

## Incorporation de fichiers ou d'images dans les ressources d'un assembly

(J-C Armici [jca.developpez.com](http://jca.developpez.com), [www.unvrai.com](http://www.unvrai.com))

**Cet article remplace le précédent. On peut toujours faire plus simple!**

**Les nouvelles parties sont indiquées en rouge dans la marge.**

### Remarque préliminaire:

*Ce qui suit a été réalisé avec Visual Studio 2005 professionnel (en anglais, car la version française n'est pas encore disponible)*

### Objectif:

Le but de cet exemple est de montrer comment incorporer, ou inclure, des données dans exécutables. L'intérêt pouvant être de distribuer un seul et unique fichier contenant des données de plusieurs types et formats. Nous nous limiterons ici à un fichier texte et une image.

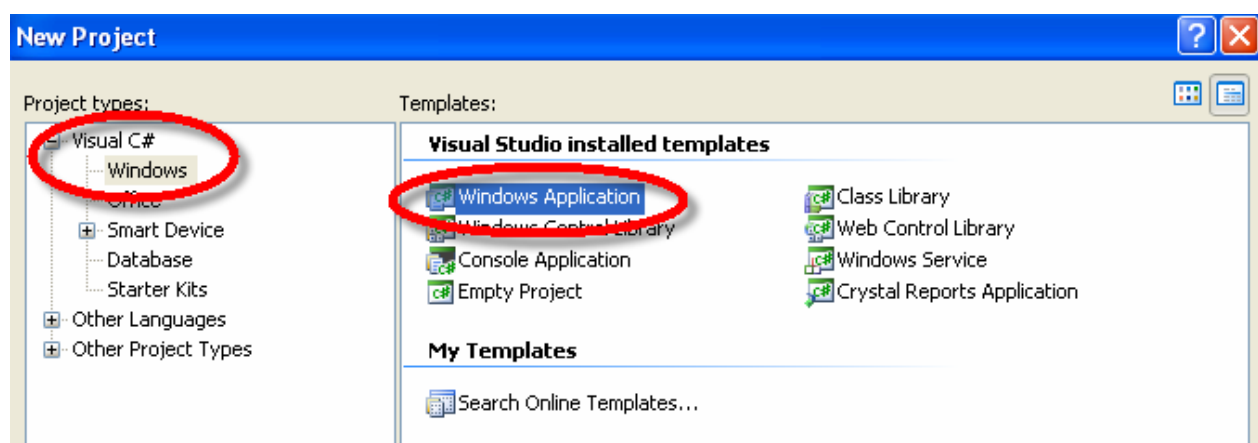
Le travail est constitué de deux étapes:

- placer les informations dans les ressources utilisées par le projet
- accéder à ces ressources au moment de l'exécution

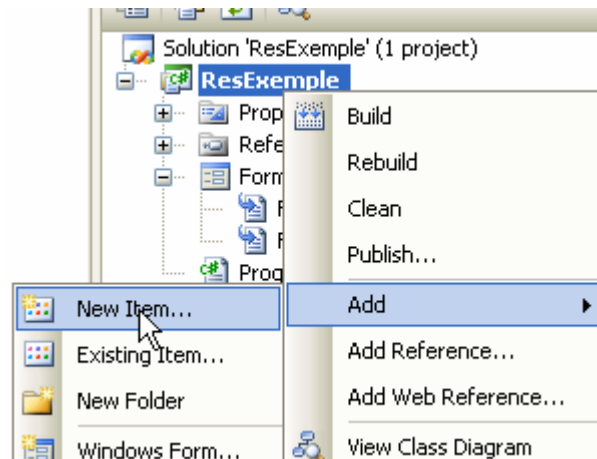
### Comment placer des informations dans les ressources d'un projet ?

Voici, point par point comment réaliser ce projet:

