

Curs 3

2018/2019

# Programarea aplicațiilor web

# Introducere

---

# PAW

- Programarea aplicațiilor web
  - An V RC
    - 2C/1L/1P

# Nota

- An V
  - 33% E
  - 66% Aplicatii
    - 33% L (0%)
    - 33% P (66%)

# Documentatie

- <http://rf-opto.etti.tuiasi.ro> + WWW



The screenshot displays the website for RF-OPTO at ETTI. The header includes the ETTI logo, the text 'RF-OPTO', and the University of Technical Sciences 'Gheorghe Asachi' Iasi logo. A navigation menu lists 'Main', 'Courses', 'Master', 'Staff', 'Research', and 'Students'. Below the menu, there are links for 'Radiocommunication Systems', 'Microwave IC', and 'Web Design'. The main content area features the course title 'Modern Web Applications Design' and the following details:

**Course: TMPAW (2014-2015)**  
**Course Coordinator:** Assist.P. Dr. Radu-Florin Damian  
**Code:** DS507RC  
**Discipline Type:** DOMS; Alternative, Synthesis  
**Credits:** 7  
**Enrollment Year:** 5, Sem. 10

**Activities**  
**Course:** Instructor: Assist.P. Dr. Radu-Florin Damian, 2 Hours/Week, Specialization Section, Timetable:  
**Laboratory:** Instructor: Assist.P. Dr. Radu-Florin Damian, 1 Hours/Week, Half Group, Timetable:

**Evaluation**  
Type: **Examen**  
**A:** 33.33%, (Test/Colloquium)  
**B:** 33.33%, (Seminary/Laboratory/Project Activity)  
**D:** 33.33%, (Homework/Specialty papers)

**Grades**  
Not yet

**Attendance**  
Not yet

**Materials**  
**Course Slides**

# CURS

I.	HTML si XHTML (recapitulare)	1 oră
II	CSS	2 ore
III	<del>Baze de date, punct de vedere practic</del>	<del>1 oră</del>
IV	<del>Limbajul de interogare SQL</del>	<del>4 ore</del>
V	PHP - HyperText Preprocessor	8 ore
VI	XML - Extended Mark-up Language si aplicatii	4 ore
VII	Conlucrare intre PHP/MySql, PHP/XML, Javascript/HTML	2 ore
VIII	Exemple de aplicatii	6 ore
	Total	28 ore

# rf-opto.etti.tuiasi.ro

- [http://rf-opto.etti.tuiasi.ro/master\\_it.php](http://rf-opto.etti.tuiasi.ro/master_it.php)
- Laborator 2011-2012
  - Exercitii/Performanta MySql
  - optional

Web Design

# Concepte generale



# Concepte

- Steve Krug: “**Don't Make Me Think**”
- Utilizatorii scaneaza pagina, nu o citesc
- Informatia trebuie redusa la minimul necesar in majoritatea locurilor
- “Daca ceva e greu de utilizat, mai bine nu o utilizez”
- Utilizatorii au comportament de **rechin**
- Originalitatea **nu e** intotdeauna **recomandata**

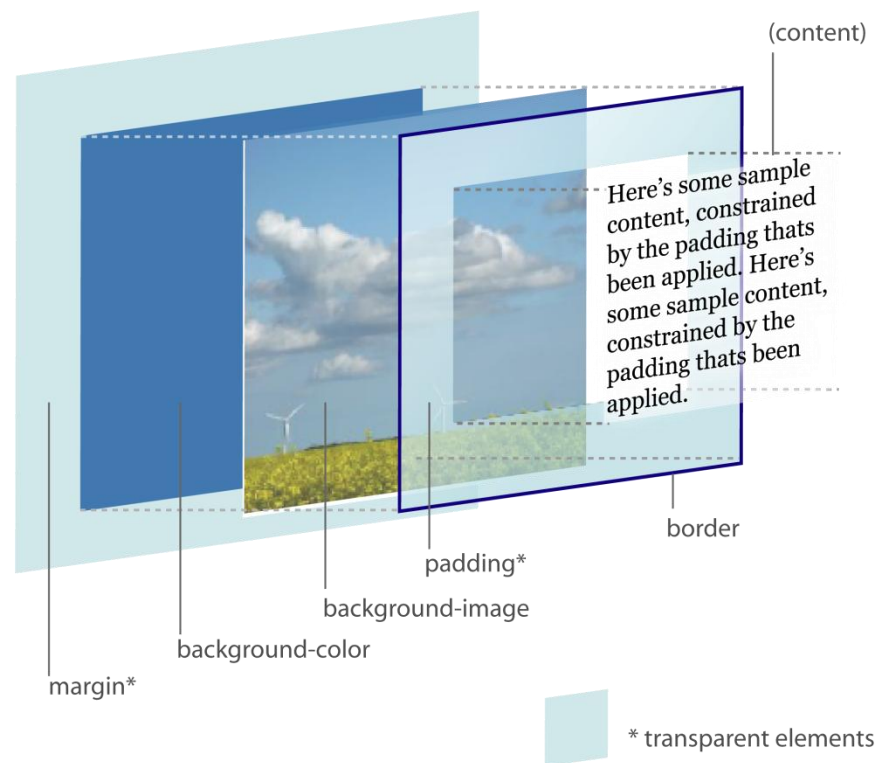
Capitolul II

**CSS**

# CSS Box Model

- Orientat in jurul conceptului de "cutie" – Box model

THE CSS BOX MODEL HIERARCHY

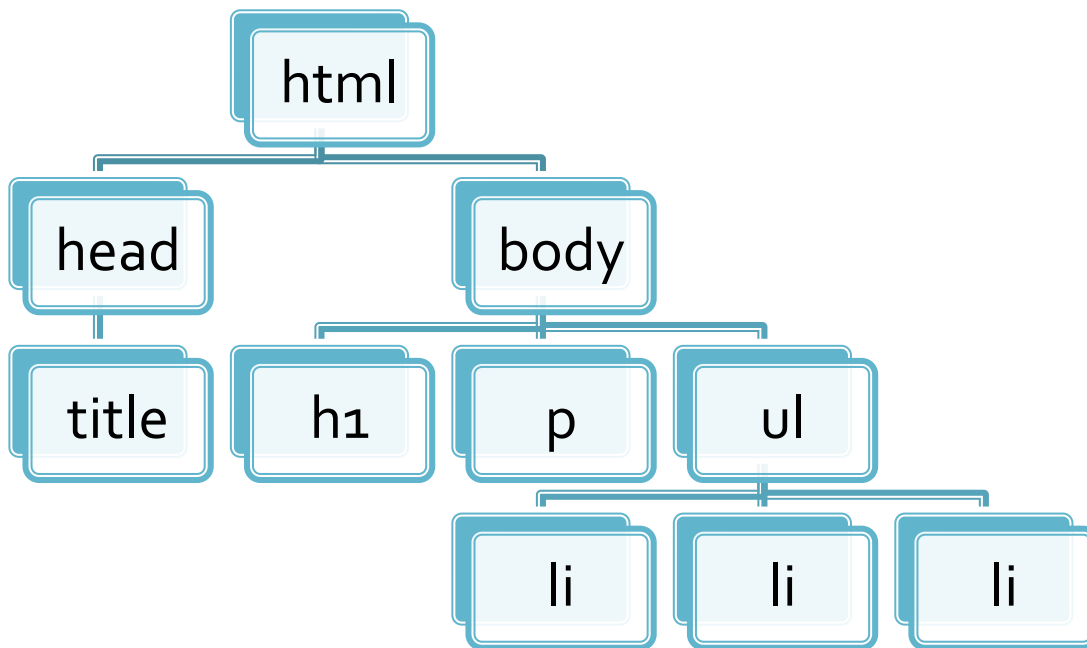


# Pozitionare

- `position`: absolute | fixed | relative | static | inherit;
- `top, right, bottom, left`: auto | [valoare] | [%] | inherit;
- `z-index`: [valoare intreaga]; - mai mare = deasupra

# DOM

- DOM – Document Object Model: structura de tip graf



```
<html>  
<title>pagina mea</title>  
<body>  
<h1>Compozitori:</h1>  
<p>  
<ul>  
<li> elvis costello  
<li> johannes brahms  
<li> georges brassens  
</ul>  
</body>  
</html>
```

# Relatie cu Javascript in aplicatii

- Javascript poate accesa prin intermediul obiectului DOM atasat documentului HTML elementele din structura arbore DOM si modifica proprietatile corespunzatoare
- se deschide astfel calea spre aplicatii dinamice

# CSS Zen Garden

- <http://www.csszengarden.com/>
- un fisier html comun
- schimbarea formei permisa numai prin intermediul CSS

# CSS Zen Garden – HTML original

## css Zen Garden

### The Beauty of CSS Design

A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the

Download the sample [html file](#) and [css file](#)

### The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CS

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless ef

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with  
Become one with the web.

### So What is This About?

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, ;  
one will load the style sheet into this very page. The code remains the same, the only thing that has changed is th

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrate  
placed in the hands of those able to create beauty from structure. To date, most examples of neat tricks and hac



# CSS Zen Garden



## Css Zen Garden

*the beauty of css design.*

*A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.*

[DOWNLOAD THE SAMPLE HTML FILE AND CSS FILE](#)

### *The Road To Enlightenment*

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C,

SELECT A DESIGN:

[Under the Sea!](#)  
by Eric Stoltz

[Make 'em Proud](#)  
by Michael McAghon and  
Scotty Reif'snyder

Consideratii generale

# Aplicatii

# Aplicatii

Favorites BCC e-SMART

BANCA COMERCIALA CARPATICA **BCC e-SMART**  
internet banking inteligent

Conturi Plati Depozite **Rapoarte** Setari

Rulare raport Vizualizare rapoarte

**ATENTIE!**  
Va reamintim ca BCC nu solicita informatii confidentiale (user, parola, numar de card, data expirarii cardului, codul PIN) prin e-mail. Aceste informatii nu trebuie divulgate nimanui, sub niciun motiv.  
Pentru alte lamuriri, puteti apela 0800.807.807 (numar accesibil din retea Romtelecom).  
**Aveti posibilitatea sa alegeti procesarea in regim de urgenta a platilor. Aceasta optiune se comisiona suplimentar.**

Rulare raport

Raport

Ruleaza raportul

imediat

la data 03.03.2010 ora 23:59

Ruleaza

top

surati activitate pe Internet, va rugam sa consultati periodic documentul [SECURITATE INTERNET](#) \*\*\* BCC informeaza ca SWIFT poate furniza autor

# Aplicatii

The screenshot displays the BCR online banking portal. At the top, there is a navigation bar with links for Home, Contact, English, Contact center, and a search bar. Below this is a menu with categories like 24 Banking, Persoane fizice, Private banking, Timeri, PFA, Micro, Corporatii, IMM, Municipalitati, Despre noi, Cariere, and Presa. A secondary menu includes Lista de conturi, Conturi curente, Economisire, and Finantare.

The main content area is titled "Ordin de Plata - Creare" (Payment Order - Create). It shows a table with account details:

Stare cont	Tip	Numar cont	Sold disponibil	Valuta
Activ	Conturi curente	[Redacted]	[Redacted]	RON

Below the table, there are dropdown menus for "Sablon personal" and "Sablon furnizor", and a text field for "IBAN beneficiar". A "Verificare IBAN" button is present. A note states: "Nota: Pentru a ordona o plata, va rugam fie sa introduceti IBAN beneficiar, fie sa selectati un sablon personal sau pentru furnizori. Atentie: Transferurile se pot efectua doar catre conturi deschise in aceeași valuta!"

On the left side, there is a sidebar with a "Logout" button and a "Favorite" list containing items like "Ordin de Plata", "Istoricul tranzactiilor", "Lista ordine de plata", "Sabloane", "Plata repetitiva", "Schimb Valutar", "Cumpara produse", "Deschidere cont curent", "Constituire depozite la termen", "Deschidere cont de economii", "Cumparare CDD", "Ataseaza card de debit", "Aplica pentru un credit", "Curs valutar", "Mesaje(4)", "Contul meu de CLICK 24Banking (Favorite)", and "Demo Click 24 Banking".

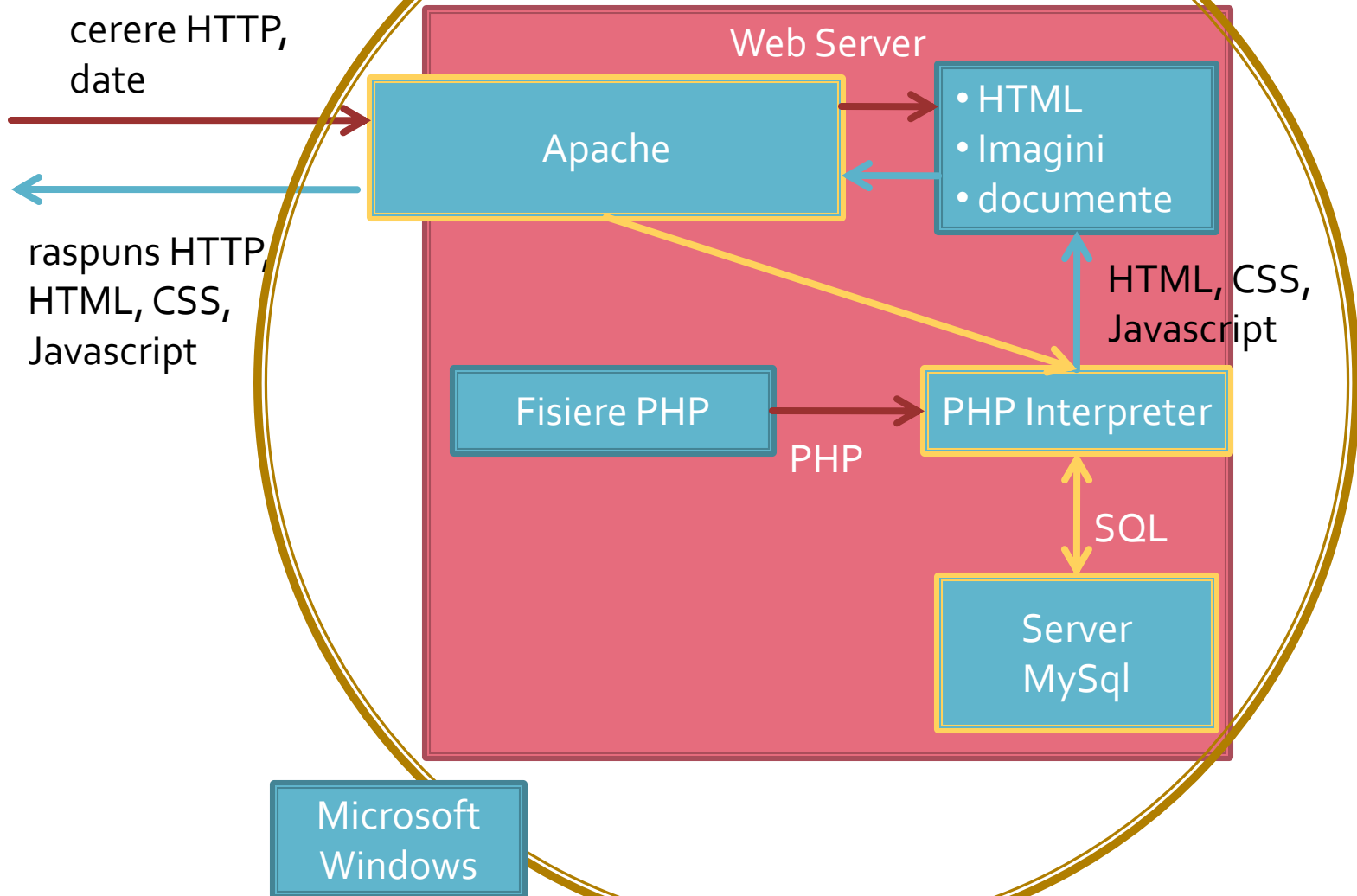
At the bottom, there are several promotional tiles: "Tarife si comisioane", "Intrebari si Raspunsuri", "24 Banking", and "Contact center".

