

Web 3

Lesson 11: Post/Redirect/Get

OP HET EXAMEN...



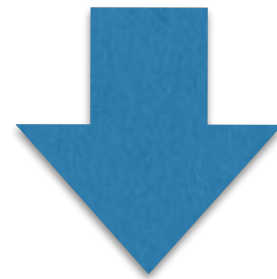
- ☑ Leg uit wat het probleem is dat Post/Redirect/Get oplost?
- ☑ Leg de figuren uit in deze slides.
- ☑ Hoe moet je Post/Redirect/Get implementeren?
- ☑ Wat is het verschil tussen forward en sendRedirect?
- ☑ ...

EXAMPLE

What if
I was submitting
an order, ...?

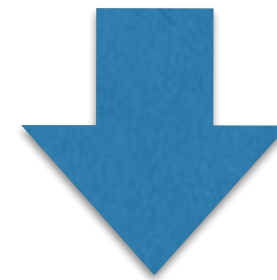
Your userid

Your password



submit

Welcome to this test site ...



refresh

Welcome to this test site ...



Confirm

To display this page, Firefox must send information that will repeat any action (such as a search or confirmation) that was performed earlier.

Cancel

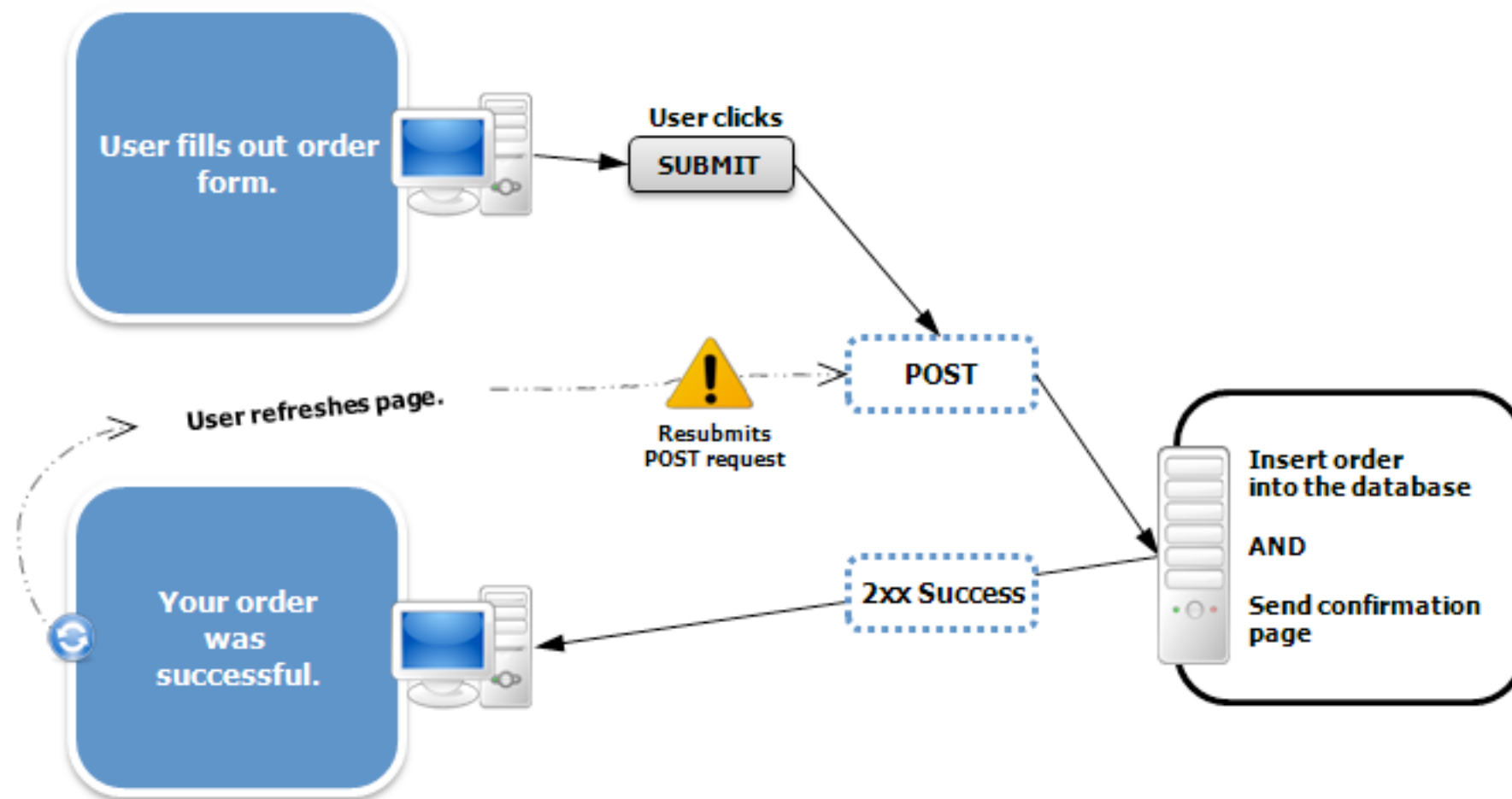
Resend



CODE

```
protected void doPost(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException {
    String action = request.getParameter("action");
    if (action.equals("login")) {
        String userId = request.getParameter("userid");
        String password = request.getParameter("password");
        if (service.getUserIfAuthenticated(userId, password);) {
            ...
            request.getRequestDispatcher("loggedin.html").forward(request, response);
        }
    }
    else
    {
        ...
        request.getRequestDispatcher("index.html").forward(request, response);
    }
}
}
```

PROBLEM



PROBLEM

- If you refresh this page
 - You can send the POST again
 - If this is a confirmation of an order you are placing ...
 - you will order it twice or more !?!?

SOLUTION

POST/**R**EDIRECT/**G**ET

After submitting a form, ...

- Do NOT forward the request
- But redirect the response
- = PRG

HTTP STATUS CODES

REMEMBER COMPUTER NETWORKS

- 1xx Informational
- 2xx Success
- 3xx Redirection
- 4xx Client Error
- 5xx Server Error