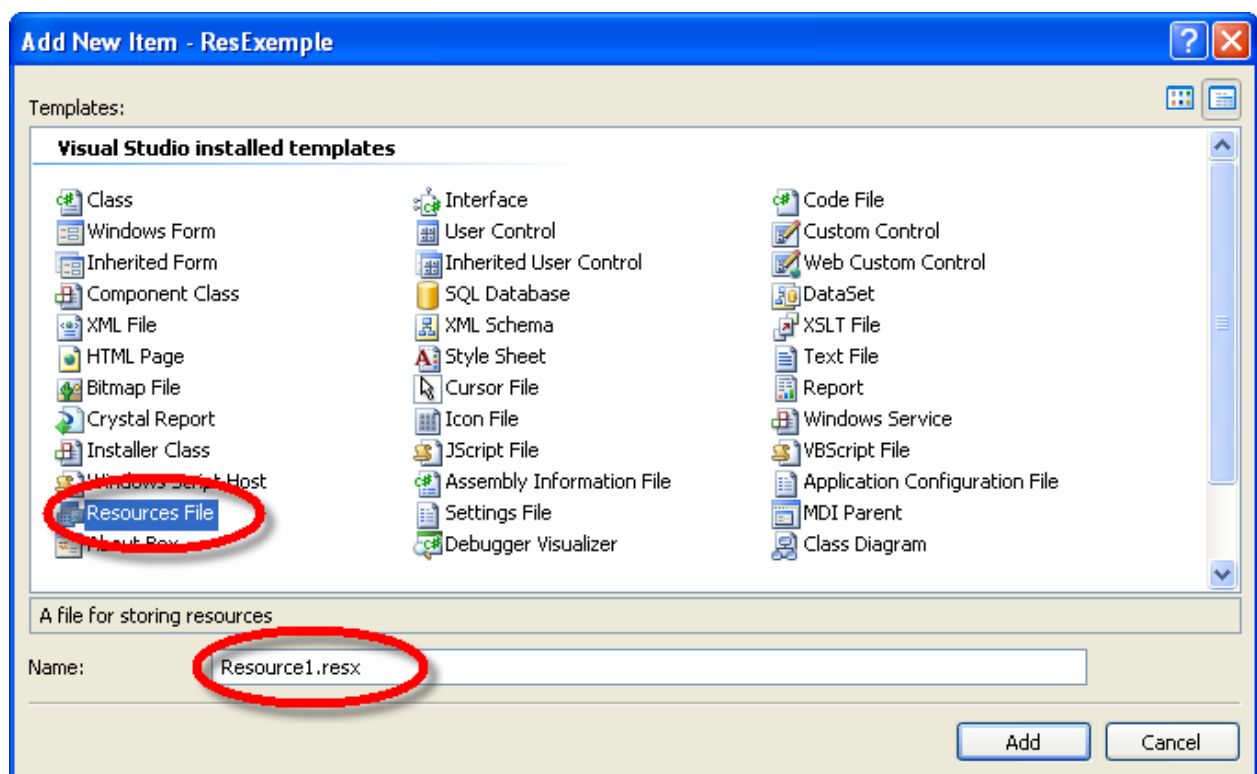
- commencer un nouveau projet en C# de type **Windows Application**



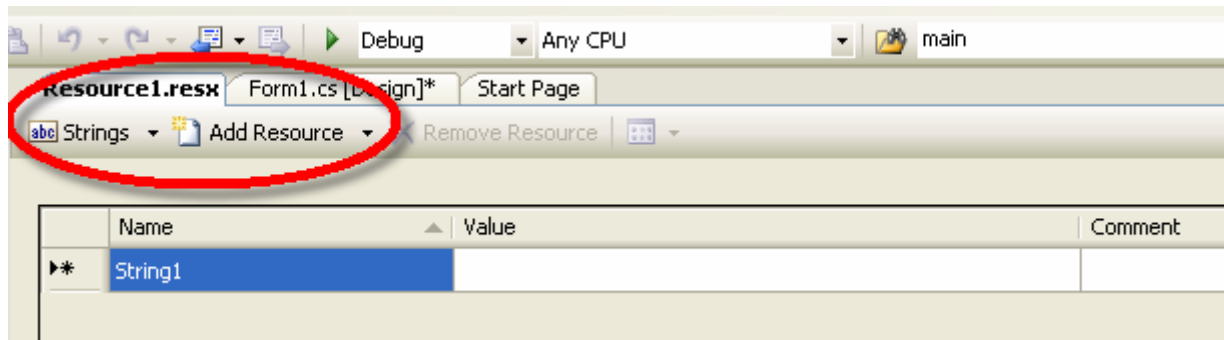
- dans l'explorateur de solution, faire un clic droit sur le nom du projet, puis **Add** et **New Item...**



