vinci
HAUTE ÉCOLE LÉONARD DE VINCI

institut paul lambin

C# : WCF

Modèle de programmation orienté service

vinci
HAUTE ÉCOLE LÉONARD DE VINCI

Sommaire

- SOAP vs REST
- Introduction SOA
- SOA : principes
- SOA & WCF
- Première Application WCF
- Test
- Hébergement – considérations
- Configuration du service - pour aller plus loin

vinci
HAUTE ÉCOLE LÉONARD DE VINCI

SOAP vs REST

- SOAP
  - Multi-protocole (HTTP, TCP, ….)
  - Support transaction
  - Sécurité accrue (contrats)
  - Rigidité (contrats)
- REST
  - Uniquement HTTP
  - Simple (facile pour exposer une API)
  - Trt données -> XML, JSON, ….

vinci
HAUTE ÉCOLE LÉONARD DE VINCI

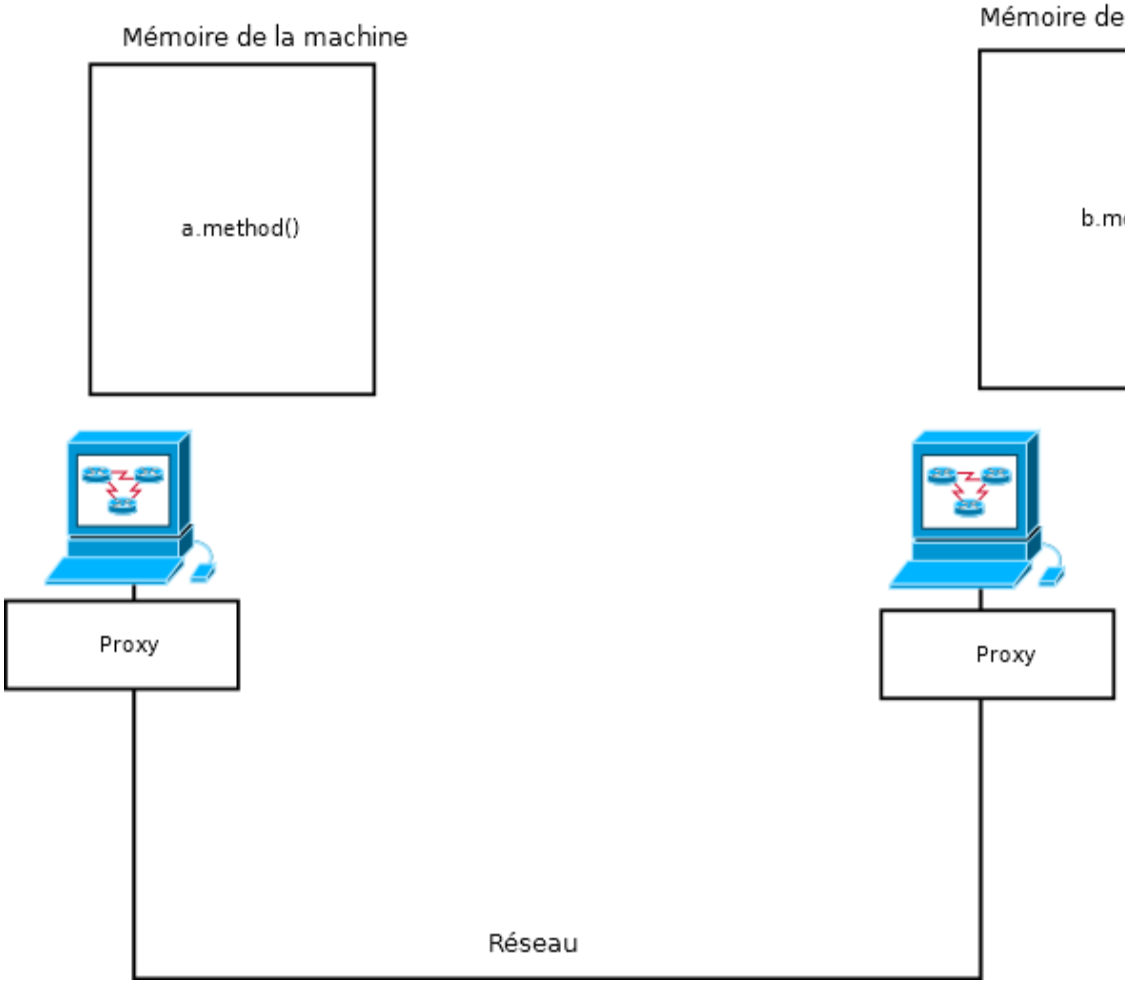
SOAP vs REST

- Le framework . NET
  - SOAP
    - Via WCF
  - REST
    - Via WCF REST
    - Via ASP.NET Web Api (avenir)

Introduction SOA

- SOA : architecture orientée service
- Un peu d'histoire
  - Ordi personnel mono-utilisateur
  - Ordi multi-utilisateur et multi-tâches -> nécessité d'un moyen de communication
  - Développement et harmonisation des communications réseaux informatiques
  - Développement de mécanisme de communication inter-processus
    - COM, DCOM, Corba, RMI … (LAN)
    - Web Services (WAN)

Problématique SOA



Services rendus par le proxy

Sérialisation des données et désérialisation
Envoi et réception des requêtes par le réseau
...

vinci
HAUTE ÉCOLE LÉONARD DE VINCI

SOA : principes

- **La définition du service est explicite**
- **Les services sont autonomes**
- **Les clients et les services ne partagent que des contrats**
- **La compatibilité est basée sur les règles**

WCF & SOA : objectifs de WCF

- Windows Communication Foundation
  - Interopérabilité entre différentes technologies
  - Unification des technologies de communication .Net
  - Support pour le développement d'application orienté service

vinci
HAUTE ÉCOLE LÉONARD DE VINCI

WCF & SOA : objectifs de WCF