The footer contains the text: "BANCA COMERCIALA ROMANA - SOCIETATE ADMINISTRATA IN SISTEM DUALIST, Bucuresti, B-dul Regina Elisabeta nr.5, Sector 3 | mentiuni legale - © 2008 BCR SA - Toate drepturile rezervate | site map

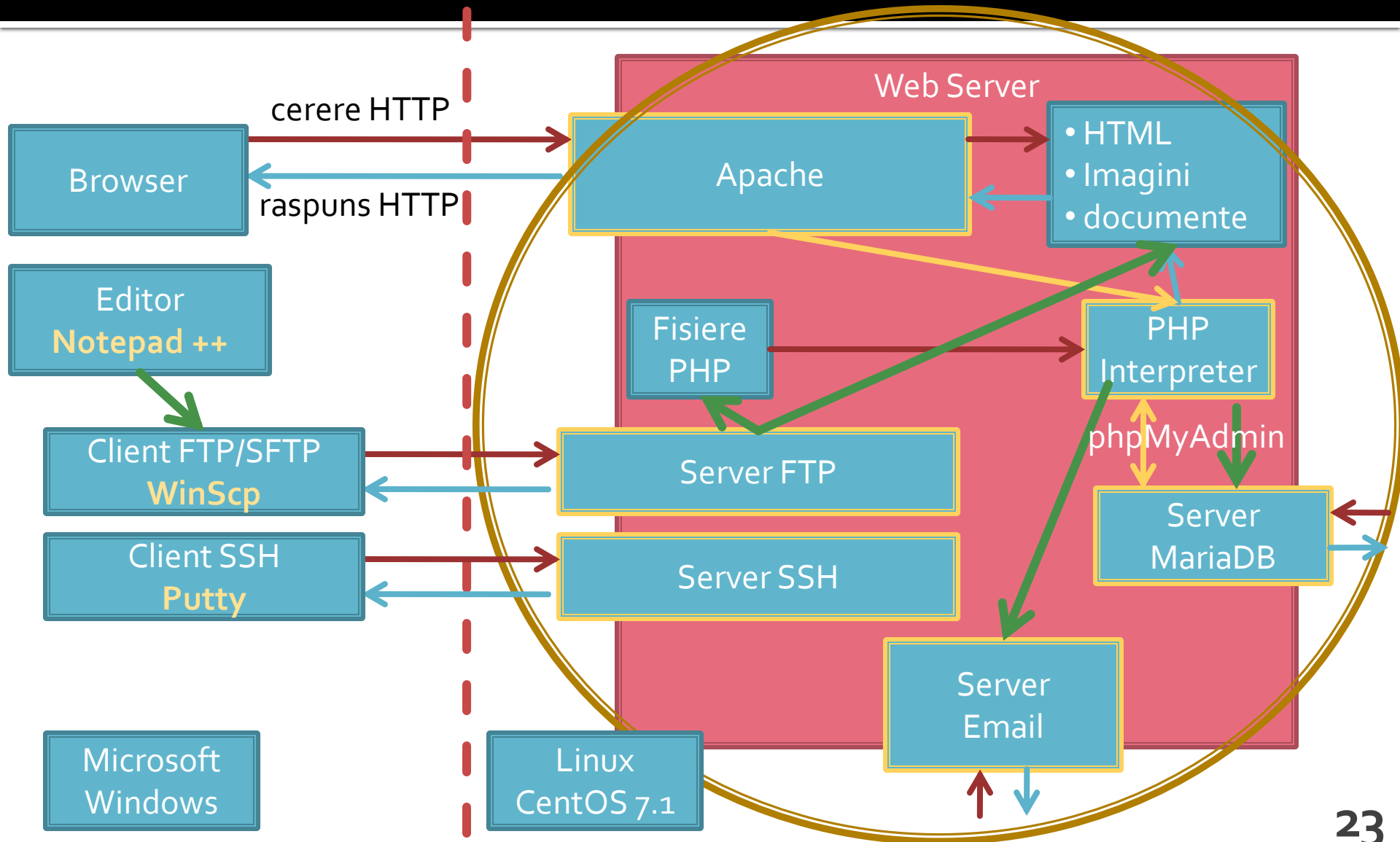




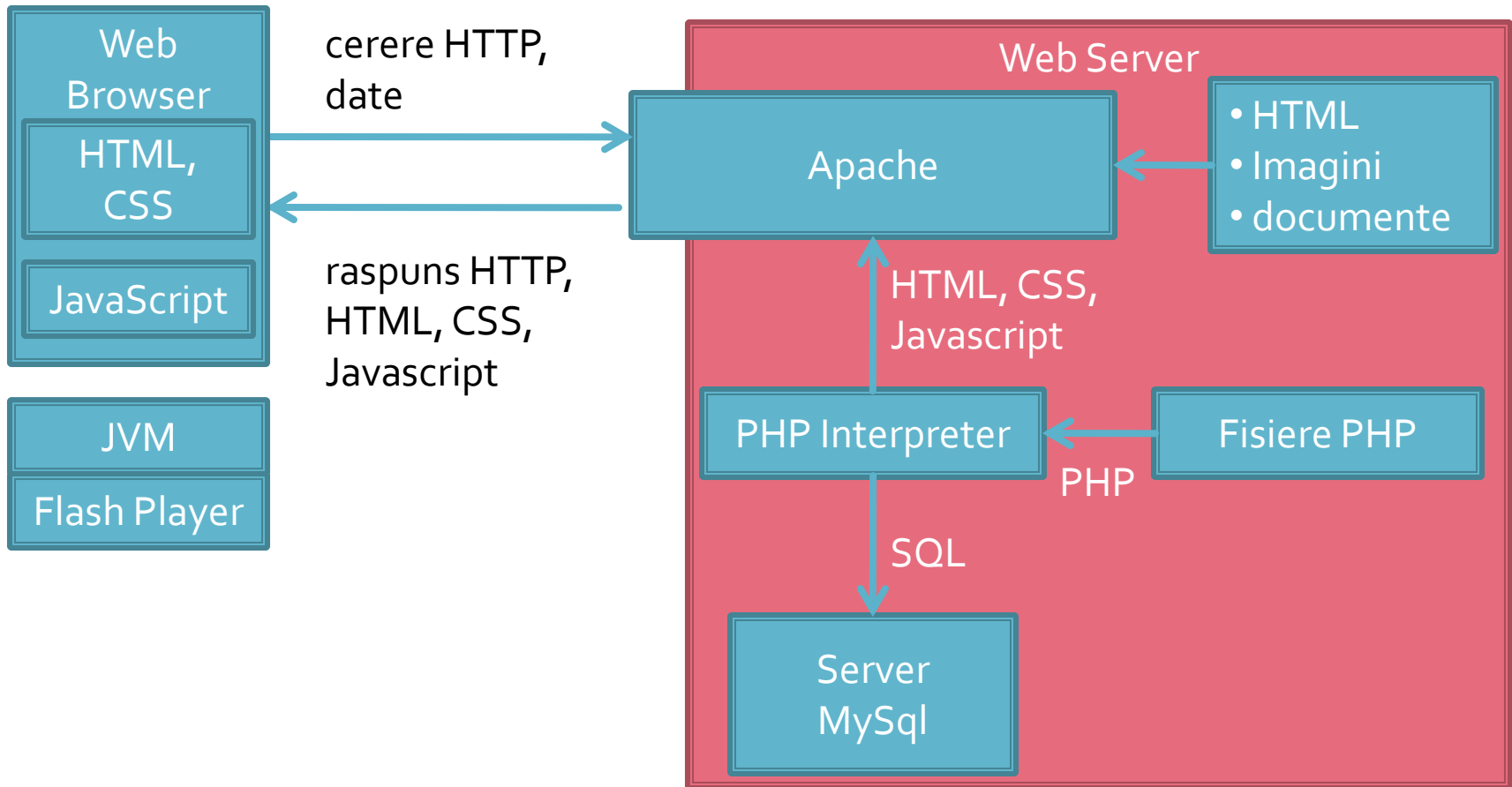
# WAMP



# LAMP



# Client/Server Scripting



client  
side  
scripting

server  
side  
scripting



# Aplicatie Web

- presupune prelucrarea unor date si oferirea unui document personalizat (rezultat al datelor respective)
- datele pot fi obtinute:
  - de la utilizator
  - o sursa externa (baze de date)
  - **combinatie** utilizator/baze de date

# Forme in HTML

- necesare pentru ca utilizatorul sa poate trimite date server-ului
- `<form>...</form>`
- Attribute specifice:
  - action: adresa documentului care preia datele
    - `<form action="<?php echo $_SERVER['PHP_SELF'];?>">`
    - `<form action="actiuni/fisier2.html">`
  - method: modalitatea de transmitere a datelor: post sau get
    - `<form method="post" action= ... >`

# Metode de transmitere

- **post** datele sunt transmise in bloc
- **get** datele sunt atasate adresei documentului de procesare : [results.php?prob=81&an=2009](http://results.php?prob=81&an=2009)
- **get** trebuie folosit numai cand datele sunt "idempotente",
  - nu cauzeaza efecte colaterale
  - nu modifica starea server-ului (baze date, etc)
- se poate simula realizarea unei forme (**get**) prin scrierea corespunzatoare a link-urilor

# Elemente de interactiune cu utilizatorul

- in interiorul etichetei `<form>...</form>`
  - `input`
  - `select/option`
  - `textarea`
  - `button`
- **toate** elementele vor avea un nume
  - atribut: `name=""`
  - numele va fi intalnit in uri-ul generat prin get, sau in numele variabilei trimisa prin post
- trimiterea datelor se face prin intermediul unui buton cu tipul `type="submit"`

# INPUT

- `<input .../>`
- Attribute:
  - type: text | password | checkbox | radio | submit | reset | file | hidden | image | button
  - name: numele variabilei
  - value: valoarea trimisa server-ului la selectie (valoarea initiala in anumite cazuri)
  - checked/src/size/maxlength in functie de tip

# Example

```
<input name="textfield" type="text" value="ceva" />
```

```
<input name="Ok" type="submit" value="Trinite" />
```

```
<label><input name="check" type="checkbox" value="5" checked />check1</label>
```

```
<label><input name="RG1" type="radio" value="a" checked="checked" />but1</label>
```

```
<label><input name="RG1" type="radio" value="b" />but2</label>
```

```
<input name="hid" type="hidden" value="6" />
```

ceva

Trinite



check1



but1



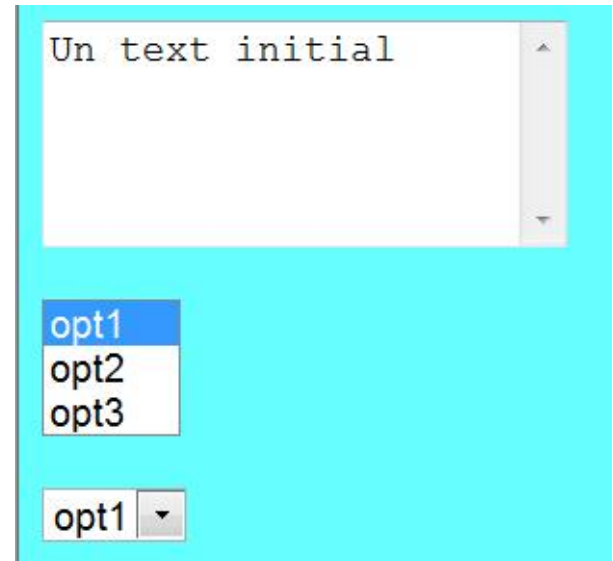
but2

# TEXTAREA/SELECT

```
<textarea name="textarea" cols="20" rows="5">Un text  
initial</textarea><br /><br />
```

```
<select name="select." size="3">  
  <option value="1" selected="selected">opt1</option>  
  <option value="2">opt2</option>  
  <option value="3">opt3</option>  
</select><br /><br />
```

```
<select name="select...">  
  <option value="1" selected="selected">opt1</option>  
  <option value="2">opt2</option>  
  <option value="3">opt3</option>  
</select>
```



The image displays three distinct HTML form controls on a light blue background. At the top is a text area containing the text "Un text initial". Below it is a list box with three options: "opt1", "opt2", and "opt3", where "opt1" is currently selected and highlighted in blue. At the bottom is a dropdown menu with "opt1" selected and a small downward-pointing arrow on the right side.

