

INF4375 - Paradigmes des échanges Internet

Historique des communications distantes

Jacques Berger

# Objectifs

Introduire les web services

# Prérequis

Communication réseau

# Sophismes

Le réseau est fiable

Le temps d'attente est nul

La bande passante est infinie

Le réseau est sécuritaire

# Sophismes

La topologie du réseau est fixe

Il n'y a qu'un seul administrateur

Le coût de transport est nul

Le réseau est homogène

# RPC

Remote Procedure Call

Partage de comportement inter-applications

# RPC

Basé sur l'appel de fonctions locales

Connu des programmeurs

Simple à imaginer, à visualiser

Appels synchrones

Architecture client-serveur

Le serveur contient les fonctions distantes

# RPC

## Appel RPC vs appel local

Le réseau peut être en problème  
Que faire si la réponse n'arrive pas?

Le réseau est TRÈS lent  
Incomparable à un appel local

Le réseau n'est pas sécuritaire  
La sécurité doit être gérée par l'application



# RPC

## Déroulement nominal

Le programme client fait un appel local au stub.

Le stub sérialise la requête et la transmet sur le réseau.

Le skeleton reçoit la requête sérialisée et la désérialise.

Le skeleton envoie la requête à l'implémentation de la procédure.

La procédure s'exécute et retourne la réponse au skeleton.

Le skeleton sérialise la réponse et la transmet sur le réseau.

Le stub reçoit la réponse sérialisée et la désérialise.

La réponse est retournée au programme client.

# RPC

Granularité des appels élevée

Conservation de l'état de la conversation du côté client

Gestion des transactions à la main

# RPC

Penser à la sécurité lors du design

Format de sérialisation

Compression des requêtes et réponses

Rendre la signature des fonctions disponibles

# RMI

Remote Method Invocation

RPC pour Java

Permet de distribuer des objets

Problème récurrent avec les firewalls

Restreint au monde de la JVM

# CORBA

Common Object Request Broker Architecture

Permet de faire du RPC inter-plateforme

# CORBA

Nouveau concept : IDL  
Interface Description Language

Décrire l'interface des services  
Décrire les signatures des fonctions  
Permet la génération de proxy en C++, Java, et  
plusieurs autres

# CORBA

## Inconvénients

Trop complexe à programmer, à apprendre

Aucune sécurité

Aucune cohérence architecturale

Plus de ressources pour des fonctionnalités équivalentes

Écrasé par XML et Microsoft

# DCOM

Distributed Component Object Model

Réponse de Microsoft à CORBA

Abandonné par Microsoft après quelques années



# XML-RPC

Encoder les appels de fonction en XML

Nom de la fonction

Type de retour

Liste de paramètres

Nom, type

# XML-RPC

La norme spécifie :

- Le format d'une requête

- Le format d'une réponse

- Les types de données

- Comment encoder une requête/réponse dans un message HTTP

# XML-RPC

Spécification très simple

Automatiquement interopérable à cause de XML

Aussitôt repris par Microsoft comme base à SOAP

# Web Services

Ensemble de 3 normes XML

SOAP

Simple Object Access Protocol

WSDL

Web Service Description Language

UDDI

Universal Description Discovery & Integration

# Communication

Conçu pour utiliser n'importe quel protocole de communication

Seul HTTP est supporté

Habituellement synchrone

# Description d'un service

## Norme WSDL

«WSDL-first» development :

Génération des proxy à partir du WSDL

«WSDL-last» development :

Génération du WSDL à partir de  
l'implémentation du service

Façon habituelle de faire

# WSDL

Version courante : WSDL 2.0

Permet de décrire l'interface d'un service

Syntaxe très complexe

Structure très complexe

# WSDL

En résumé, on fait le lien entre :

Des types de données

Une signature de méthode

Un style d'échange (RPC ou document)

Un protocole de communication (HTTP)

Un style d'encodage

Une URL



# Styles d'échanges

Unidirectionnel (message)

Bidirectionnel

RPC

Document

# Styles d'échanges

## RPC

Norme XML permettant de décrire un appel de méthode

Le contenu de l'enveloppe SOAP doit être un fragment de XML respectant la section 7 de la norme SOAP

# Styles d'échanges

## Document

Le contenu de l'enveloppe SOAP doit être un fragment XML bien-formé

# Découverte de service

UDDI

Annuaire de services

Implémentations

Java : uddi4j

Ruby : uddi4r

Python : uddi4py

# SOAP

Simple Object Access Protocol

Acronyme original, maintenant un nom commun

Version courante : SOAP 1.2

Langage XML permettant de décrire une enveloppe

# SOAP

Enveloppe

    Méta-données

    Données

    Ajouts (attachments)

# SOAP

## Méta-données (headers)

Contient de l'information à propos des données

Destinées aux intermédiaires :

Routage, logging, sécurité, filtres, etc.

Optionnelles

# SOAP

Méta-données (headers)

N'ont pas besoin d'être traitées, sauf exceptions

Ne doivent pas contenir d'instructions pour le service



# SOAP

Données (body)

Destinées au destinataire ultime

Contient :

- Le document XML

- L'appel RPC

- La réponse RPC

- Gestion d'erreurs

# SOAP

## Données – gestion d'erreurs

Un élément du body sert à rapporter les erreurs  
fault (doit obligatoirement être traité)

Champs :

faultcode : indique le type de la faute

faultstring : description lisible de la faute

faultfactor : URI de la source de la faute

detail : détails à propos de la faute

# SOAP

Modèle de données

Optionnel

Types XSD simples

int, string, Date, etc.

Compound types

Structs

Arrays

# SOAP

Modèle de données

Structs

Semblable à un enregistrement

Sans méthode

Membres publiques

Comme une struct C++ (concept)

# SOAP

Modèle de données

Arrays

Éléments dans un vecteur, identifiable par une position (pas par un nom)

Peuvent être imbriqués

Valeurs peuvent être des types simples, des structs ou des arrays

# SOAP

Styles de communication

RPC

Invocation synchrone

# SOAP

## Styles de communication

### Document

Message-oriented

Plus bas niveau d'abstraction

Exige plus de programmation

Le paramètre d'entrée est un document XML

La réponse peut être n'importe quoi

Très flexible

# SOAP

## Encodage

Traduction d'un protocole à un autre

### SOAP Encoding

Sérialisation et désérialisation des types du modèle de données

Standard SOAP 1.1



# SOAP

## Encodage

### Literal

Aucun encodage

Habituellement, un XSD décrit le format et les types du message

### Literal XML

Conversion directe d'un arbre DOM en enveloppe SOAP et vice versa

# SOAP

## Encodage

### XMI

XML Metadata Interchange

Très peu utilisé

# Plus loin...

SOAP 1.2

<http://www.w3.org/TR/soap/>

WSDL 2.0

<http://www.w3.org/TR/wsdl20/>