# Web Architecture
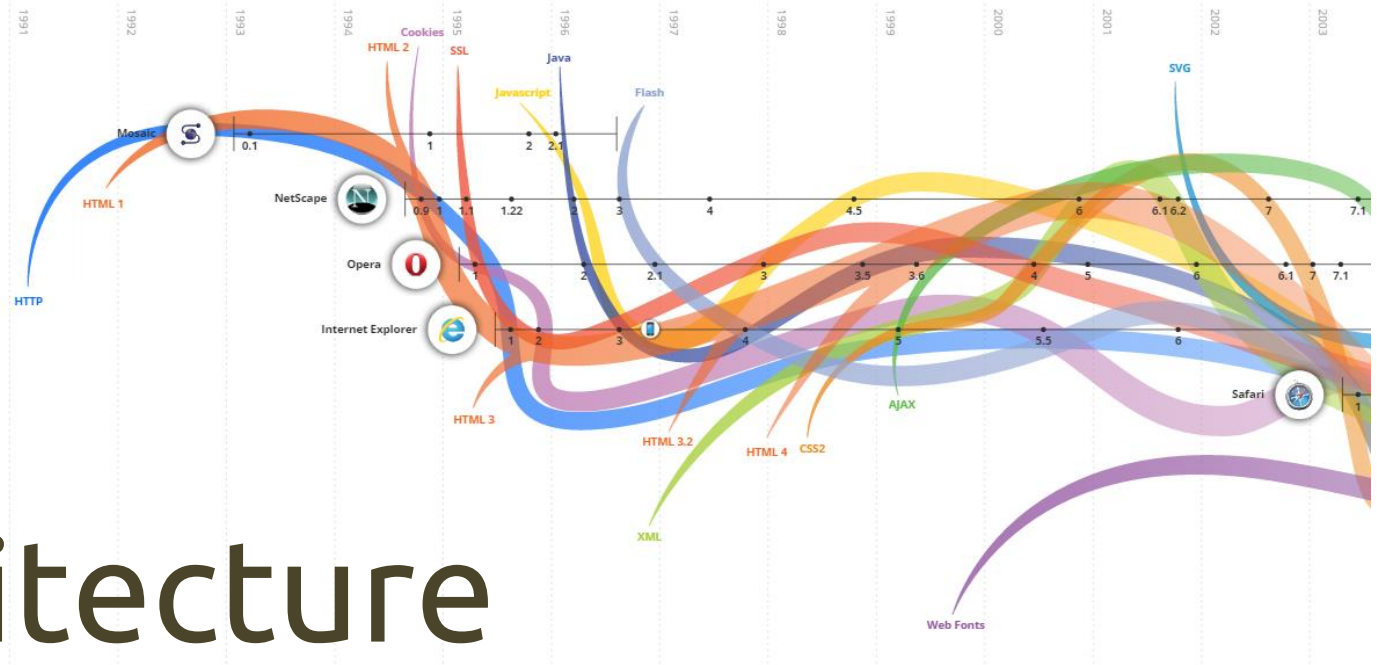
## AN INTRODUCTION

Quick overview to the Web architecture and technologies

e-Lite

# General Architecture

Client

Server(s)

**Internet**

Typically, a web browser.
It may also be:
- mobile application
- desktop application
- other server application
- …

**Logical server**
a process that runs on a host that relays information to a client upon requests

**Physical server**
a host computer on a network that holds information and respond to requests for those information

# General Architecture

- What types of information do clients and servers exchange?

- How can these information be found?

- How are these information sent?

# General Architecture

- What types of information do clients and servers exchange?
    - HTML (Hyper Text Markup Language) files
    - Multimedia files
    - CSS, JavaScript, … files
- How can these information be found?
    - URL (Uniform Resource Locator)
- How are these information sent?
    - With HTTP (Hyper Text Transfer Protocol) requests and responses
    - TCP/IP is used for data transfer

