

Java EE

Cours de 2^e année ingénieur
Spécialisation « Génie Informatique »

fabien.romeo@fromeo.fr
<http://www.fromeo.fr>

Présentation du cours

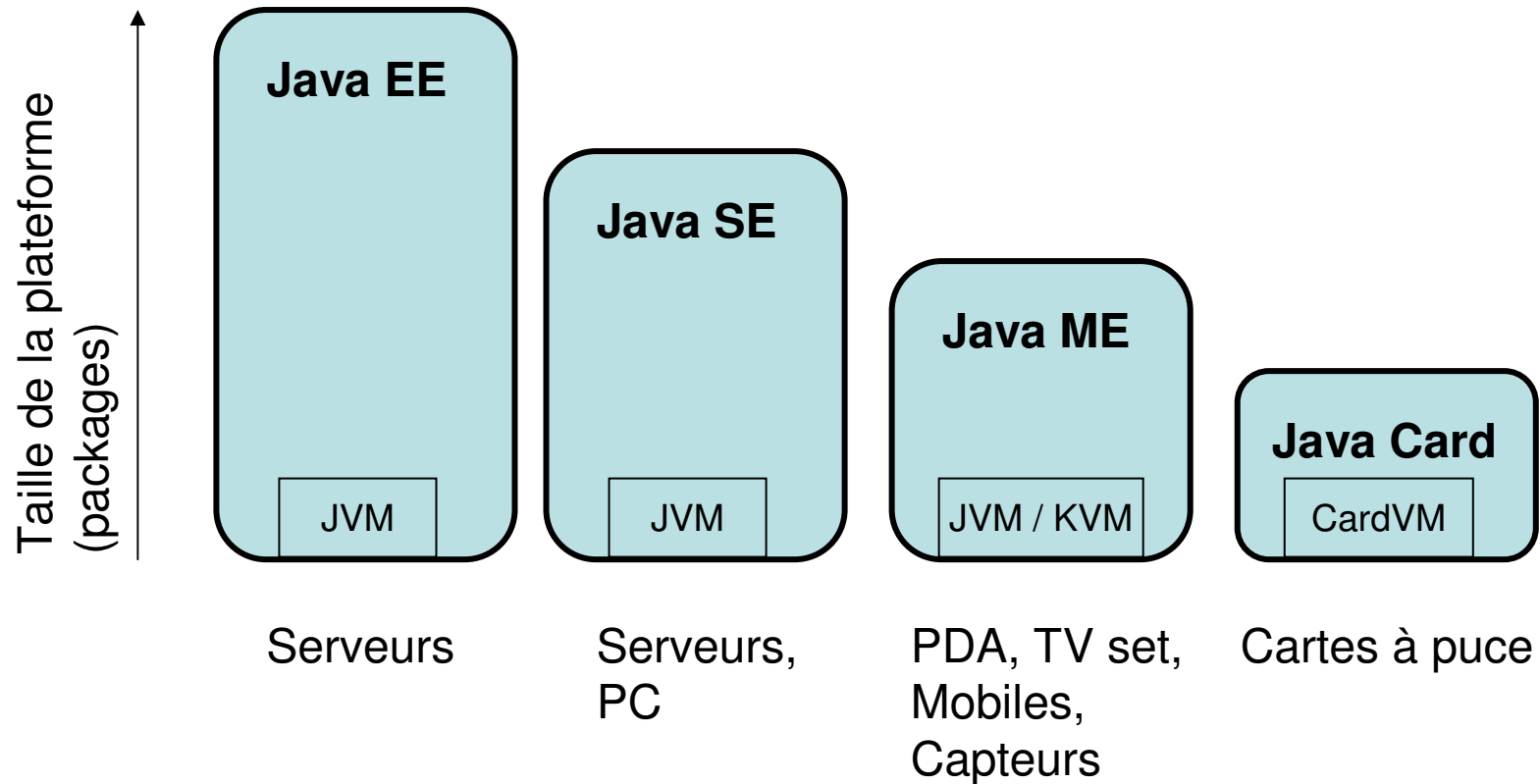
- Pôle GL-I2 « Génie logiciel »
 - 20h sur 8 semaines (1h cours + 1h30 TP)
- Objectifs
 - Développement d'applications Web **robustes**
 - « *Ne pas réinventer la roue* » → utilisation d'un framework ¹ (standard)
 - Apprentissage d'une partie de Java EE
 - Servlet, JSP et EL/JSTL
- Prérequis
 - Maîtrise du langage Java (Java SE)
 - Maîtrise du développement Web **client**
 - XHTML (au moins balises de structure et formulaires)
 - CSS et XML sont un plus

¹ dans COO, utilisation des Design Patterns

Java EE ?

