Web Information Systems

https://bit.ly/PolitoSIA

SOftEng
http://softeng.polito.it

Version 2.0.0 – November 9, 2020

© Fulvio Corno, Marco Torchiano, 2020

Licensing Note

This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 2.5 International License.

To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

You are free: copy and redistribute the material in any medium or format; remix, transform, and build upon the material

Under the following conditions:

Attribution. You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

Non-commercial. You may not use this work for commercial purposes.

ShareAlike – If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original..

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

Acknowledgments

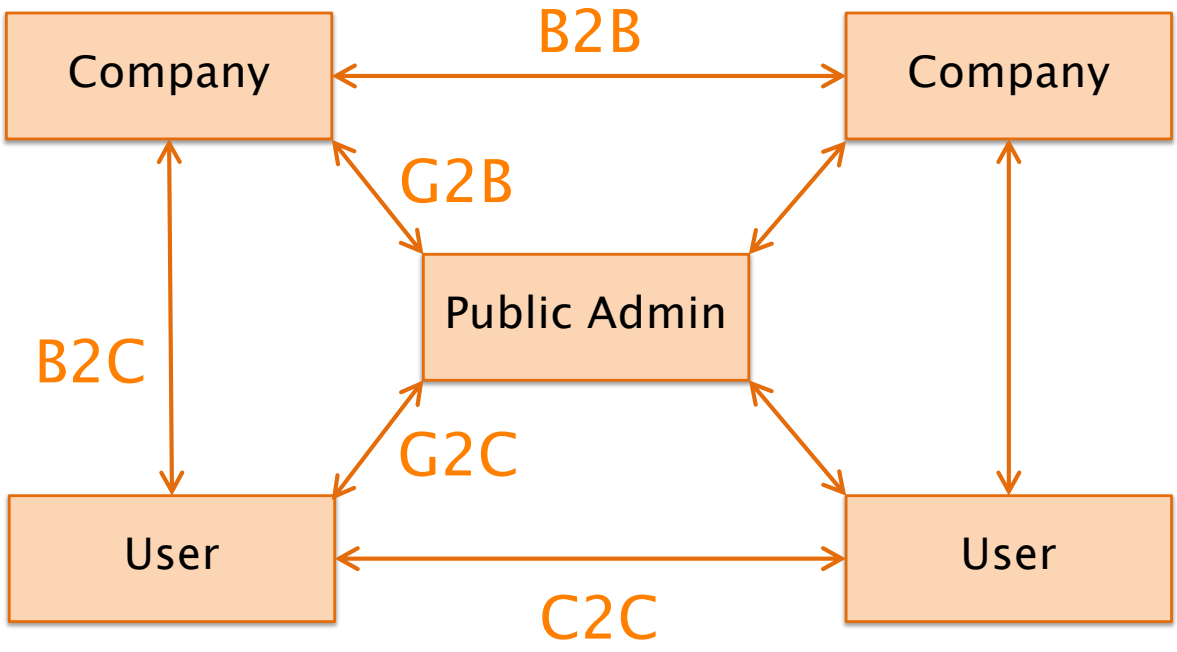
- This set of slides are derived from those authored by Prof. Fulvio Corno for previous issues of the course “Sistemi Informativi Aziendali” at Politecnico di Torino
- Many thanks to Fulvio for kindly sharing his materials

WEB-BASED IS

Definition

- Web Information System (WIS)
  - Communication between computers and hosts takes place in the Internet or through a Virtual Private Network (VPN) based on the internet standards
  - Access to information and services is supported by a program that manage the user interface, known as browser

Actors



Collaboration models

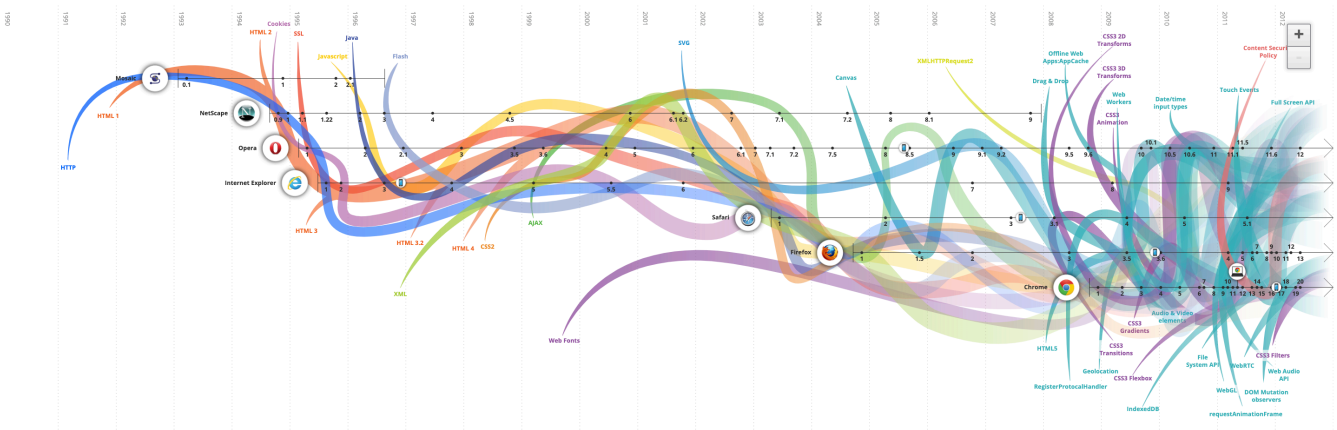
- B2B (business to business): collaboration among companies
- B2C (business to consumer): on-line shops
- C2C (consumer to consumer): auctions, buy-sell notices
- Government to business : on-line taxes, services to companies
- Government to citizens : on-line taxes

Levels of complexity

- Informative sites
  - Who we are / Products / Services / Contacts
  - Newsletter, Journal, Blog, …
- Ordering sites
  - Selection, configuration, purchase
- Management systems
  - CRM, SCM, ERP, MRP, …
- Autonomous systems
  - Negotiation, transaction, monitoring
- Portals, marketplace, marketspace
  - Aggregate many related companies/products

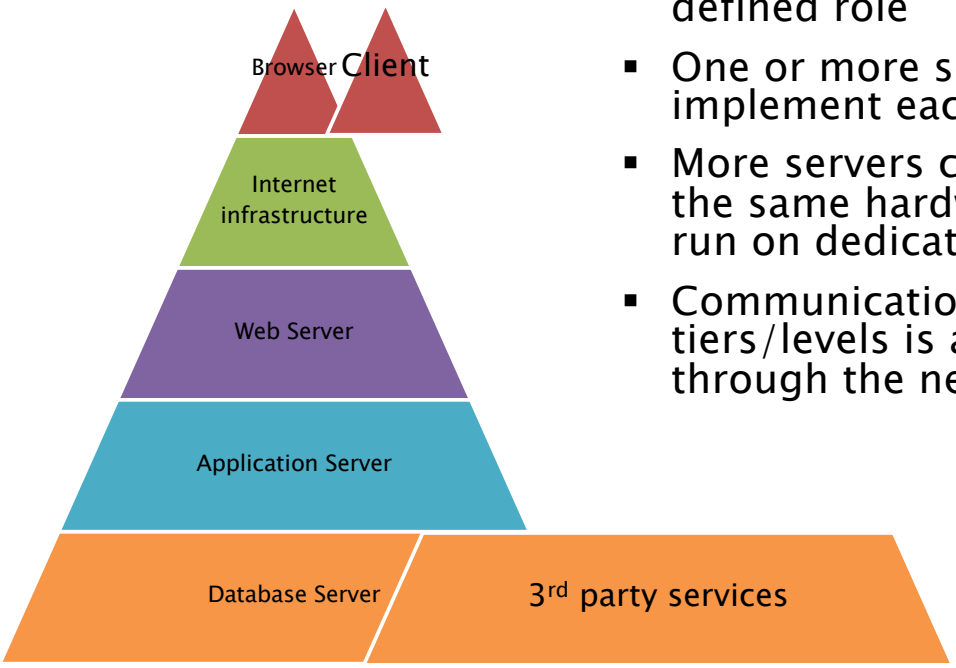
WEB ARCHITECTURE

Evolution of web architectures



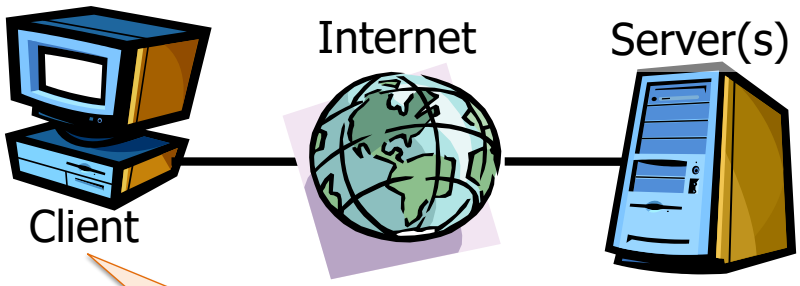
http://www.evolutionoftheweb.com

N-tier (N-level) architecture



- Each level/tier has a well defined role
- One or more servers implement each tier/layer
- More servers can share the same hardware or can run on dedicated devices
- Communication between tiers/levels is achieved through the network

General base architecture



Historically, a web browser.
Nowadays also:
- Mobile app
- Desktop app
- Other server application

Components

- One or more connections to the Internet by means of an Internet Service Provider (ISP).
- One or more servers implementing each tier/level of the architecture.
- One or more physical networks for interconnecting the servers.
- One or more network devices (router, firewall, switch) which implement communication and security policies.