	ASMX	.NET Remoting	Enterprise Services	WSE	System. Messaging	System. Net	WCF
Interoperable Web Services	X						X
Binary .NET –.NET Communication		X					X
Distributed Transactions, etc.			X				X
Support for WS-* Specifications				X			X
Queued Messaging					X		X
RESTful Communication						X	X

vinci
HAUTE ÉCOLE LÉONARD DE VINCI

Rappel architecture distribuée

- Sérialisation
  - Classe de base de c#
  - Classes utilisateurs
  - Listes
- Utilisation de proxy pour invoquer une méthode

Première application WCF: Etapes

- Définition de contrats
  - Contrat de service /opération
  - Contrat de données
- Implémentation des contrats
- Hébergement du service
- Configuration du service (point d'accès finaux)
- Consommation du service par une application cliente

Définition de contrat WCF

```
using System;
// les 3 espaces suivants sont la base de la création d'un service WCF
using System.Runtime.Serialization;
using System.ServiceModel;
using System.ServiceModel.Web;

namespace WCFMediatheque
{
    // Cette interface est un contrat de service WCF
    [ServiceContract]
    public interface IMediathequeService
    {
        // Cette méthode sera exposée au travers du service WCF
        [OperationContract]
        IList<Film> GetAll();

        [OperationContract]
        bool AddFilm(int id, string name);

    }

}
```

Définition de contrat de données WCF

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
// DataContrat et DataMember -> sérialisation
using System.Runtime.Serialization;
namespace WCFMediatheque
{
    // cette classe sera sérialisable
    [DataContract]
    public class Film
    {
        // attribut sérialisé
        [DataMember]
        private string name { get; set; }
        // attribut sérialisé
        [DataMember]
        private string category { get; set; }
        // attribut non sérialisé
        private string director { get; set; }
    }
}
```

Implémentation du contrat WCF