200 OK

404 Not Found

418 I'm a teapot

500 Internal Server Error

HTTP REDIRECTS

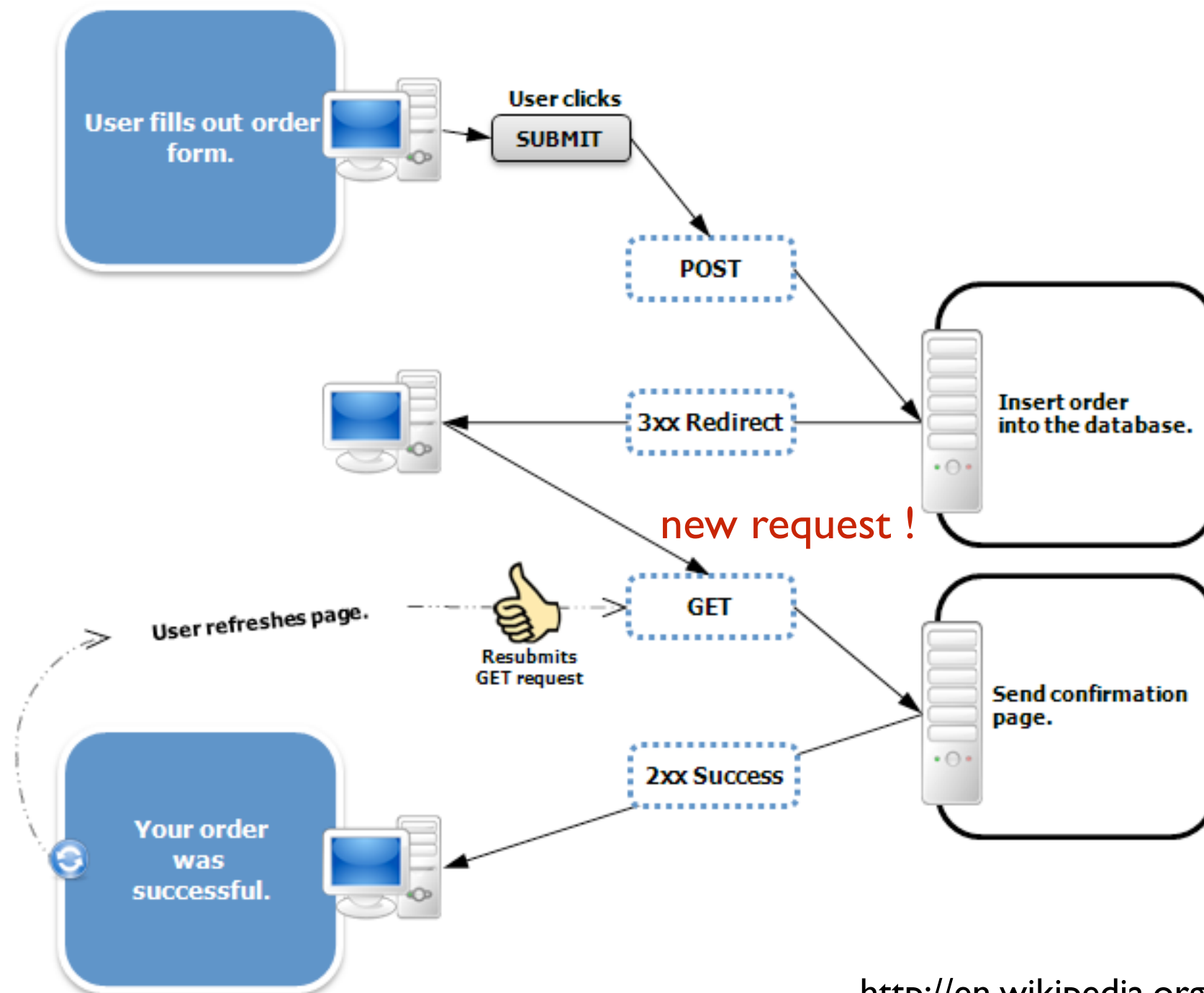
- Responses with a status code of **3xx**.
- When receiving a redirect response, the **browser** uses the new URL provided and immediately loads it
- Most of the time, the redirection is **transparent to the user**, besides a small performance hit.

CODE

```
protected void doPost(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException {
    String action = request.getParameter("action");
    if (action.equals("login")) {
        String userId = request.getParameter("userid");
        String password = request.getParameter("password");
        if (service.getUserIfAuthenticated(userId, password);) {
            ...
            response.sendRedirect("Controller?action=welcome");
        }
    }
    else
    {
        ...
        request.getRequestDispatcher("index.html").forward(request, response);
    }
}
}
```

SOLUTION

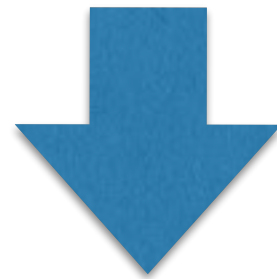
POST/REDIRECT/GET



EXAMPLE

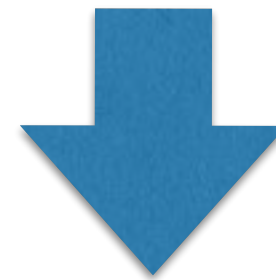
Your userid

Your password



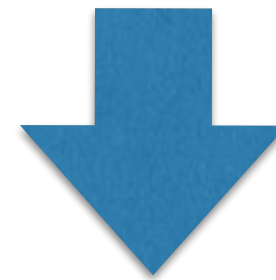
submit

Welcome to this test site ...



refresh

Welcome to this test site ...



refresh

Welcome to this test site ...

DIFFERENCES?

request.forward()