# Getting started with HTML...



https://developer.mozilla.org/docs/Learn/HTML/Introduction_to_HTML

# URL: Example

`http://elite.polito.it/courses/01qzp-ami`

DNS

130.192.27.201

TCP

Contact server

URI Rewriting

/var/www/html/elite/courses/01qzp-ami

File system

HTML file contents
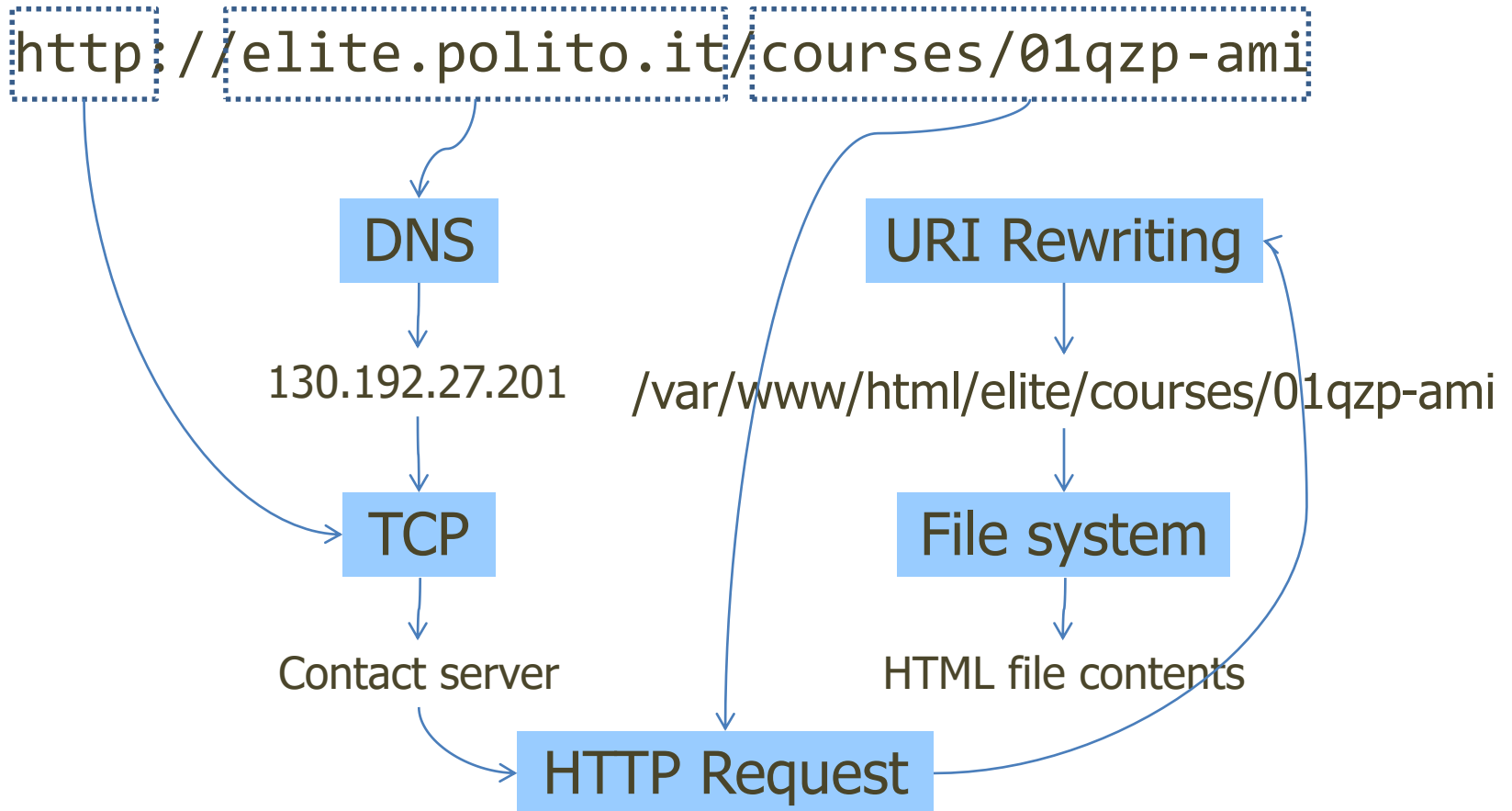
HTTP Request

# URL Basics

- **Scheme** **Hostname** **Query**
  http://www.sadev.co.za/users/1/contact

- **Scheme** **Hostname** **Query**
  http://www.sadev.co.za?user=1&action=contact