Communication Protocol

- Set of rules to transfer information between two (or +) parties
  - Syntax and format
  - Semantics
  - General procedure (steps)
  - Synchronization mechanisms
  - Error recovery methods

Network Protocols Stack

- HTTP, HTTPS
  - Transfer of content
- TCP
  - Reliable transfer of variable length data
- IP
  - Allow transfer of data though a network
- WiFi, Ethernet, Bluetooth, 4G, 5G
  - Local networking (logical access, medium access, physical details)

Server

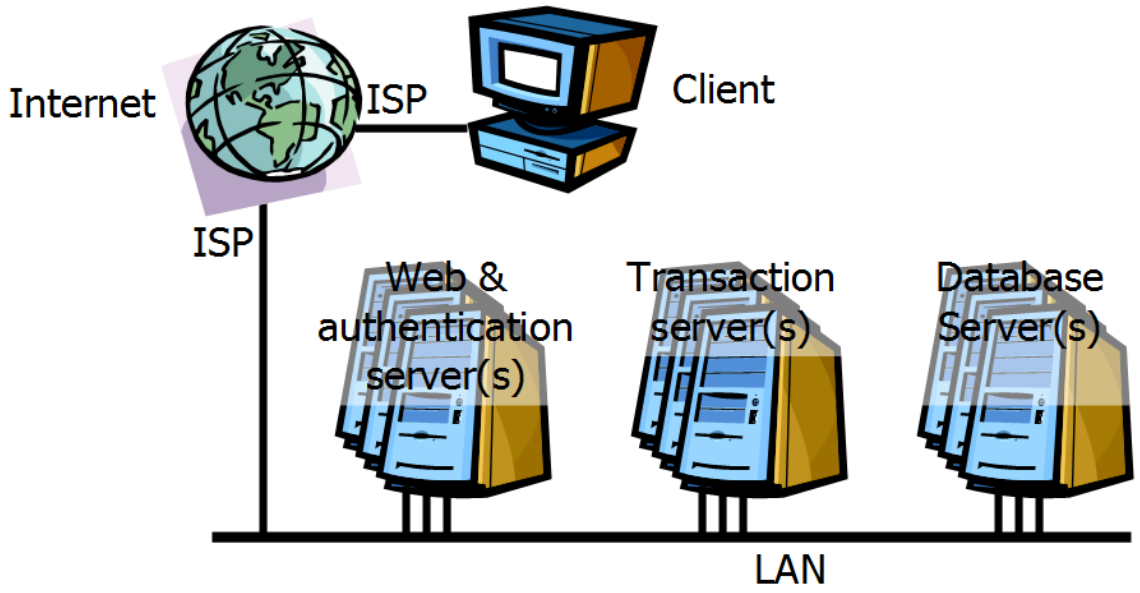
- Logical
  - A process that runs on a host that relays information to a client upon the client sending it a request.
- Physical
  - A host computer on a network that holds information (e.g., web sites) and responds to requests for information

BASIC HTTP SERVER

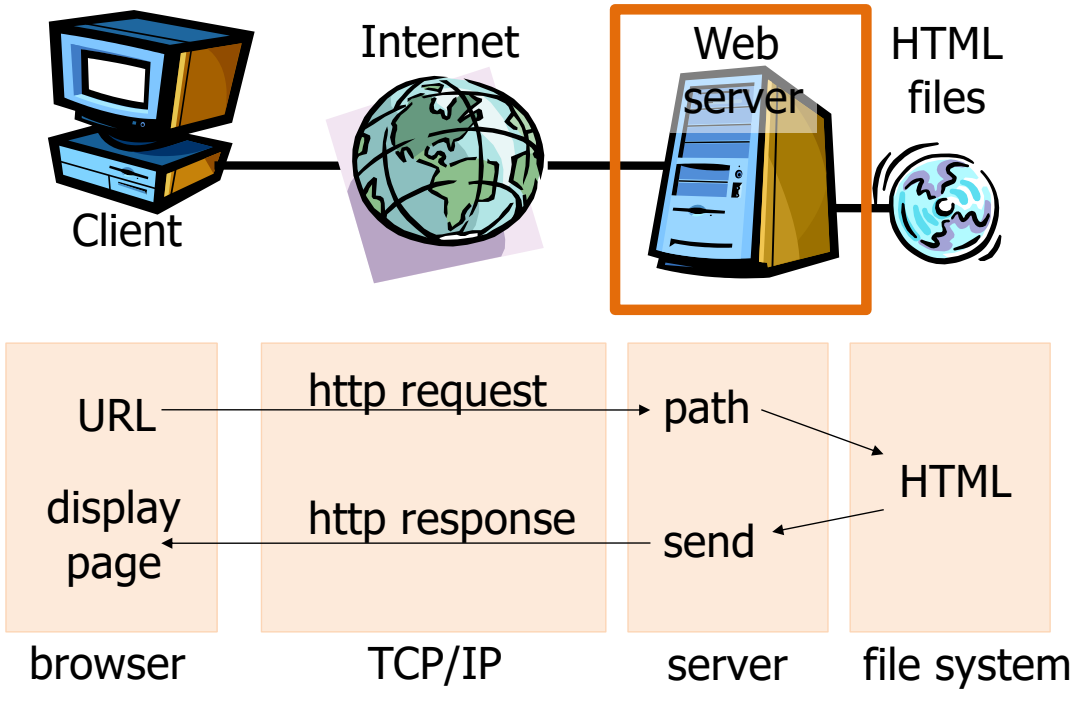
Web server

- Manages the HTTP protocol
  - Receives client requests
  - Reads static pages from the file system
  - Activates the application server for dynamic pages (server-side)
  - Sends an HTML file back to the client
- One HTTP connection for each request
- Multi-process, Multi-threaded or Process pool

General Architecture



Example

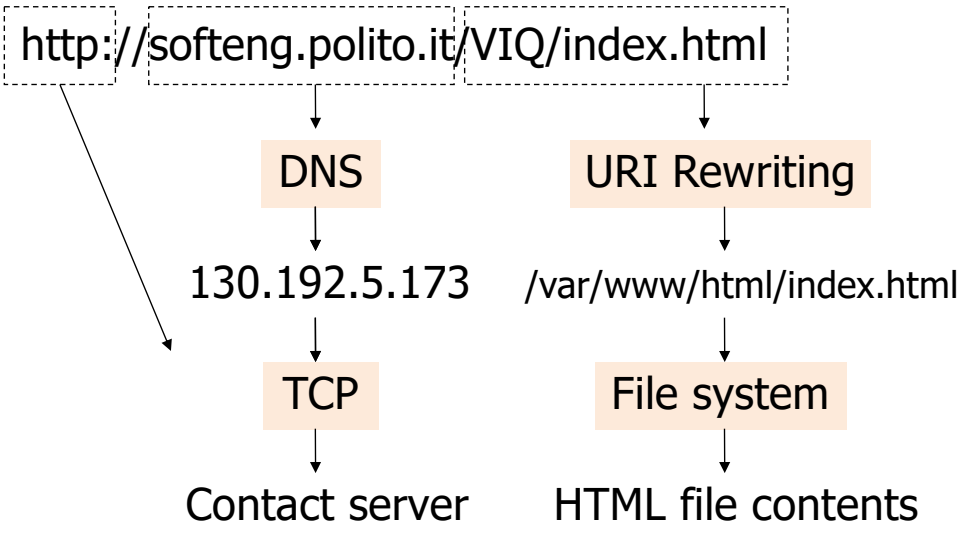


Adopted standards

- URL (uniform resource locator) for finding web pages
- HTTP (hyper text transfer protocol) for client-server interaction
- HTML (hyper text markup language) for writing web pages

URL

RFC 2396
http://www.w3.org/Addressing/



URI Basics

- Scheme: http
  Hostname: www.sadev.co.za
  Query: /users/1/contact
  http://www.sadev.co.za/users/1/contact
- Scheme: http
  Hostname: www.sadev.co.za
  Query: ?user=1&action=contact
  http://www.sadev.co.za?user=1&action=contact
- Scheme: http
  Userinfo: rob:pass
  Hostname: bbd.co.za
  Port: 8044
  http://rob:pass@bbd.co.za:8044
- Scheme: https
  Hostname: bbd.co.za
  Query: /index.html
  Fragment: about
  https://bbd.co.za/index.html#about

HTTP protocol

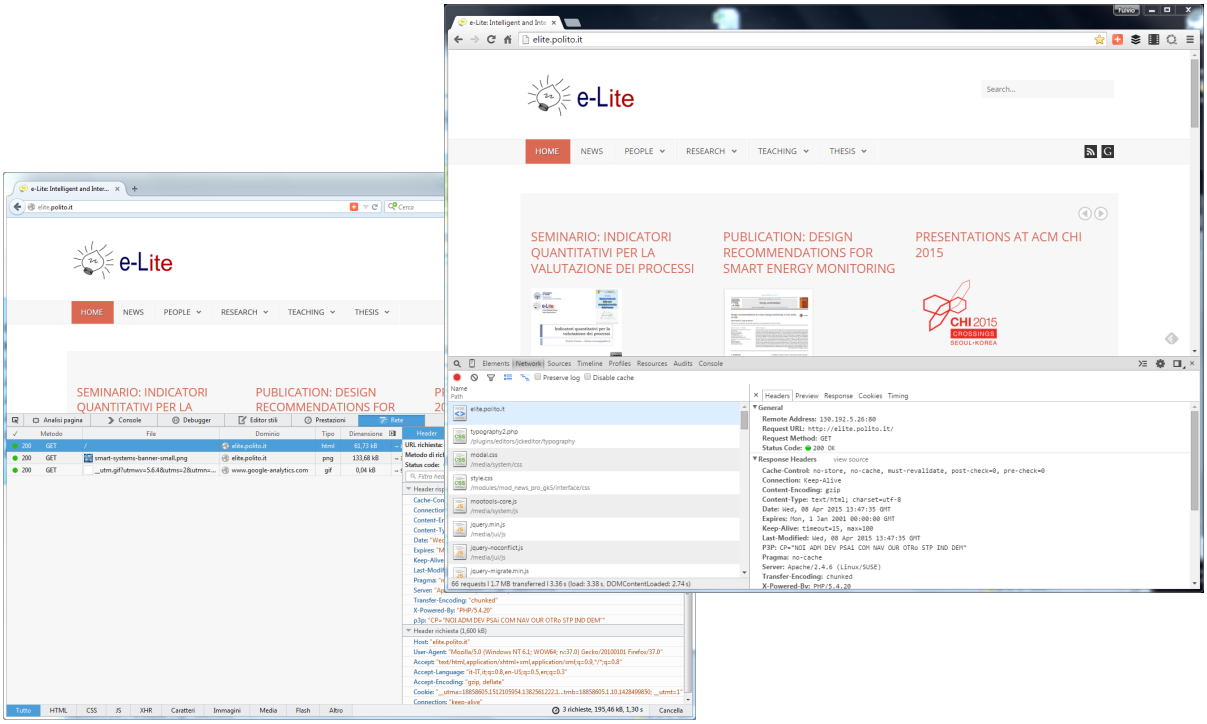
RFC 2616, RFC 2617
http://www.w3.org/Protocols

```
GET /VIQ/index.html HTTP/1.0
Accept: text/html
Accept: image/gif
User-Agent: FireChrome SuperBrowser 9.45
```

```
HTTP/1.0 200 OK
Date: Monday, 01-Jan-2001 00:00:00 GMT
Server: Apache 1.3.0
MIME-Version: 1.0
Last-Modified: 31-Dec-2000
Content-type: text/html
Content-length: 3021

<HTML> . . .
```

Browser developer tools



Performance measures

- Latency: time required for providing a 0 byte http page. Includes the server activation time, the request decoding time, the file access time, the transmission time and the time for closing the connection.
  - Unit of measure: http/s or s/http
- Throughput: maximum speed at which infinite-sized pages can be sent.
  - Unit of measure: Bytes (Mbytes)/s
- RPS: max(#Requests / s)