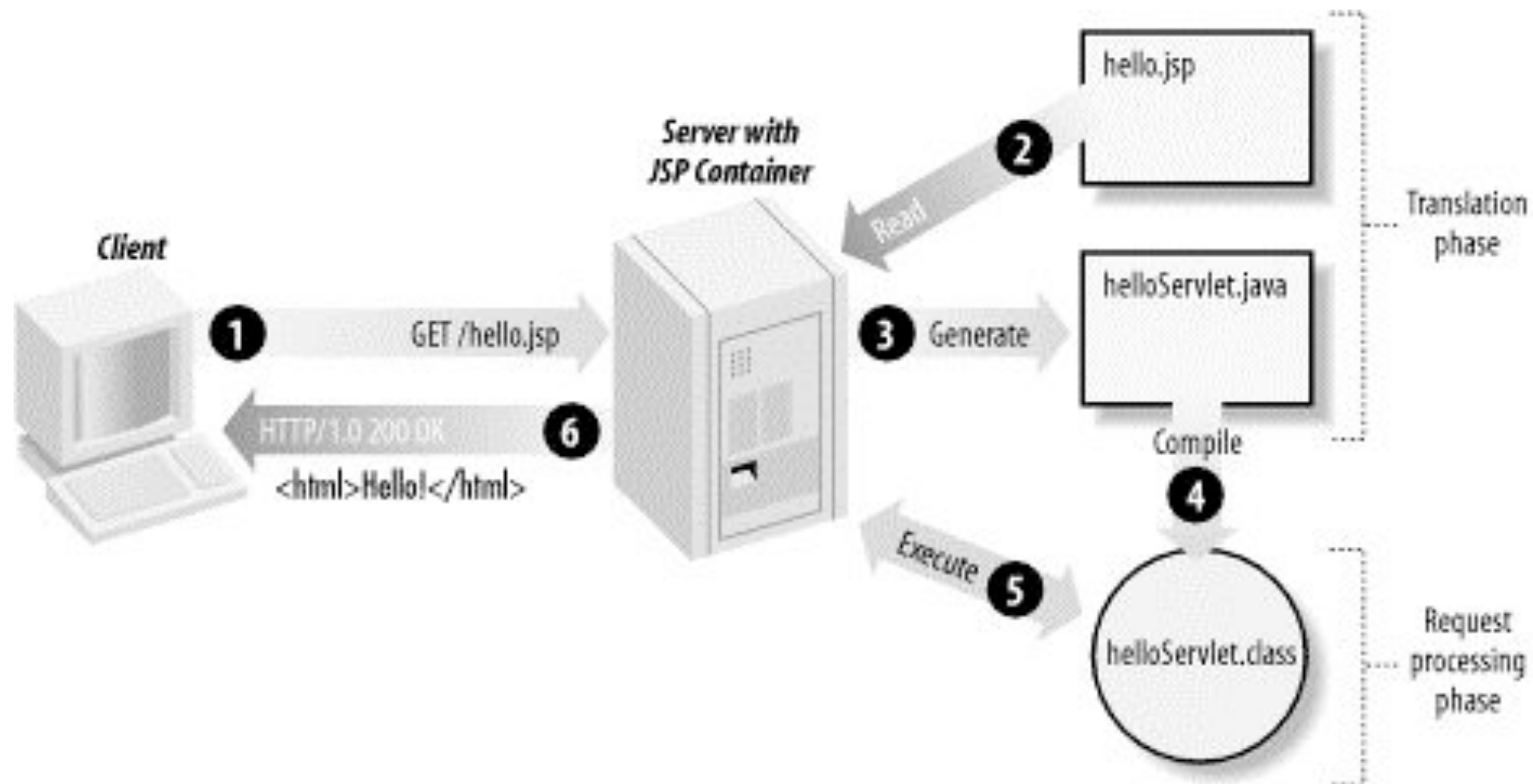
- The forward() method is executed on the server side.
- It can be used within server.
- information in the request is still available
- ...

response.sendRedirect()

- The sendRedirect() method is executed on the client side.
- It can be used within and outside the server.
- information in the request is lost
- ...