- Java Enterprise Edition est un framework
 - riche (Java SE + nombreuses API)
 - ouvert (specs. du Java Community Process)
 - dédié au développement, au déploiement et à l'exécution d'applications Internet **modernes** (nécessaires aux entreprises)
- Favorise la séparation des préoccupations
 - Code métier vs. Propriétés non-fonctionnelles
 - QoS, persistance (JPA), administration (JMX), sécurité, transaction (JTS/JTA), ...

La galaxie Java



- Une édition n'inclut pas forcément la totalité d'une édition de plus « petite taille »

Le développement Web

- Le World Wide Web
 - Un SI public et universel déployé sur Internet
 - Un langage : HTML (ou XHTML)
 - Un modèle de communication : client-serveur
 - Un protocole : HTTP
- Pages statiques
 - Pages HTML préparées à l'avance
 - Le serveur renvoie les pages sans effectuer de traitement particulier
- Pages dynamiques
 - Pages HTML générées par le serveur
 - Le serveur construit la réponse en fonction de la requête de l'utilisateur

Java et le développement Web

- Différentes technologies « Java » permettent de faire du développement Web à différents niveaux

	Développement	
	Côté client	Côté Serveur
.class autonome	applet	servlet
Source Java mixé avec code html	<i>JavaScript</i>	JSP

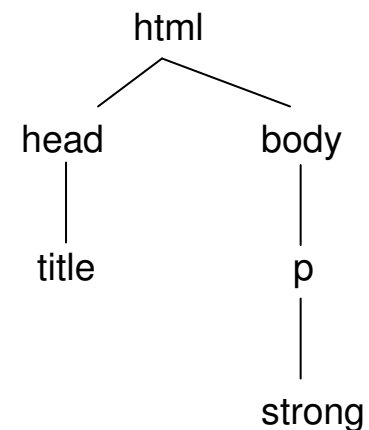
↑ ↑

Pages statiques **Pages dynamiques**

HTML (rappel)

- Langage de balisage, non propriétaire (W3C)
- Conçu pour afficher des documents sur le Web
- Liens hypertextes possibles entre les documents
- XHTML assure maintenant la compatibilité avec XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
  <head>
    <title>XHTML 1.0 valide !</title>
  </head>
  <body>
    <p>Une page XHTML 1.0 <strong>valide</strong>.</p>
  </body>
</html>
```



Quelques balises

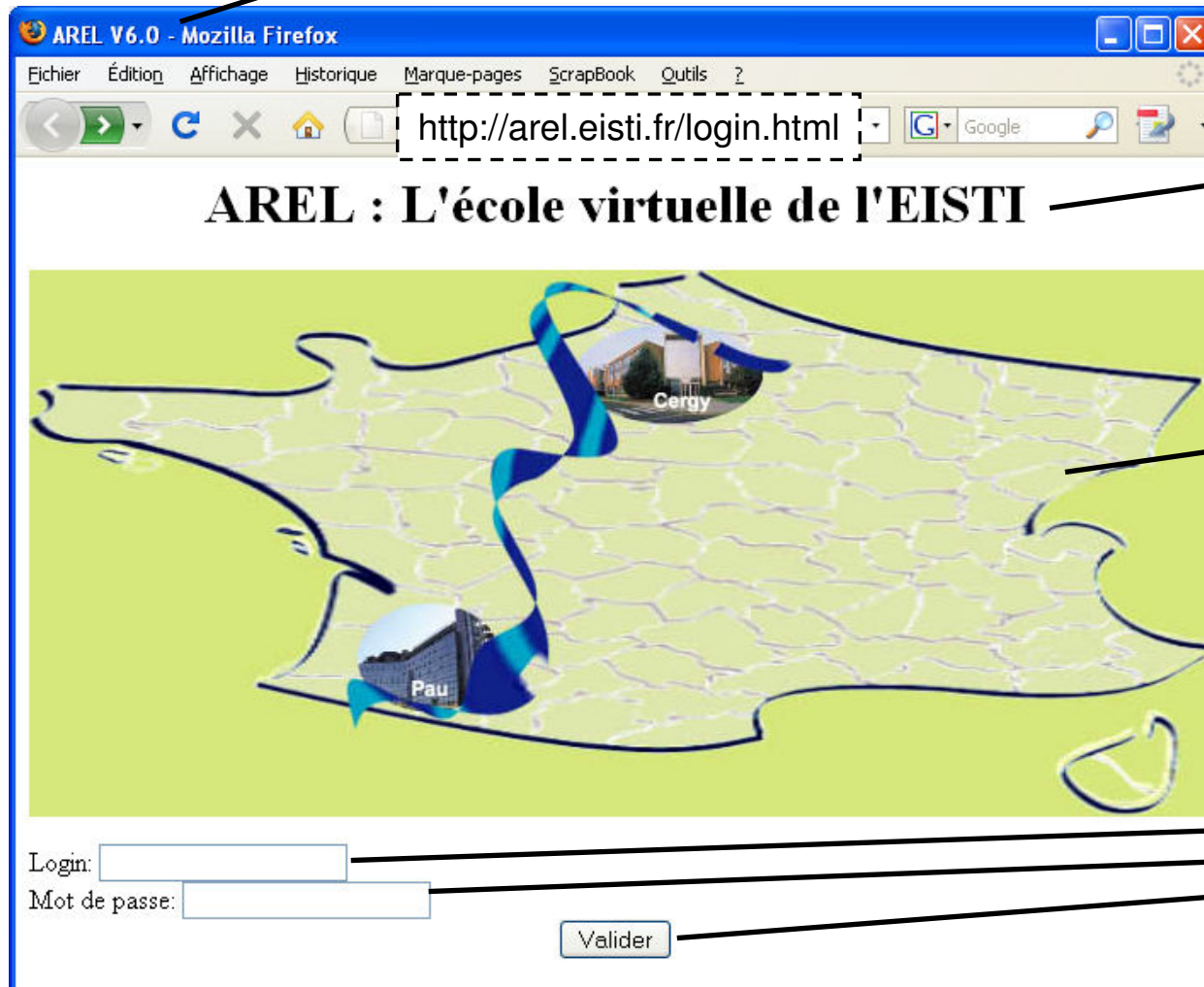
- `<!-- -->` commentaires
- `<a>` ancre (hyperlien href)
- `<body>` corps du document
- `<br>` *line break*
- `<form>` formulaire
- `<h1>` titre 1
- `<head>` entête
- `<html>` limite le document
- `<input type>` boutons et champs de saisie
- `<p>` paragraphe
- `<title>` titre

Ex : AREL V6 (code HTML)

