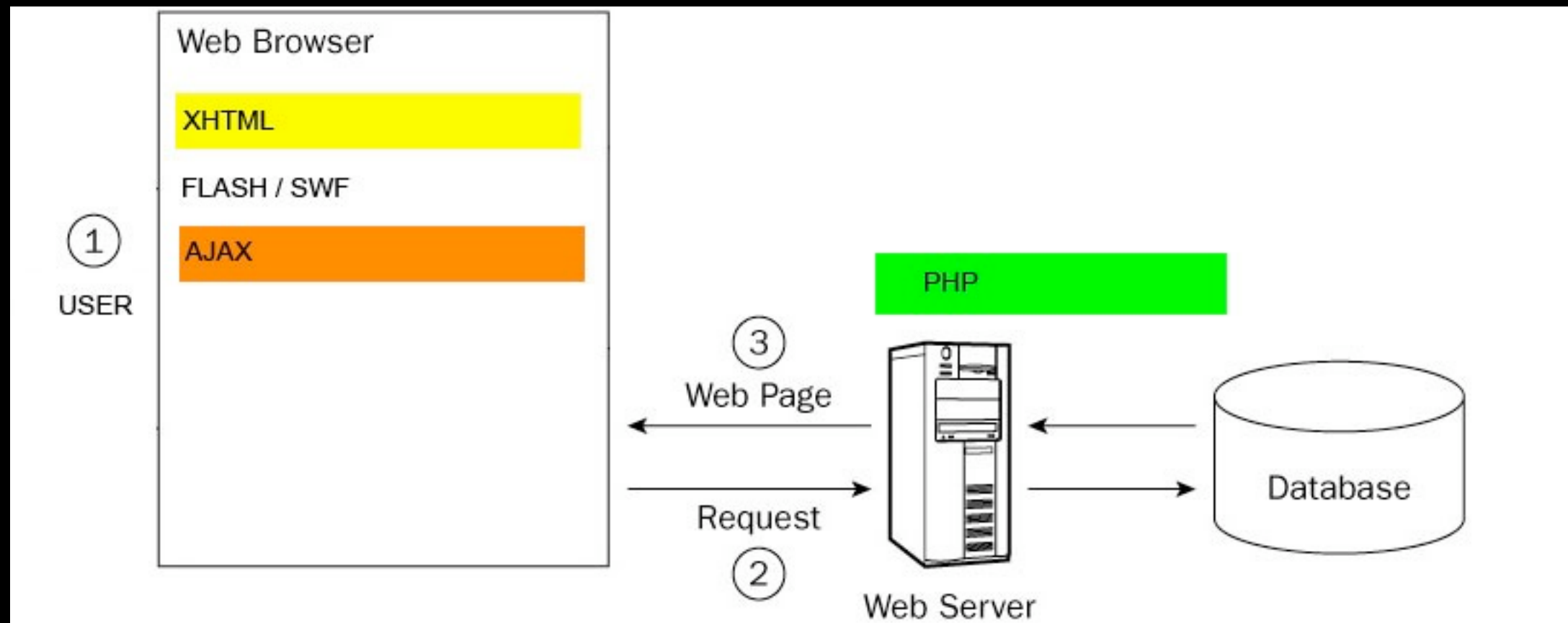


Ajax

communication client - server

andreaferracani@gmail.com



- **XHTML**: definisce la formattazione della pagina nel browser
- **FLASH**: aggiunge funzioni interattive alla pagina XHTML e può comunicare con PHP
- **AJAX**: aggiunge funzioni interattive alla pagina XHTML e può comunicare con PHP
- **PHP**: comunica con FLASH /AJAX e con il Database

Fondamenti di Ajax

- Invia richieste ad un server attraverso l'oggetto XMLHttpRequest
- Gestisce le risposte in XML o JSON
- E' in grado di recuperare un riferimento ai TAG nella pagina HTML
- Genera HTML interpretando la risposta del Server

XMLHttpRequest

Codice XHTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "  
http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd>  
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">  
  
  <head>  
  
    <link rel="stylesheet" href="../../../css/style.css" type="text/css" />  
  
    <!-- CHIAMATA A JAVASCRIPT -->  
    <script type="text/javascript" src="json2.js"></script>  
    <script type="text/javascript" src="mysql_3.js"></script>  
  
  </head>
```

XMLHttpRequest

Codice XHTML

```
//funzione che viene eseguita al caricamento della pagina
<body onload="process()">