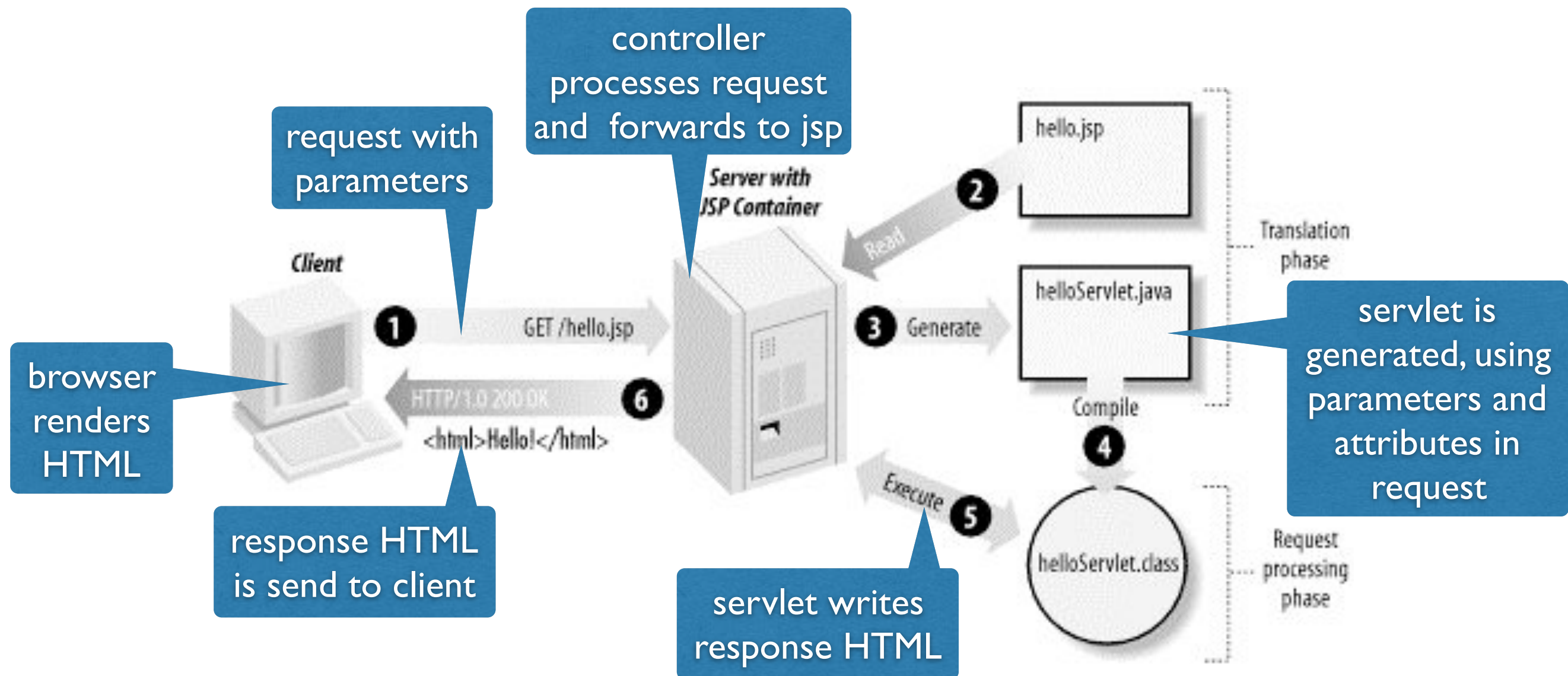
INFORMATION LOST ?