```
<html>
<!-- AREL V6.0 -->
<head>
  <title>AREL V6.0</title>
</head>
<body>
  <h1 align="center">AREL : L'école virtuelle de l'EISTI</h1>
  <p><center>
    
  </center></p>
  <form action="log">
    Login: <input type="text" name="param1"/><br/>
    Mot de passe: <input type="password" name="param2"/><br/>
    <center>
      <input type="submit" value="Valider"/>
    </center>
  </form>
</body>
</html>
```

Ex : AREL V6 (affichage)

```
<title>AREL V6.0</title>
```



```
<h1 ...> ... </h1>
```

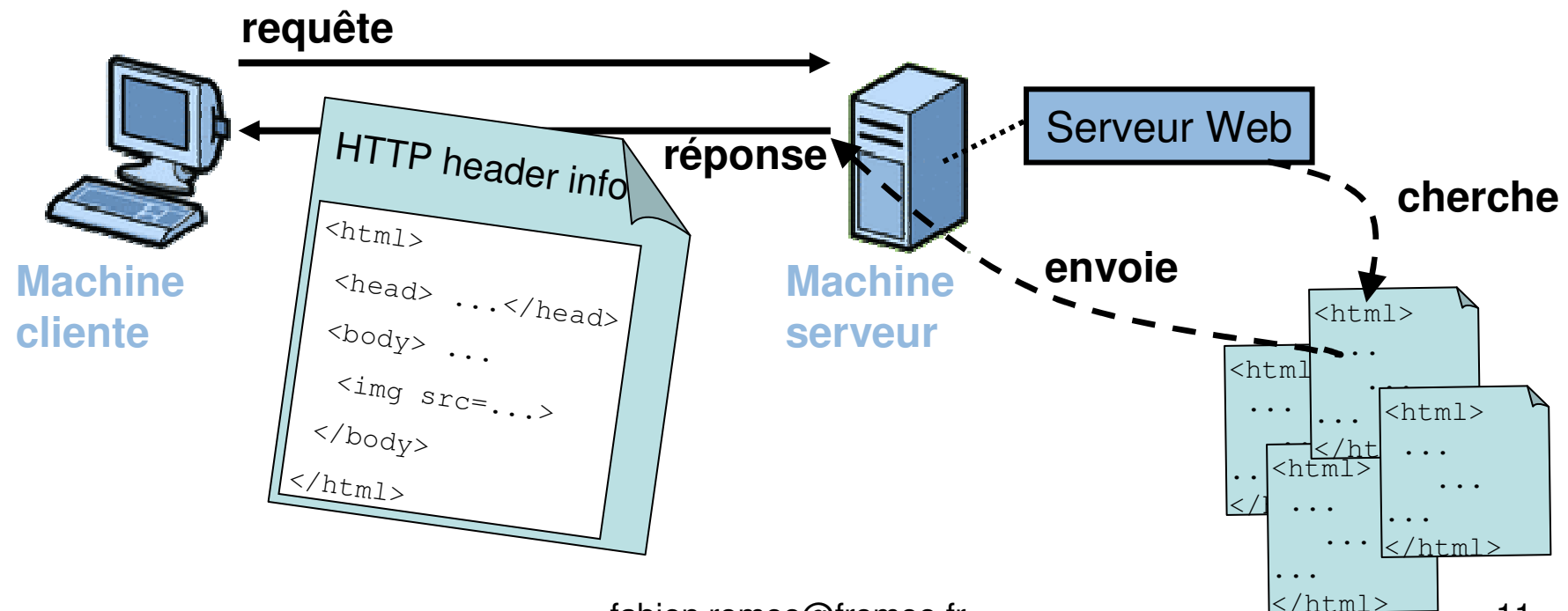
```
<img ... />
```

