

# Applications orientées données (NSY135)

## 3 – Environnement Java

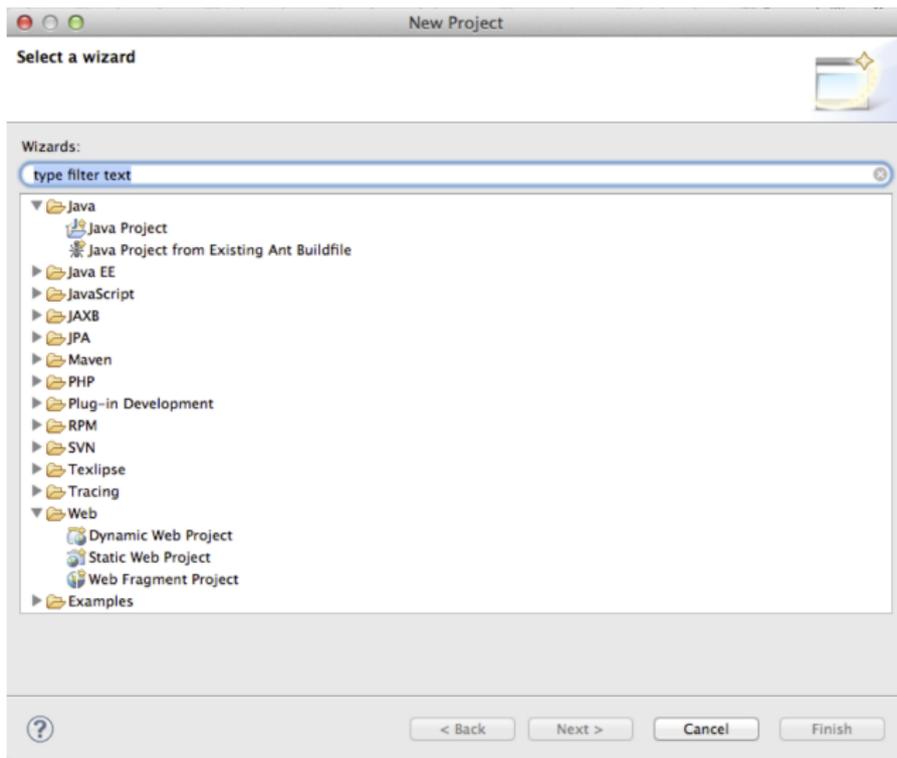
Auteurs: Raphaël Fournier-S'niehotta et Philippe Rigaux  
([philippe.rigaux@cnam.fr](mailto:philippe.rigaux@cnam.fr),[fournier@cnam.fr](mailto:fournier@cnam.fr))

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

# Plan du cours

## S3 Création d'une application JEE

# Création du projet



# Création du projet

**New Dynamic Web Project**

**Dynamic Web Project**  
Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application.

Project name:

Project location

Use default location

Location:

Target runtime

Dynamic web module version

Configuration

A good starting point for working with Apache Tomcat v7.0 runtime. Additional facets can later be installed to add new functionality to the project.

EAR membership

Add project to an EAR

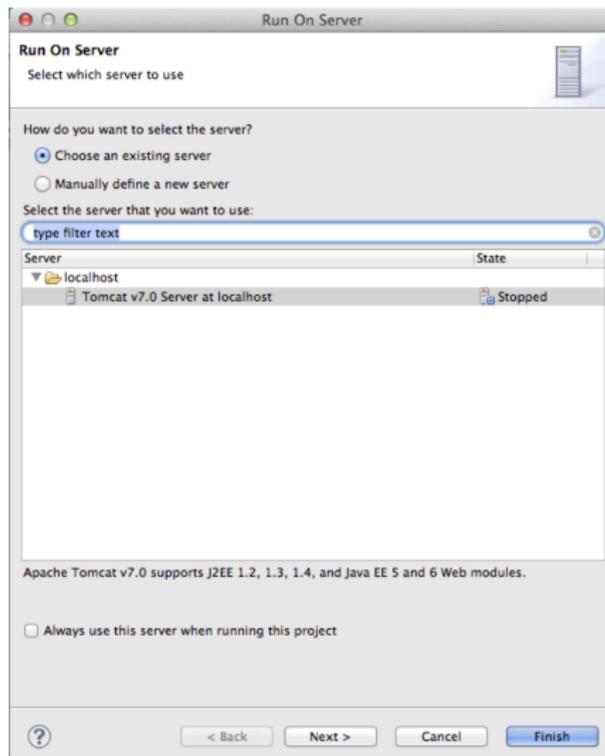
EAR project name:

Working sets

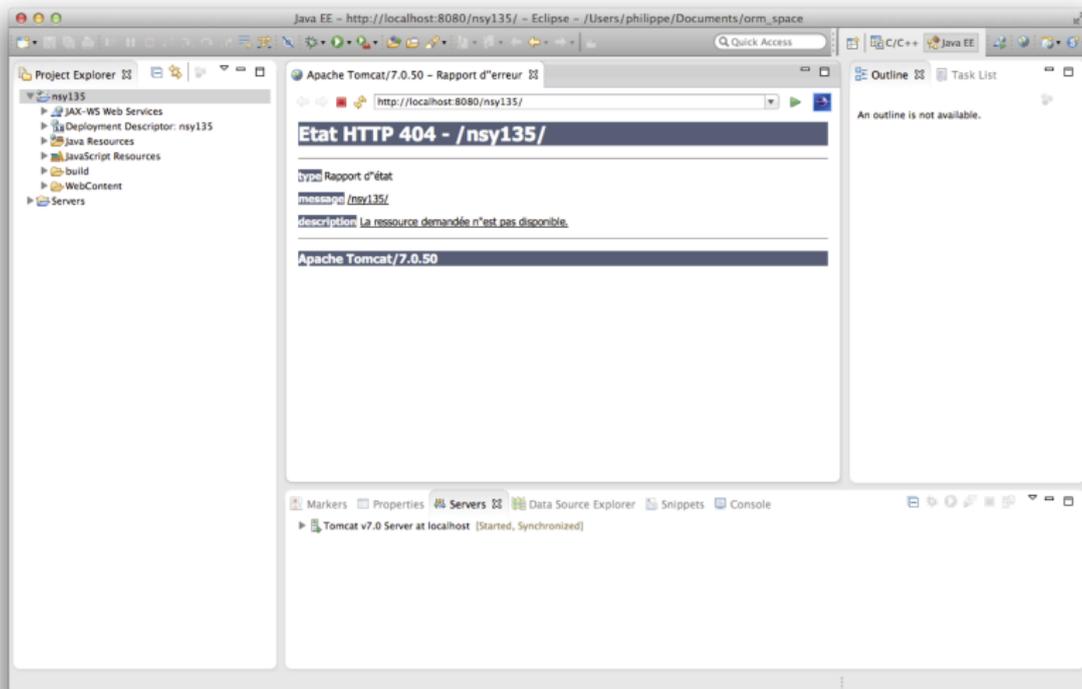
Add project to working sets

Working sets:

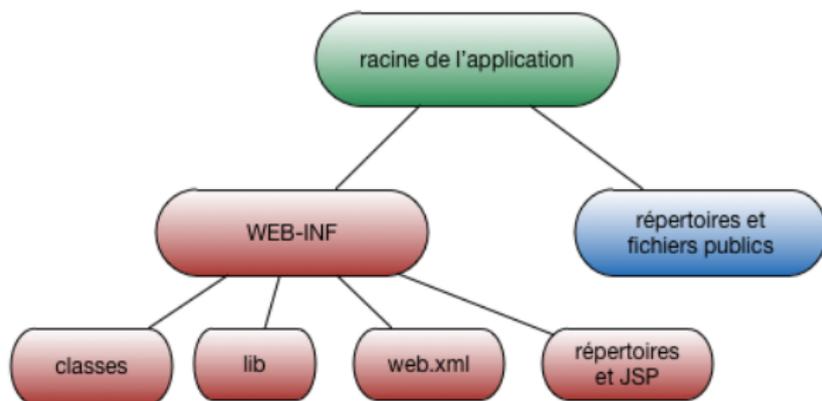
# Exécuter sur le serveur



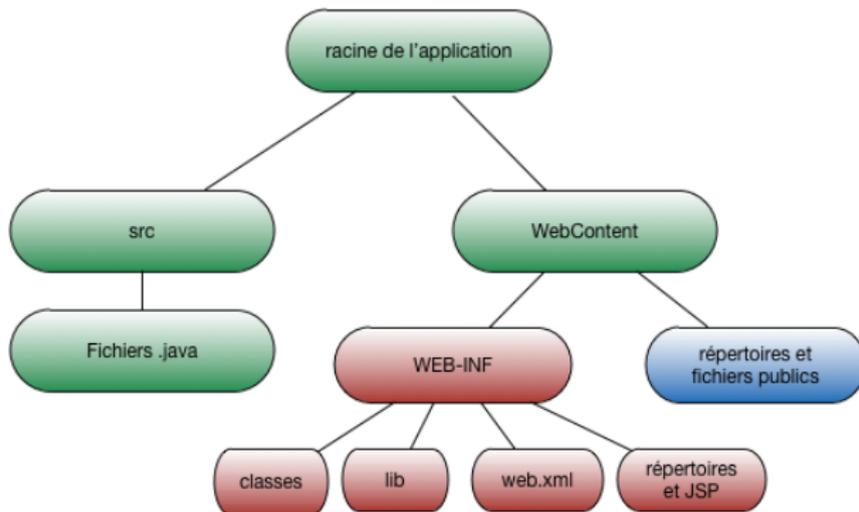
# Erreur 404



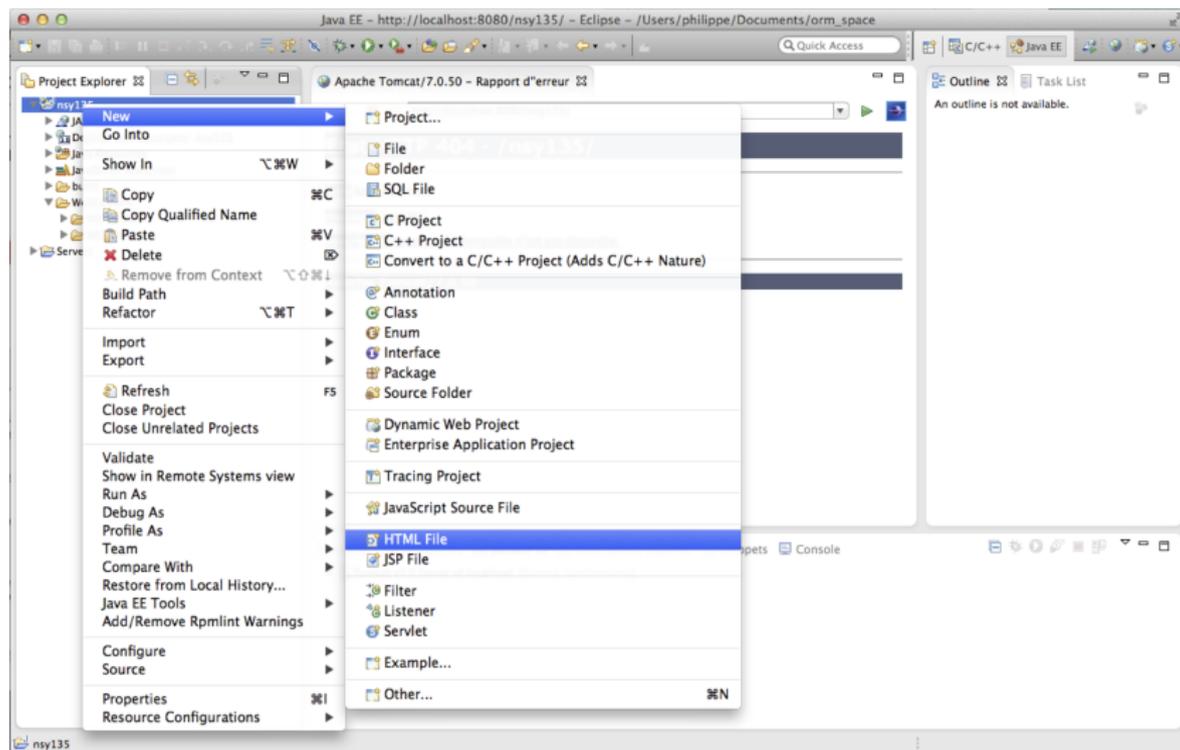
## Structure d'une application JEE



## Structure d'une application JEE dans Eclipse



# Nouveau document HTML



# WebContent/maPage.html

---

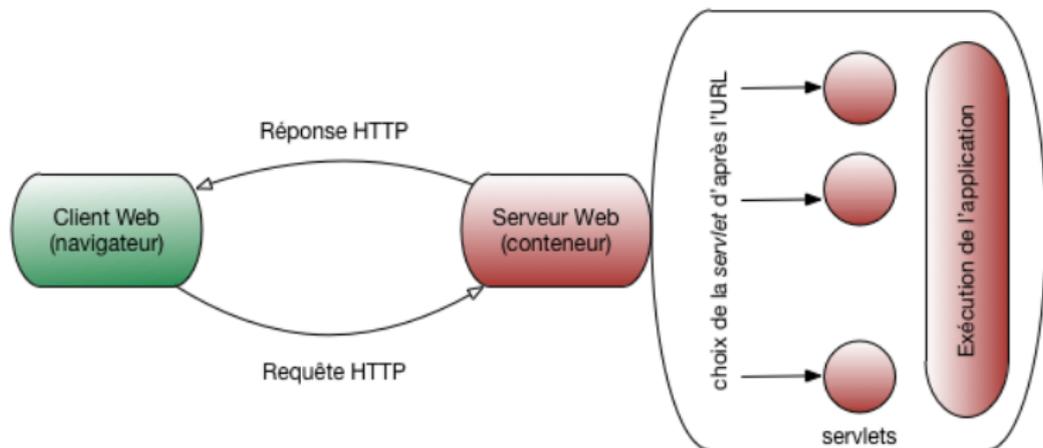