# BUTTON

```
<button type="button" onClick="do ( );">Click Here</button>
```

```
<button type="submit" value="infoOnly">Request Info</button>
```

```
<button type="reset">Clear</button>
```

```
<button type="submit" id="sender"  
value="infoOnly">Request<br />Info<br /></button>
```

```
<button type="submit" id="sender"  
value="infoOnly"><p>Request</p><p><b><i>Info</i></b></p>  
</button>
```

- introdus pentru a oferi posibilitatea introducerii de continut mai complex (text formatat, imagini) in interiorul butoanelor de pe forma

Click Here

Request Info 

 Clear

Request  
Info



Request

*Info*





# Exemplu – forma

- Mic magazin online
- Formular de comanda cu procesarea comenzii

```
<html>
<head>
<title>Magazin online XXX SRL</title>
</head>
<body>
<h1>Magazin online XXX SRL</h1>
<h2>Realizati comanda</h2>
<form action="rezultat.html" method="post">
<table border="0">
<tr bgcolor="#cccccc"><td width="150">Produs</td><td width="15">Cantitate</td></tr>
<tr><td>Carti</td><td align="center"><input type="text" name="carti_cant" size="3" maxlength="3" /></td></tr>
<tr><td>Caiete</td><td align="center"><input type="text" name="caiete_cant" size="3" maxlength="3" /></td></tr>
<tr><td>Penare</td><td align="center"><input type="text" name="penare_cant" size="3" maxlength="3" /></td></tr>
<tr><td colspan="2" align="center"><input type="submit" value="Trimite" /></td></tr>
</table>
</form>
</body>
</html>
```

## Magazin online XXX SRL

### Realizati comanda

Produs	Cantitate
Carti	<input type="text" value="1"/>
Caiete	<input type="text" value="2"/>
Penare	<input type="text" value="3"/>

# Exemplu – raspuns static

- fisier html
- fisierele HTML sunt doar “servite” de server
- in aparenta a existat o procesare, real **nu**

```
<html>
<head>
<title>Magazin online XXX SRL</title>
</head>
<body>
<h1>Magazin online XXX SRL</h1>
<h2>Rezultate comanda</h2>
<p>Comanda receptionata</p>
</body>
</html>
```

**Magazin online XXX SRL**

**Rezultate comanda**

Comanda receptionata

# CURS

I.	HTML si XHTML (recapitulare)	1 oră
II	CSS	2 ore
III	<del>Baze de date, punct de vedere practic</del>	<del>1 oră</del>
IV	<del>Limbajul de interogare SQL</del>	<del>4 ore</del>
V	PHP - HyperText Preprocessor	8 ore
VI	XML - Extended Mark-up Language si aplicatii	4 ore
VII	Conlucrare intre PHP/MySql, PHP/XML, Javascript/HTML	2 ore
VIII	Exemple de aplicatii	6 ore
	Total	28 ore

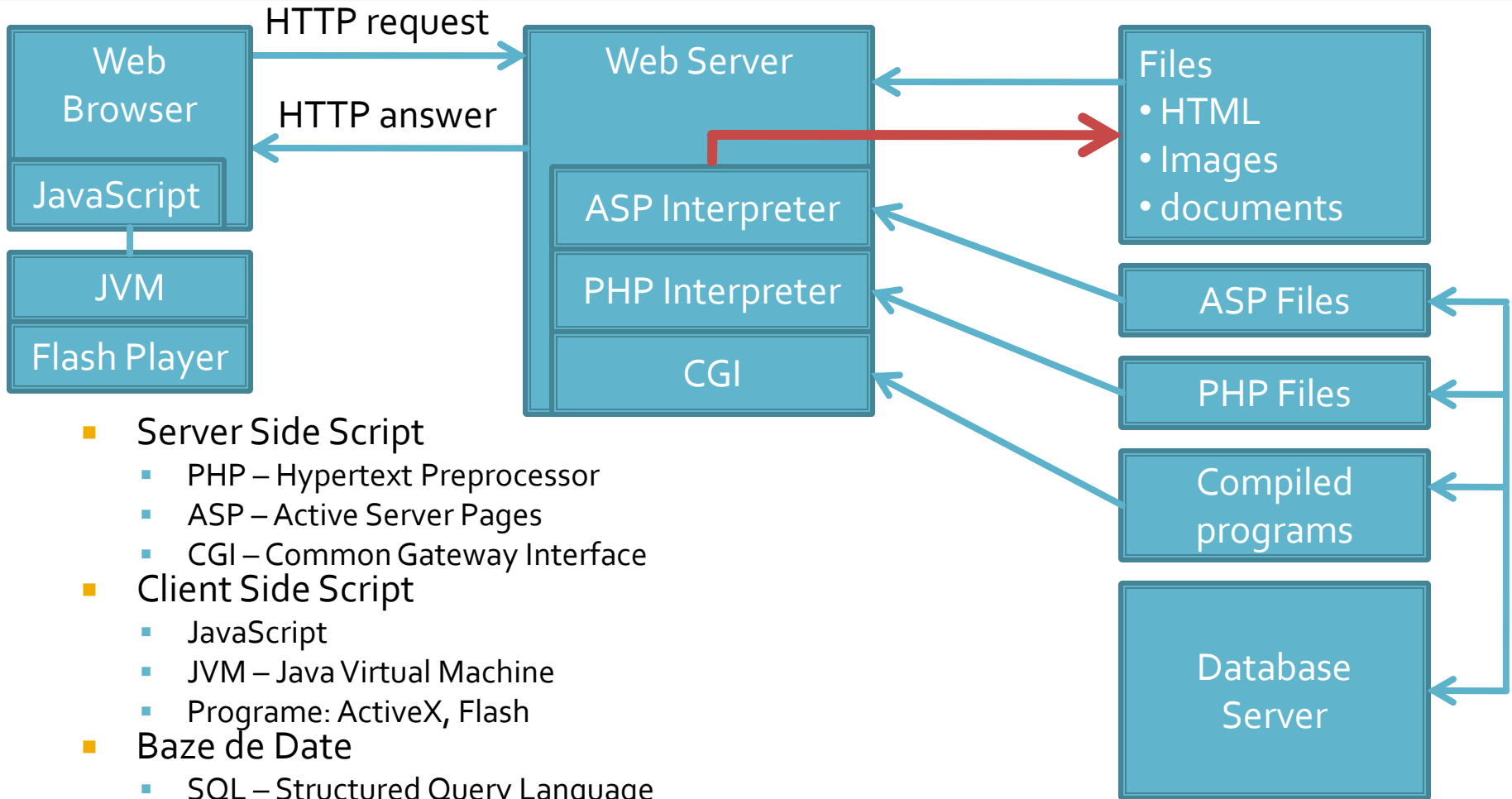
Hypertext PreProcessor

**PHP**

# PHP

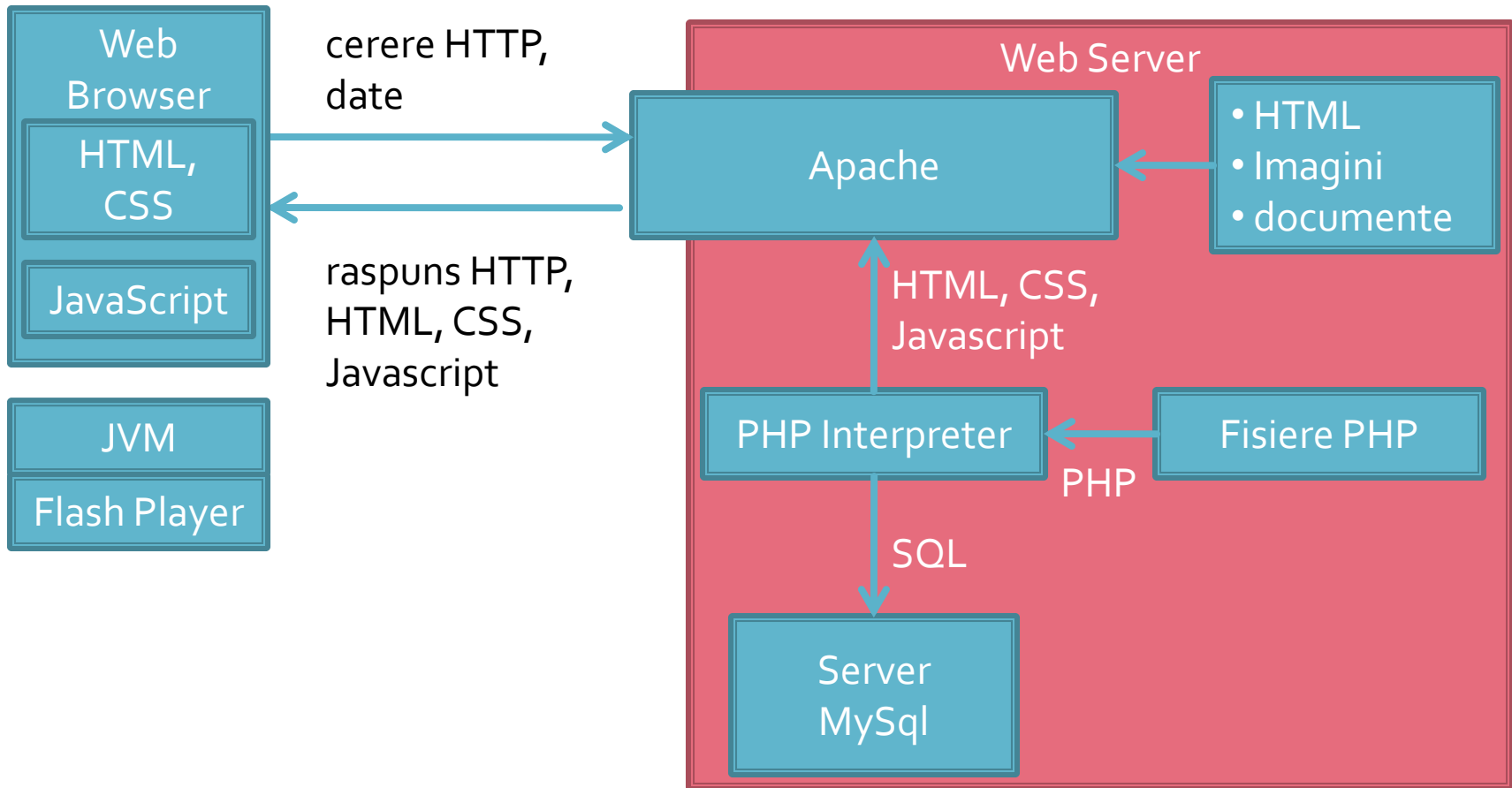
- Hypertext PreProcessor - acronim recursiv
  - initial – Personal Home Page / Form Interpreter
  - 1995 – 1.0
- versiune curenta: 5.6.40
  - 2019-03-05
- versiune curenta: 7.2.15
  - 2019-03-05
- limbaj de scripting de uz general,
- rulare pe server (**server-side scripting**)
- open source

# Web server Technology



- **Server Side Script**
  - PHP – Hypertext Preprocessor
  - ASP – Active Server Pages
  - CGI – Common Gateway Interface
- **Client Side Script**
  - JavaScript
  - JVM – Java Virtual Machine
  - Programme: ActiveX, Flash
- **Baze de Date**
  - SQL – Structured Query Language
  - MySql – open Source
  - Microsoft SQL Server
  - Oracle

# Client/Server Scripting

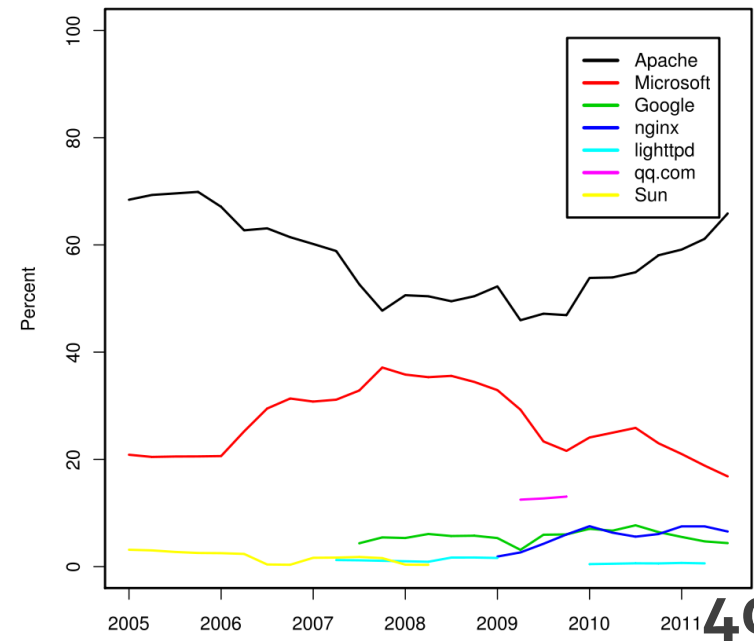
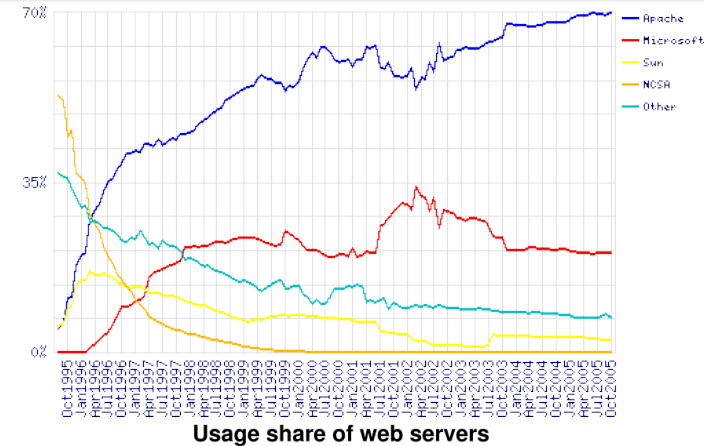


client  
side  
scripting

server  
side  
scripting

# Tehnologia server-elor Web

- PHP – Hypertext Preprocessor
  - initially – Personal Home Page
  - open source
  - C++
  - Apache
- ASP – Active Server Pages
  - Microsoft
  - VBasic
  - IIS
- Java/JavaScript
  - Sun
  - Java Virtual Machine