```
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.ServiceModel.Web;
using System.Text;

namespace WCFMediatheque
{
    // REMARQUE : vous pouvez utiliser la commande Renommer du menu Refactoriser pour changer le nom de classe "Service1" dans le code, le fichier svc et le fichier de configuration.
    public class MediathequeServiceImpl : IMediathequeService
    {
        public IList<Film> GetAll()
        {
            FilmsDAO fd = new FilmsDAO();
            return fd.GetAll();
        }


       ...
```

Implémentation du contrat WCF : remarques

- WCF utilise la sérialisation -> les attributs doivent donc posséder un accesseur en lecture (get) et un accesseur en écriture (set) pour permettre cette sérialisation.
- Si certaines classes sont générées avec LINQ to SQL, il faut dire à LINQ d'ajouter l'attribut « DataMember » à toutes les propriétés ainsi que la propriété « SerializationMode » à Unidirectionnal pour permettre la sérialisation.

Hébergement du service

- WCF Service Host
  - FichierImpl.svc (afficher le balisage)

```
<%@ ServiceHost Language="C#" Debug="true"
Service="WCFMediatheque.MediathequeServiceImpl"
CodeBehind="MediathequeServiceImpl.svc.cs" %>
```

- IIS
- Déployé par défaut dans IIS embarqué de Visual studio 2013 (développement)

Consommation du service par une application cliente

- Simple Classe
- Ajouter une référence de service
  - Attention aux listes sérialisées par défaut en array
- Utilisation de la classe proxy créée par WCF

```
MediathequeSVC.MediathequeServiceClient proxy = new
MediathequeSVC.MediathequeServiceClient();
```

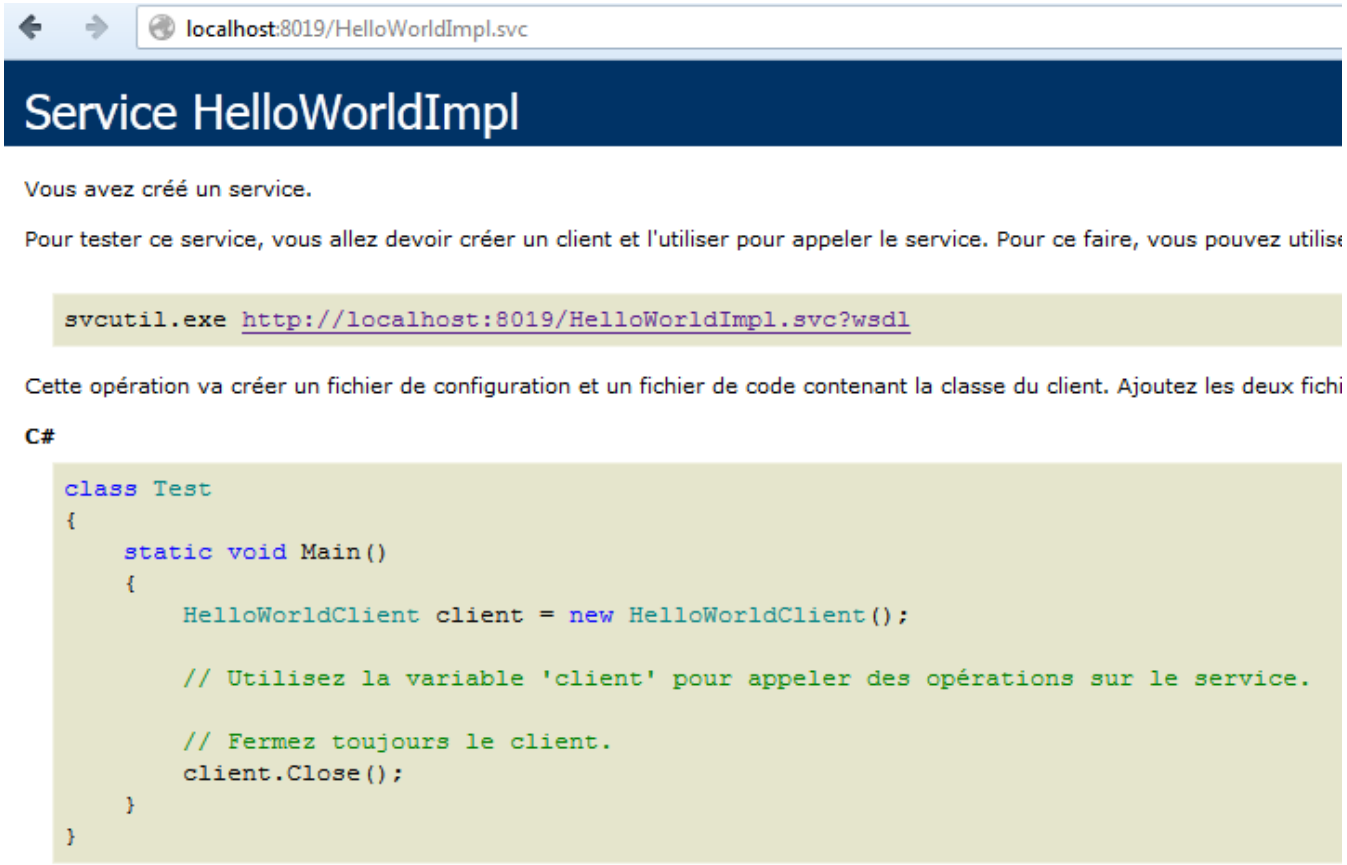
```
IList<WCFMediathequeClient.MediathequeSVC.Film> lst = proxy.GetAll();
```

Hébergement du service – InstanceContextMode

- Objet InstanceContext
- Modes :
  - Per Call
  - Per Session
  - Single
- Attention par défaut -> Per Call (le web service est recyclé à chaque appel)

