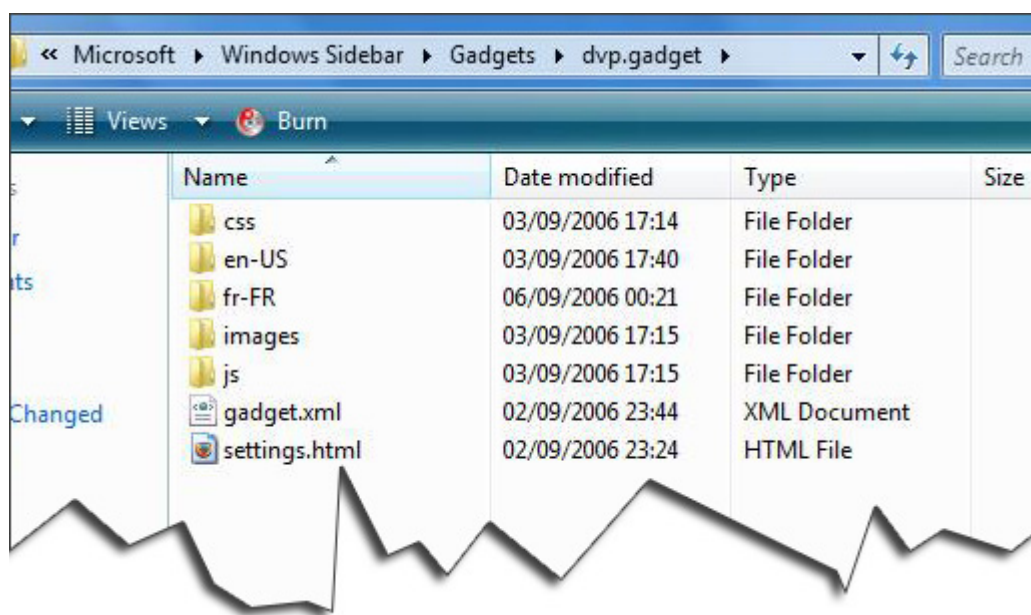
Dans cette dernière partie, nous allons voir deux derniers points qui se révéleront essentiels pour le développement d'un Gadget de qualité.

5.1 - La localisation

En informatique, le terme *localisation* se rapporte à la langue de l'utilisateur. Ainsi, nous allons créer un gadget qui s'adaptera tout seul à la langue de l'utilisateur.

Nous avons ici un problème, c'est que nous affichons le mot "Rechercher". Mais que se passerait-il si le système sur lequel il était installé était en anglais? Notre utilisateur serait alors bien perdu. Nous allons faire en sorte que notre Gadget affiche un texte différent selon que l'ordinateur soit en français ou en anglais.

Chaque culture (langue) est représentée par deux parties, la langue et le pays. Pour la France, nous avons fr-FR, pour la Belgique, nous avons fr-BE, et pour les Etats-Unis, nous avons en-US. Nous allons donc créer un dossier nommé fr-FR et un dossier nommé en-US. Nous allons alors déplacer le fichier `dvp.html` dans le dossier fr-FR, puis en faire une copie que nous collerons dans en-US. Nous l'éditerons alors pour changer "Rechercher" par "Search". Notre gadget pourra alors être en français ou en anglais selon la langue par défaut du système.



Note: Tous les fichiers n'ont pas besoin d'être localisés. Dans notre cas, le fichier JavaScript, les images et même le fichier `settings.html` n'ont pas besoin d'être changés quelle que soit la langue de l'utilisateur. Ces fichiers peuvent rester en un unique exemplaire à la racine du gadget. Pas besoin de changer les chemins dans le fichier `dvp.html`, le moteur de gadgets ira les chercher tout seul.

5.2 - L'effet dockable

Vous ne l'avez peut-être pas remarqué, mais les gadgets peuvent être placés sur le bureau, simplement par glisser-déposer. A ce moment là, ils peuvent soit rester dans le même état que lorsqu'ils étaient accrochés à la Sidebar, soit changer totalement de forme.

Nous allons donc définir deux états pour notre gadget: l'état docké et l'état non docké. Pour l'état docké, c'est simplement l'état dans lequel nous l'avons développé jusqu'à maintenant. Pour l'état non docké, nous allons agrandir le gadget et lui donner un fond et un contour. Le fonctionnement interne reste totalement identique.

Dans le fichier `dvp.js`, déclarons deux méthodes que nous accrocherons aux événements de dock et de undock.

`dvp.js`

```
System.Gadget.onUndock = WhenUndocked;
System.Gadget.onDock = WhenDocked;
```

Créons alors la fonction pour l'état non docké:

`dvp.js`

```
function WhenUndocked()
{
    System.Gadget.beginTransition();
    with(document.body.style)
    {
        marginTop = "30px";
        marginLeft = "30px";
        width=383;
        height=197;
        backgroundImage="url('../images/fond.png')";
    }

    with(lblRechercher.style)
    {
        fontSize="17px";
    }

    with(txtSearch.style)
    {
        fontSize="17px";
        width = "290px";
    }
    System.Gadget.endTransition(System.Gadget.TransitionType.none,0);
}
```

Voyons ce que signifie le code. Comme vous l'avez deviné, la méthode `WhenUndocked()` est appelée chaque fois que le gadget est pris de la Sidebar pour être déposé sur le bureau.

Sinon voici la signification des modifications:

document.body.style: pour le gadget en lui-même, je lui définis de nouvelles dimensions (383x197) et je change son fond. Puis je définis une marge à gauche et en haut pour recentrer les éléments (label, Textbox, bouton)

lblRechercher.style: pour le label, j'agrandis sa police

txtSearch.style: pour la Textbox, je redéfinis sa largeur ainsi que la taille du texte. Voici maintenant la méthode lorsque le gadget est replacé sur la Sidebar

`dvp.js`

```
function WhenDocked()
{
    System.Gadget.beginTransition();
    with(document.body.style)
```

dvp.js

```
{
  width=130,
    height=85,
  backgroundImage="url('../images/bg.gif')";
  marginTop = "3px";
    marginLeft = "5px";
}
with(lblRechercher.style)
{
  fontSize="13px";
}

with(txtSearch.style)
{
  fontSize="13px";
  width = "120px";
}

with (imgSearch.style)
{
  width = "16px";
}

System.Gadget.endTransition(System.Gadget.TransitionType.none,0);
}
```

Pas grand chose à expliquer cette fois-ci. Nous redonnons les valeurs d'origine. Voici une image montrant les deux états de notre Gadget.



Et voilà, vous savez pratiquement tout sur les gadgets, et vous pouvez dorénavant leur donner la forme que vous souhaitez et leur faire faire ce que vous désirez.


6 - Conclusion


Comme vous avez pu le voir, il n'est pas bien compliqué de développer son propre Gadget et vous n'avez pas d'excuse de ne pas avoir le gadget de vos rêves à partir du moment où vous acceptez de mettre les mains dans le cambouis.

N'oubliez pas, si votre gadget est fonctionnel, n'hésitez pas à le mettre en ligne sur le site MicrosoftGadget, il peut servir à d'autres personnes ;)

7 - Liens et téléchargements

Voici quelques liens qui vous aideront lors du développement de vos Gadgets:

Des Gadgets en téléchargement libre :  [ici](#)

Un forum d'entraide sur les gadgets :  [ici](#)

Les sources du Gadget que nous avons développé durant cet article : [télécharger](#)

8 - Remerciements

J'aimerais remercier **Maximilian** pour ses commentaires et corrections apportées à l'article.