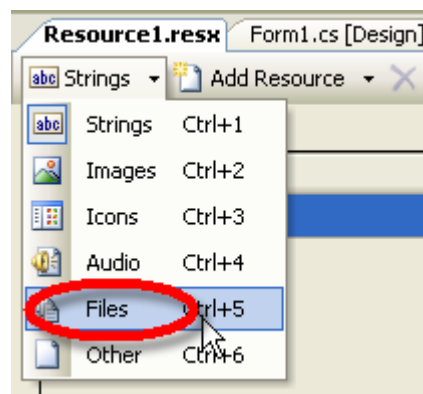
- dans la fenêtre qui apparaît, choisir **Resources File**. Au besoin renommer l'élément (par défaut le nom est **Resource1.resx**)



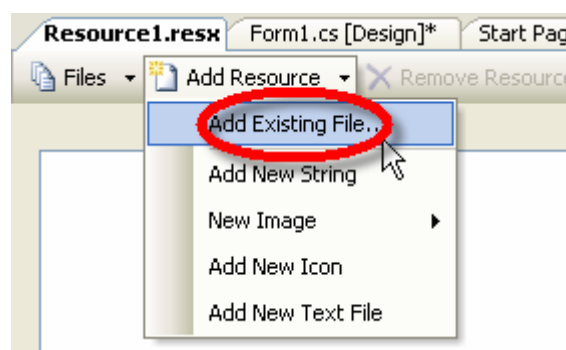
- un nouvel onglet apparaît dans la partie centrale de Visual Studio



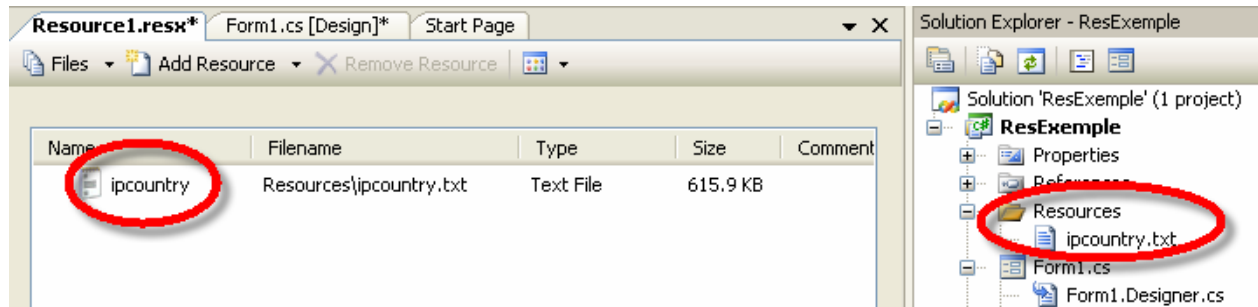
- comme nous allons placer un fichier texte dans les ressources, il faut choisir **Files**



- puis **Add resource / Add existing file...**



- après avoir choisi un fichier texte (ici **ipcountry.txt**), un nouveau répertoire a été créé dans l'arborescence du projet. Ce répertoire **Resources** contient une copie du fichier spécifié



- la première étape est terminée, la ressource (fichier **ipcountry.txt**) a été ajoutée au projet. On remarquera que la ressource porte le nom **ipcountry**. Ce nom peut être modifié. A noter également qu'un double-clic sur la ressource permet de visualiser le fichier dans un nouvel onglet de Visual Studio.

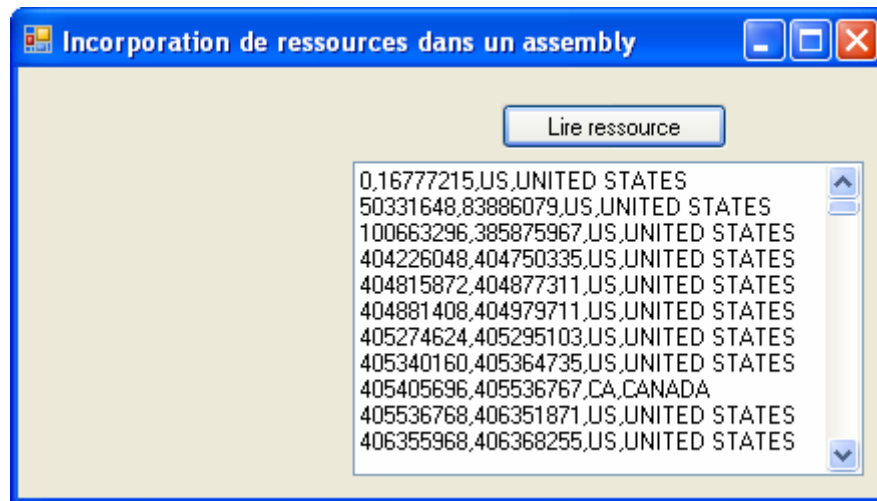
### Comment accéder au fichier placé dans les ressources au moment de l'exécution ?