```
<input type="..." .../>
```

```
<input type="..." .../>
```

Pages statiques

- Fonctionnement normal d'un serveur Web seul
 - Le serveur cherche la page dans le système de fichiers
 - La page est renvoyée au client **telle quelle**



Limites d'un serveur Web seul

- Pas de contenu dynamique

```
<html>
  <head>
    <title>Clock</title>
  </head>
  <body>
    Il est toujours 12:12.
  </body>
</html>
```

statique

VS.

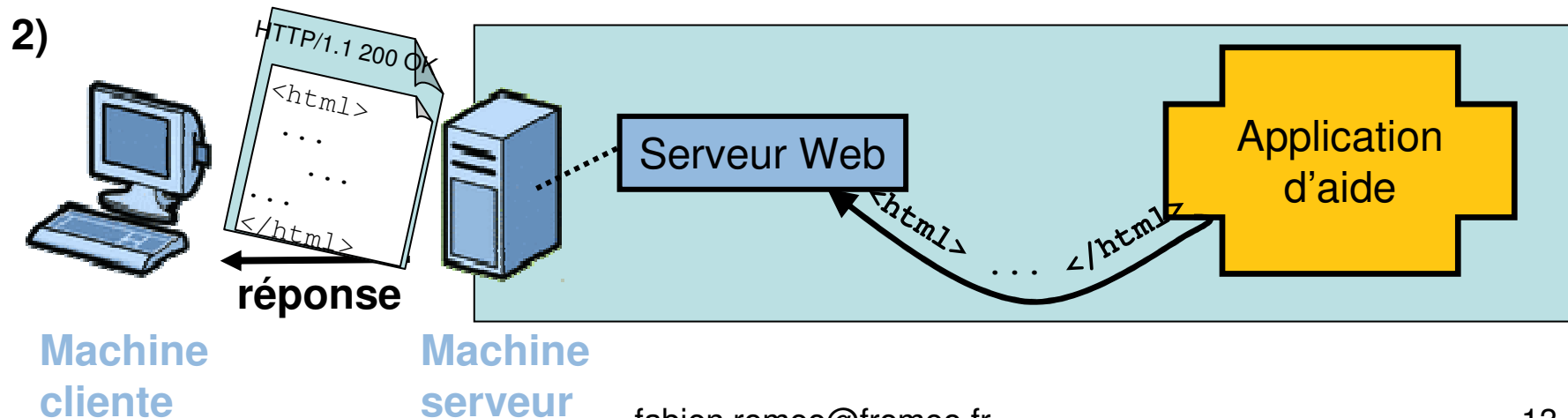
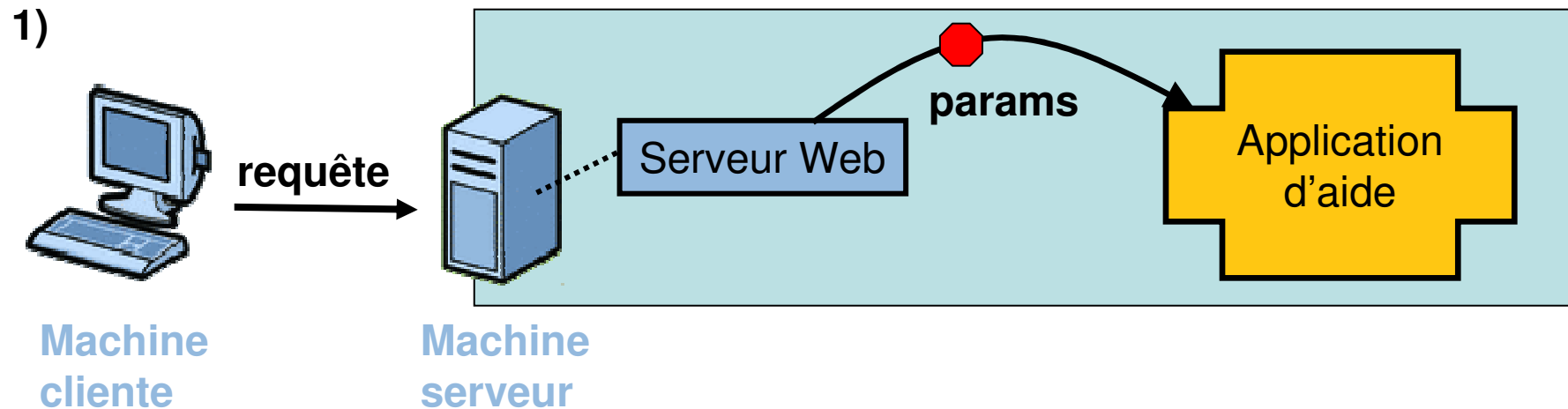
```
<html>
  <head>
    <title>Clock</title>
  </head>
  <body>
    Il est [getTimeOnServer].
  </body>