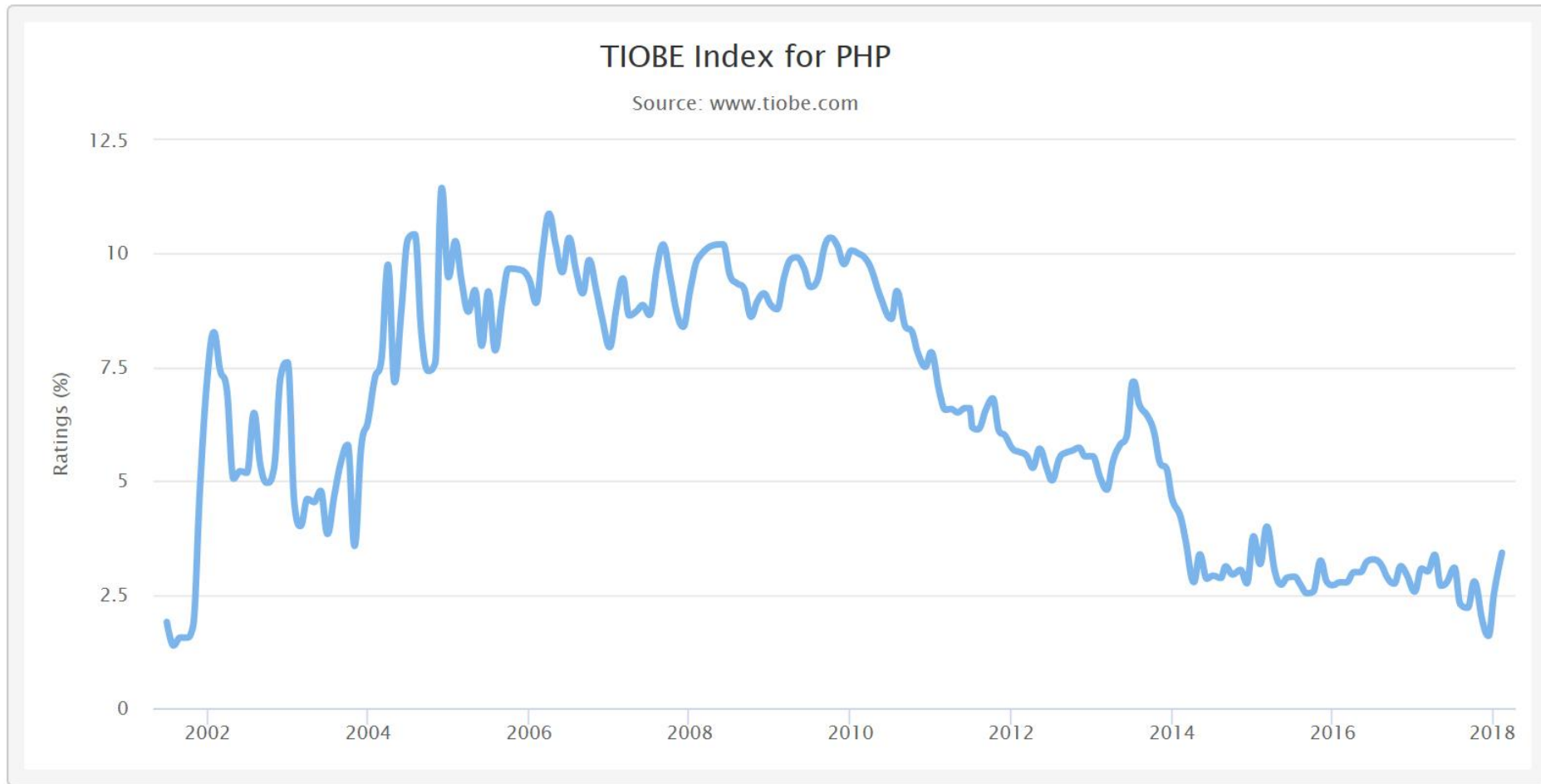
# TIOBE Programming Community Index for February 2018

Position Feb 2018	Position Feb 2017	Delta in Position	Programming Language	Ratings Feb 2018	Delta Feb 2017
1	1	=	<a href="#">Java</a>	14.99%	-1.69%
2	2	=	<a href="#">C</a>	11.86%	3.41%
3	3	=	<a href="#">C++</a>	5.73%	0.30%
4	5	↑	<a href="#">Python</a>	5.17%	1.12%
5	4	↓	<a href="#">C#</a>	4.45%	-0.45%
6	8	↑	<a href="#">Visual Basic .NET</a>	4.07%	1.25%
7	6	↓	<a href="#">PHP</a>	3.42%	0.35%
8	7	↓	<a href="#">JavaScript</a>	3.17%	0.29%
9	9	=	<a href="#">Delphi/Object Pascal</a>	2.59%	0.11%
10	11	↑	<a href="#">Ruby</a>	2.53%	0.38%

# TIOBE Programming Community Index for March 2010

Position Mar 2010	Position Mar 2009	Delta in Position	Programming Language	Ratings Mar 2010	Delta Mar 2009	Status
1	1	=	<a href="#">Java</a>	17.509%	-2.29%	A
2	2	=	<a href="#">C</a>	17.279%	+1.42%	A
3	4	↑	<a href="#">PHP</a>	9.908%	+0.42%	A
4	3	↓	<a href="#">C++</a>	9.610%	-0.75%	A
5	5	=	<a href="#">(Visual) Basic</a>	6.574%	-1.71%	A
6	7	↑	<a href="#">C#</a>	4.264%	-0.06%	A
7	6	↓	<a href="#">Python</a>	4.230%	-0.95%	A
8	9	↑	<a href="#">Perl</a>	3.821%	+0.40%	A
9	10	↑	<a href="#">Delphi</a>	2.684%	-0.03%	A
10	8	↓↓	<a href="#">JavaScript</a>	2.651%	-0.96%	A

# Evolutie PHP

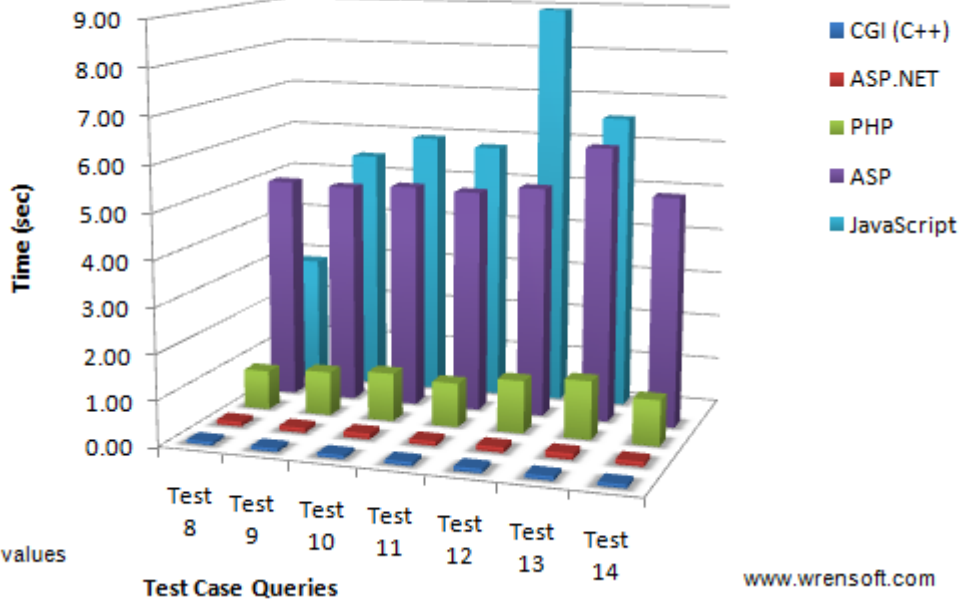


# Avantaje PHP

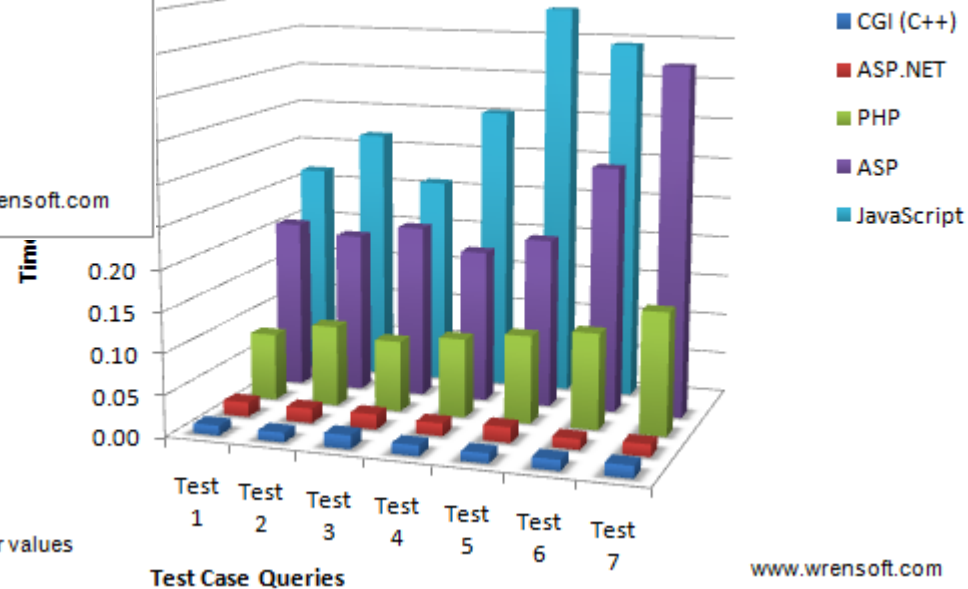
- Performanta ridicata
- Interfata cu multe sisteme de baze de date
- Costuri reduse
- Biblioteci incluse pentru majoritatea operatiunilor uzuale in aplicatii web
- Usurinta in invatare (C)
- Portabilitate
  - Disponibilitate a codului sursa
- Exemple disponibile in comunitate
- Suport disponibil

