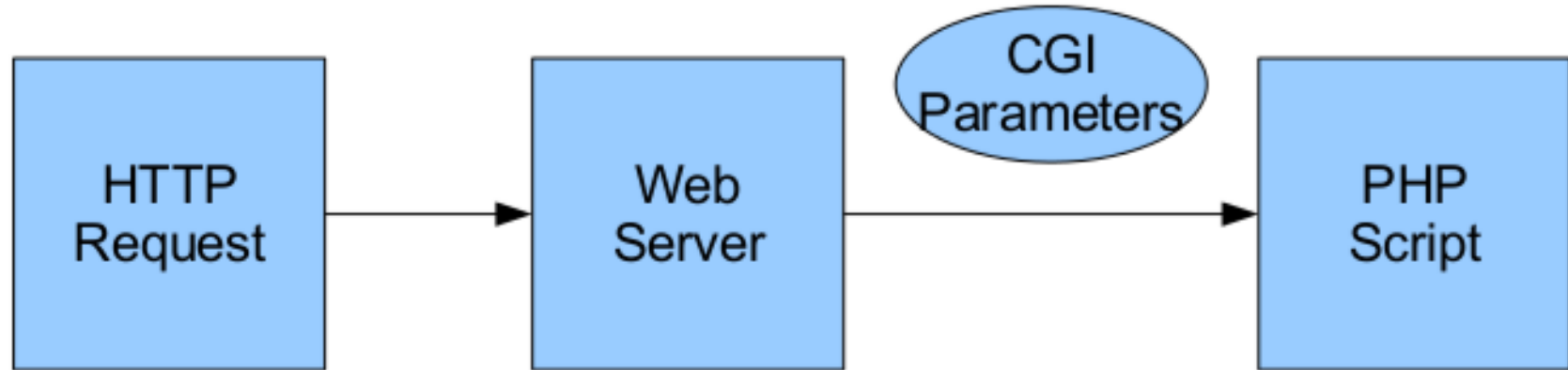


# **More PHP Details**

# In This Lecture

- Built in variables.
- Form processing using PHP.

# In PHP



- The Web server passes HTTP request info as parameters to be used by the PHP script.
- These CGI parameters are automatically “stored” in built-in variables for ready use in PHP.

# `$_GET` and `$_POST`

(formerly `$HTTP_GET_VARS*`, `$HTTP_POST_VARS*`)

- Associative array containing data submitted to the script using GET and POST method respectively.
  - `$_GET['name']` is the value of data associated with '**name**' say in
    - GET `process.php?name=pedro` HTTP/1.1
  - `$_POST['name']` is the value associated with '**name**' say in

POST `process.php` HTTP/1.1

{Other headers}

**name=pedro**&other=juan

# Form Processing in PHP

- On the browser: everything that has value is submitted
  - Textfield value, textarea value, submit button used, etc.
  - **Name** attribute is particularly useful for easy reference to form data.

# `$_FILES`

- Associative array of items uploaded to current script via POST.
- Multi-dimensional array.

# Processing textfields and other Single Valued Data items.

- Straightforward: use the name as used in the form to refer to the data item.

```
<form  
  action='process.php'  
  method='get'>  
  
  <input type='text'  
        name='fname' />  
  
  <input type='submit' />  
  
</form>
```

---

```
//In process.php  
  
$fname= $_GET['fname'];
```



# Forms with Multiple Submit buttons

```
<form action='process.php'  
method='get'>
```

```
<input type='submit'  
      name='open'  
      value='Open' />
```

```
<input type='submit'  
      name='close'  
      value='Close' />
```

```
</form>
```

```
//in process.php
```

```
if (isset($_GET['open']))  
{
```

```
    //if the Open submit  
    //was used used
```

```
}
```

```
//isset function checks
```

```
//for existence of var.
```



# Submitting Multi-valued data items

(check box groups and multi-select boxes)

- For PHP to effectively process multi-valued data items, the name in the form should contain “[]” so values are automatically saved as arrays.

```
<form action="..." >
  <input type='checkbox'
        name='choices[]'
        value='red' /> Red
  <input type='checkbox'
        name='choices[]'
        value='blue' />Blue
  ...
</form>
```

# Submitting Multi-valued data items

(check box groups and multi-select boxes)

```
//in process.php
```

```
$c = $_GET['choices'];
```

```
//$c is an array
```

```
//so $_GET['choices'][0] is possible
```

```
foreach ($c as $index=>$value) {
```

```
    echo "Choice No. $index is  
    $value";
```

```
}
```

# File uploads

<form


enctype="multipart/form-data"

method="post"

action="processupload.php">

<input type='hidden' name='MAX\_FILE\_SIZE'  
value='3000' />

PHP-specific requirement  
Must precede file input field



Send file: <input type='file' name='ufile' />

<input type='submit' />

</form>

# File uploads

```
//To get the file details
echo "Orig name of file " . $_FILES['ufile']['name'] .
    "<br/>";
echo "File Type: " . $_FILES['ufile']['type'] . "<br/>";
echo "Size: " . $_FILES['ufile']['size'] . " bytes <br/>";
echo "File name in server is " .
    $_FILES['ufile']['tmp_name'] . "<br/>";
echo "Error : " . $_FILES['ufile']['error'] . "<br/>";
//then you can use file functions in PHP
//such as move_upload_file( );
```

# Multiple File Uploads

```
<form action="file-upload.php" method="post"  
enctype="multipart/form-data">
```

```
  Send these files:<br />
```

```
  <input name="userfile[]" type="file" /><br />
```

```
  <input name="userfile[]" type="file" /><br />
```

```
  <input type="submit" value="Send files" />
```

```
</form>
```