Dans cette seconde étape nous plaçons dans notre interface (**Form1**) un **TextBox** et un **Button**. Ce dernier permettant de lire le fichier depuis la ressource et de le placer dans le **TextBox**. Voici le code effectuant cette opération:

```
private void btnRessource_Click(object sender, EventArgs e)
{
    txtRes.Text = ResExe.Resource1.ipcountry;
}
```

Comme on peut le voir, Visual Studio nous permet d'utiliser la ressource à l'aide d'une syntaxe très simple. **ResExe** est le nom du projet, **Resource1** celui du fichier de ressources et **ipcountry** le nom de la ressource désirée.

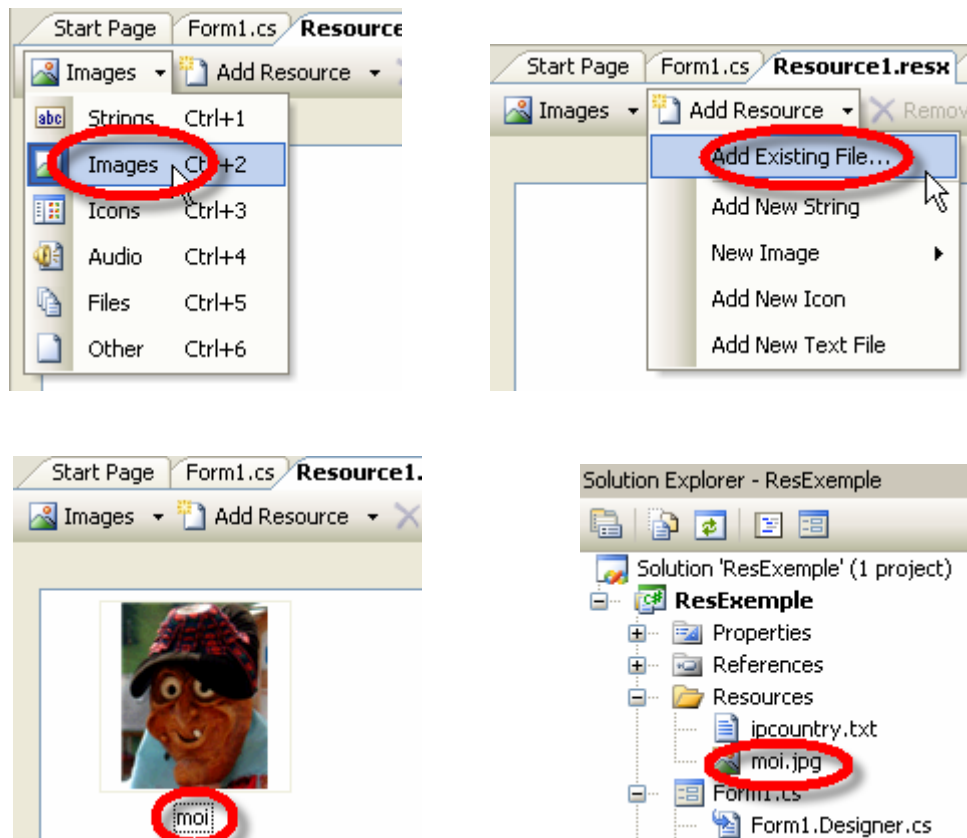
On peut alors lancer le projet, cliquer sur le bouton Lire ressource. On obtient alors ceci:



### Comment faire la même chose avec une image ?

La manière de faire étant analogue, voici avec moins de détails comment placer une image dans les ressources d'une application.

Toujours dans l'onglet **Resource1.resx**, on choisit cette fois d'incorporer une image:



A ce stade l'image fait partie des ressources du projet.

Pour ce qui est de l'utilisation de cette image, nous allons simplement l'afficher dans un **PictureBox**. Nous faisons appel à l'événement **Paint** de la fenêtre (**Form1**). Voici le code:

```
private void Form1_Paint(object sender, PaintEventArgs e)
{
    pictureBox1.Image = ResExe.Ressources1.moi;
}
```

Voici le résultat:



## Conclusion

Au travers de ce petit exemple on peut imaginer intégrer assez facilement toutes sortes d'objets dans les ressources d'une application C#.