# Performanta

**Zoom search benchmarking**  
(Medium site - 60K pages, 2.5 million words)

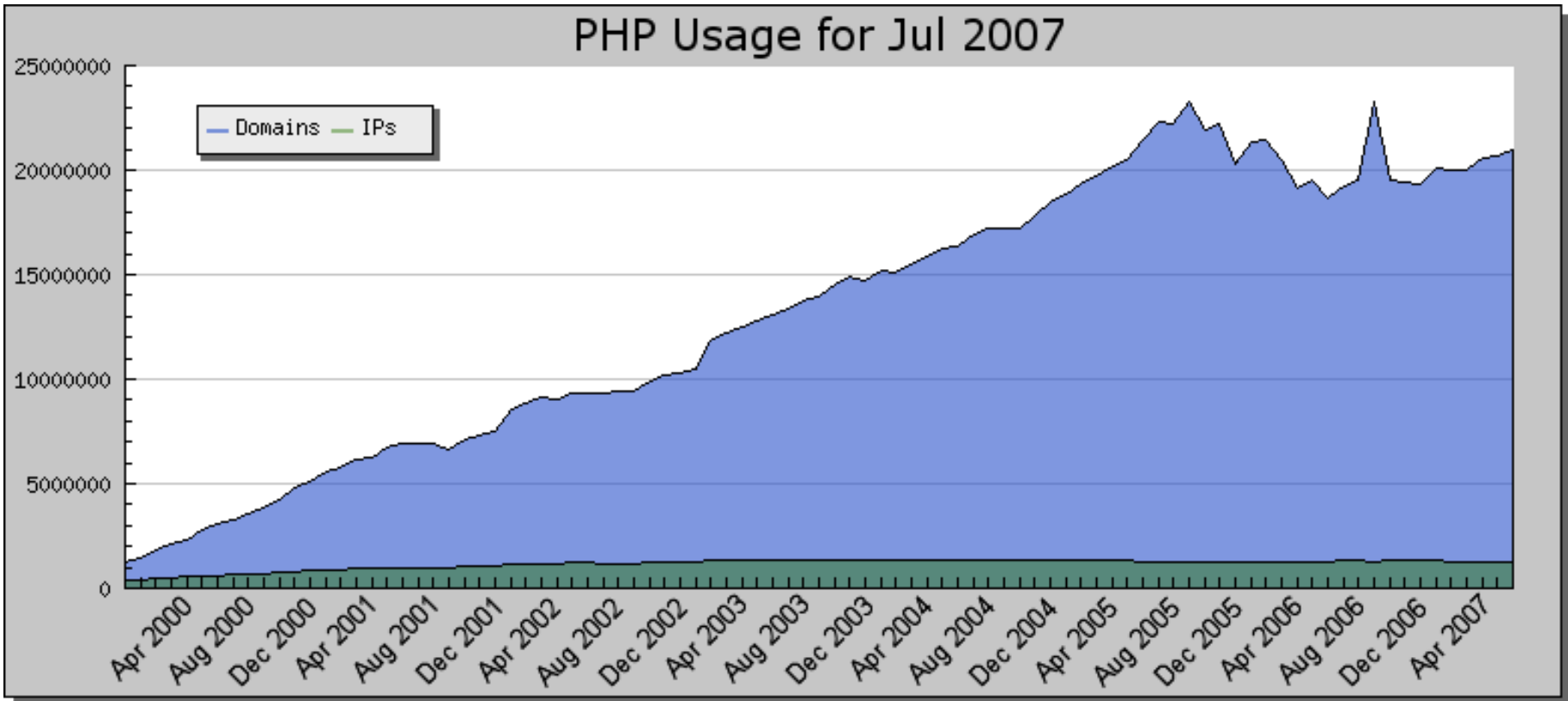


**Zoom search benchmarking**  
(Small site - 400 pages, 266K words)

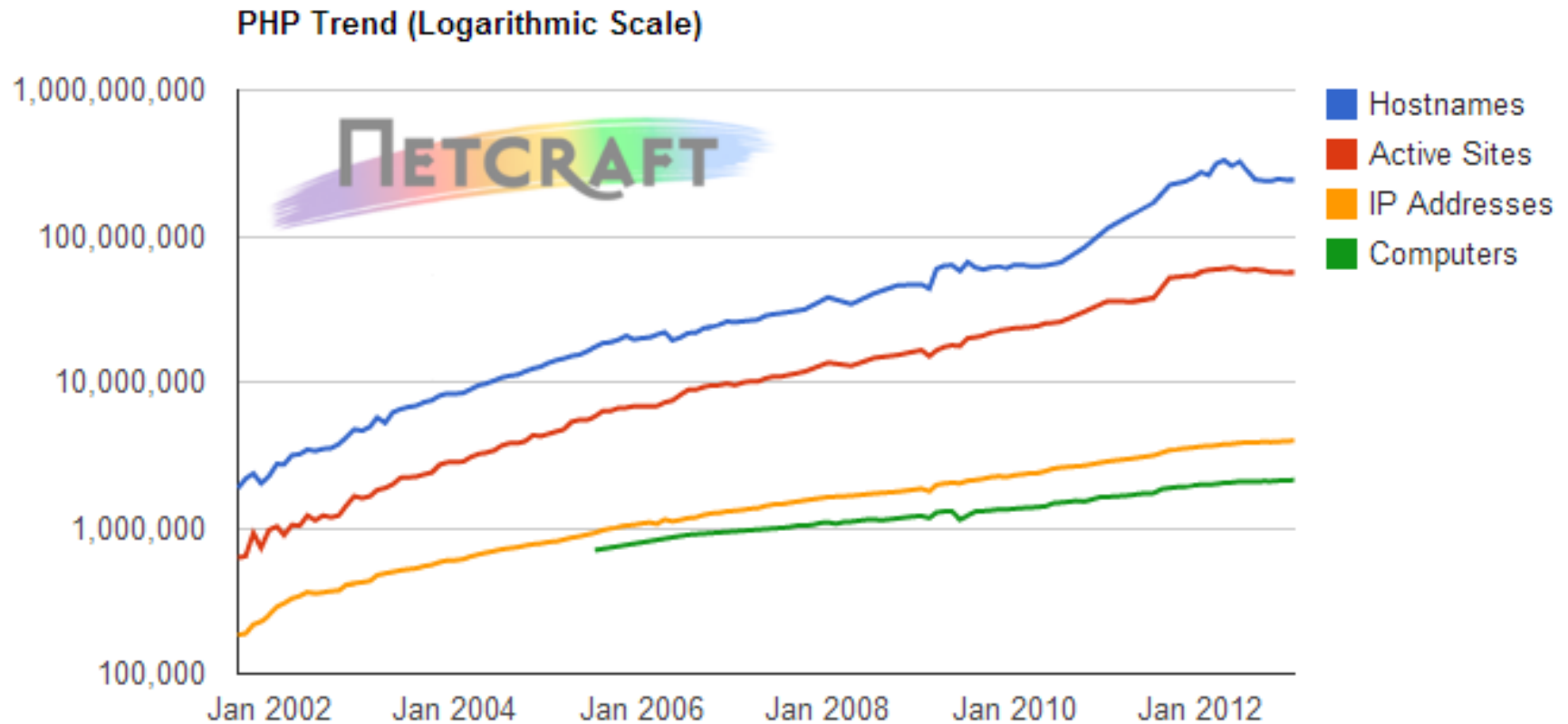


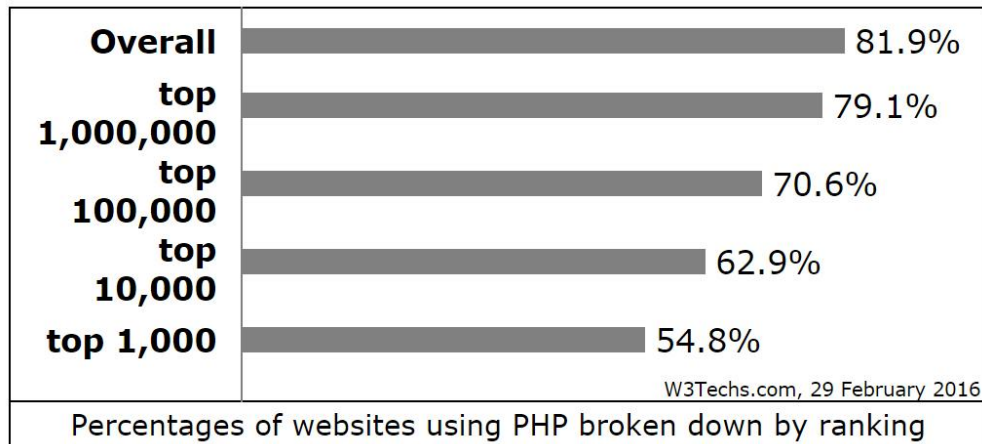
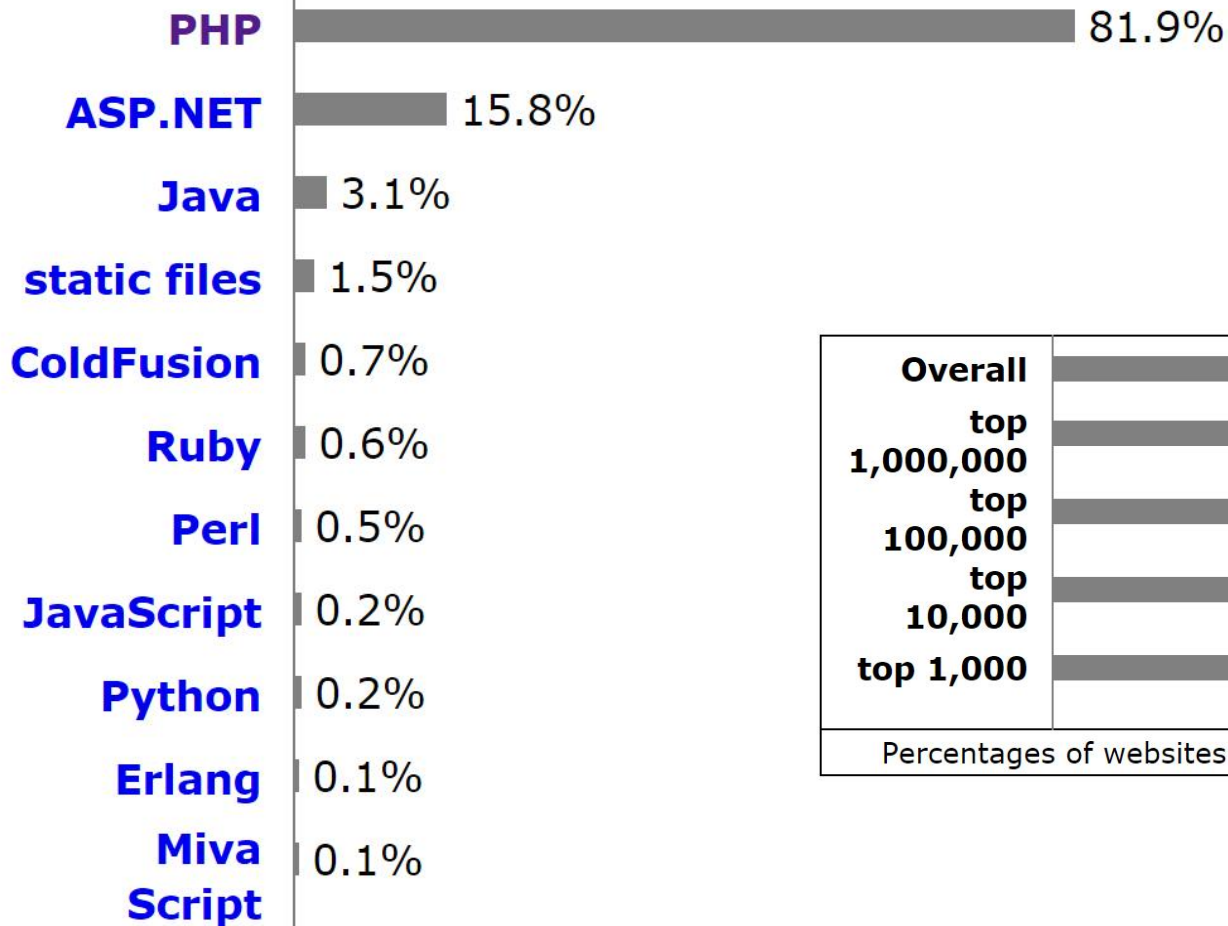
# Raspiandire

PHP Usage for Jul 2007



# Rasbandire

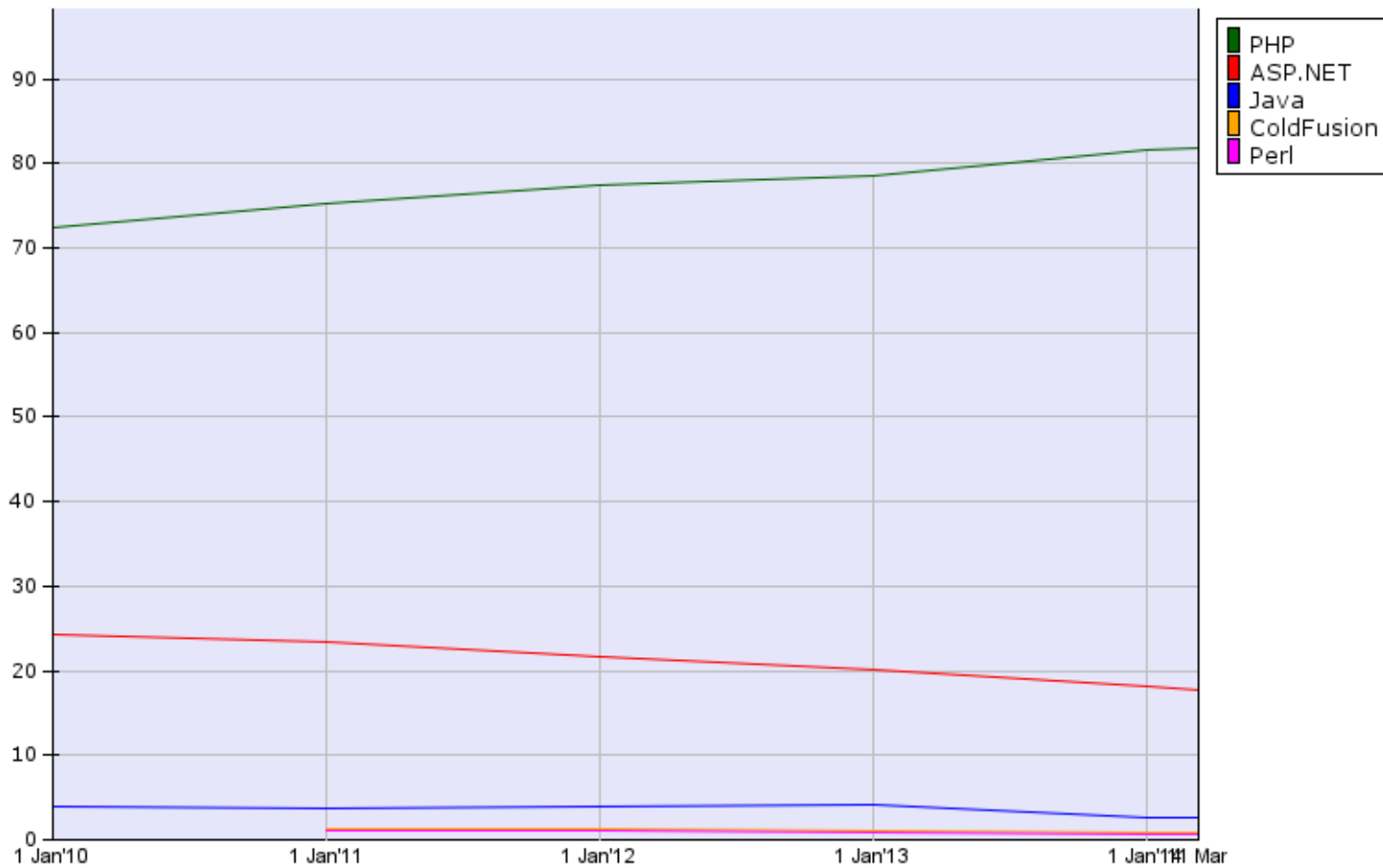




W3Techs.com, 29 February 2016

Percentages of websites using various server-side programming languages





Usage of server-side programming languages for websites, 11 Mar 2014, W3Techs.com

# Server Side Scripting

	2010 1 Jan	2011 1 Jan	2012 1 Jan	2013 1 Jan	2014 1 Jan	2014 11 Mar
PHP	72.5%	75.3%	77.3%	78.7%	81.6%	81.9%
ASP.NET	24.4%	23.4%	21.7%	20.2%	18.2%	17.8%
Java	4.0%	3.8%	4.0%	4.1%	2.7%	2.7%
ColdFusion		1.3%	1.2%	1.1%	0.8%	0.8%
Perl		1.1%	1.0%	0.8%	0.6%	0.6%
Ruby	0.5%	0.5%	0.6%	0.5%	0.4%	0.5%
Python	0.3%	0.3%	0.2%	0.2%	0.2%	0.2%
JavaScript			<0.1%	<0.1%	0.1%	0.1%

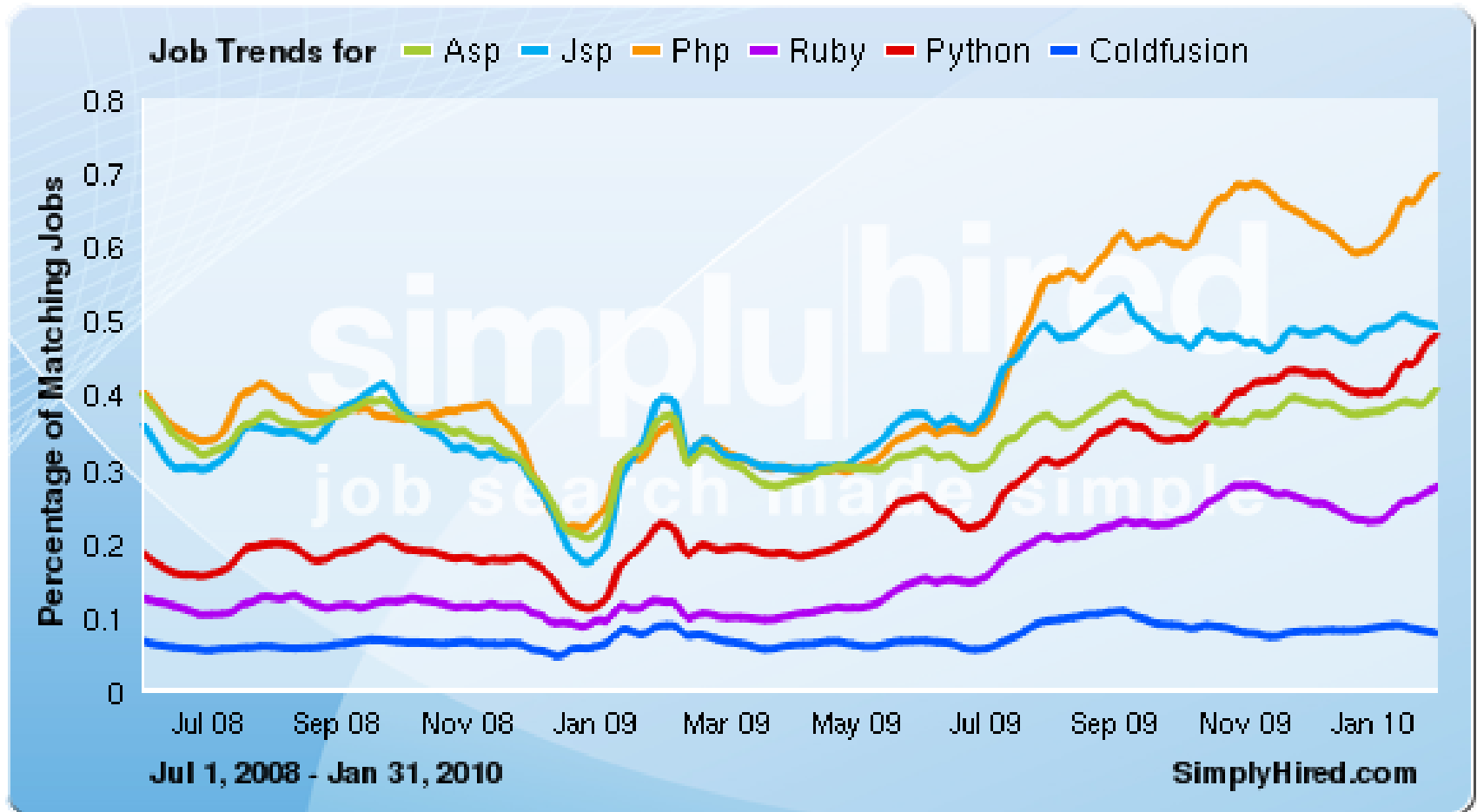
# Server Side Scripting

	2015 1 Feb	2015 1 Jul	2016 1 Jan	2016 15 Feb
PHP	82.0%	81.8%	81.7%	81.8%
ASP.NET	17.0%	16.9%	16.0%	15.9%
Java	2.8%	3.0%	3.0%	3.0%
static files			1.6%	1.6%
ColdFusion	0.7%	0.7%	0.7%	0.7%
Ruby	0.6%	0.6%	0.6%	0.6%
Perl	0.5%	0.5%	0.5%	0.5%
Python	0.2%	0.2%	0.2%	0.2%