</html>
```

dynamique

- Pas de sauvegarde de données sur le serveur
 - Traitement de formulaires :
besoin d'une application d'aide au serveur Web
 - pour évaluer les paramètres reçus
 - pour générer une réponse appropriée

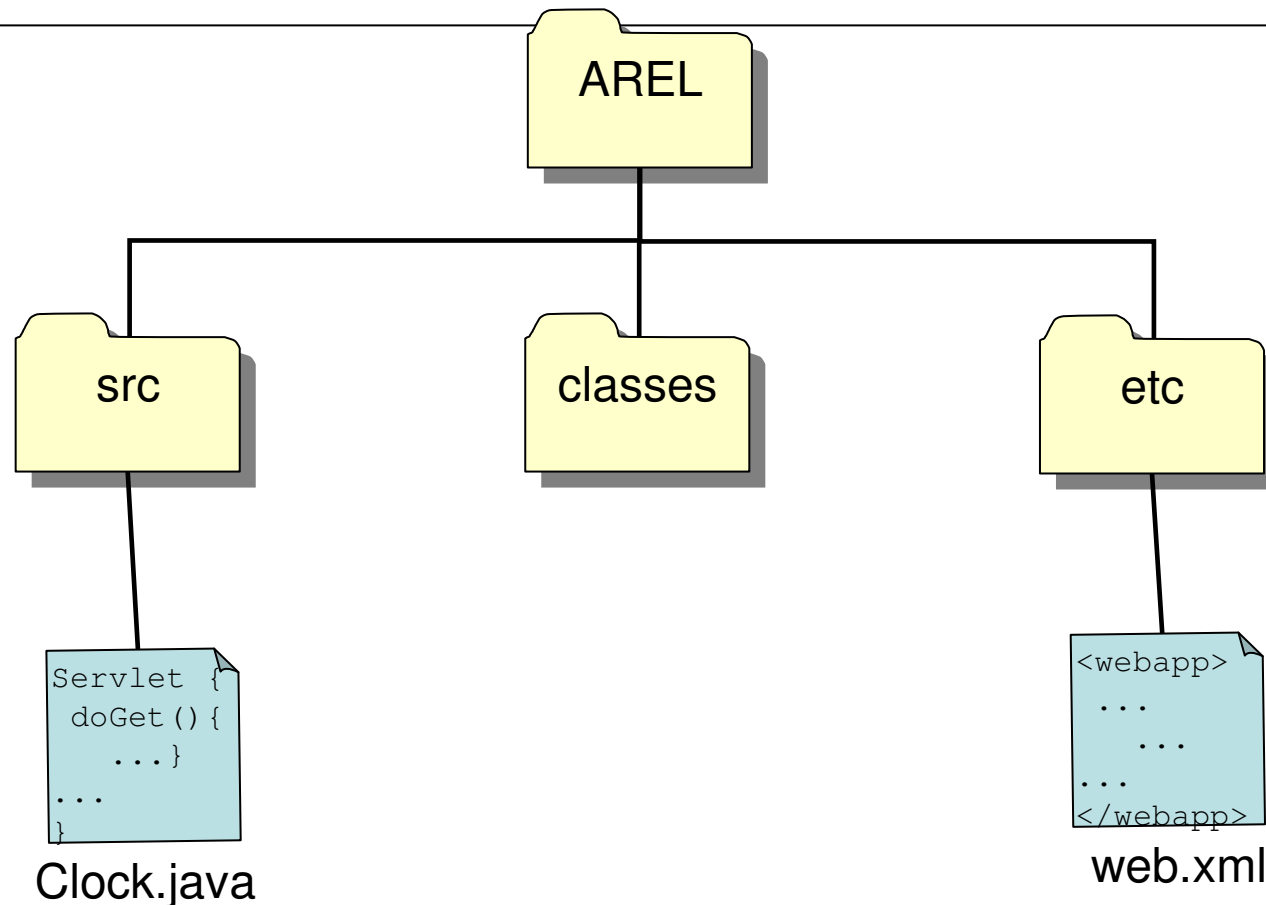
Pages dynamiques

- Le serveur Web a besoin d'aide pour faire du dynamique
 - Autrefois, les CGI (Common Gateway Interface)
 - Aujourd'hui, un conteneur Java EE avec servlets (ex : Tomcat)



Un servlet en 9 étapes (1/9)

- Créer l'arborescence suivante



Un servlet en 9 étapes (2/9)

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
```