    <div class="content">

        <h2>PHP e MYSQL</h2>
        <p>Questa lista di libri è caricata dinamicamente dal
        <strong>database</strong> con <strong>AJAX</strong>!</p>

        <!-- CONTENITORE DELLA RISPOSTA DEL SERVER -->
        <div id="myDivElement"></div>

    </div>

</body>
</html>
```

XMLHttpRequest

Codice AJAX

```
// contiene il riferimento all'oggetto XMLHttpRequest, creo l'istanza
var xmlHttp = createXmlHttpRequestObject();

// funzione per creare l'istanza cross browsers
function createXmlHttpRequestObject() {

    // contiene il riferimento all'oggetto XMLHttpRequest
    var xmlHttp;

    // se il browser è Internet Explorer 6 o più vecchio
    if(window.ActiveXObject) {
        try {
            xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");
        } catch (e) {
            xmlHttp = false;
        }
    }
}
```

XMLHttpRequest

Codice AJAX

```
// se è Mozilla o altri
else {
    try {
        xmlhttp = new XMLHttpRequest();
    } catch (e) {
        xmlhttp = false;
    }
}

// ritorna l'oggetto o errore
if (!xmlhttp) {
    alert("Error creating the XMLHttpRequest object.");
} else {
    return xmlhttp;
}
}
```

XMLHttpRequest

Codice AJAX

```
// funzione che apre la connessione con PHP attraverso XMLHttpRequest e  
mette l'oggetto in attesa della risposta
```

```
function process() {  
    // se esiste l'oggetto XMLHttpRequest  
    if (xmlHttp) {  
  
        // cerca di connetterti al server  
        try {  
            // apri la connessione al file PHP  
            xmlHttp.open("GET", "index.php", true);  
  
            //metti in attesa  
            xmlHttp.onreadystatechange = handleRequestStateChange;  
  
            //invia la richiesta  
            xmlHttp.send(null);  
        }  
        // messaggio di errore  
        catch (e) {  
            alert("Can't connect to server:\n" + e.toString());  
        }  
    }  
}
```

XMLHttpRequest

Codice AJAX

```
// funzione che controlla lo stato della richiesta sul server

function handleRequestStateChange() {

    // quando readyState è 4, la risposta è completata
    if (xmlHttp.readyState == 4) {

        // controllo che lo stato della richiesta sia OK
        if (xmlHttp.status == 200) {
            try {

                // fai qualcosa con la risposta del server
                handleServerResponse();
            } catch(e) {
                // mostra un messaggio di errore
                alert("Error reading the response: " + e.toString());
            }
        } else {
            // mostra un messaggio di stato
            alert("There was a problem retrieving the data:\n" +
                xmlHttp.statusText);
        }
    }
}
```

XMLHttpRequest

Codice AJAX

```
// gestisce la risposta del server

function handleServerResponse() {

    // leggo la risposta con una libreria
    responseJSON = JSON.parse(xmlHttp.responseText);

    // genero HTML output
    var html = "";

    // itero nell'array e creo HTML
    for (var i=0; i<responseJSON.books.length; i++) {
        html += responseJSON.books[i].title + "<br/>";
    }

    // ottengo una refernce al contenitore
    myDiv = document.getElementById("myDivElement");

    // inserisco l'HTML
    myDiv.innerHTML = "<p>Server says: </p>" + html;
}
```

XMLHttpRequest

Codice PHP

```
//File che serve a fare la query al Database, formattare i dati e  
restituirli ad Ajax
```

```
<?php
```

```
// connettiti al database
```

```
$mysqli = new mysqli(DB_HOST, DB_USER, DB_PASSWORD, DB_DATABASE);
```

```
// formula la query
```

```
$query = 'SELECT book_id, book_title FROM books';
```

```
// esegui la query
```

```
$result = $mysqli->query($query);
```

```
// cicla sul risultato
```

```
while ($row = $result->fetch_array(MYSQLI_ASSOC)) {
```

```
    // estrai book id e title
```

```
    $book_id = $row['book_id'];
```

```
    $book_title = $row['book_title'];
```

```
    // output dei dati per ajax lato client
```

```
    echo 'Name of user #' . $user_id . ' is ' . $user_name .
```

```
    '<br/>';
```

```
}
```

XMLHttpRequest

Codice PHP

```
// chiudi l'input stream
$result->close();

// chiudi la connessione al db
$mysqli->close();
?>
```