Delay time

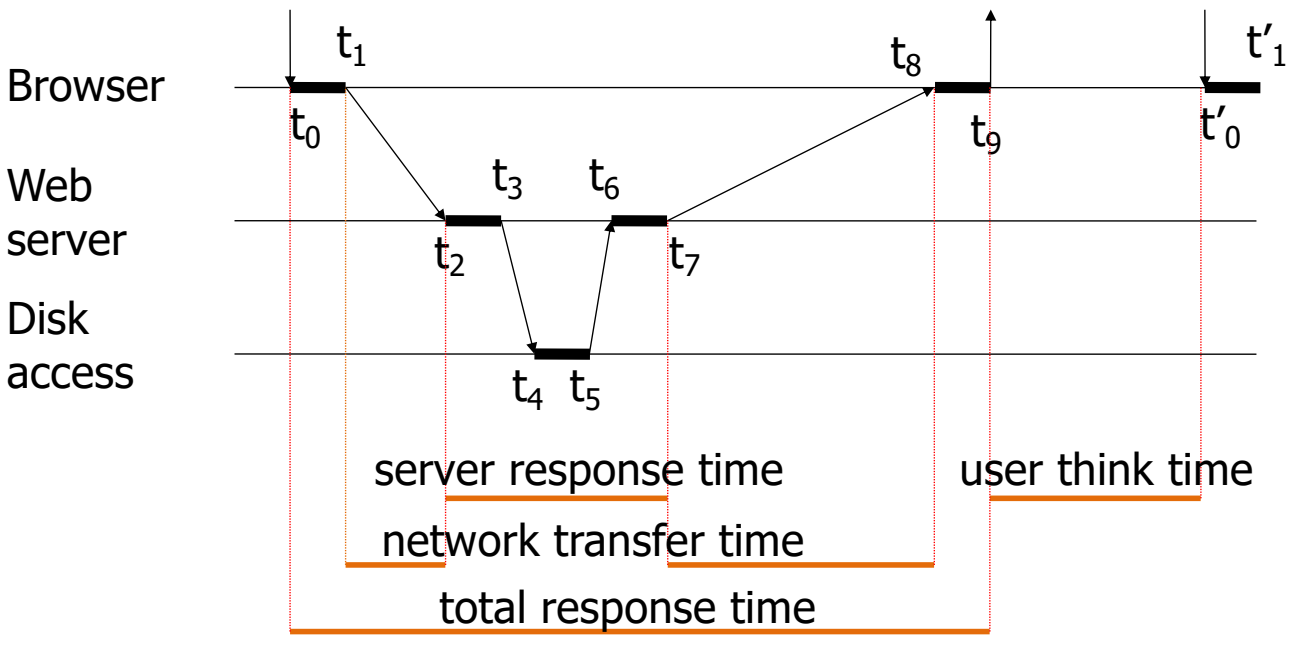
$$Delay = Latency + \frac{Size}{Throughput}$$

- Valid if all components are not overloaded

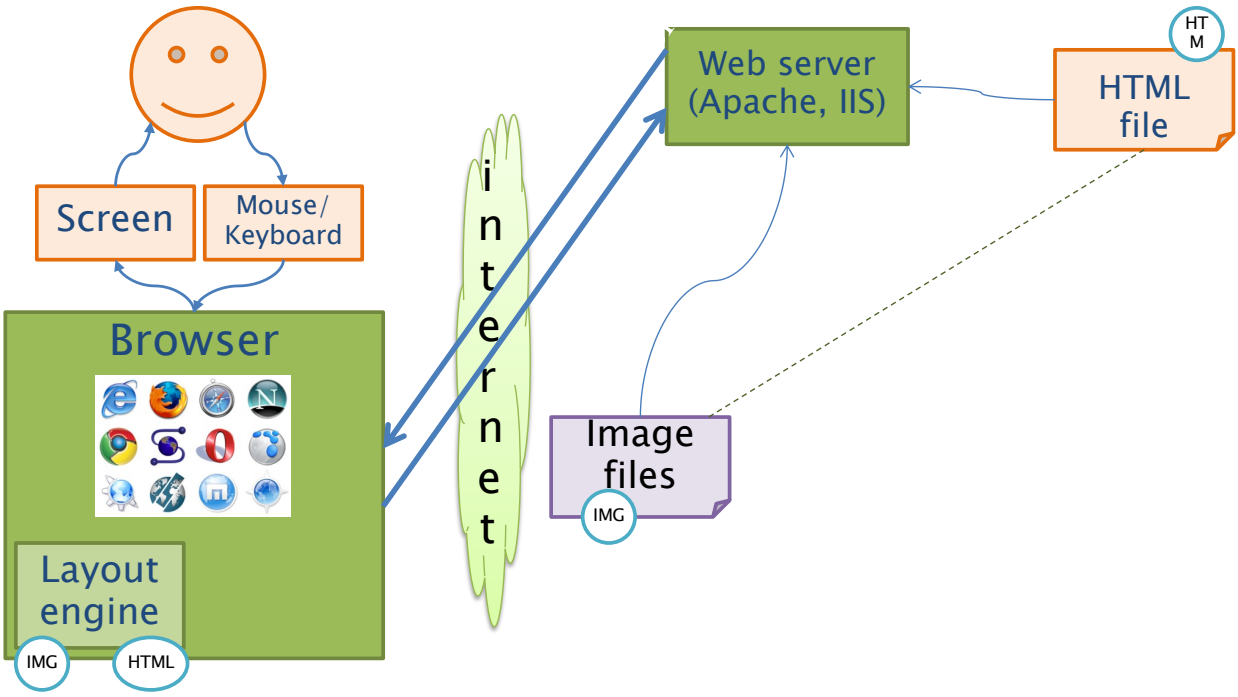
- Example:
  - Latency: 100ms
  - Size: 100kBytes
  - Throughput: 800kBytes/s

$$Delay = 0.1s + \frac{100kB}{800\frac{kB}{s}} = 0.225s$$

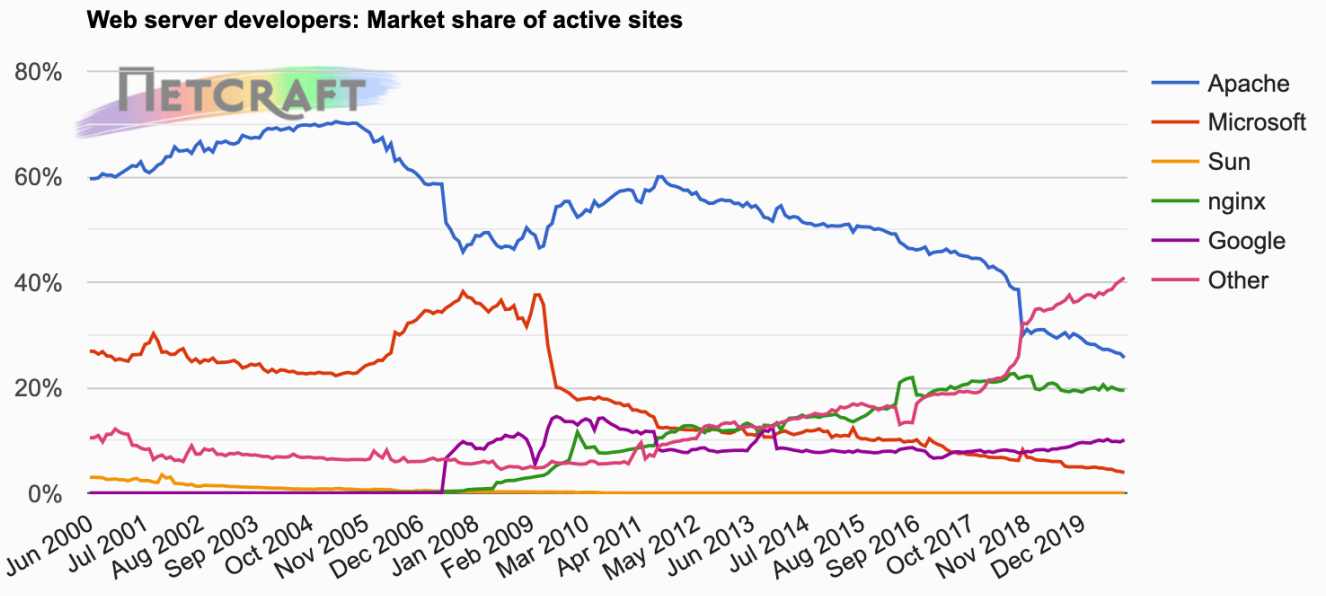
Static web transaction



Basic web architecture

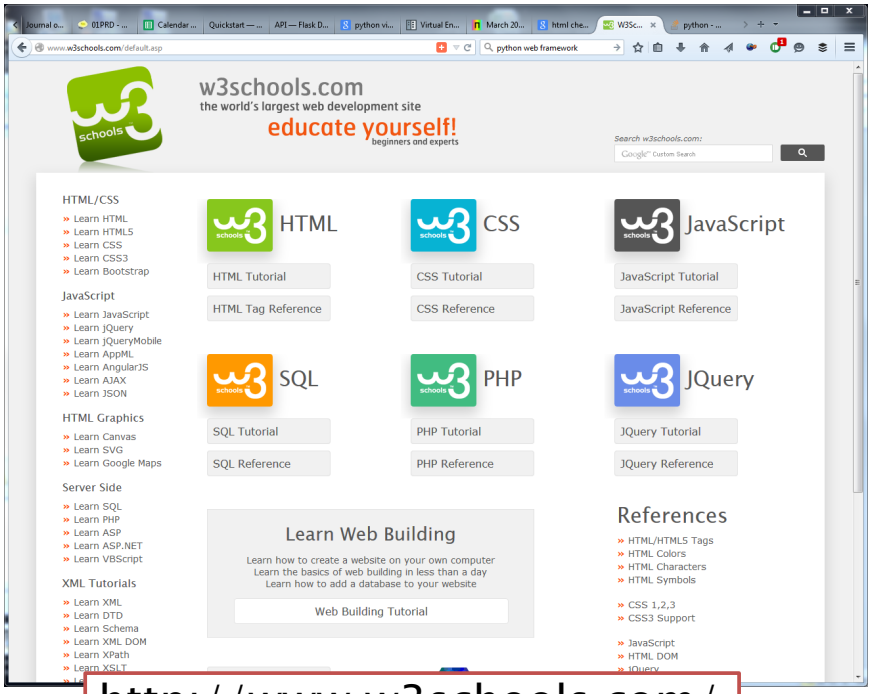


Market share of active sites



Source: https://news.netcraft.com/archives/2020/10/21/october-2020-web-server-survey.html