```
ServiceContract(InstanceContextMode=
                InstanceContextMode.Single])
```

Tester la configuration



vinci
HAUTE ÉCOLE LÉONARD DE VINCI

Configuration du service

- Points de terminaison
  - L'adresse du service (dépend du protocole utilisé)
  - La liaison du service(comment le client peut se connecter)
    - Protocole de transport (HTTP, HTTPS, TCP, MSGQ, …)
    - Format d'encodage des messages (binaire, UTF-8, …)
    - Exigences de sécurité du service
    - Exigences transactionnelles du service (commit/ rollback)
    - Fiabilités des communications avec le service (réseau et sessions)
  - Contrat mis en œuvre par le service (publication)

Configuration du service WCF

- Web.config ou App.config
  - Points accès finaux
    - Address
    - Binding (liaison de services)
    - Contract
  - Notion de comportement
  - Fichier de configuration simple
  - Editeur de configuration WCF

vinci
HAUTE ÉCOLE LÉONARD DE VINCI

Liaisons de service

- Un ou plusieurs éléments liaisons
- Un élément de liaison == un canal
- Description du protocole de transport
- Description de l'encodage des messages
- Gère un aspect non fonctionnel (sécurité, …)
- …

Liaisons de service prédéfinies

- basicHttpBinding
  - Protocole HTTP ou HTTPS
  - Encodage des messages en XML
  - Compatible avec les Web Services ASMX
- wsHttpBinding
  - Protocole HTTP ou HTTPS
  - Sessions sécurisées et transactions distribuées
  - Encodage des messages XML ou MTOM

Liaisons de service prédéfinies

- netTcpBinding
  - Protocole TCP (plus rapide qu'HTTP)
  - Encodage des messages en binaire
  - Communications sécurisées, transactions distribuées
- mexHttpBinding
  - Liaison particulière permettant d'exposer aux clients les méthodes disponibles pour un service

Comportements de service

- Etendre les fonctionnalités d'un service
  - Activation de la publication des métadonnées
  - Mode debug ou pas pour les messages
  - ….

vinci
HAUTE ÉCOLE LÉONARD DE VINCI

Démo HelloWorld