# Sites

- [Facebook.com](https://www.facebook.com)
- [Wikipedia.org](https://www.wikipedia.org)
- [Qq.com](https://www.qq.com)
- [Taobao.com](https://www.taobao.com)
- [Sina.com.cn](https://www.sina.com.cn)
- [Wordpress.com](https://www.wordpress.com)
- [Vk.com](https://www.vk.com)
- [Weibo.com](https://www.weibo.com)
- [Babylon.com](https://www.babylon.com)
- [Mail.ru](https://www.mail.ru)

# Angajare

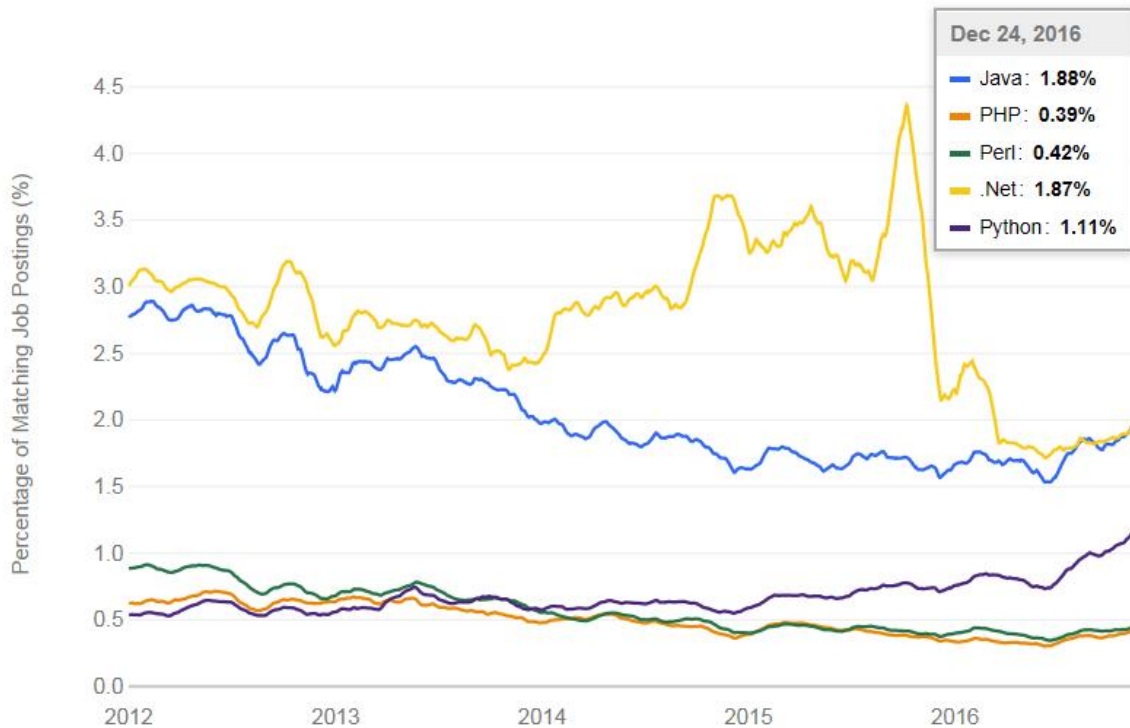


# Angajare

## Java, PHP, Perl, .Net, and Python Job Trends

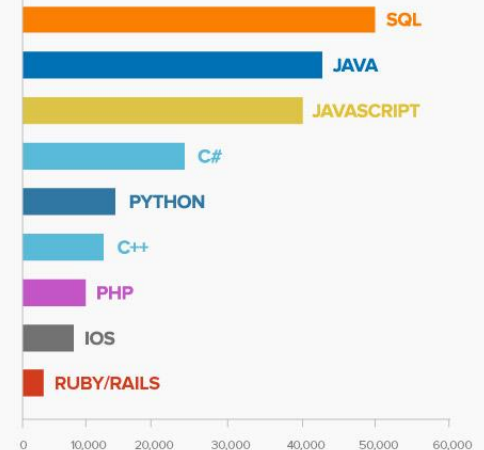
Java x PHP x Perl x .Net x Python x + Add Term Find Trends

### Job Postings



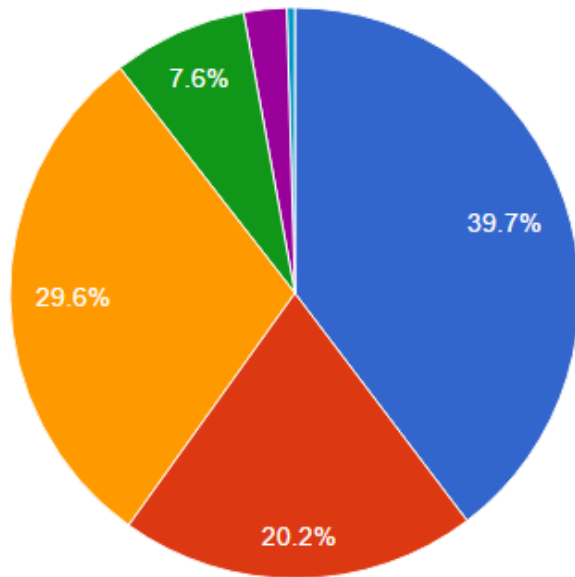
### Languages ranked by number of programming jobs

Data from Indeed.com 2016



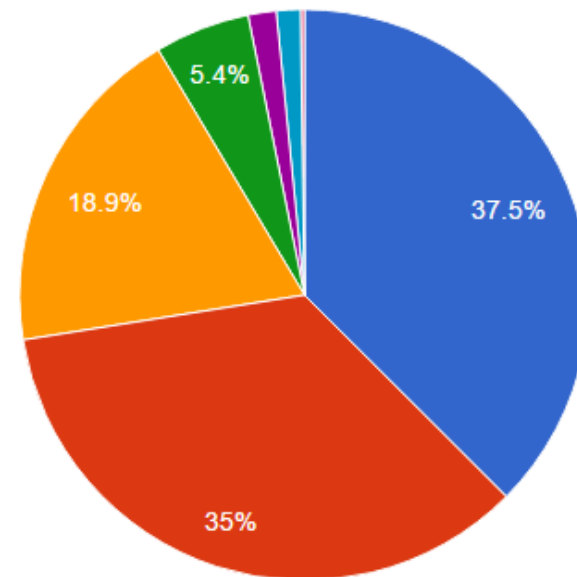
# PHP 7, Systeme

PHP Versions Grouped 2016-05



- PHP 5.6
- PHP 7.0
- PHP 5.5
- PHP 5.4
- PHP 5.3
- PHP 7.1

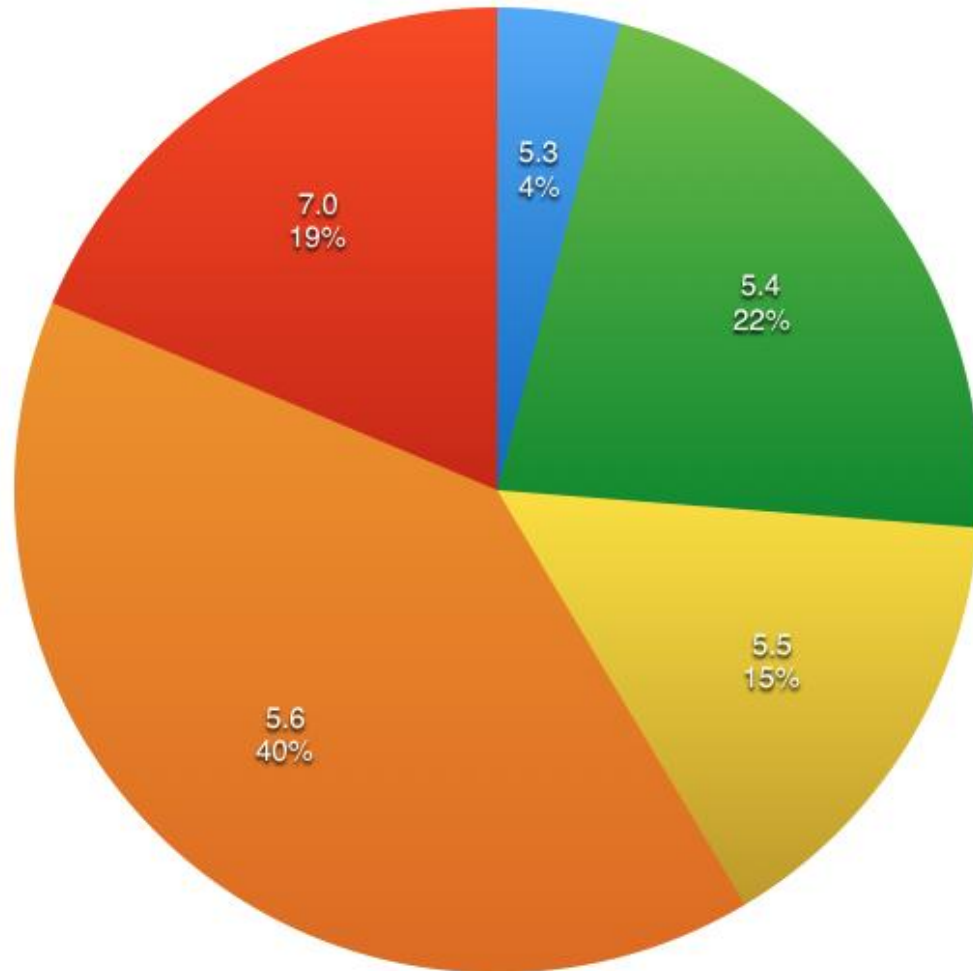
PHP Versions Grouped 2016-11



- PHP 5.6
- PHP 7.0
- PHP 5.5
- PHP 5.4
- PHP 5.3
- PHP 7.1
- PHP 7.2

# PHP 7, Aplicatii

PHP versions in commercial projects on Semaphore, 2016





# Ex. Server

## Server Information

### Server Information

Item	Detail
Hosting Package	5G
Server Name	server72
cPanel Version	62.0 (build 8)
Apache Version	2.4.25
PHP Version	5.6.30
MySQL Version	10.0.27-MariaDB-cll-lve
Architecture	x86_64
Operating System	linux
Shared IP Address	188.212.252.74
Path to Sendmail	/usr/sbin/sendmail
Path to Perl	/usr/bin/perl
Perl Version	5.16.3
Kernel Version	3.10.0-427.18.2.lve1.4.27.el7.x86_64

## Server Information

Item	Detail
Hosting Package	personal
Server Name	server11
cPanel Version	60.0 (build 37)
Apache Version	2.2.27
PHP Version	5.4.28
MySQL Version	5.5.52-cll
Architecture	x86_64
Operating System	linux
Shared IP Address	88.198.167.40
Path to Sendmail	/usr/sbin/sendmail
Path to Perl	/usr/bin/perl
Perl Version	5.10.1
Kernel Version	2.6.32-458.23.2.lve1.2.48.el6.x86_64

# Ex. Server

## System Information

System Item	Details	Status
Server Load	12.708984 (32 CPUs)	✓
Memory Used	60.9% (160,698,628 of 263,854,760)	✓
Swap Used	10.47% (7,028,300 of 67,108,860)	✓

## Disk Information

Device	Mount Point	Usage	Status
/home	/home	79% (867,228,844 of 1,152,542,444)	✓
/	/	41% (153,355,872 of 393,757,580)	✓
/tmp	/tmp	1% (52,332 of 3,997,376)	✓
/var/tmp	/var/tmp	1% (52,332 of 3,997,376)	✓
/boot	/boot	61% (277,346 of 487,634)	✓