HTML in 5 minutes



http://www.w3schools.com/

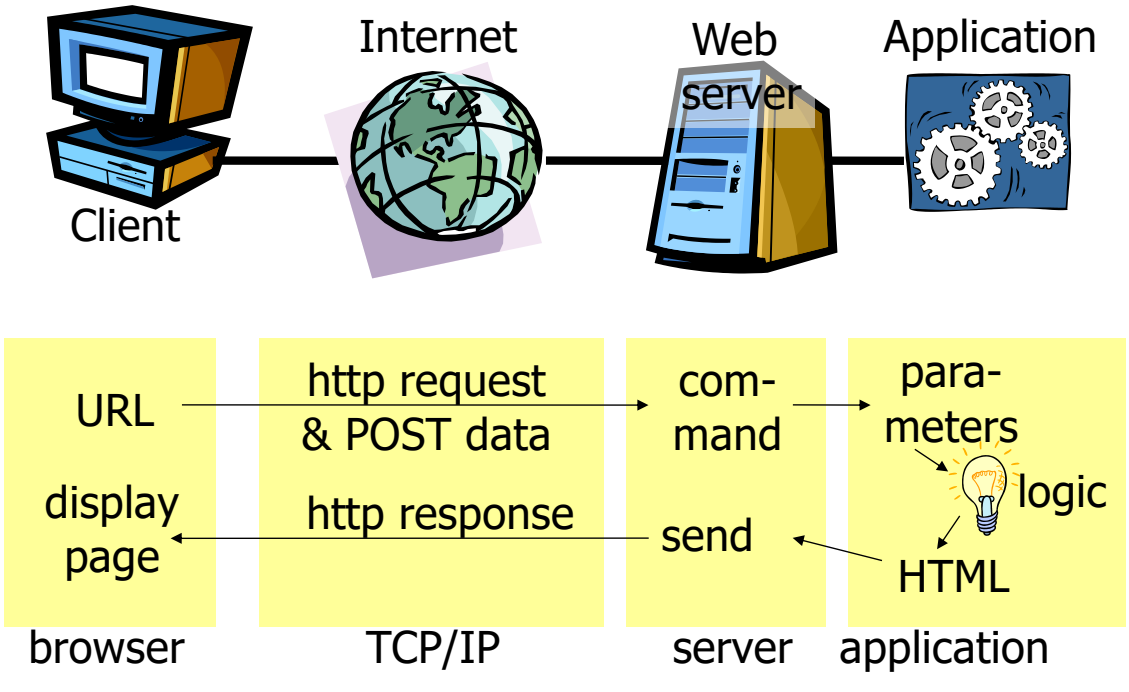
Web Information Systems – Part 2

APPLICATION SERVER

Application server

- Dynamic page generation
- Manages the site business logic
- It's the middle tier between the client browser and the data residing on a database
- Implements the session mechanisms
- Different technologies and architectures are available

Dynamic web transaction



Adopted standards

- HTTP-POST for sending user-specified data
- CGI (common gateway interface), ISAPI (internet information server application programming interface), server-side script, java servlet for integrating application logic into web servers
- ASP (active server pages), PHP, Ruby as languages for application development
- Cookies for storing the state of a session

Sessions

- A session is a set of related HTTP transactions performed by the same user
- They allow a continuous (state-full) interaction with the web application upon a state-less protocol (HTTP/S)
  - Each HTTP exchange is independent
- They are usually based on cookies

Cookies

- A cookie is a pair name and value
- Cookies are sent by the web server
  - Usually upon the first interaction
- The browser keep a list of
  - Host – Cookie (name + value)
- When a browser sends an request to a host is sends along all the cookies previously received from that host

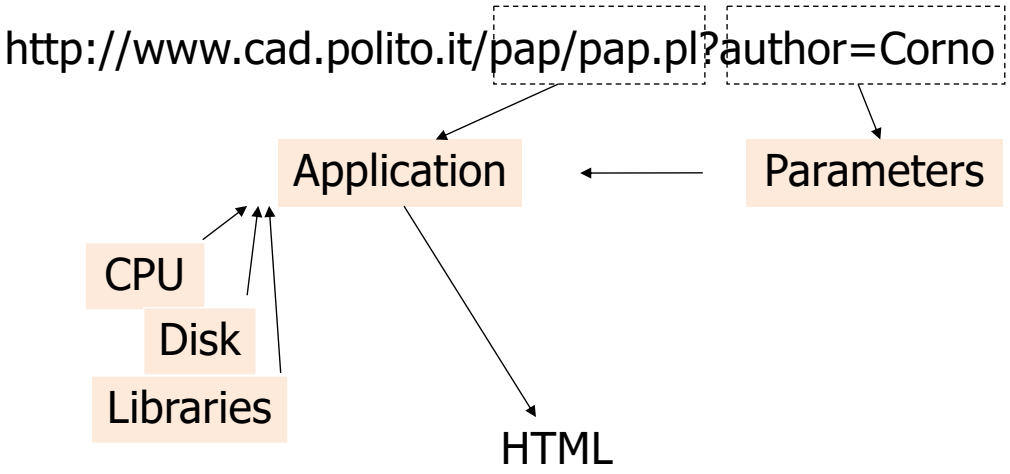
Session cookies

- When the web server receives a request, it looks for a session cookie
- If none is received, then a new session must be started
  - A new unique session cookie is generated to identify the session
  - The cookie is sent back to the browser with the response
- If one is present, it means a session is already in progress
  - All the operations are linked to the current session

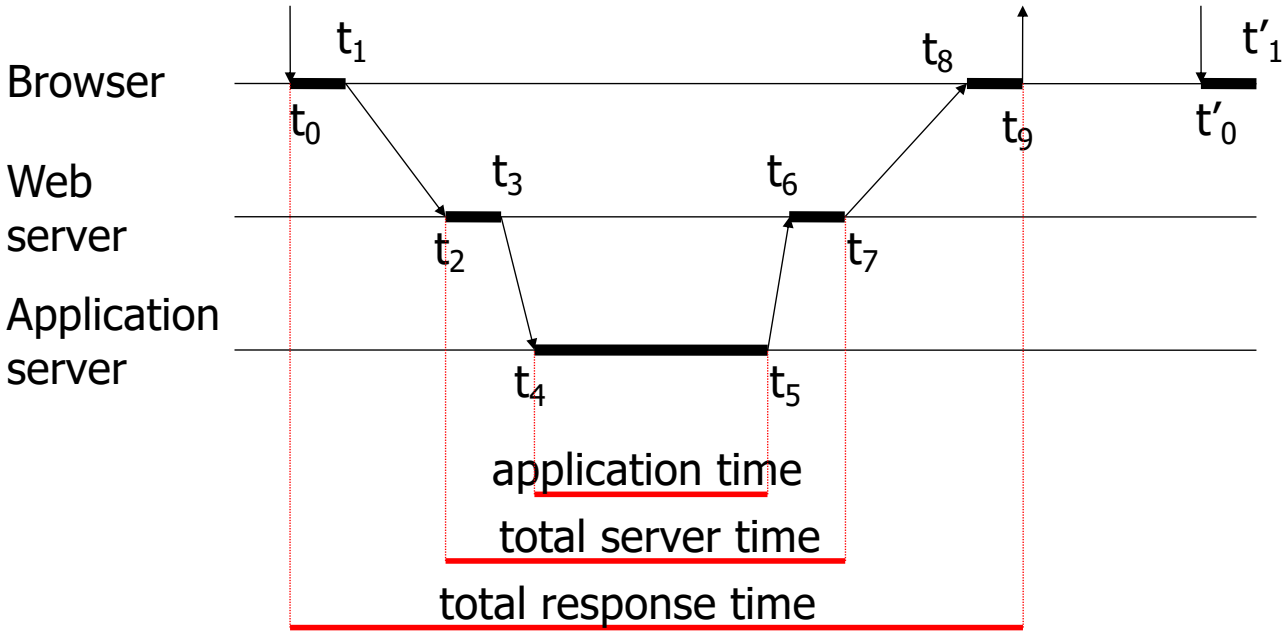
Session cookies

- When the user is identified (e.g. through a login) then the session is linked to the user
  - All following operations will be relative to the user linked to the session
- Cookies have an expiration time
  - After expiration they are not sent back to the server anymore
  - Thus sessions have a time limit

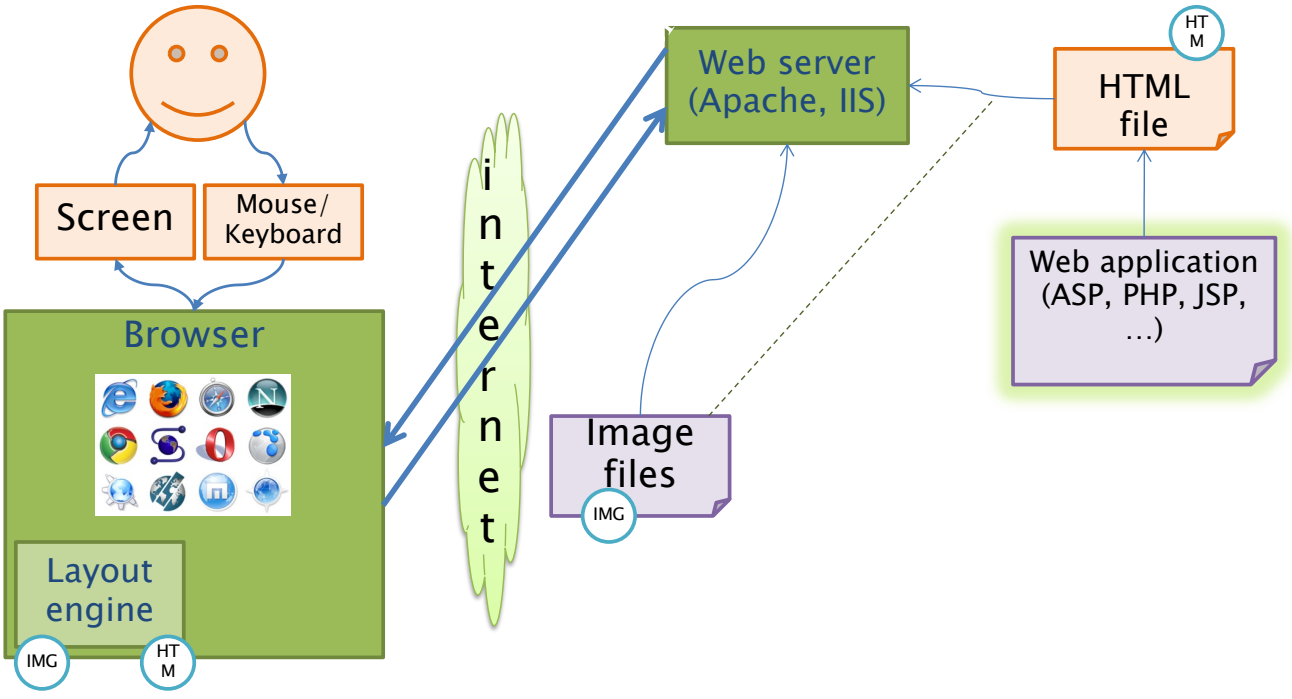
URL (HTTP GET)



