

Applications orientées données (NSY135)

2 – Applications Web Dynamiques

Auteurs: Raphaël Fournier-S'niehotta et Philippe Rigaux
(philippe.rigaux@cnam.fr, fournier@cnam.fr)

Département d'informatique
Conservatoire National des Arts & Métiers, Paris, France

Plan du cours

S1 Bases sur le fonctionnement du Web

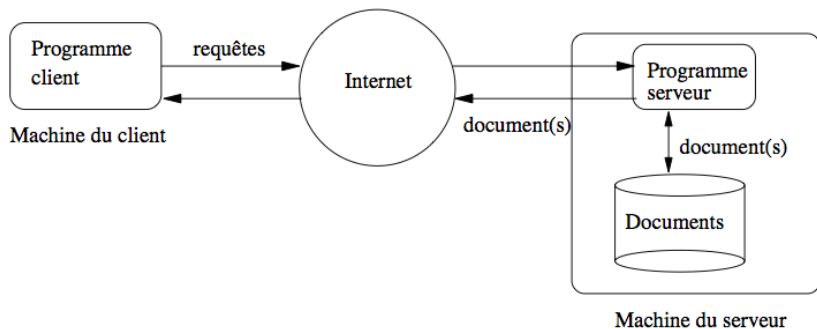
Le World-Wide Web

- Le **World-Wide Web** (ou WWW, ou Web) est un très grand système d'information, réparti sur un ensemble de **machines** connectées par le réseau Internet.
- Ce système est essentiellement constitué de **documents hypertextes** et de documents **multimedia**: textes, images, sons, vidéos, etc.
- Chaque machine propose un ensemble plus ou moins important de documents qui sont transmis sur le réseau par l'intermédiaire d'un **programme serveur**.
⇒ Exemple : Apache, un serveur **open source** très couramment utilisé pour des sites simples (blogs, sites statiques)
- Ce programme serveur dialogue avec un **programme client** qui peut être situé n'importe où sur le réseau
⇒ Le programme client prend le plus souvent la forme d'un **navigateur** (Firefox, Internet Explorer, Safari, Opera)

Fonctionnement du Web

- Le dialogue entre un programme serveur et un programme client s'effectue selon des règles précises qui constituent un **protocole**
⇒ Le protocole du Web est HTTP
- Tout site web est constitué, matériellement, d'une machine connectée à l'Internet équipée du programme serveur tournant en permanence
- Le programme serveur est en attente de requêtes transmises à son attention sur le réseau par un programme client
- Quand une requête est reçue, le programme serveur l'analyse afin de déterminer quel est le document demandé, recherche ce document et le transmet au programme client
⇒ Le programme client peut demander au programme serveur d'exécuter un programme, en fonction de paramètres, et de lui transmettre le résultat

Fonctionnement du Web



Le protocole HTTP

- HTTP, pour **HyperText Transfer Protocol**, est un protocole extrêmement simple, basé sur TCP/IP, initialement conçu pour échanger des documents hypertextes.
- Voici par exemple une requête envoyée à un serveur Web :

```
GET /myResource HTTP/1.1  
Host: www.example.com
```

- Et voici une possible réponse à cette requête :

```
HTTP/1.1 200 OK  
Content-Type: text/html; charset=UTF-8  
  
<html>  
  <head><title>myResource</title></head>  
  <body><p>Hello world!</p></body>  
</html>
```

Le protocole HTTP (2/2)

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=UTF-8

<html>
  <head><title>myResource</title></head>
  <body><p>Hello world!</p></body>
</html>
```

- Un message HTTP est constitué de deux parties: **l'entête** et **le corps**, séparées par une ligne blanche.
- Le programme client qui reçoit cette requête affiche le corps du message en fonction des informations contenues dans l'entête. Si le code HTTP est 200 par exemple, il procède à l'affichage. Un code 404 indique une ressource manquante
- D'autres aspects du protocole HTTP incluent la protection par mot de passe, la négociation de contenu (le client indique quel type de document il préfère recevoir), des **cookies**, informations déposées côté client pour mémoriser les séquences d'échanges (sessions)

Les URLs

- Toutes les **ressources** sur le Web sont identifiées par une **URL** (**Uniform Resource Locator**), de la forme suivante :

https://www.example.com:443/chemin/vers/doc?nom=orm&type=latex#fragment

- le **protocole** https est variante de HTTP comprenant un chiffrement des échanges
- l'**hôte** (*hostname*, ou domaine, avec le sous-domaine www)
⇒ Un des services du réseau Internet (le DNS) convertit ce nom d'hôte en adresse IP, permettant la communication entre machines
- le **port**. Chacun des ports correspondant à l'un des services gérés par le serveur. pour HTTP, le port par défaut est 80 (8080 pour Tomcat)

Les URLs

- Toutes les **ressources** sur le Web sont identifiées par une **URL** (**Uniform Resource Locator**), de la forme suivante :

https://www.example.com:443/chemin/vers/doc?nom=orm&type=latex#fragment

- le **chemin vers la ressource**. Il suit la syntaxe d'un chemin d'accès à un fichier dans un système de fichiers. S'il est omis, l'URL désigne la **racine** du site web.
 - ⇒ Dans les sites simples, "statiques", ce chemin correspond de fait à un emplacement physique vers le fichier contenant la ressource.
 - ⇒ Dans des applications dynamiques, les chemins sont virtuels et conçus pour refléter l'organisation logique des ressources offertes par l'application.
- la liste des **paramètres** (**query string**)
- le **fragment** (une sous-partie) du document

Le langage HTML

- Les documents échangés sur le Web peuvent être de types très divers.
- Le principal type est le **document hypertexte**, un texte dans lequel certains mots, ou groupes de mots, sont des **liens**, ou **ancres**, référençant d'autres documents.
- Le langage qui permet de spécifier des documents hypertextes, et donc de fait le principal langage du Web, est HTML.
⇒ HTML est dédié à la **présentation** des documents d'un site (c'est-à-dire la définition de leur structure), et ne constitue pas un langage de programmation. Il peut être couplé au langage CSS pour définir le "style" d'un site (templates)
- Il existe de nombreux éditeurs de documents HTML qui facilitent la saisie des balises et fournissent une aide au pré-positionnement des différentes parties du document

Exemple de page HTML

```
<?xml version="1.0" encoding="utf8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" >
<head>
<title>Gestion de la persistance dans les applications</title>
<link rel='stylesheet' href="template.css" type="text/css"/>
</head>
<body>

<h1>Cours Frameworks/ORM</h1>

<hr/>

Ce cours aborde les ORM comme
<a href="http://hibernate.org">Hibernate</a>,
dans le cadre du developpement d'applications modernes
construites avec des <i>frameworks</i>.

</body>
</html>
```