- Ecrire un servlet Clock.java et le mettre dans src

```
public class Clock extends HttpServlet {
    public void doGet(HttpServletRequest request,
                      HttpServletResponse response)
        throws IOException {
        PrintWriter out = response.getWriter();
        java.util.Date today = new java.util.Date();
```

Servlet classique :
redéfinit doGet()
de HttpServlet

```
String docType = "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 \"+
                  \"Transitional//EN\">\n";
```

Code HTML
incorporé
dans Java

```
    out.println(docType);
    out.println("<html>");
    out.println("<head>\n<title>Clock</title>\n</head>");
    out.println("<body>\n" +
                "<h1>Time on server</h1>\n" +
                "<p>" + today + "</p>\n" +
                "</body>");
    out.println("</html>");
```

```
}
```

```
}
```

Un servlet en 9 étapes (3/9)

- Créer un descripteur de déploiement web.xml et le mettre dans etc

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<web-app version="2.4" xmlns="http://java.sun.com/xml/ns/j2ee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
  http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">
```

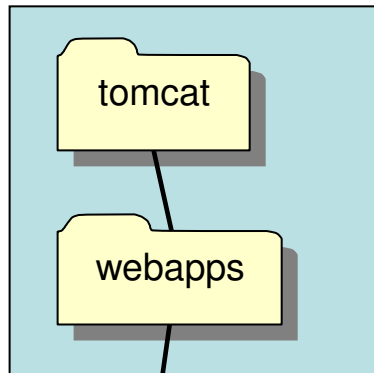
```
<servlet>
  <servlet-name>The Clock</servlet-name>
  <servlet-class>Clock</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>The Clock</servlet-name>
  <url-pattern>/Serv1</url-pattern>
</servlet-mapping>
</web-app>
```

classe Java

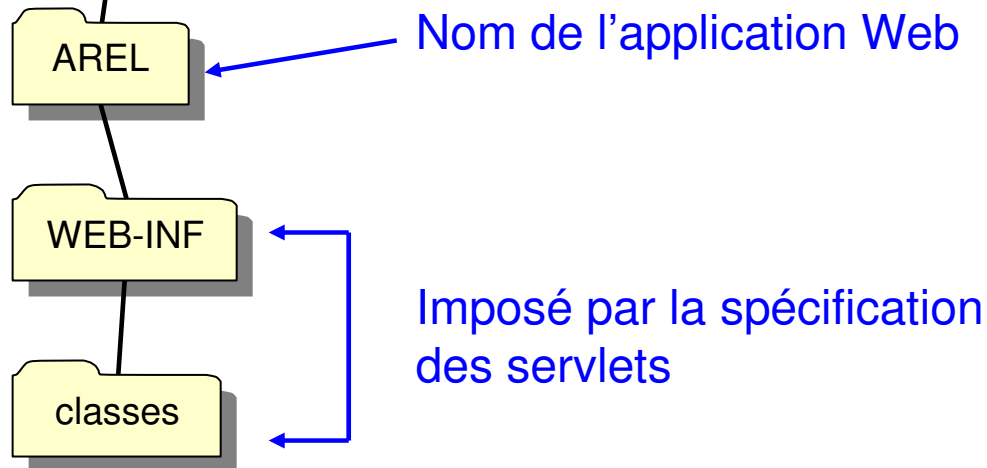
nom utilisé par le client dans sa requête

<servlet-name> permet de lier <servlet> et <servlet-mapping>

Un servlet en 9 étapes (4/9)



- Créer l'arborescence suivante **dans** l'arborescence existante de tomcat

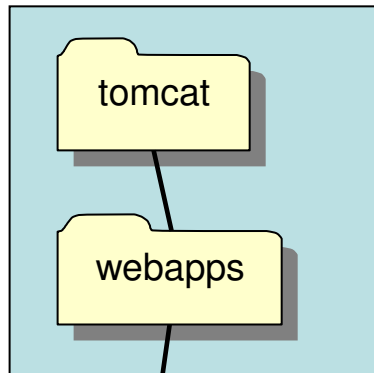


Un servlet en 9 étapes (5/9)

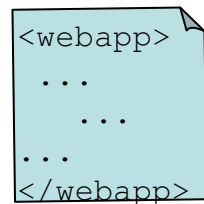
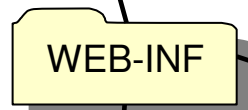
- A partir du répertoire AREL, compiler le servlet