Dynamic web transaction

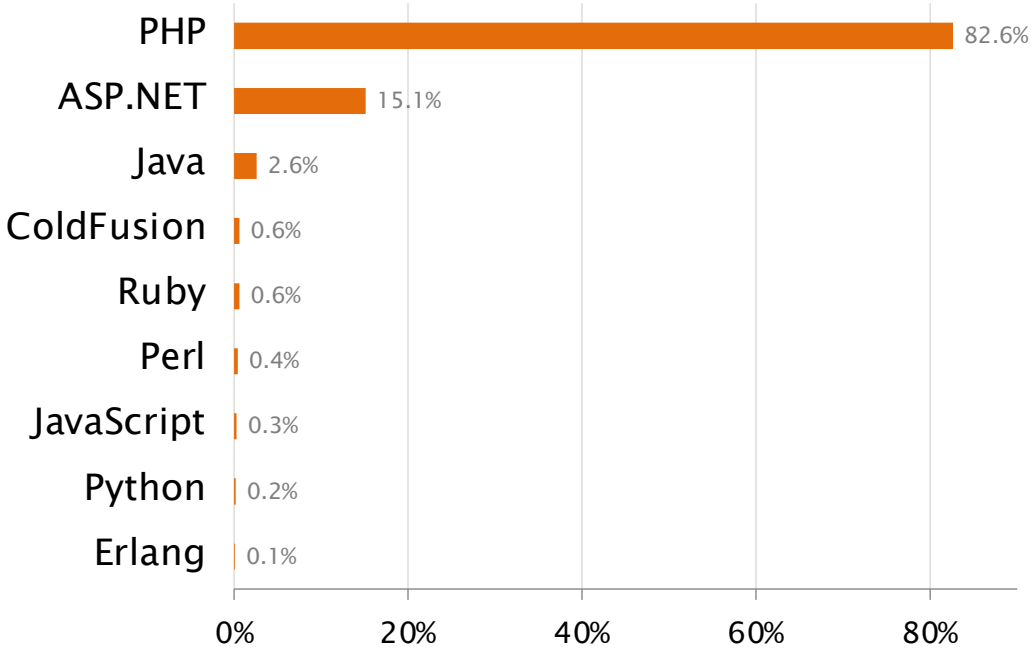


General web architecture



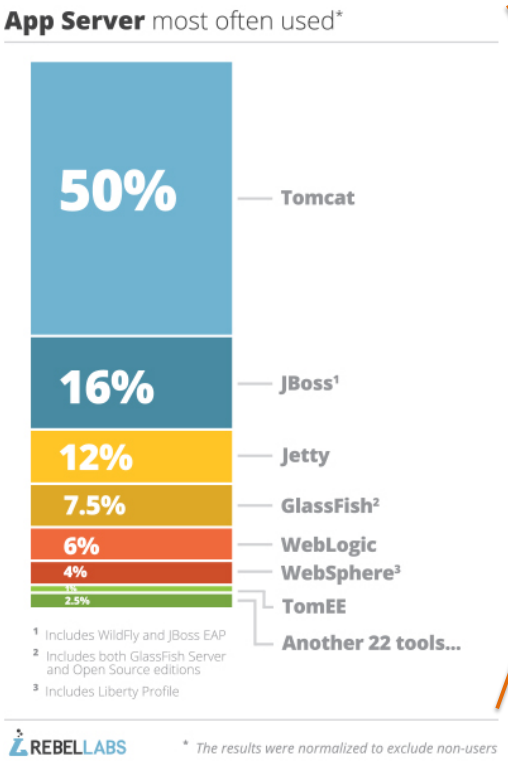
Application Servers

Percentages of websites using various server-side programming languages



https://w3techs.com/technologies/overview/programming_language/all

Application Servers



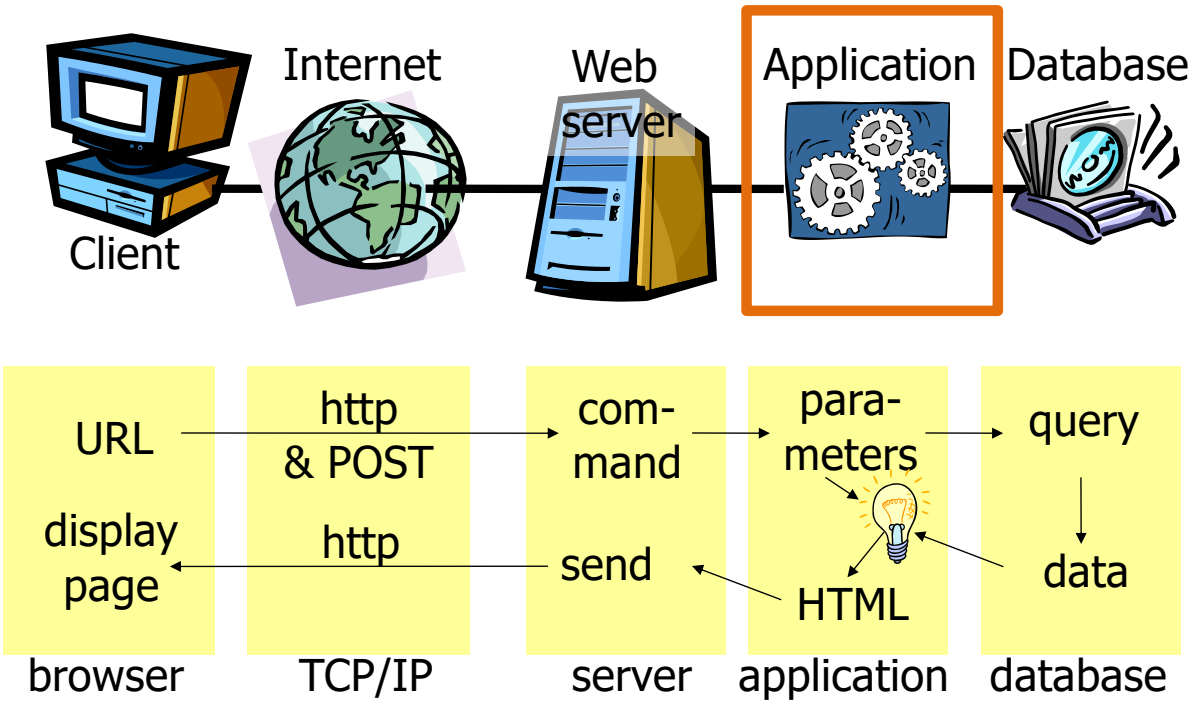
- Several different technologies
  - PHP
  - .Net
  - Java EE
  - Python
  - Ruby
  - Node.js

DATABASE SERVER

Database server

- Stores the data on which the application server works.
- Executes the queries issued by the application server:
  - Updates stored data
  - Inserts new data
  - Provides back query results
- The most frequent/complex queries can be implemented internally as stored procedures

Example



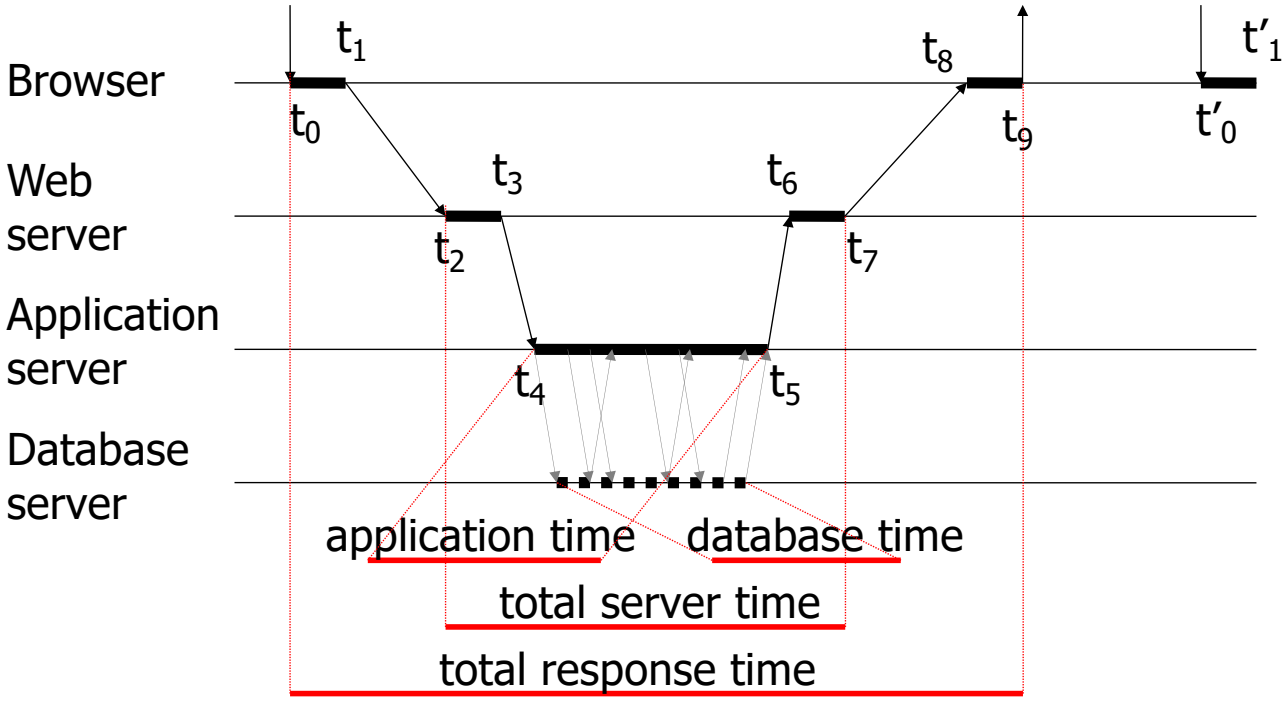
Adopted standards

- SQL (structured query language), ODBC (open database connectivity) to access data bases
- No-SQL for non-relational databases
  - SPARQL for triple-based knowledge bases

Database server

- Queries are often in SQL
  - **`SELECT * FROM table;`**
  - ....
- Adopts the relational database model
  - Other models can be used
    - Object model
    - Triple model
- The most advanced/complete solutions are called Transaction servers

Database-driven transaction



Example (PHP)

The application composes the query

```
<?php
$query = "SELECT doc_id FROM key_doc_index, keywords WHERE
key_doc_index.key_id = keywords.id AND keywords.key =
$_REQUEST["query"];";
```

