Parul® University

NAAC A++

Enterprise Programming using JAVA

Chapter-2: Servlets

Prof. ARNIKA PATEL
Assistant Professor
Department of CSE

Parul® University

NAAC A++

Content

INDEX

Servlets Configuration

ServletConfig in Servlet

javax.servlet.ServletConfig is an interface as a part of servlet API. For every Servlet class in our application, the web container will create one ServletConfig object and the web container will pass this object as an argument to the **public void init(ServletConfig config)** method of our Servlet class object.

Servlets Configuration

ServletConfig in Servlet

Some of the important points on ServletConfig are:

- ServletConfig is an object containing some initial parameters or configuration information created by the Servlet Container and passed to the servlet during initialization.
- ServletConfig is for a particular servlet, which means one should store servlet-specific information in web.xml and retrieve it using this object.

Servlets Configuration

ServletConfig in Servlet

Methods in the ServletConfig Interface

1. public abstract java.lang.String **getServletName()**
2. public abstract javax.servlet.ServletContext **getServletContext()**
3. public abstract java.lang.String **getInitParameter(java.lang.String)**
4. public abstract java.util.Enumeration<java.lang.String> **getInitParameterNames()**

Servlets Context

ServletContext is the object created by Servlet Container to share initial parameters or configuration information to the whole application.

Example: Suppose, the name of one's job portal is "NewWebsite.tg".

Showing the website name at the top of webpages delivered by different servlets, one needs to store the website name in every servlet inviting redundancy.

Servlets Context

Since the information shared by ServletContext can be accessed by every Servlet, it is better to go with ServletContext and retrieve the website name using getServletContext.getInitParameter("Name") whenever required.

ServletConfig	ServletContext
ServletConfig is servlet specific	ServletContext is for whole application
Parameters of servletConfig are present as name-value pair in <init-param> inside <servlet>.	Parameters of servletContext are present as name-value pair in <context-param> which is outside of <servlet> and inside <web-app>
ServletConfig object is obtained by getServletConfig() method.	ServletContext object is obtained by getServletContext() method.
Each servlet has got its own ServletConfig object.	ServletContext object is only one and used by different servlets of the application.
Use ServletConfig when only one servlet needs information shared by it.	Use ServletContext when whole application needs information shared by it

Servlets Collaboration

- The exchange of information among servlets of a particular Java web application is known as **Servlet Collaboration**.
- This enables passing/sharing information from one servlet to the other through method invocations.

Servlets Collaboration

What are the principle ways provided by Java to achieve Servlet Collaboration?

The servlet api provides two interfaces namely:

1. javax.servlet.RequestDispatcher
2. javax.servlet.http.HttpServletResponse

Servlets Collaboration

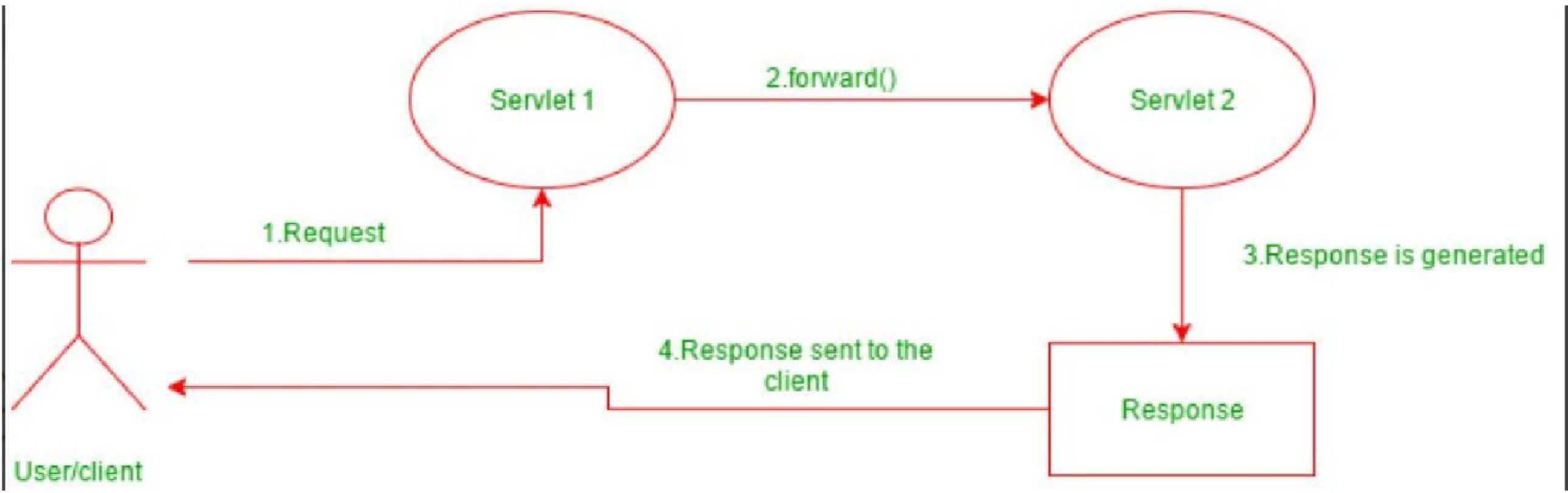
Using RequestDispatcher Interface

The RequestDispatcher interface provides the option of dispatching the client's request to another web resource, which could be an HTML page, another servlet, JSP etc. It provides the following two methods:

public void forward(ServletRequest request, ServletResponse response)throws ServletException, java.io.IOException:

Servlets Collaboration

Using RequestDispatcher Interface



Servlets Collaboration

Using RequestDispatcher Interface

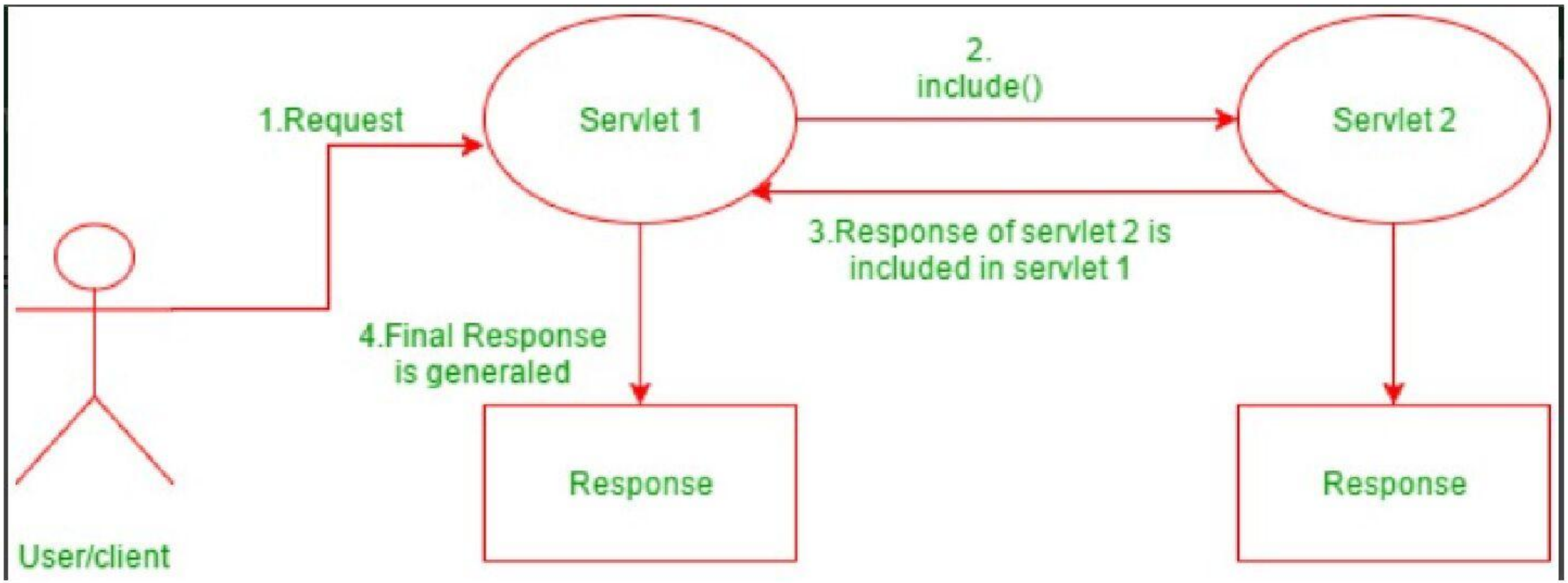
public void include(ServletRequest request, ServletResponse response)throws ServletException, java.io.IOException:

The include() method is used to include the contents of the calling resource into the called one.

When this method is called, the control still remains with the calling resource. It simply includes the processed output of the calling resource into the called one.

Servlets Collaboration

Using RequestDispatcher Interface



Servlets Collaboration

Using HttpServletResponse Interface

The HttpServletResponse interface is entrusted with managing Http responses. To achieve servlet collaboration, it uses the following method:

public void sendRedirect(String URL)throws IOException;

This method is used redirect response to another resource, which may be a servlet, jsp or an html file. The argument accepted by it, is a URL which can be both, absolute and relative. It works on the client side and uses the browser's URL bar to make a request.

Servlets Collaboration

What is the difference between forward() method of RequestDiispatcher and sendRedirect() of HttpServletResponse?

forward()	sendRedirect()
It works on the server side	It works on the client side
It sends the same request and response objects to another resource.	It always send a new request
It works only within the server.	It can be used within and outside the server.

PPT Content Resources Reference Sample:

1. **Book Reference**

   Jim Farley, William Crawford, David Flanagan. Java Enterprise in a Nutshell, O'Reilly

2. **Book Reference**

   Rocha, R., Purificação, J. (2018). Java EE 8 Design Patterns and Best Practices: Build Enterprise-ready Scalable Applications with Architectural Design Patterns. Germany: Packt Publishing..

3. **Website Reference**

   https://www.scribd.com/document/268349254/Java-8-Programming-Black-Book .

4. **Sources**

   https://developers.redhat.com/topics/enterprise-java

5. **Article**

   https://www.researchgate.net/publication/276412369_Advanced_Java_Programming

Parul® University | NAAC GRADE A++

https://paruluniversity.ac.in/