```
% cd AREL  
% javac -classpath ~tomcat/lib/servlet-api.jar  
-d classes src/Clock.java
```

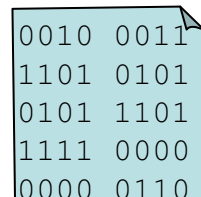
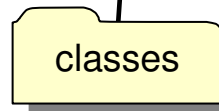
Un servlet en 9 étapes (6/9)



- Copier Clock.class dans WEB-INF/classes
- Copier web.xml dans WEB-INF



web.xml



Clock.class

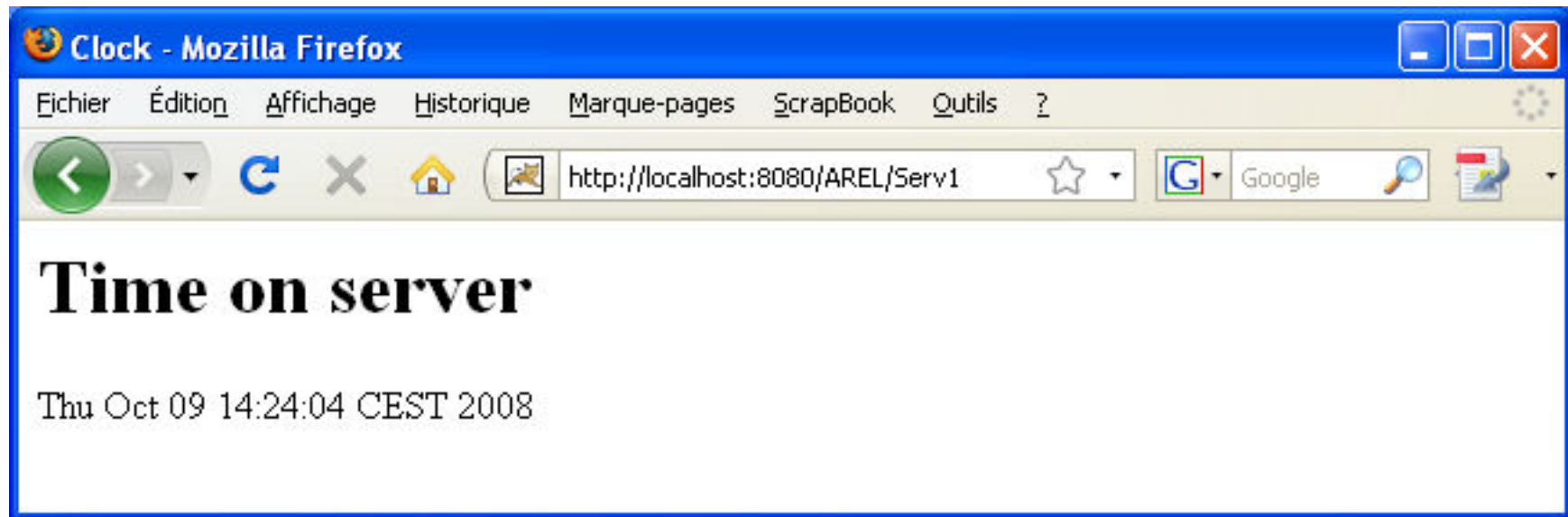
Un servlet en 9 étapes (7/9)

- A partir du répertoire de tomcat, démarrer tomcat

```
% cd tomcat  
% bin/startup.sh
```

Un servlet en 9 étapes (8/9)

- Lancer un navigateur et entrer l'URL suivante
 - `http://localhost:8080/AREL/Serv1`
- Le navigateur affiche :



Un servlet en 9 étapes (9/9)

- Fermer Tomcat à chaque modification de la classe servlet ou du descripteur de déploiement

```
% cd tomcat  
% bin/shutdown.sh
```

Problème : HTML dans Java

1. C'est laid !