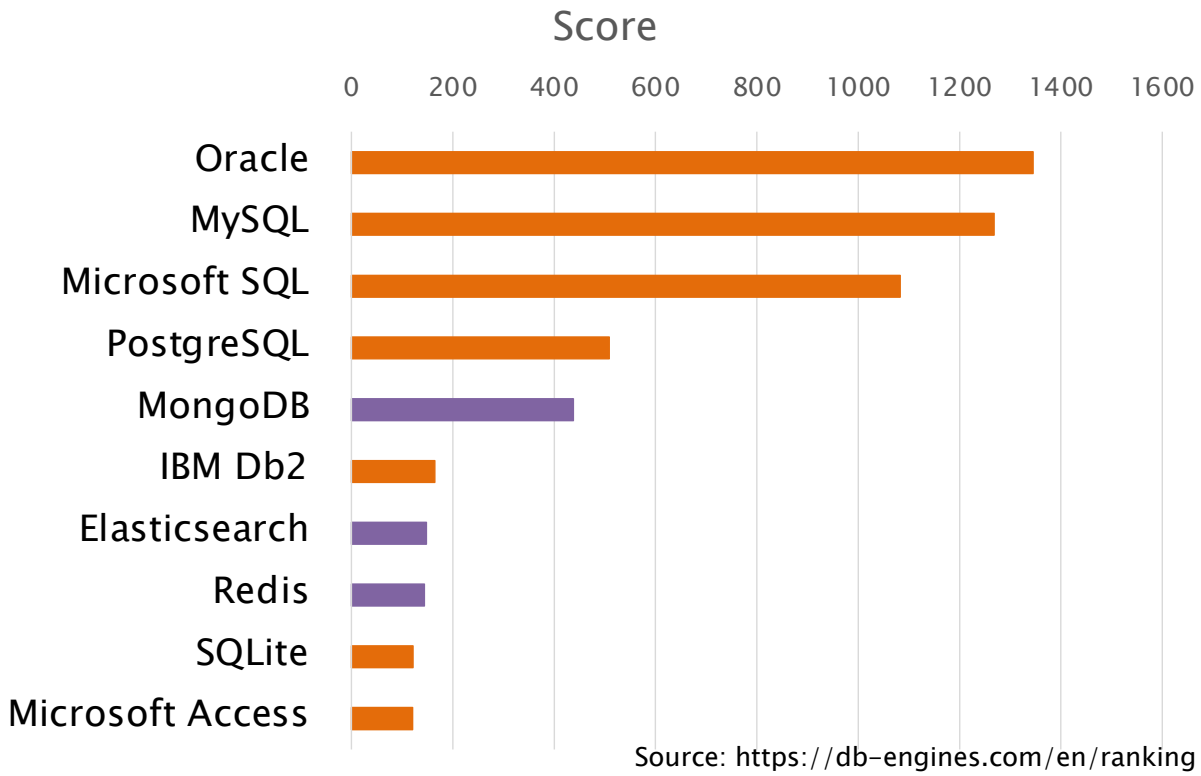
The query is sent to the db-server and a rowset containing the results is returned

```
$rowset = mysql_query($query);
```

```
while($row = mysql_fetch_row($rowset))
{
//elaborate data
}
?>
```

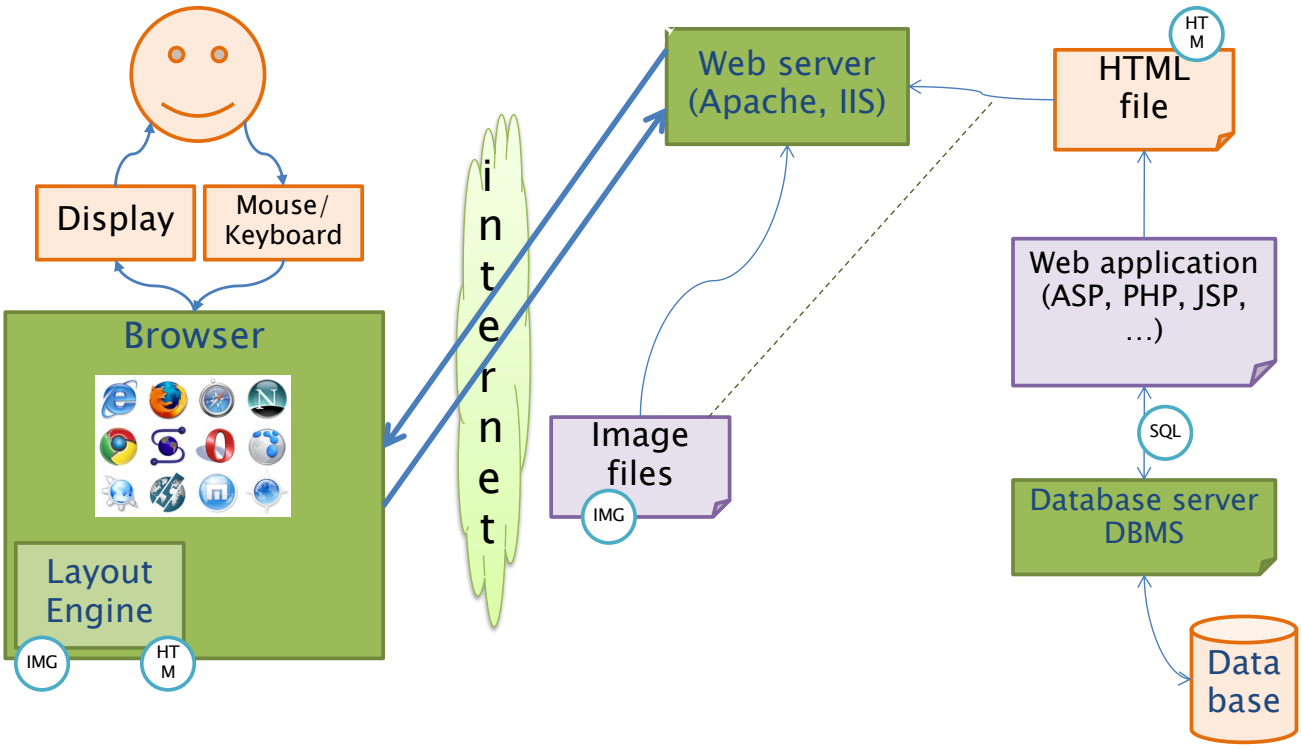
The application processes the data

DB Server diffusion



Source: https://db-engines.com/en/ranking

General web architecture



CLIENT-SIDE APPLICATIONS

Client-side programming

- Making a web page dynamic
  - Able to change the page content **after** it was loaded by the server
  - Able to interact with the user, on the browser
  - Able to "augment" the default interactions provided by the browser
- Examples:
  - Animations on the page
    - e.g. menus, accordions, slideshows, hide/show, …
  - Client-side form validation

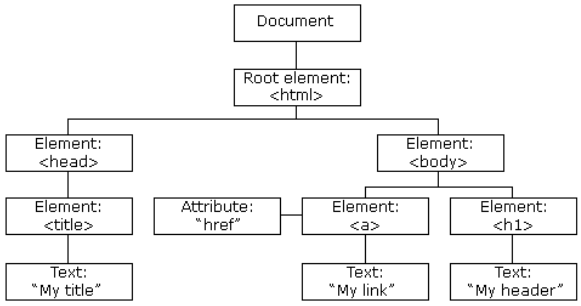
Client-side programming

- Requires:
  - A programming language accepted by all browsers
  - A program embedded in the web page
  - An execution engine in the browser
- Limitations:
  - All data needed by the program must be known beforehand (when the page is loaded)
  - The program must have a restricted access to the execution environment

Document Object Model (DOM)

- Standard data structure for representing the web page content
- Supported by all browsers
- Javascript programs can read & modify the DOM
- Abstracts
  - Browser
  - HTML
- The HTML DOM is a standard for how to get, change, add, or delete HTML elements

"The W3C Document Object Model (DOM) is a platform and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document."

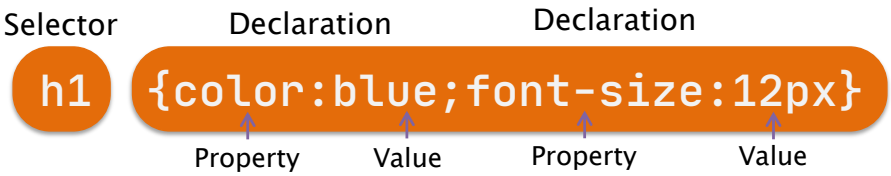


Separating layout from content

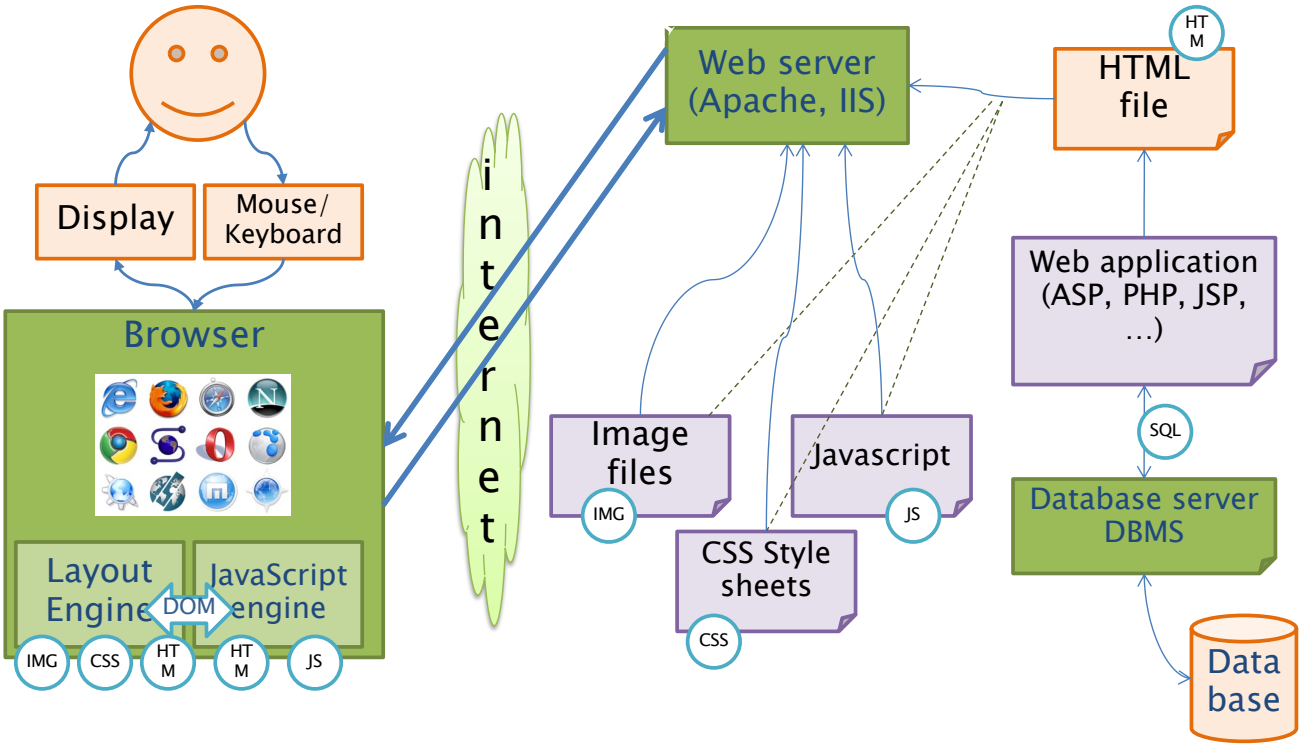
- Goals:
  - Allow the definition of complex layouts
  - Adapt web pages to different resolutions
  - Adapt web pages to different devices (e.g., mobile)
  - Adapt web pages to different preferences (e.g., color schemes)
  - Adapt web pages to different media (e.g., text vs video)
  - In a standard way
- 'Adapt' means:
  - Resize, Reflow, Show/Hide, Substitute, Animate, Highlight, Move, …
- Solution: Cascading Style Sheets (CSS)