- **Scheme** **Userinfo** **Hostname** **Port**
  http://rob:pass@bbd.co.za:8044

- **Scheme** **Hostname** **Query** **Fragment**
  https://bbd.co.za/index.html#about

# Web server (logical)

- Manages the HTTP protocol (handles requests and provides responses)
  - Receives client requests
  - Reads static pages/contents from the filesystem
  - Activates the application server for dynamic pages/content generation (server-side)
  - Provides an file (HTML, or other) back to the client
- One HTTP connection for each request
- Multi-process, Multi-threaded or Process pool

# Example

Web Server

Client

**Internet**

| browser | TCP/IP | server | file system |
|---------|--------|--------|-------------|
| URL | HTTP request →| path | |
| | | | HTML |
| display page | ← HTTP response | send | |
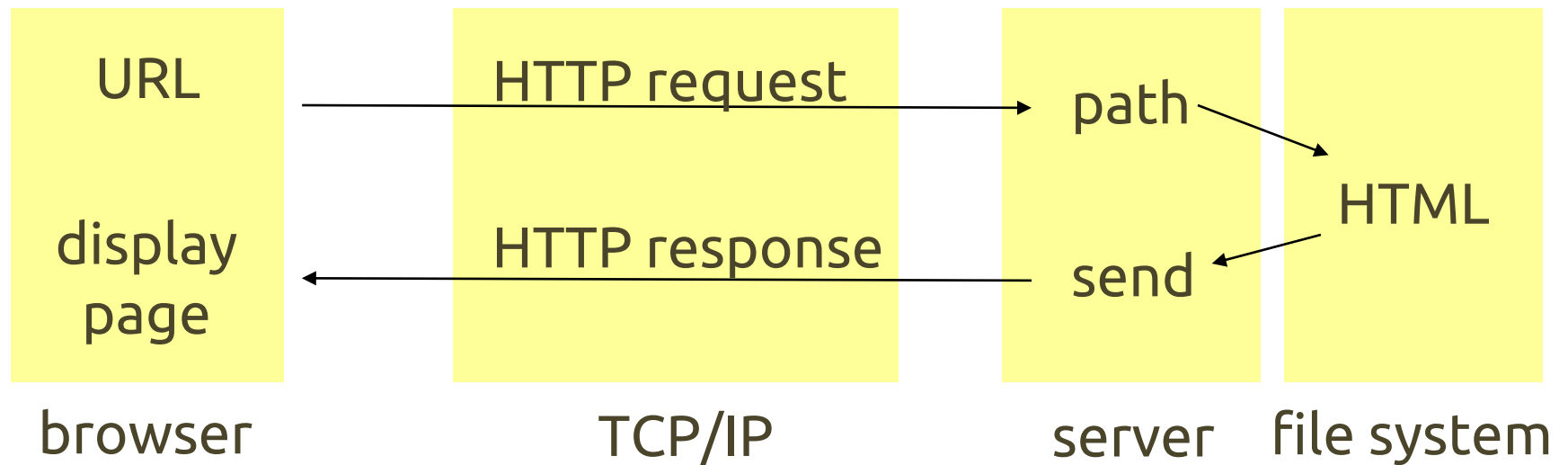
# HTTP protocol

```
GET / HTTP/1.1
Host: elite.polito.it
User-Agent: Mozilla/5.0
Accept: text/html,appl
Accept-Language: it-I
Accept-Encoding: gzi
Cookie: __utma=1885
Connection: keep-a
```

```
HTTP/1.0 200 OK
Cache-Control: no-store, no-cache, must-revalidate,
Connection: Keep-Alive
Content-Encoding: gzip
Content-Type: text/html; charset=utf-8
Date: Wed, 08 Apr 2016 13:36:24 GMT
Expires: Mon, 1 Jan 2020 00:00:00 GMT
Keep-Alive: timeout=15, max=100
Last-Modified: Wed, 08 Apr 2016 13:36:24 GMT
Pragma: no-cache
Server: Apache/2.4.6 (Linux/SUSE)
Transfer-Encoding: chunked
X-Powered-By: PHP/5.6.30
p3p: CP="NOI ADM DEV PSAi COM NAV OUR OTRo STP IND DEM«

<!DOCTYPE html>
<html>
<head>
. . . . . .
```

