



# JAKARTA EE

## Les servlets



JAKARTA EE

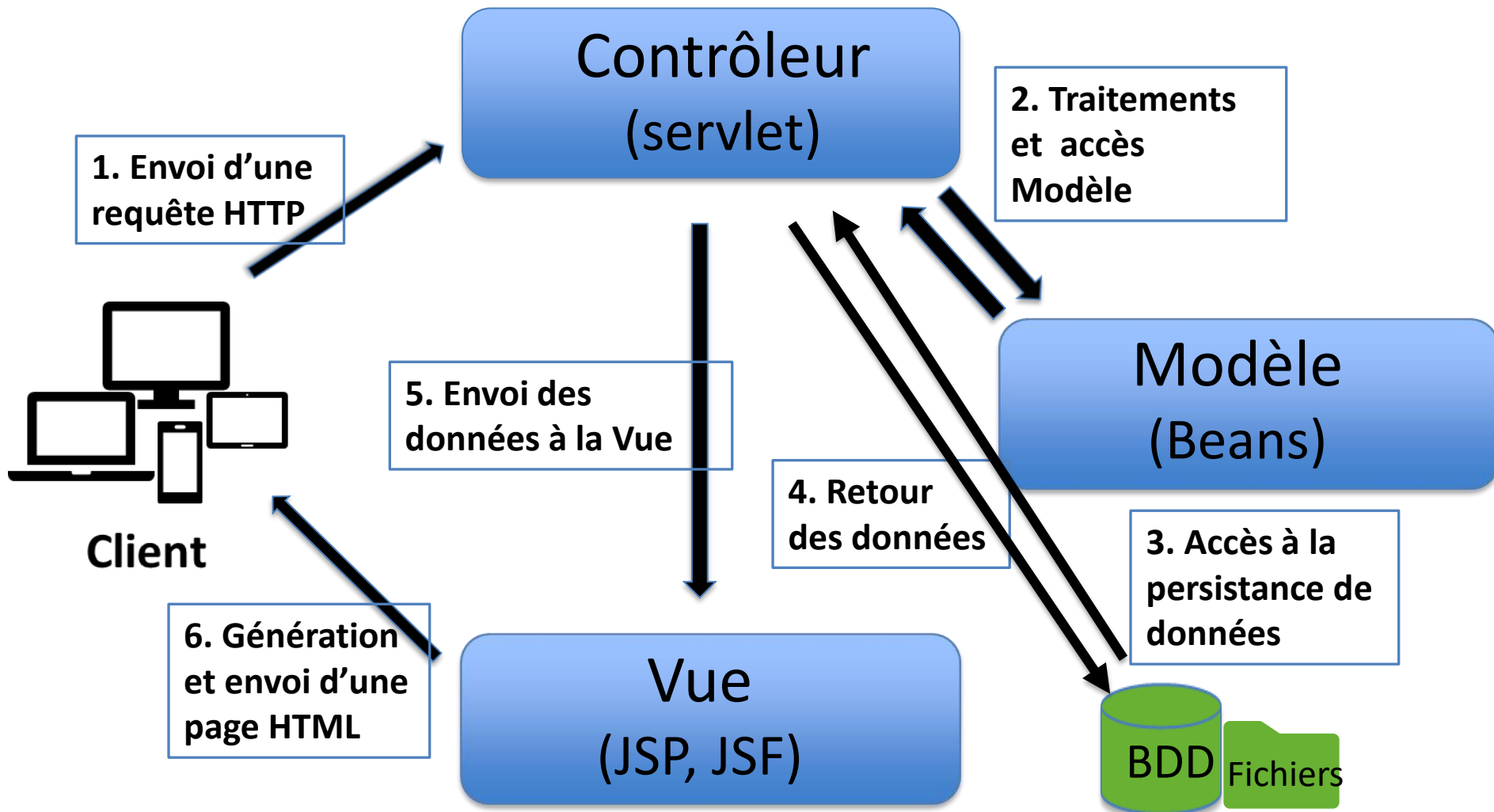
1. Jakarta EE : Les servlets
2. Jakarta EE : Rôle des servlets
3. Les servlets : Cycle de vie
4. Les servlets : héritage
5. Les servlets : La méthode init()
6. Les servlets : Les méthodes doGet() et doPost()
7. Les servlets : la classe HttpServletRequest
8. Les servlets : la classe HttpServletResponse

# 1. Jakarta EE : Les servlets



- Un ou une servlet est un objet d'une classe Java traitant les requêtes provenant du Web, généralement des requêtes HTTP ou HTTPS
- Il peut utiliser les API Java et fait le lien entre le Modèle et la Vue
- Il est compilé, multi-threadé et s'exécute dans un conteneur de servlet (serveur Web)

## 2. Jakarta EE : Rôle des servlets



### 3. Les servlets : Cycle de vie



- ➡ 1. Le client envoie la requête HTTP au serveur Web qui transfère la requête au moteur de servlets
- ➡ 2. Le moteur vérifie que la servlet estinstanciée, sinon il crée une instance de la servlet et appelle la méthode **init()** de celle-ci
- ➡ 3. Le moteur appelle ensuite la méthode **service()** de la servlet. Celle-ci reçoit deux objets en paramètres : un objet `ServletRequest` et un objet `ServletResponse` de la requête HTTP.