REMEMBER WEB 2



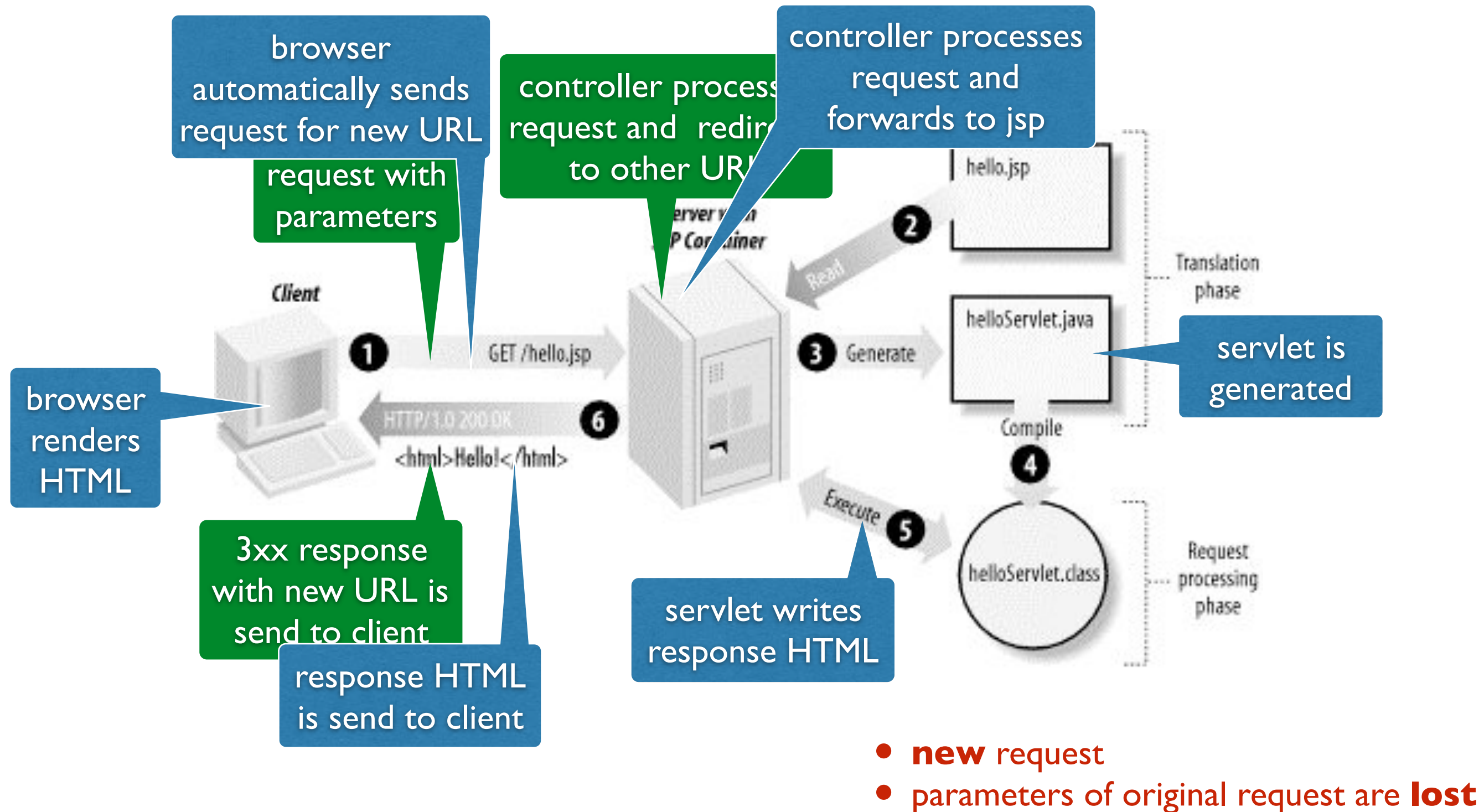
FORWARD REQUEST

NO INFORMATION LOST



REDIRECT RESPONSE

INFORMATION LOST




```
public class Controller extends HttpServlet {
    protected void processRequest(HttpServletRequest request, HttpServletResponse
        response) throws ServletException, IOException {
        String destination = "index.jsp";
        String action = request.getParameter("action");

        if (action == null) {
            destination = "index.jsp";
        }
        else if (action.equals("overview")) {
            destination = getOverview(request, response);
        }
        else if (action.equals("signUp")) {
            destination = "signUp.jsp";
        }
        else if (action.equals("confirmSignup")) {
            destination=confirmSignup(request,response);
        }
        else if (...) {
            ...
        }
        request.setAttribute("action", action);
        RequestDispatcher view = request.getRequestDispatcher(destination);
        view.forward(request, response);
    }
}
```

WARNING

Move to helper methods !



```
public class Controller extends HttpServlet {
    protected void processRequest(HttpServletRequest request, HttpServletResponse
        response) throws ServletException, IOException {
        String destination = "index.jsp";
        String action = request.getParameter("action");

        if (action == null) {
            destination = "index.jsp";
        }
        else if (action.equals("overview")) {
            destination = getOverview(request, response);
        }
        else if (action.equals("signUp")) {
            destination = "signUp.jsp";
        }
        else if (action.equals("confirmSignup")) {
            destination=confirmSignup(request,response);
        }
        else if (...) {
            ...
        }
    }
}
```

?