CSS

- "Declarations" applied to some "Selector"
  - **Selectors** identify portions of the ~~web page~~ DOM
  - **Declarations** set the value of some "properties"
  - **Properties** control aspect, position, layout, behavior



General web architecture



Client-side, server-side, databases

Programming languages used in most popular websites*

Websites	Popularity (unique visitors per month)[1]	Front-end (Client-side)	Back-end (Server-side)	Database
Google.com[2]	1,600,000,000	JavaScript	C, C++, Go,[3] Java, Python	BigTable,[4] MariaDB[5]
Facebook.com	1,100,000,000	JavaScript	Hack, PHP (HHVM), Python, C++, Java, Erlang, D,[6] Xhp,[7] Haskell[8]	MariaDB, MySQL,[9] HBase Cassandra[10]
YouTube.com	1,100,000,000	JavaScript	C, C++, Python, Java,[11] Go[12]	BigTable, MariaDB[5][13]
Yahoo	750,000,000	JavaScript	PHP	MySQL, PostgreSQL[14]
Amazon.com	500,000,000	JavaScript	Java, C++, Perl[16]	Oracle Database[17]
Wikipedia.org	475,000,000	JavaScript	PHP, Hack	MySQL[*citation needed*], MariaDB[18]
Twitter.com	290,000,000	JavaScript	C++, Java, Scala, Ruby on Rails[19]	MySQL[20]
Bing	285,000,000	JavaScript	ASP.NET	Microsoft SQL Server
eBay.com	285,000,000	JavaScript	Java,[21] JavaScript[22]	Oracle Database
MSN.com	280,000,000	JavaScript	ASP.NET	Microsoft SQL Server
Microsoft	270,000,000	JavaScript	ASP.NET	Microsoft SQL Server
Linkedin.com	260,000,000	JavaScript	Java, JavaScript,[23] Scala	Voldemort[24]
Pinterest	250,000,000	JavaScript	Django (a Python framework),[25] Erlang	MySQL, Redis [26]
WordPress.com	240,000,000	JavaScript	PHP, JavaScript [27] (Node.js)	MariaDB, MySql

https://en.wikipedia.org/wiki/Programming_languages_used_in_most_popular_websites

AJAX

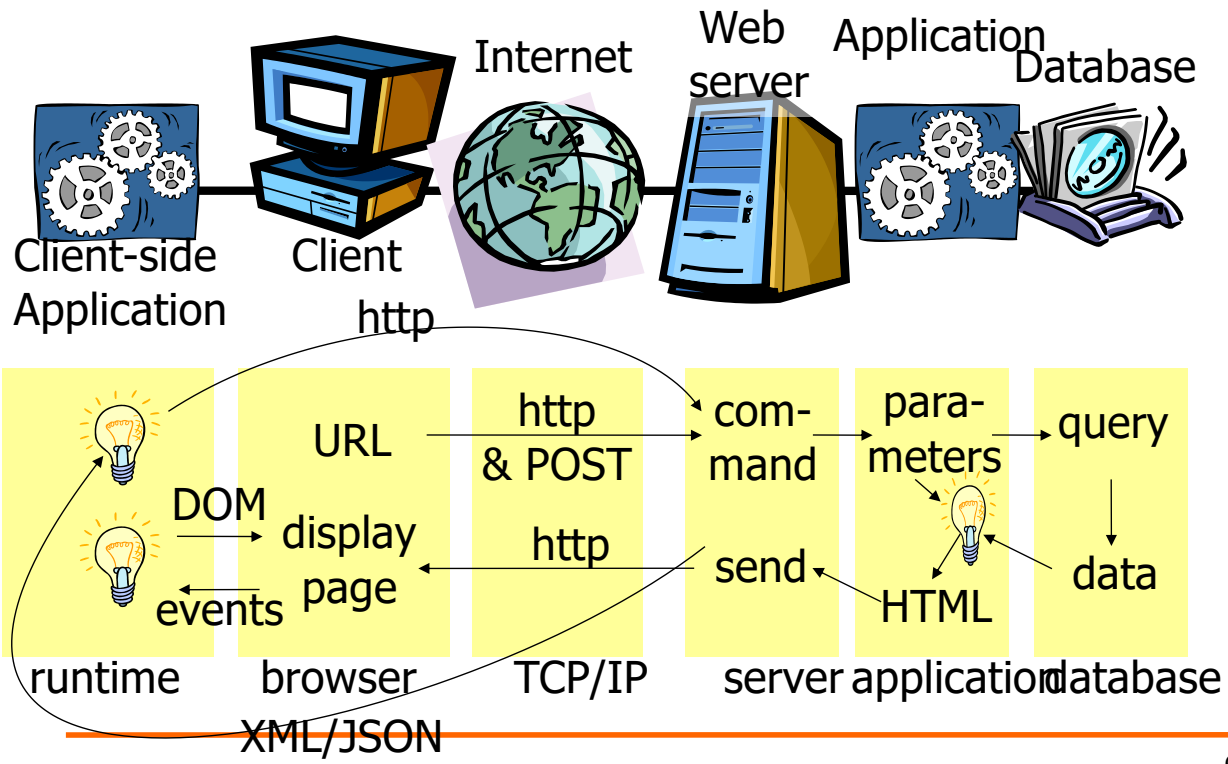
AJAX

- Asynchronous Javascript And Xml
- Describes an approach to design web applications where JS makes asynchronous requests to the server in order to update the web page
  - Does not have to use XML necessarily
- Modern AJAX applications use the REST approach to interact with servers

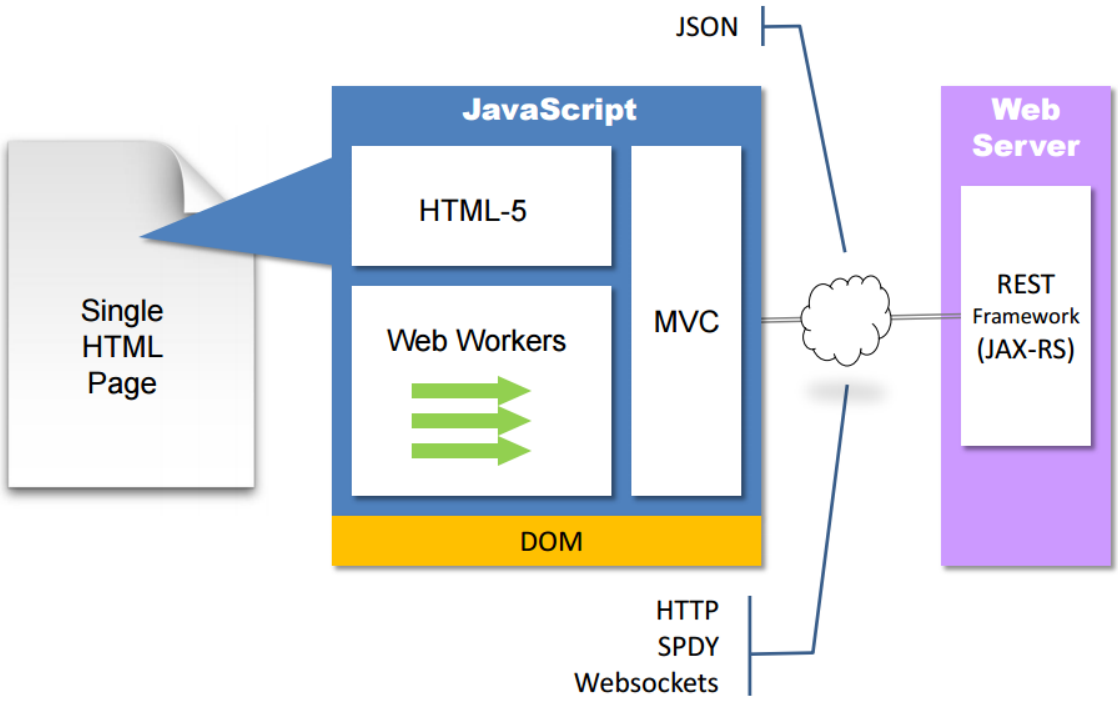
Web 2.0

- Web applications support social interaction models
- Peer exchange and user-contributed content instead of rigid publisher/reader pattern
  - Online communities
- Rich, dynamic, interactive user interfaces
- Integration of contents across web sites (mashups)