Cette méthode **service()** appelle automatiquement soit **doGet()**, soit **doPost()** en fonction de la méthode HTTP utilisée. Ces méthodes génèrent le résultat.

- ➡ 4. Le moteur retourne le résultat généré au serveur Web qui le renvoie au client.

## 4. Les servlets : héritage



### ► Packages :

javax.servlet.\*

javax.servlet.http.\*

### ► Etend de :

javax.servlet.http.HttpServlet

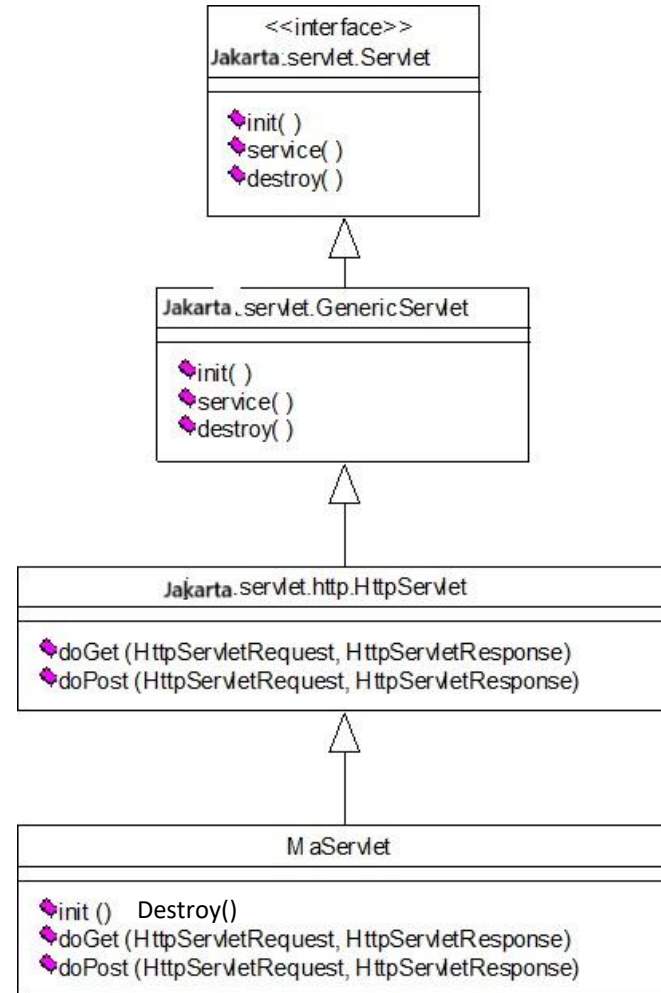
### ► Méthodes :

init()

destroy()

doGet()

doPost()



## 5. Les servlets : La méthode init()



### La méthode init()

- ➡ Elle est appelée lors du premier appel de la servlet
- ➡ Elle peut avoir 2 signatures :
  - `Public void init() throws ServletException`
  - `Public void init(ServletConfig config) throws ServletException`

L'objet de la classe `ServletConfig` permet de récupérer les valeurs des paramètres de configuration

## 6. Les servlets : Les méthodes doGet() et doPost



### Les méthodes doGet() et doPost()

- En fonction du type de requête HTTP reçue (get ou post), la méthode doGet() ou doPost() sera appelée
- doGet() et doPost() ont 2 paramètres
  - HttpServletRequest
  - HttpServletResponse



## 6. Les servlets : Les méthodes doGet() et doPost()



- ▶ Elles doivent effectuer le même traitement et la bonne pratique est de leur faire invoquer une méthode commune, ici **processRequest**

@Override

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

```
    processRequest(request, response);
```

```
}
```

@Override

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

```
    processRequest(request, response);
```

```
}
```

## 7. Les servlets : la classe HttpServletRequest



### L'objet de la classe HttpServletRequest

- Il est passé en paramètre des méthodes doGet() et doPost()
- Récupère les paramètres passés au serveur Web par le client via la requête HTTP
- Envoie à la vue les paramètres pour la génération de la page HTML

## 7. Les servlets : la classe HttpServletRequest



- Possède des méthodes dont :
  - `getParameter(String nameParameter)` : retourne la valeur du paramètre `nameParameter`
  - `getParameterValues(String nameParameter)` : retourne la liste des valeurs du paramètre `nameParameter` (exemple : liste à choix multiples)
  - `getParameterMap()` : retourne une liste des paramètres avec leur valeur associée (clé, valeur). Elle possède une méthode `containsKey(" clé ")` permettant de savoir si le paramètre se trouve dans la requête
  - `setAttribute(String nameAttribute, attributeValue)` : envoie à la JSP la valeur de l'attribut

## 8. Les servlets : la classe HttpServletResponse



### L'objet de la classe HttpServletResponse

- ➡ Il est passé en paramètre des méthodes doGet() et doPost()
- ➡ Il est composé des réponses envoyées par le serveur au navigateur
- ➡ Il possède des méthodes dont :
  - getWriter() : pour écrire des données de type texte dans la réponse
  - getOutputStream() : pour écrire des données binaires
  - sendRedirect(url) : permet une redirection



Copyright © 2025 Afpa.  
Tous droits réservés.