```
out.println(docType);
out.println("<html>");
out.println("<head>\n<title>Clock</title>\n</head>");
out.println("<body>\n" +
            "<h1>Time on server</h1>\n" +
            "<p>" + today + "</p>\n" +
            "</body>");
out.println("</html>");
```

2. Les ingénieurs EISTI connaissent Java, mais les développeurs Web, NON !!

- Difficile de séparer les différentes tâches du développement (Web code vs. Business code)

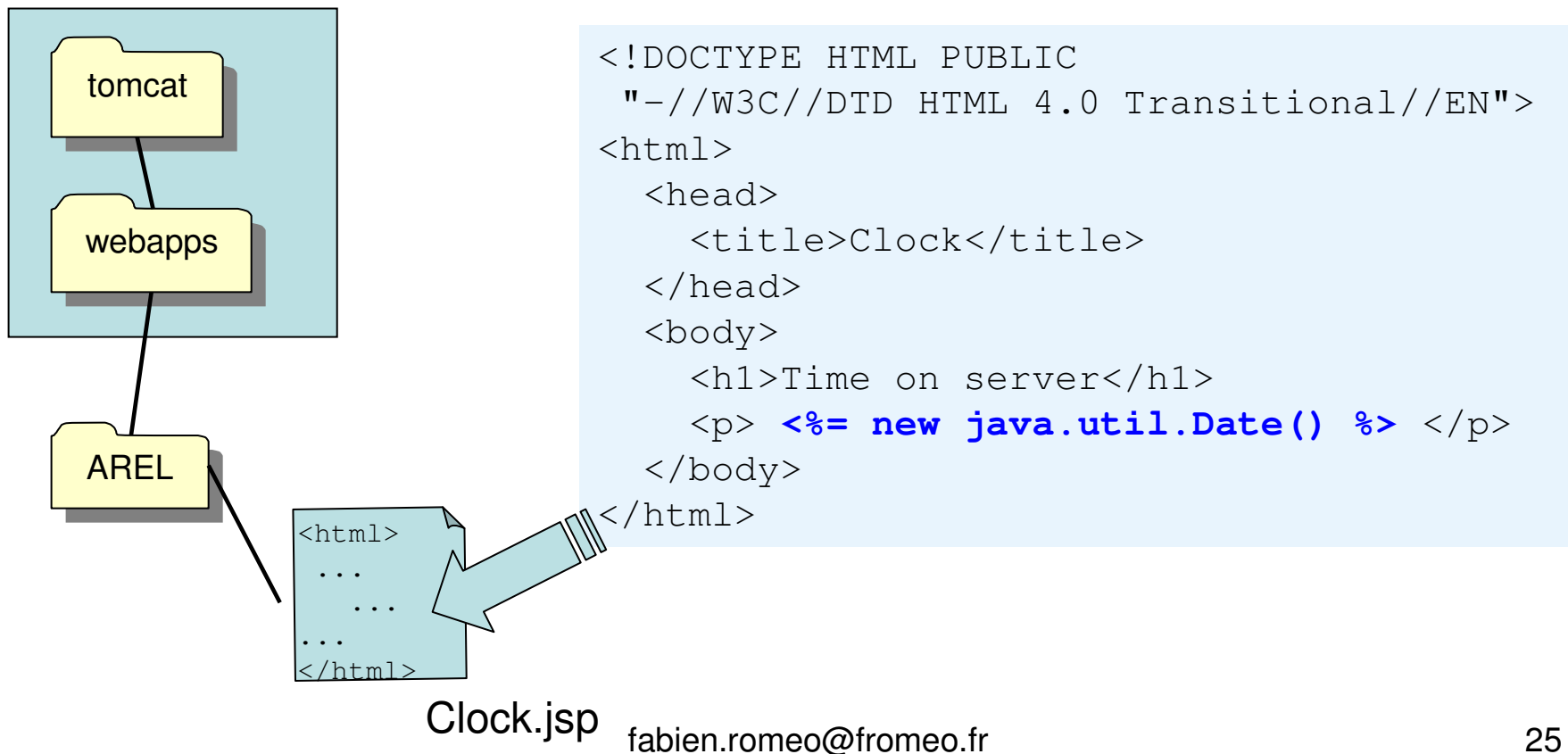
Solution : Java dans HTML

- Une JSP est identique à une page HTML sauf que l'on peut rajouter du Java dedans

```
<!DOCTYPE HTML PUBLIC
  "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
  <head>
    <title>Clock</title>
  </head>
  <body>
    <h1>Time on server</h1>
    <p> <%= new java.util.Date() %> </p>
  </body>
</html>
```

Une JSP en 1 étape

- Ecrire une JSP Clock.jsp et la mettre dans AREL (le répertoire de notre application Web)



Environnements de développement

- Le cycle *Développement-Déploiement-Exécution* est trop complexe à votre goût ?
- Les IDE sont là pour vous assister !