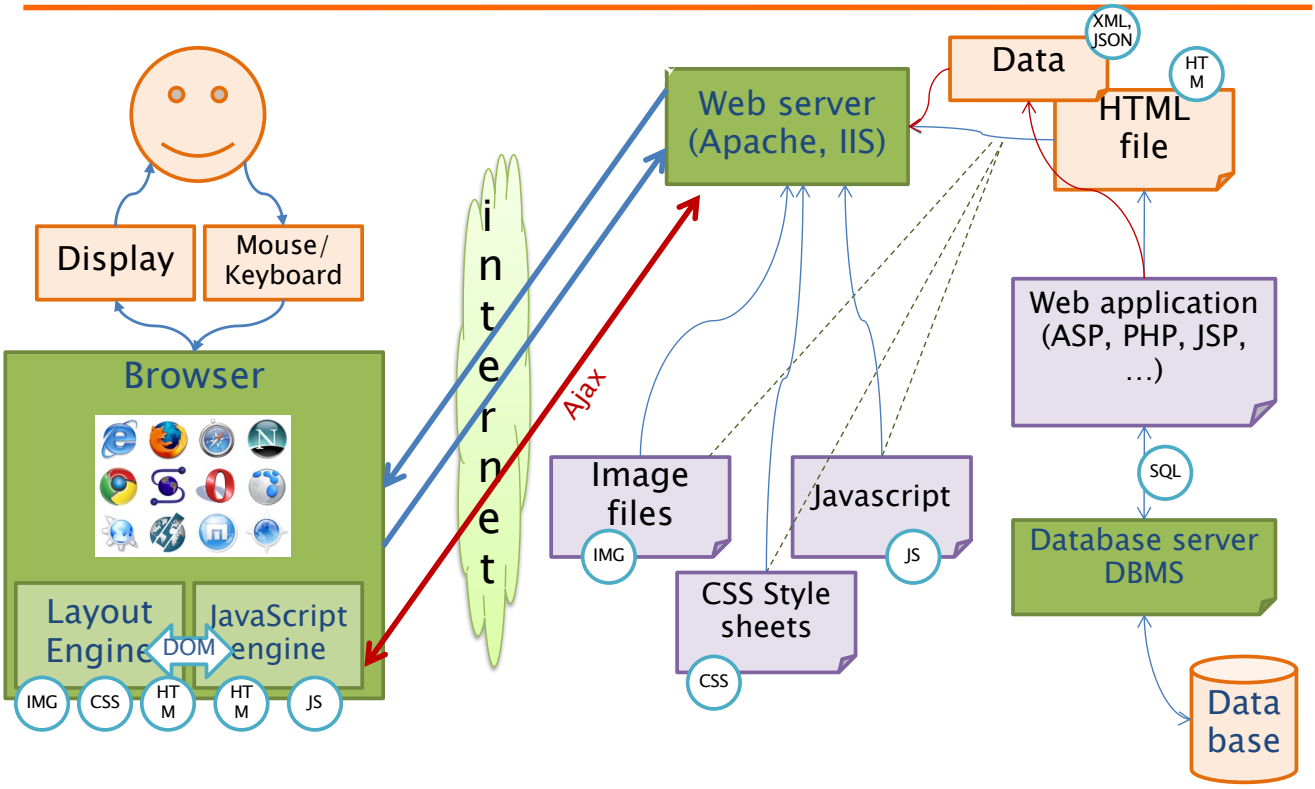
Rich-Client Asynchronous Transactions



Single Page Applications (SPA)



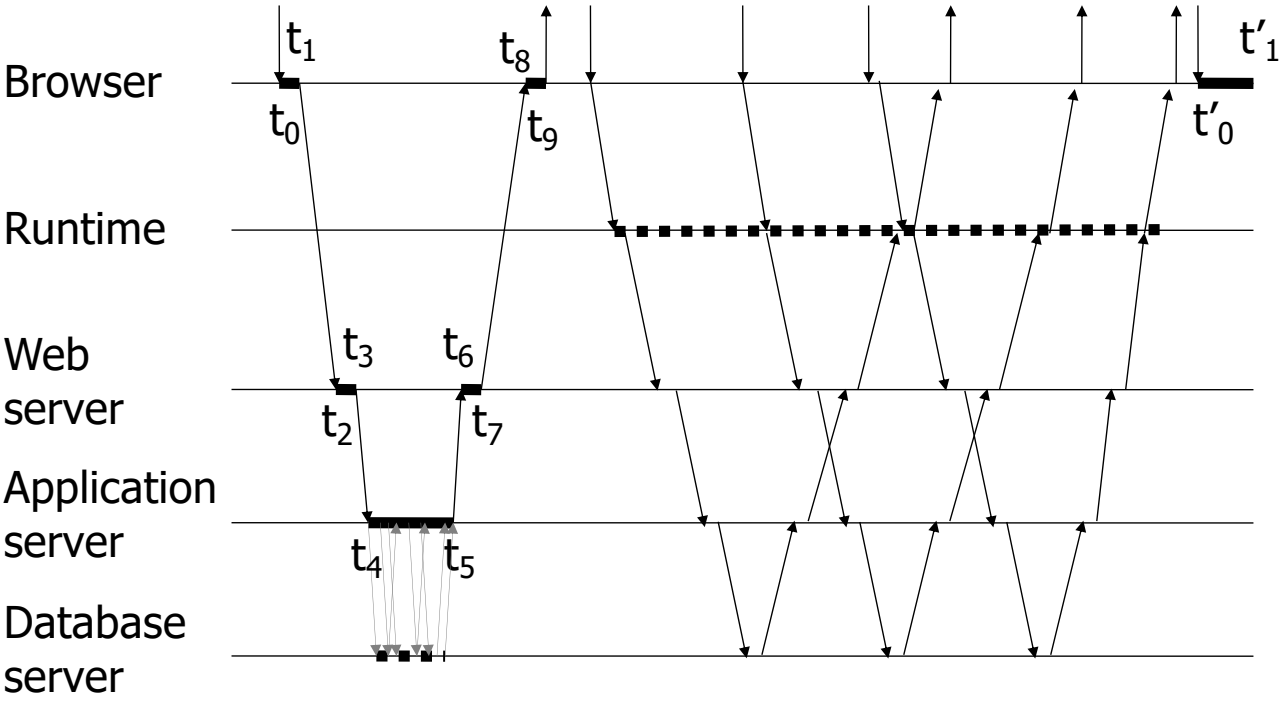
General web architecture



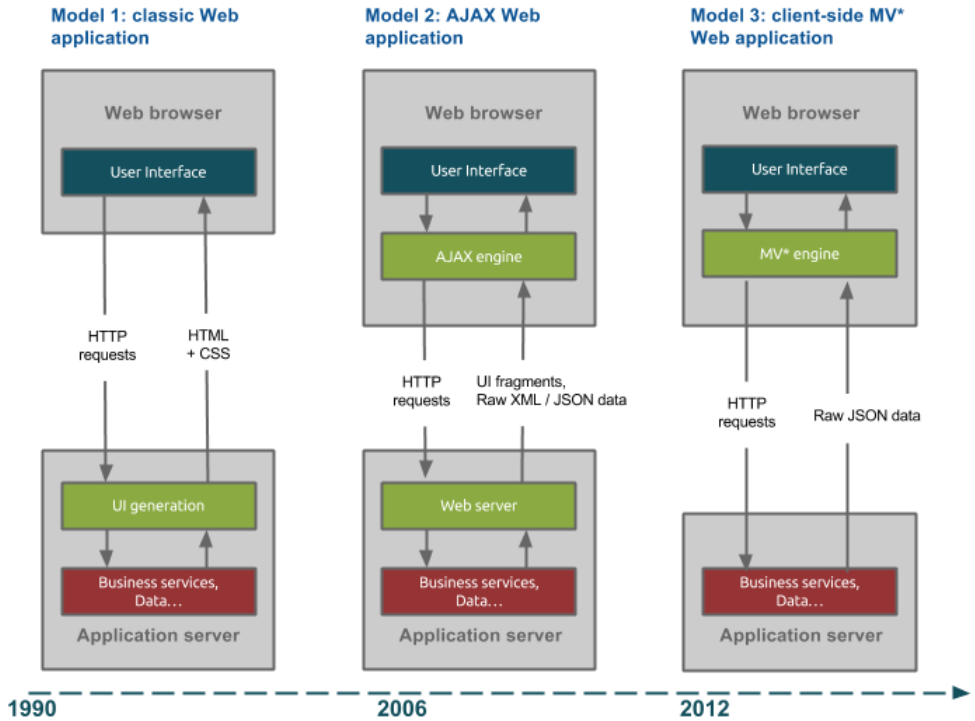
Adopted standards

- Dynamic HTML: DOM, Javascript, CSS
  - JavaScript, Flash to handle a runtime environment on the browser
  - DOM (XHTML Document Object Model) to allow on-the fly modification of the web page
  - CSS 2.1 to modify attribute and handle objects
- AJAX: Asynchronous Javascript and XML
  - XMLHttpRequest for asynchronous communication to the server
  - Data transfer formats: JSON, XML, RDF, RSS, Atom, FOAF, ...
- Mash-up technology

Rich-client transaction



Web application architectures



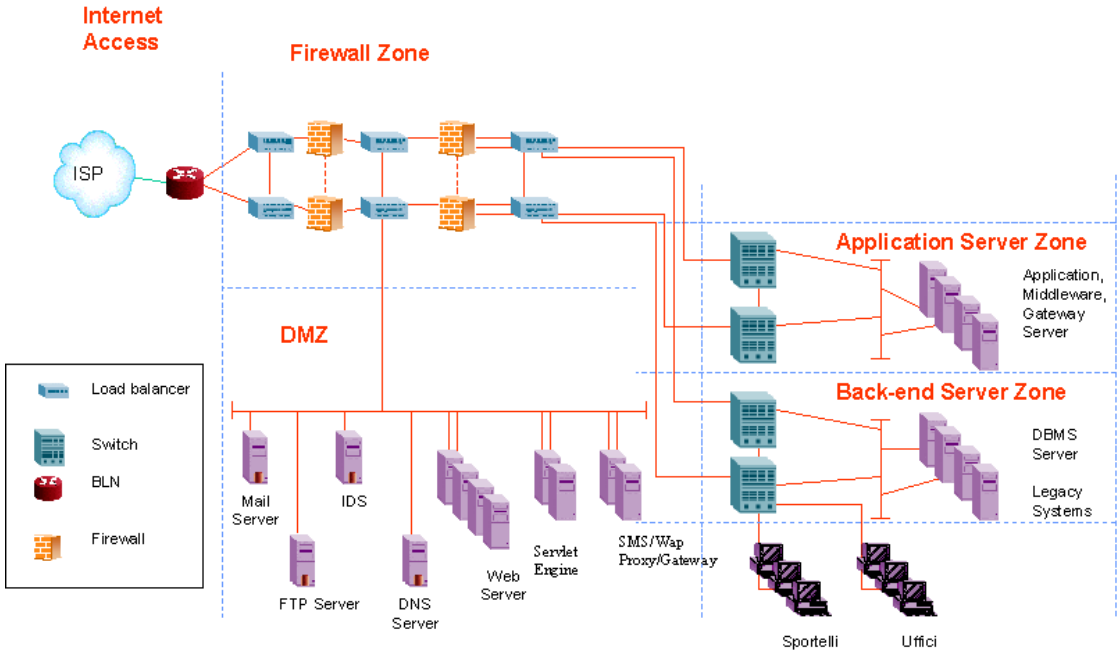
http://blog.octo.com/en/new-web-application-architectures-and-impacts-for-enterprises-1/

The real word is different...

- The users
- Functionality
- Flexibility
- Portability
- Reliability
- Security
- Integrity
- Maintenance
- Performance
- Scalability

- Costs
- Maintainance
- Development times
- Interactions with existing systems
- Interactions with the “physical” world

Ordering site – example



References

- Aa.Vv. "Come Funzione Internet" EDRi Papers, Centro Nexa
  - http://nexa.polito.it/nexafiles/ComeFunzionaInternet.pdf
- RFC 2396 - Uniform Resource Identifiers (URI): Generic Syntax
  - https://www.ietf.org/rfc/rfc2396.txt
- RFC 2616 - Hypertext Transfer Protocol – HTTP/1.1
  - https://www.w3.org/Protocols/rfc2616/rfc2616.html