

Web services : développement de services web avec .net

1 Création de service web avec C#

On va ici créer un service web simplissime avec .NET.

Notre service web permettra simplement de récupérer la date du jour (utiliser la classe `System.DateTime`).

1. Dans la solution de votre choix, clic droit, ajouter → nouveau site web.
2. Vous obtenez (entre autres) d'une part un répertoire pour le code applicatif (`App_Code`) contenant `Service.cs` (un genre de `HelloWorld`, qui contient des services identifiés par `[WebMethod]`), et d'autre part un fichier `.asmx` qui permet le déploiement de ce service, en précisant en quel langage il est écrit (ici C#), et dans quelle classe et quelle assembly se trouve la méthode à invoquer. Son code ressemble à :

```
<%@ WebService Language="C#" CodeBehind="~/App_Code/Service.cs" Class="Service" %>
```

3. Quand on démarre le projet (juste par un `run` du projet), le serveur de développement ASP.net se met en route, en local, et un navigateur s'ouvre sur la page `asmx`. Vous obtenez une interface de test de votre service web, vous pouvez aussi visualiser le `wsdl` qui est publié et le SOAP qui sera échangé. Jetez-y un coup d'œil.
4. Définissez maintenant un service web qui fournit la date du jour (ce qui est follement utile).

2 Le retour de la bibliothèque, avec des services web

Reprenons l'exemple de la bibliothèque. Dans sa version actuelle, la communication entre un client et la bibliothèque se fait via .net remoting. Il s'agit ici que cette communication puisse aussi passer par service web. Cela a entre autre l'avantage de pouvoir développer un client en Java, et de réaliser facilement des clients légers.

Dans votre bibliothèque, proposez certains services comme service web (par exemple : la recherche de livre par ISBN et par auteur, et l'ajout de commentaire), et réalisez un client C# pour ces services. Pensez bien que les types échangés doivent être simples mais bien structurés. Pensez aussi qu'il n'est pas raisonnable de prévoir que pour obtenir une information, on échange de nombreux messages entre le client et le serveur (pensez au temps que cela prendrait).