```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01
    Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Ma première page HTML</title>
  </head>
  <body>
    Ma première page HTML pour mon application JEE.
  </body>
</html>
```

---

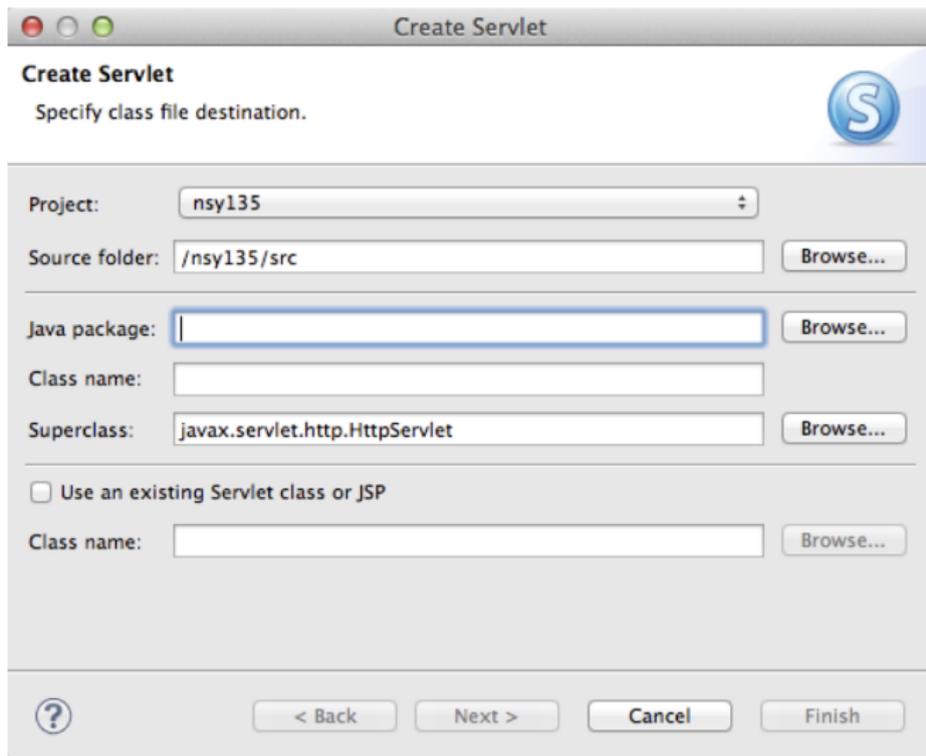
# Servlet

- (rappel) on cherche à développer une application **dynamique**
- générant du code HTML à la volée
  