# Browser developer tools

Web Architecture

# 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
- **#Requests / s**

# Dynamic Web Transaction

Client

Web Server

Internet

| URL and data | HTTP request with data | com mand | parameters |
| display page | HTTP response | send | logic |
| | | | HTML |

browser          TCP/IP          server    application
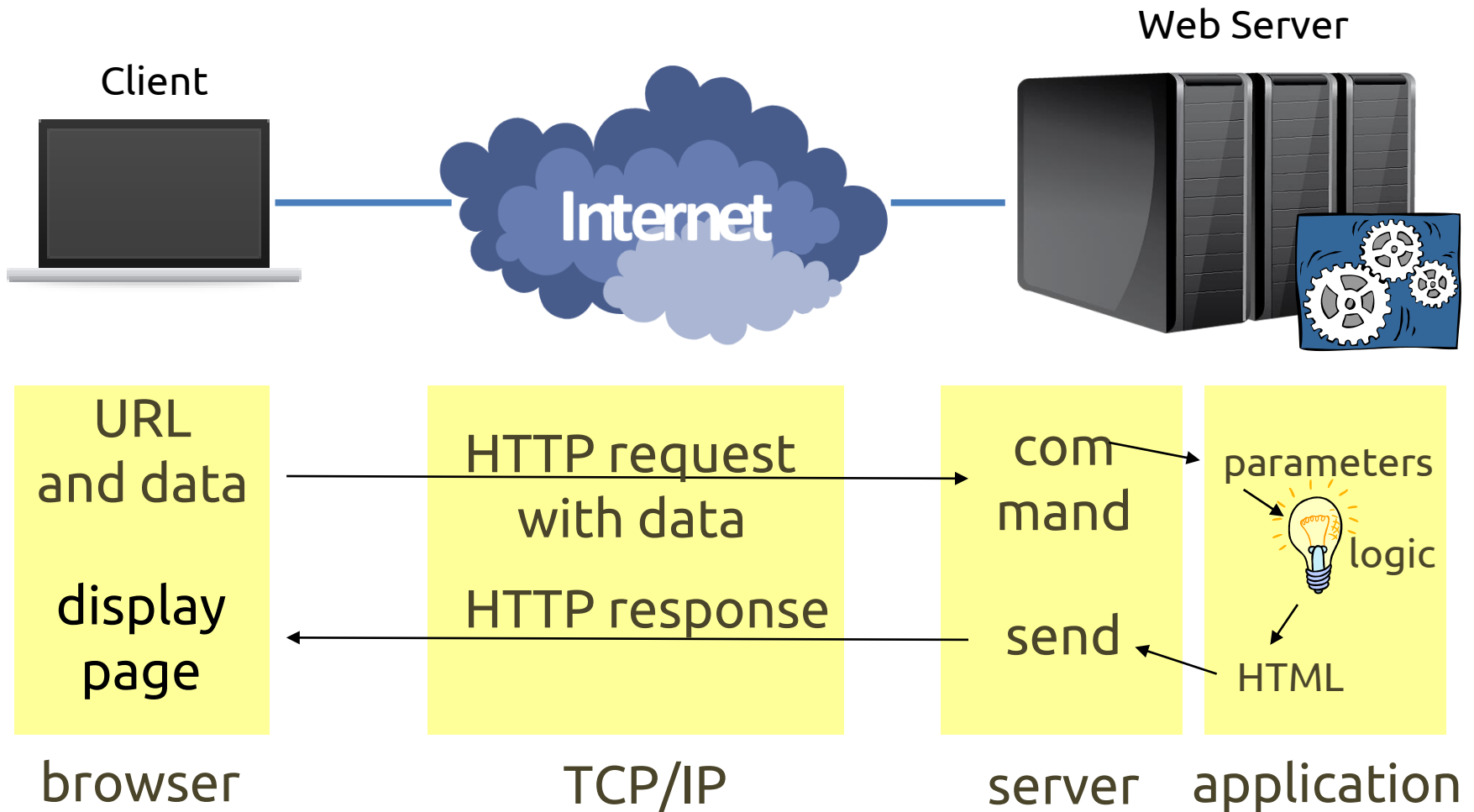
# Application server