## Server Information

### Processor Information

Total processors: 32

#### Processor #1

Vendor

GenuineIntel

Name

Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz

Speed

2600.343 MHz

Cache

20480 KB

#### Processor #2

Vendor

GenuineIntel

Name

Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz

Speed

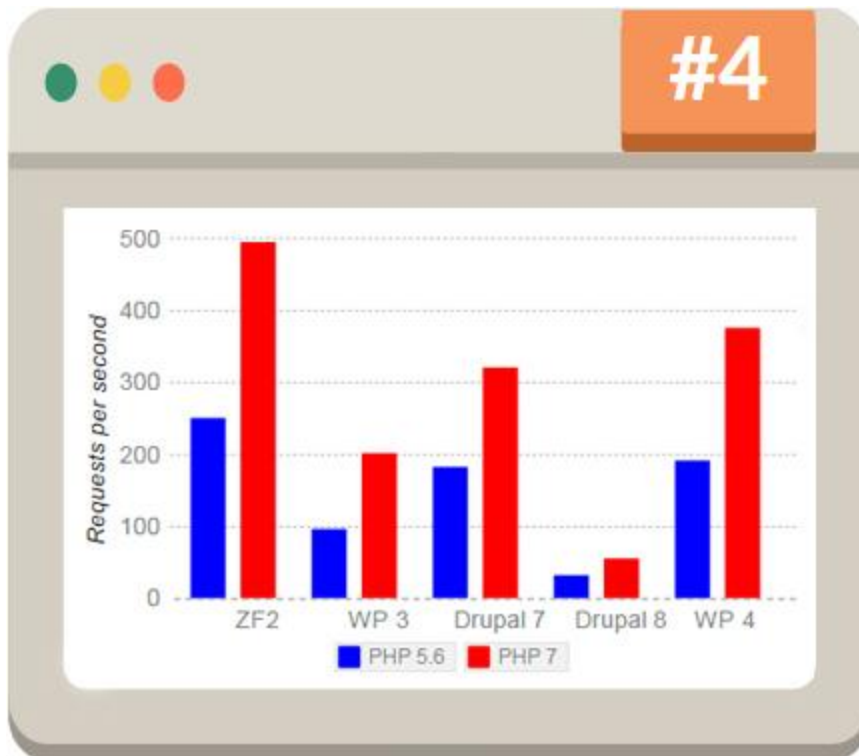
2599.968 MHz

Cache

20480 KB

#### Processor #3

# PHP 7, performanta



5

THINGS YOU  
MUST KNOW  
ABOUT

*php7*

# PHP - Concepte

- limbaj interpretat – compilat “on the fly” de interpretorul PHP de pe server
- script-urile PHP contin sursele
  - exista posibilitatea pre-compilarii surselor pentru spor de viteza
  - [Hip-Hop for PHP / Facebook](#)
  - [HipHop Virtual Machine / Facebook](#)
- orientat spre aplicatii web dinamice (biblioteci)
- poate fi integrat in HTML – utilizarea tipica

# Integrare

```
539         <td><?php echo $row['Documente'];?>&nbsp;</td>
540         <td><?php echo $row['user_creat'];?>&nbsp;</td>
541         <td class="smaller"><a href="control_lot.php?id=<?php echo $row['ID_LOT'];?>">dezactiveaza</a><br /><a href=
"control_lot.php?id=<?php echo $row['ID_LOT'];?>">modifica</a></td>
542     </tr>
543     <?php $index++;
544     } while ( $row = mysql_fetch_assoc($result));?>
545 </table><?php
546 }
547 else
548 {
549     echo "<p>Nu exista loturi active</p>";
550 }
551 ?>
552
553 <p class="title">Loturi inactive</p>
554 <?php
555 $query = "SELECT l.*, c.`nume_user` AS `user_creat`
556         FROM `lot` AS l
557         LEFT JOIN `users` AS c ON (l.`User`=c.`id_user`)
558         WHERE l.`Activ` = 0 ORDER BY l.`ID_LOT` DESC";
559 $result = mysql_query($query);
560 $total=0;
561 if ($result && (mysql_num_rows($result) > 0))
562     {
563     $total=mysql_num_rows($result);
564     $row = mysql_fetch_assoc($result);
565     }
566 if ($total>0)
567 {?>
568 <table align="center">
569     <tr class="lista_titlu">
570         <td align="center">Nr. </td>
```

# Separare cod PHP

- un fisier sursa PHP este un fisier HTML (in general) cu sectiuni de cod PHP
- interpretorul PHP cauta sectiunile pe care trebuie sa le interpreteze si interiorul lor proceseaza instructiuni ca fiind PHP
- ce se gaseste in **exteriorul** acestor sectiuni este trimis spre server-ul web **nemodificat**

# Separare cod PHP

- `<?php ... ?>`
  - stil XML – implicit, disponibil intotdeauna, recomandat
- `<? ... ?>`
  - scurt, este de obicei dezactivat
  - lipsa in PHP 7
- `<script language="php"> ... </script>`
  - stil script, disponibil
- `<% ... %>`
  - stil ASP, de obicei dezactivat
  - lipsa in PHP 7

# Variante de integrare

- echo .... afiseaza un text la "iesire" (echivalent cu printf() din C)
- poate realiza procesarea datelor
- in exemplu se trimite spre iesire un sir static (echivalent cu puts() din C)
- "iesire" in marea majoritate a cazurilor reprezinta datele trimise clientului de server-ul web
- "iesire" poate fi considerata de obicei:
  - documentul curent
  - pozitia curenta



# Variante de integrare

- Toate variantele ofera aceeasi sursa HTML pentru browser
- E **recomandata** cea care lasa structura HTML nemodificata si doar datele dinamice sunt rezultatul procesarii
- Codul HTML + PHP e interpretat mult mai elegant in editoarele WYSIWYG

```
<h2>Rezultate comanda</h2>  
<?php echo '<p>Comanda receptionata</p>';?>
```

```
<h2>Rezultate comanda</h2>  
<p><?php echo 'Comanda receptionata';?></p>
```

```
<?php echo '<h1>Magazin online XXX SRL</h1>';?>  
<?php echo '<h2>Rezultate comanda</h2>';?>  
<?php echo '<p>Comanda receptionata</p>';?>
```

```
<?php  
echo '<h1>Magazin online XXX SRL</h1>';  
echo '<h2>Rezultate comanda</h2>';  
echo '<p>Comanda receptionata</p>';  
?>
```

# PHP – instructiuni

- instructiunile PHP trebuie sa se termine cu ;
  - exceptie: se poate omite ; la sfarsitul blocului de cod php: ... `echo 'ceva' ?>`
- instructiunile pot sa fie scrise pe acelasi rand (fara trecerea la linia noua)
  - `echo 'ceva1'; echo 'ceva2'; ...`
  - nu este in general recomandat

# PHP – comentarii

- comentariile in PHP respecta regulile C si Perl
- la sfarsit de linie:
  - `// echo ceva; //comentariu`
  - `# echo ceva; #comentariu`
- comentariu bloc
  - `/* ... */`
  - `/* un comentariu  
pe mai multe  
linii */`

# PHP – constante

- Ca orice limbaj de programare PHP se bazeaza pe utilizarea
  - constante
  - variabile
  - functii
- Definirea constantelor:
  - `define('PRETCARTE', 100);`
  - "case sensitive"
  - prin conventie, numai cu litere mari
  - `echo PRETCARTE; // 100`

# PHP – variabile

- variabila – semnul \$ urmat de un nume
- numele e “case sensitive”
- o greseala frecventa e uitarea semnului \$
  - PHP Notice: Use of undefined constant an – assumed \$an (sau ‘an’) in D:\\Server\\
- Tipuri de date
  - scalar
  - compus
  - special

# PHP – tipuri de date

- scalar
  - boolean
  - integer
  - float (double)
  - **string**
- compus
  - array
  - object
- special
  - resource
  - NULL

# PHP – tipuri de date

- **declararea** variabilelor **nu** e necesara decat cand se declara un domeniu de definitie (variabile globale)
  - `global $a, $b;`  
`$c=$a+$b;`
- eliberarea memoriei nu este necesara, se face automat la terminarea executiei

# PHP – tipuri de date

```
$var = expresie
```

- Controlul variabilelor se face automat, “on the fly”
  - Daca \$var nu era definita anterior, in urma atribuirii se defineste de tipul dat de rezultatul expresiei
  - Daca \$var era definita, de un anumit tip (oarecare), in urma atribuirii devine de tipul dat de rezultatul expresiei
  - La finalizarea executiei script-ului se elimina variabila din memorie (automat)



# PHP – tipuri de date

- tipul de date este in totalitate dependent de ceea ce se stocheaza
- PHP reactualizeaza tipul pentru a putea primi ceea ce se stocheaza

```
<?php
echo $variabila ; // tip Null, neinitializat – valoare NULL (doar)
$variabila = "0"; // $variabila tip string (ASCII 48)
$variabila += 2; // $variabila tip integer (2)
$variabila = $variabila + 1.3; // $variabila tip float (3.3)
$variabila = 5 + "10 obiecte"; // $variabila tip integer (15)
$var2=5; // $var2 tip integer (5)
$variabila=$var2."10 obiecte"; // $variabila tip string "510 obiecte"
?>
```

# PHP – operatori

- In general similari celor din C/C++
- Operatori
  - Aritmetici
  - Atribuire
  - Bit
  - Comparare
  - Incrementare/Decrementare
  - Logici
  - **Sir**

# PHP – operatori

- Aritmetici
  - $-$a$  – Negare
  - $$a + $b$  – Adunare
  - $$a - $b$  – Scadere
  - $$a * $b$  – Inmultire
  - $$a / $b$  Impartire
  - $$a \% $b$  Modulo (rest)
- Sir
  - **$$a.$b$  – Concatenare sir a si sir b**

# PHP – operatori

## ■ Atribuire

- `$a=$b`
- `$a+=$b` ( $a=a+b$ )
- `$a-=$b` ( $a=a-b$ )
- `$a/=$b` ( $a=a/b$ )
- `$a*=$b` ( $a=a*b$ )
- `$a%=$b` ( $a=a\%b$ )
- `$a.= $b` ( $a=a$  concatenat  $b$  - siruri)

# PHP – operatori

- Operatori la nivel de bit
  - similari celor din C
  - `~, &, |, ^, <<, >>`
- Operatori logici
  - ofera rezultat boolean true/false
  - similari celor din C
  - `&&, ||, !`
  - suplimentar
    - `and, or, xor` – echivalenti dar de prioritate mai mica
    - `$a=55/0 or die('impartire prin 0');`

# PHP – operatori

- Operatori de comparare
  - ofera rezultat boolean true/false
  - similari celor din C
  - == , != , > , < , <> , >= , <=
  - suplimentar
    - === identic, valoare egala **SI** de acelasi tip
    - !== “neidentic”, valoare diferita **SAU** de tipuri diferite

# Precedenta operatorilor

non-associative	clone new	<a href="#">clone</a> and <a href="#">new</a>
left	[	<a href="#">array()</a>
non-associative	++ --	<a href="#">increment/decrement</a>
right	~ - (int) (float) (string) (array) (object) (bool) @	<a href="#">types</a>
non-associative	instanceof	<a href="#">types</a>
right	!	<a href="#">logical</a>
left	* / %	<a href="#">arithmetic</a>
left	+ - .	<a href="#">arithmetic</a> and <a href="#">string</a>
left	<< >>	<a href="#">bitwise</a>
non-associative	< <= > >= <>	<a href="#">comparison</a>
non-associative	== != === !==	<a href="#">comparison</a>
left	&	<a href="#">bitwise</a> and <a href="#">references</a>
left	^	<a href="#">bitwise</a>
left		<a href="#">bitwise</a>
left	&&	<a href="#">logical</a>
left		<a href="#">logical</a>
left	? :	<a href="#">ternary</a>
right	= += -= *= /= .= %= &=  = ^= <<= >>=	<a href="#">assignment</a>
left	and	<a href="#">logical</a>
left	xor	<a href="#">logical</a>
left	or	<a href="#">logical</a>
left	,	many uses

# Precedenta operatorilor PHP7

non-associative	clone new	<a href="#">clone</a> and <a href="#">new</a>
left	[	<a href="#">array()</a>
non-associative	++ --	<a href="#">increment/decrement</a>
right	~ - (int) (float) (string) (array) (object) (bool) @	<a href="#">types</a>
non-associative	instanceof	<a href="#">types</a>
right	!	<a href="#">logical</a>
left	* / %	<a href="#">arithmetic</a>
left	+ - .	<a href="#">arithmetic</a> and <a href="#">string</a>
left	<< >>	<a href="#">bitwise</a>
non-associative	< <= > >= <>	<a href="#">comparison</a>
non-associative	== != === !== <=>	<a href="#">comparison</a>
left	&	<a href="#">bitwise</a> and <a href="#">references</a>
left	^	<a href="#">bitwise</a>
left		<a href="#">bitwise</a>
left	&&	<a href="#">logical</a>
left		<a href="#">logical</a>
right	??	<a href="#">comparison</a>
left	? :	<a href="#">ternary</a>
right	= += -= *= /= .= %= &=  = ^= <<= >>=	<a href="#">assignment</a>
left	and	<a href="#">logical</a>
left	xor	<a href="#">logical</a>
left	or	<a href="#">logical</a>
left	,	many uses



# Exemplu de separare cod php

## Raspuns dinamic

- `<form action="rezultat.php" method="post">`

```
<html>
<head>
<title>Magazin online XXX SRL</title>
</head>
<body>
<h1>Magazin online XXX SRL</h1>
<h2>Rezultate comanda</h2>
<p><?php echo 'Comanda receptionata';?></p>
</body>
</html>
```

**Magazin online XXX SRL**

**Rezultate comanda**

Comanda receptionata

```
<
<
<
</head>
<body>
<h1>Magazin online XXX SRL</h1>
<h2>Rezultate comanda</h2>
<p>Comanda receptionata</p>
</body>
</html>
```

# PHP – Functii

- conceptual similare celor din C/C++
- functiile nu trebuie declarate inainte de a fi folosite
- numele functiilor este “case-insensitive”
- un mare numar de functii cu utilitate directa in aplicatiile web exista in bibliotecile PHP
- unele biblioteci trebuie activate in momentul configurarii PHP
  - `extension=php_gd2.dll` (linia 639) // pentru functii de procesare grafica de exemplu
  - `extension=php_mysql.dll` (linia 651) // pentru functii de acces la baze de date MySql
  - `extension=php_mysqli.dll` (linia 652) // pentru functii de acces la baze de date MySql (**obligatoriu** de la PHP 5.6)

# Utilizarea functiilor PHP

- `<form action="rezultat.php" method="post">`

```
<p>Comanda receptionata la data:
```

```
<?php echo date('d/m/Y')." ora ".date('H:i');?></p>
```

**Magazin online XXX SRL**

**Rezultate comanda**

Comanda receptionata la data: 10/03/2010 ora 13:36

```
<body>
```

```
<h1>Magazin online XXX SRL</h1>
```

```
<h2>Rezultate comanda</h2>
```

```
<p>Comanda receptionata la data:
```

```
10/03/2010 ora 13:36</p>
```

```
</body>
```

# Elemente de control

- majoritatea notiunilor si sintaxei sunt similare celor din C/C++
- instructiune compusa: separata de acolade {...}
- if / else / elseif – executie conditionata

```
<?php
if ($a > $b) {
    echo "a mai mare ca b";
} elseif ($a == $b) {
    echo "a egal cu b";
} else {
    echo "a mai mic ca b";
}
?>
```

# Elemente de control

- while
- do-while
- for
- switch
- return
- break
- goto
  
- Similare cu echivalentele C/C++

```
$i = 1;  
while ($i <= 10) {  
    echo $i++;  
}
```

```
$i = 10;  
do {  
    echo $i--;  
} while ($i > 0);
```

```
for ($i = 1; $i <= 10; $i++) {  
    echo $i;  
}
```

```
switch ($i) {  
    case 0:  
        echo "i este 0";  
        break;  
    case 1:  
        echo "i este 1";  
        break;  
    default:  
        echo "i nici 1 nici 0";  
        break;  
}
```

# Elemente de control

- `include()`
- `require()`
- `include_once()`
- `require_once()`
  
- pentru inserarea **SI** evaluarea fisierului folosit ca parametru
- folosite pentru a nu multiplica sectiunile de cod comune
- `require` opreste executia script-ului curent daca fisierul parametru nu este gasit
- `..._once()` verifica daca respectivul fisier a mai fost introdus si **nu** il mai introduce inca o data

# Variabile tip string

# PHP – tipuri de date

- scalar
  - boolean
  - integer
  - float (double)
  - **string**
- compus
  - array
  - object
- special
  - resource
  - NULL



# Variabile tip string

- Scopul final al PHP e popularea cu date (sub forma de text) a campurilor existente intr-un schelet HTML
- Ca urmare datele de tip sir de caractere (string) sunt tratate mai complex decat echivalentul C/C++
  - mai multe modalitati de definire
  - mai multe modalitati de interpretare
  - **mult** mai multe functii

# Variabile tip string

- definire variabila de tip string
  - utilizare apostrof ` `
  - utilizare ghilimele " "
  - definiri tip bloc
    - heredoc <<< "X"
    - nowdoc <<<'X' (PHP>5.3.0)

# Variabile tip string ` `

- apostroful ` ` e utilizat pentru definirea sirurilor primare de caractere
  - se defineste o suita de caractere
  - prelucrarile in interiorul sirului sunt reduse
    - \' reprezinta caracterul apostrof
    - \\ si \ reprezinta caracterul backslash
    - doar atat!!!

