

Les Web Services Jusqu'où irez-vous ?

1 Utilisation d'un Service Météo

Parmi de nombreux fournisseurs de Services Web, vous trouverez sur www.dataaccess.com, le service « numberconversion » (Cf. <http://www.dataaccess.com/webserviceserver/numberconversion.wso>).

Ecrivez un programme console qui affiche les nombres de 1 à 100 en toute lettre. On mesurera et on affichera le nombre de ticks d'horloge pour chaque communication avec le service distant grâce à un timer `System.Diagnostics.Stopwatch sw`. Les méthodes et propriétés qui nous intéressent sont : `sw.Reset()`, `sw.Start()`, `sw.ElapsedTicks`, `sw.Stop()`.

Le site www.webservicex.com fournit aussi un grand nombre de Services Web. Regardons par exemple le service <http://www.webservicex.com/globalweather.aspx>. La méthode `GetWeather` est particulièrement intéressante.

Ecrivons un programme type « Windows Forms » qui prend un nom de ville, le nom du pays correspondant et affiche une image donnant le niveau d'ensoleillement / pluie / neige

Et oui ... tant qu'il y aura des services

2 Les services Web de Google

Google est un des plus gros fournisseurs de services web. Toutes les informations liées à la mise en œuvre de ces services se trouvent sur : <https://developers.google.com/>.

Les plus connus sont probablement ceux de google map : <https://developers.google.com/maps/web-services/>.

Comme vous pouvez le voir l'utilisation professionnelle de ces services Web est payante.

Heureusement pour pratiquer quelques tests et sous certaines contraintes d'utilisation, vous pouvez obtenir une clef gratuite d'essai.

Après avoir créé un compte google vous pouvez vous connecté sur votre tableau de bord <https://console.developers.google.com/home/dashboard>.

Pour cela :

- Vous devez créer un projet (Cf. barre de menu en haut à droite)
- Cliquer sur « Utiliser les APIs Google » pour en voir la liste et les activer pour votre projet
- Sur la gauche, cliquez sur “Identifiants”
- Choisissez le type d'identifiant qui convient à votre projet, comme par exemple « API key ». La clef identifie alors votre projet et fournit un accès à l'API. Vous en aurez besoin lors des appels à l'API Google.

Question : Tester le service Elevation Map

Question : Tester le service Google Translate

Les Web Services Jusqu'où irez-vous ?

3 Votre Gateway de Service Google en WS-SOAP

Comme il est dommage de ne pouvoir accéder à Google Translate avec un client généré à la volé, nous allons faire un service web de type « gateway ».

Question : Ecrivez un service web sur votre machine qui fournit un service de traduction en utilisant un client vers Google Translate. Ce service sera de type WS-SOAP et fournira un WSDL pratique pour générer le client de test.

4 Rappel : pour démarrer à la fois le client et l'hôte à partir de Visual Studio

- Créez une solution Visual Studio qui contient à la fois les projets client et serveur.
- Configurez la solution de façon à ce qu'elle démarre à la fois les processus client et serveur lorsque vous choisissez **Démarrer** dans le menu **Déboguer**.
 - Dans l'**Explorateur de solutions**, cliquez avec le bouton droit sur le nom de la solution.
 - Cliquez sur **Définir les projets de démarrage**.
 - Dans la boîte de dialogue **Propriétés de la Solution <Nom>**, sélectionnez **Plusieurs projets de démarrage**.
 - Dans la grille **Plusieurs projets de démarrage**, sur la ligne qui correspond au projet serveur, cliquez sur **Action** et choisissez **Démarrer**.
 - Sur la ligne qui correspond au projet client, cliquez sur **Action** et choisissez **Démarrer**.
 - Cliquez sur **OK**.