- on va pour cela utiliser une **Servlet** :
  - c'est un objet Java
  - avec une interface pour recevoir les requêtes d'un client Web
  - capable de répondre, en général par un document HTML
  - contenu dans le serveur ("conteneur")

# Conteneur d'application et servlets



# Création d'une servlet dans Eclipse

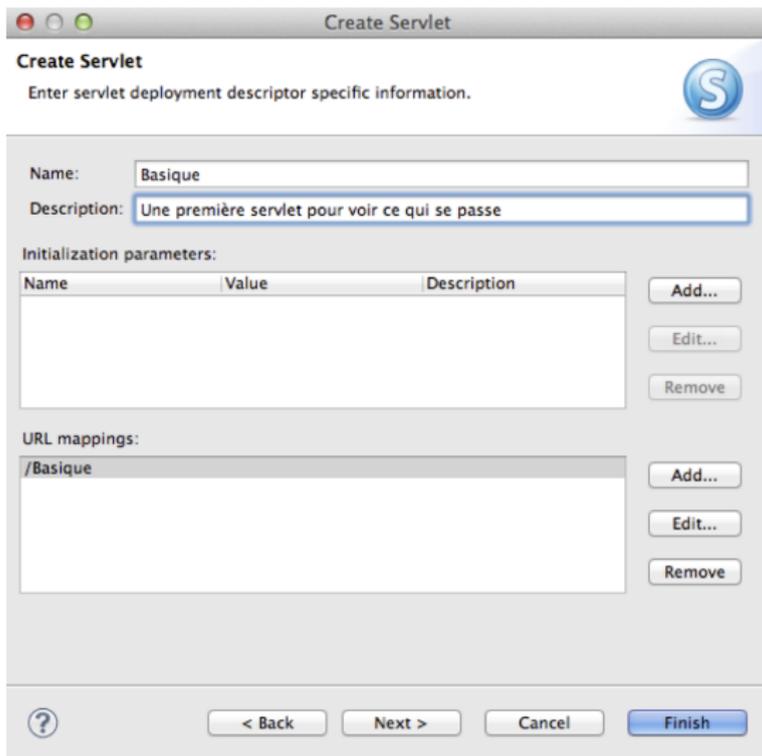


The screenshot shows the 'Create Servlet' dialog box in Eclipse. The title bar reads 'Create Servlet'. Below the title bar, the text 'Create Servlet' is followed by 'Specify class file destination.' and a blue circular icon with a white 'S'. The dialog contains several input fields and buttons:

- Project:** A dropdown menu showing 'nsy135'.
- Source folder:** A text field containing '/nsy135/src' and a 'Browse...' button.
- Java package:** An empty text field with a blue border and a 'Browse...' button.
- Class name:** An empty text field.
- Superclass:** A text field containing 'javax.servlet.http.HttpServlet' and a 'Browse...' button.
- Use an existing Servlet class or JSP**
- Class name:** An empty text field and a 'Browse...' button.

At the bottom of the dialog, there is a help icon (question mark in a circle) and four buttons: '< Back', 'Next >', 'Cancel', and 'Finish'.

# Création d'une servlet dans Eclipse



**Create Servlet**

Enter servlet deployment descriptor specific information.

Name:

Description:

Initialization parameters:

Name	Value	Description
------	-------	-------------

Add...  
Edit...  
Remove

URL mappings:

/Basique
----------

Add...  
Edit...  
Remove

? < Back Next > Cancel Finish

# Servlet

```
package tests;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class Basique
 */
@WebServlet(description = "Une première servlet pour voir ce qui se passe", urlPatterns = { "/Basique" })
public class Basique extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public Basique() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        // TODO Auto-generated method stub
    }
}
```

## GET et POST

---

```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

    response.setContentType("text/html");

    response.setCharacterEncoding( "UTF-8" );

    PrintWriter out = response.getWriter();

    out.println("Ma première <i>servlet</i> s'exécute");

}
```

---

## Exécution de notre première servlet