# Variabile tip string ""

- ghilimelele "" sunt utilizate pentru definirea sirurilor de caractere complexe
  - prelucrarile in interiorul sirului sunt mai complexe decat echivalentul C/C++
    - caracterele ASCII speciale, identic cu C++: \n, \r, \t, \\", \v, \e, \f, \x, \u
    - \" caracterul ghilimele
    - \\$ caracterul \$
    - se interpreteaza **variabile** in interiorul sirului !!!

# Variabile tip string “ ”

- caracterul \$ indica faptul ca urmeaza un nume de variabila
  - interpretorul foloseste toate caracterele care pot genera nume de variabile valide (\$x, \$x->y, \$x[y])
  - daca e nevoie de exprimare mai complexa a variabilelor (de exemplu matrici cu 2 indici x[y][z] sau cu indici neintregi) se foloseste sintaxa complexa: **{ }**

# Variabile tip string " "

- sintaxa **simpla** pentru interpretarea variabilelor in interiorul sirurilor

```
<?php
$juice = "apple";

echo "He drank some $juice juice.";
// He drank some apple juice.
echo "He drank some juice made of $juices.";
// He drank some juice made of . //s character valid pentru variabile

?>
```

# Variabile tip string ""

- sintaxa **simpla** pentru interpretarea variabilelor in interiorul sirurilor

```
<?php
$juices = array("apple", "orange", "koolaid1" => "purple");
class people {
    public $john = "John Smith";
}

$people = new people();
echo "$people->john drank some $juices[0] juice.";
// John Smith drank some apple juice.
?>
```

# Variabile tip string " "

- sintaxa **complexa** pentru interpretarea variabilelor in interiorul sirurilor **{ }**

```
<?php
$juice = "apple";

echo "He drank some juice made of $juices.";
// He drank some juice made of . //s character valid pentru variabile
echo "He drank some juice made of ${juice}s."
// He drank some juice made of apples. // {} arata unde se incheie
numele variabilei
?>
```



# Variabile tip string ""

- sintaxa **complexa** pentru interpretarea variabilelor in interiorul sirurilor **{ }**

```
<?php
$juices = array(array("apple", "orange"), "koolaid1" => "purple");
class people {
    public $name = "John Smith";
}

$obj->values[3] = new people();
echo "$obj->values[3]->name drank some $juices[0][1] juice.";
// drank some juice.
echo "{ $obj->values[3]->name } drank some { $juices[0][1] } juice.";
// John Smith drank some apple juice.
?>
```

# Variabile globale

# Variabile globale

- Variabilele globale (predefinite)
  - accesibile script-urilor PHP prin conlucrarea cu server-ul
  - Exemple:
    - `$_SERVER` — Server and execution environment information
    - `$_GET` — HTTP GET variables
    - `$_POST` — HTTP POST variables
    - `$_FILES` — HTTP File Upload variables
    - `$_REQUEST` — HTTP Request variables
    - `$_SESSION` — Session variables
    - `$_ENV` — Environment variables
    - `$_COOKIE` — HTTP Cookies

# Interactiunea cu utilizatorul

- Datele introduse de utilizator in forme se regasesc (in functie de metoda aleasa pentru forma) in una din variabilele:
  - `$_POST` – method="post"
  - `$_GET` – method="get"
  - `$_REQUEST` – ambele metode
- variabilele sunt **matrici** cu **cheia** data de atributul **name** din forma HTML
  - `<input type="text" name="carti_cant" size="3" maxlength="3" />`
  - `$_POST['carti_cant']` contine valoarea introdusa

# Laborator 2

# HTML

- se creaza o pagina care sa arate ca in schita alaturata
- forma paginilor:
  - tabel
- Se experimenteaza diversele elemente de **interactiune cu utilizatorul** (**<form>**)

culoare	<b>IMAGINE</b>	culoare
	<b>Continut</b> (cu alta culoare fundal)	

# Suplimentar

- forma din schita alaturata
- forma paginilor:
  - tabel controlat prin CSS

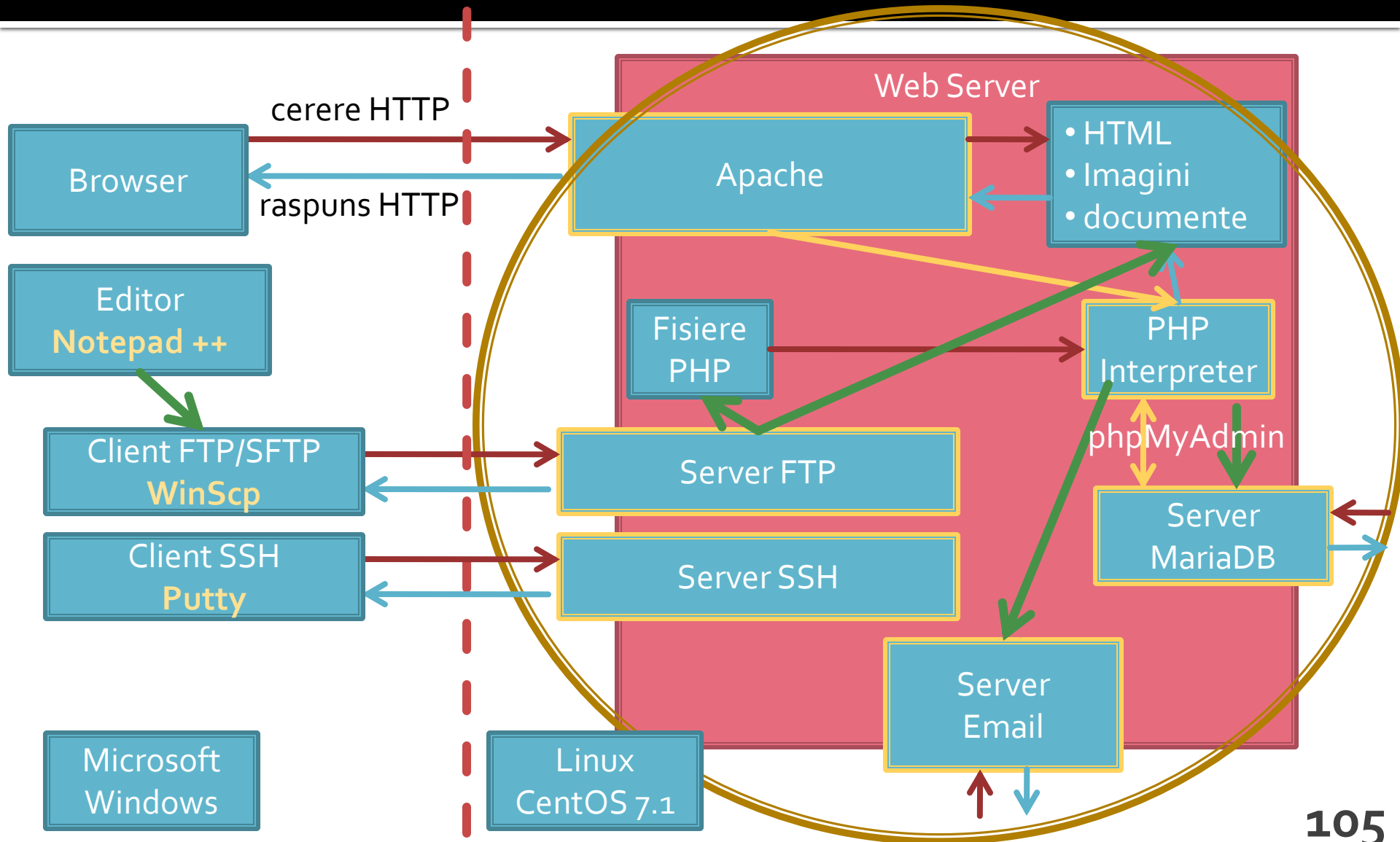
culoare	<b>IMAGINE</b>	culoare
	<b>Continut</b> (cu alta culoare fundal)	
	<b>Copyright</b> (cu alta culoare fundal)	

# Exemplu



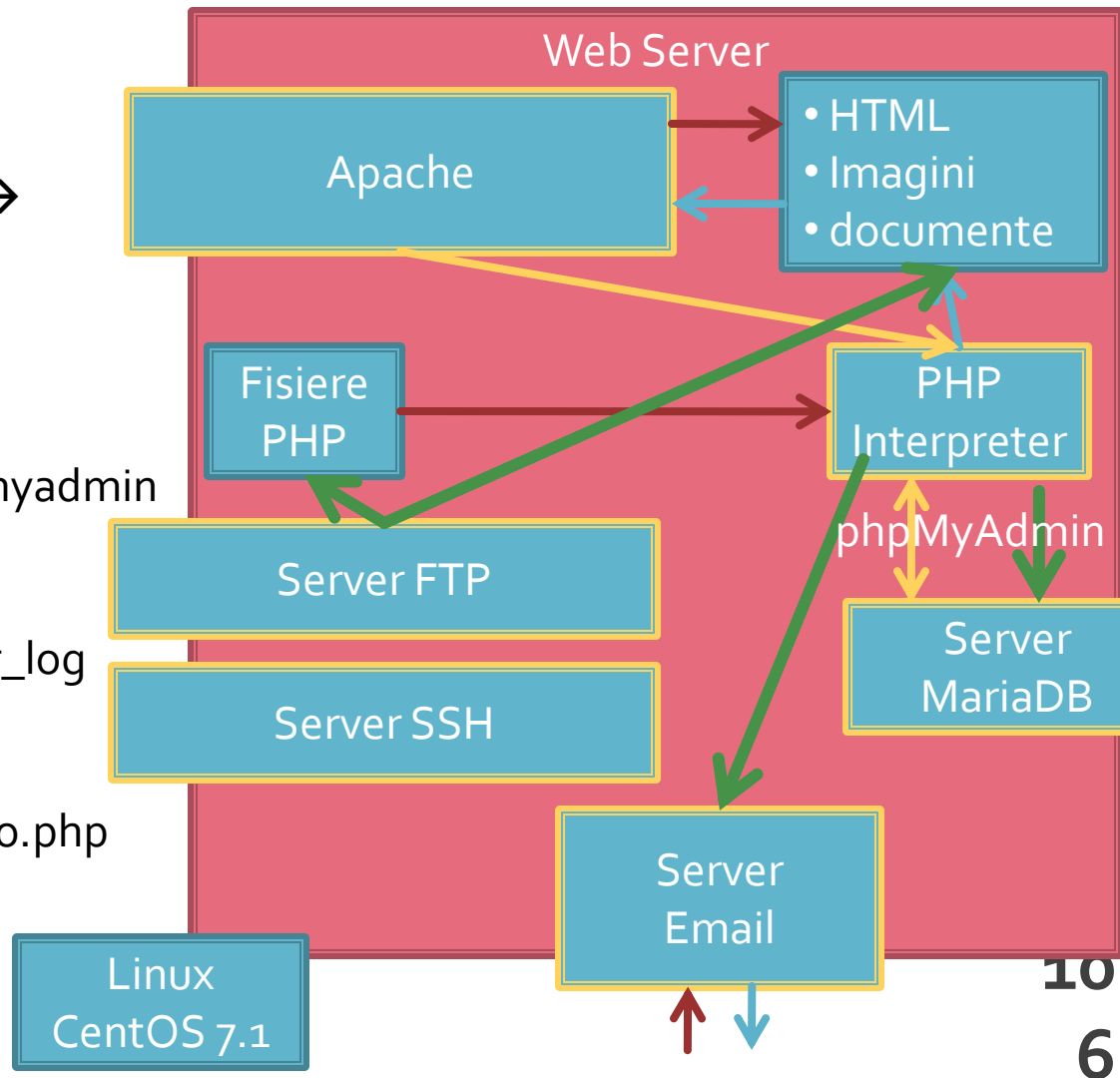


# Utilizare LAMP



# Utilizare LAMP

1. login → root:masterrc
2. ifconfig → 192.168.30.5
3. putty.exe → 192.168.30.5 → SSH → root:masterrc (remote login)
4. [alte comenzi linux dorite]
5. FTP → Winscp → SFTP → student:masterrc@192.168.30.5
6. MySql → http://192.168.30.5/phpmyadmin → root:masterrc
7. Apache Error Log →
  - 7a. putty → nano /var/log/httpd/error\_log
  - 7b. http://192.168.30.5/logfile.php (nonstandard)
8. PHP info → http://192.168.30.5/info.php



# Contact

- Laboratorul de microunde si optoelectronica
- <http://rf-opto.etti.tuiasi.ro>
- [rdamian@etti.tuiasi.ro](mailto:rdamian@etti.tuiasi.ro)