- Dynamic page generation and content generation

- Manages the site business logic

- It is the middle tier between the client browser and the data residing on a database

- Implements the session mechanisms

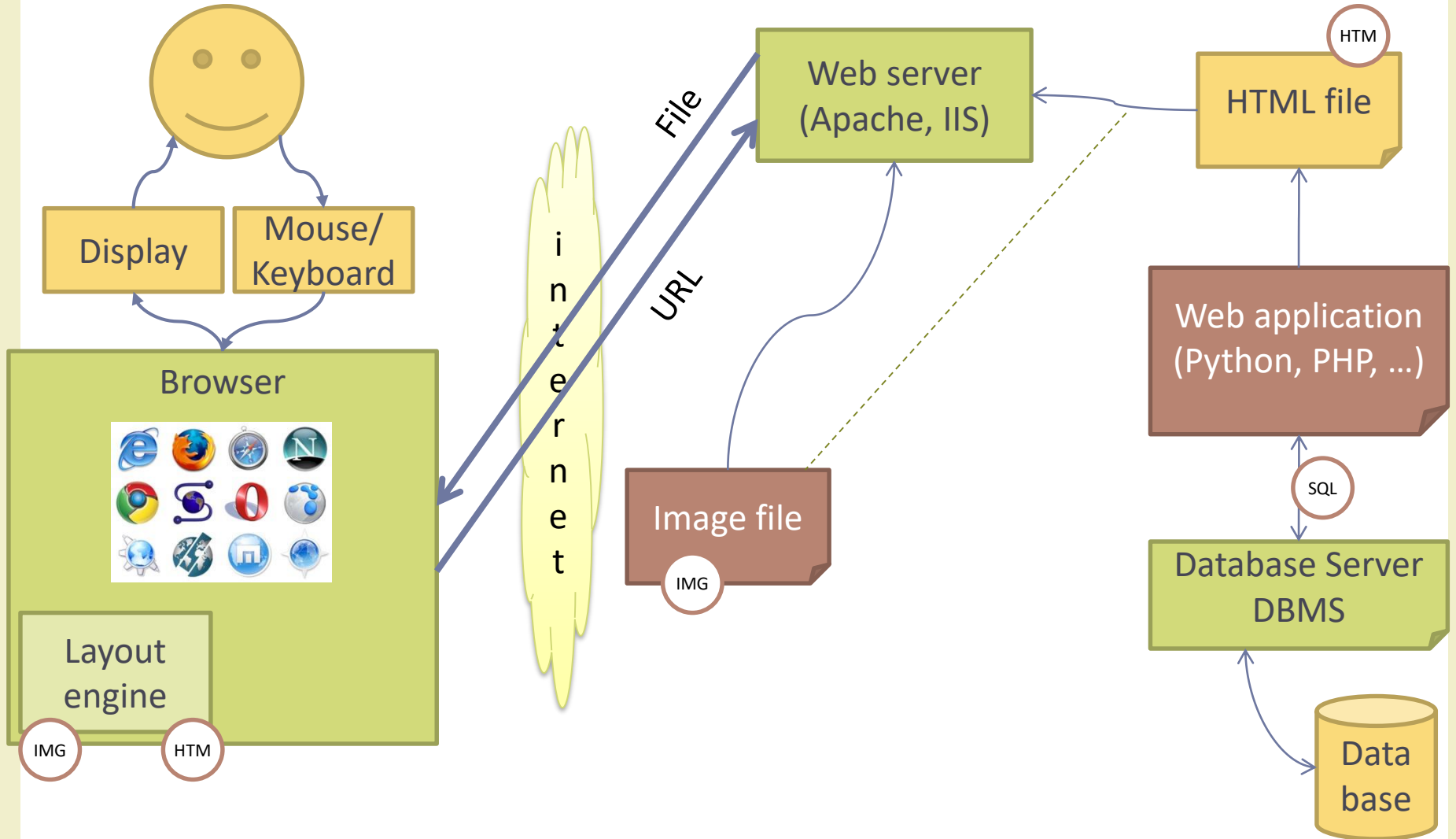- Different technologies and architectures are available
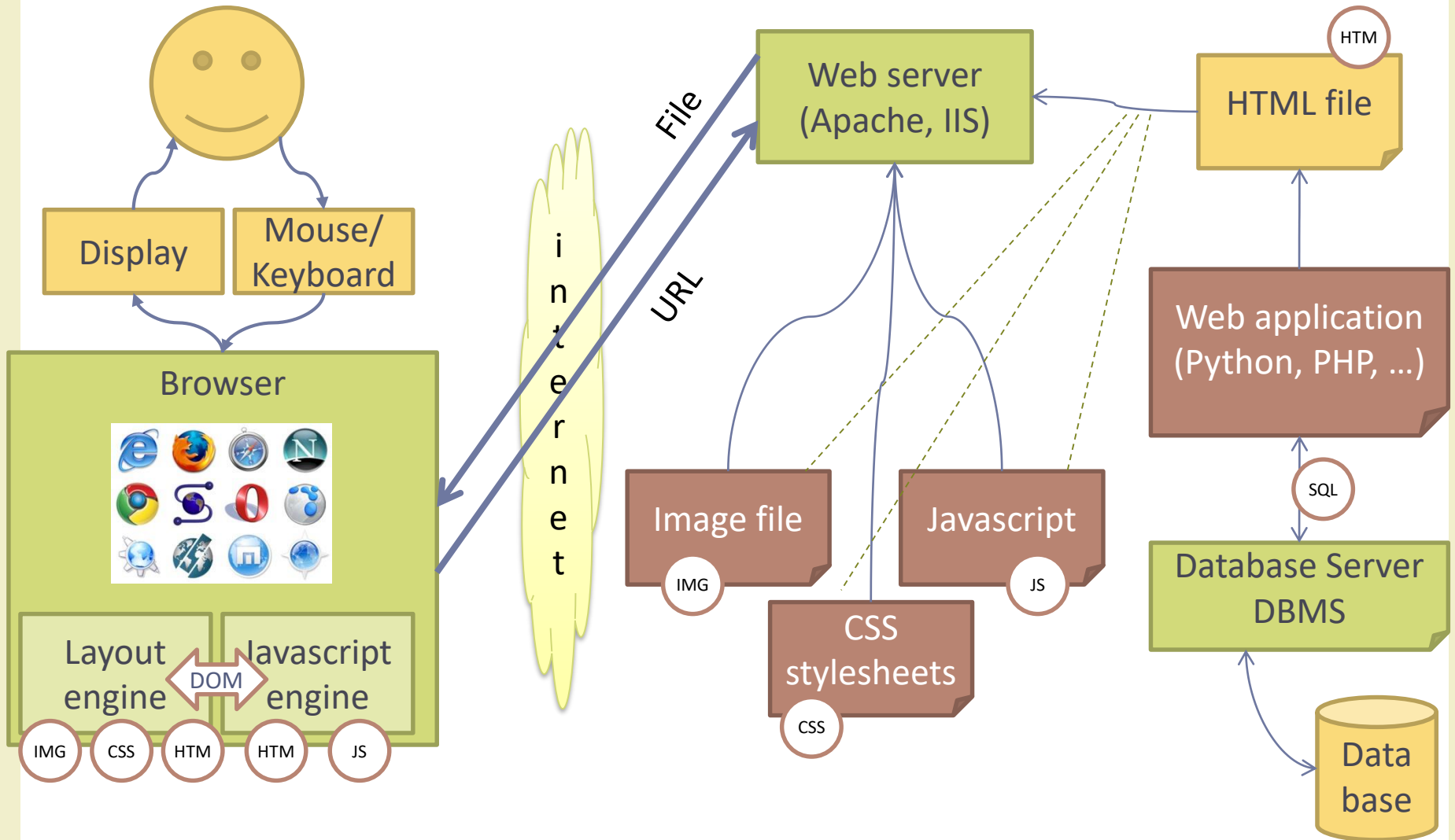
# General Architecture
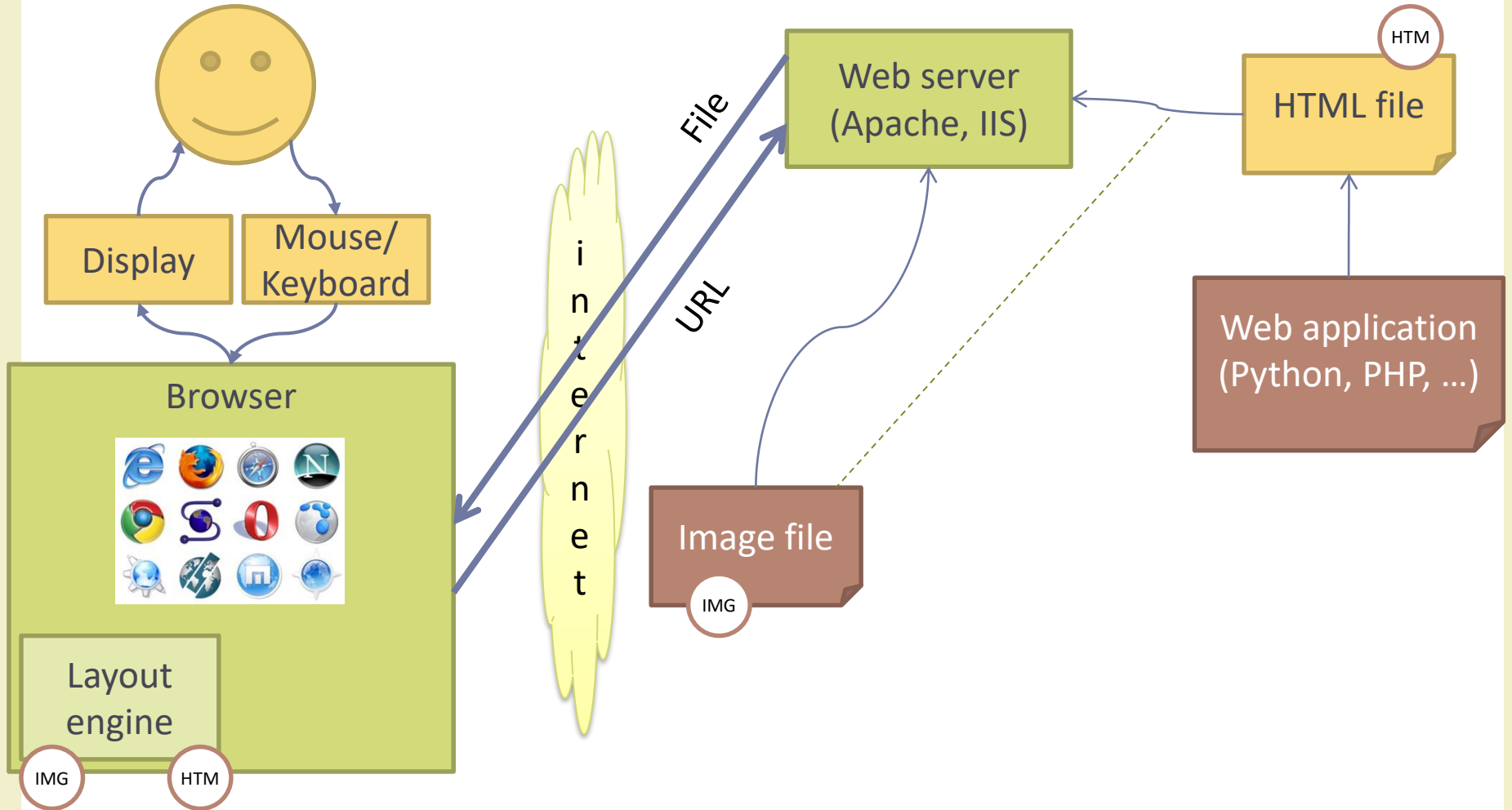
# Database server

- Stores the data on which the application server works.

- Executes the queries issued by the application server:
  - Updates the stored data
  - Inserts new data
  - Provides back query results

- The most frequent/complex queries can be implemented internally as stored procedures (pre-compiled queries with parameters)

# General Architecture



Display

Mouse/ Keyboard

Browser

Layout engine

IMG    HTM

internet

File

URL

Image file

IMG

Web server (Apache, IIS)

HTML file

HTM

Web application (Python, PHP, …)

SQL

Database Server DBMS

Data base

# General Architecture

# Why is it so "complex"?

- The Web is continuously evolving
  - it incorporates (= adds) different technologies and tools
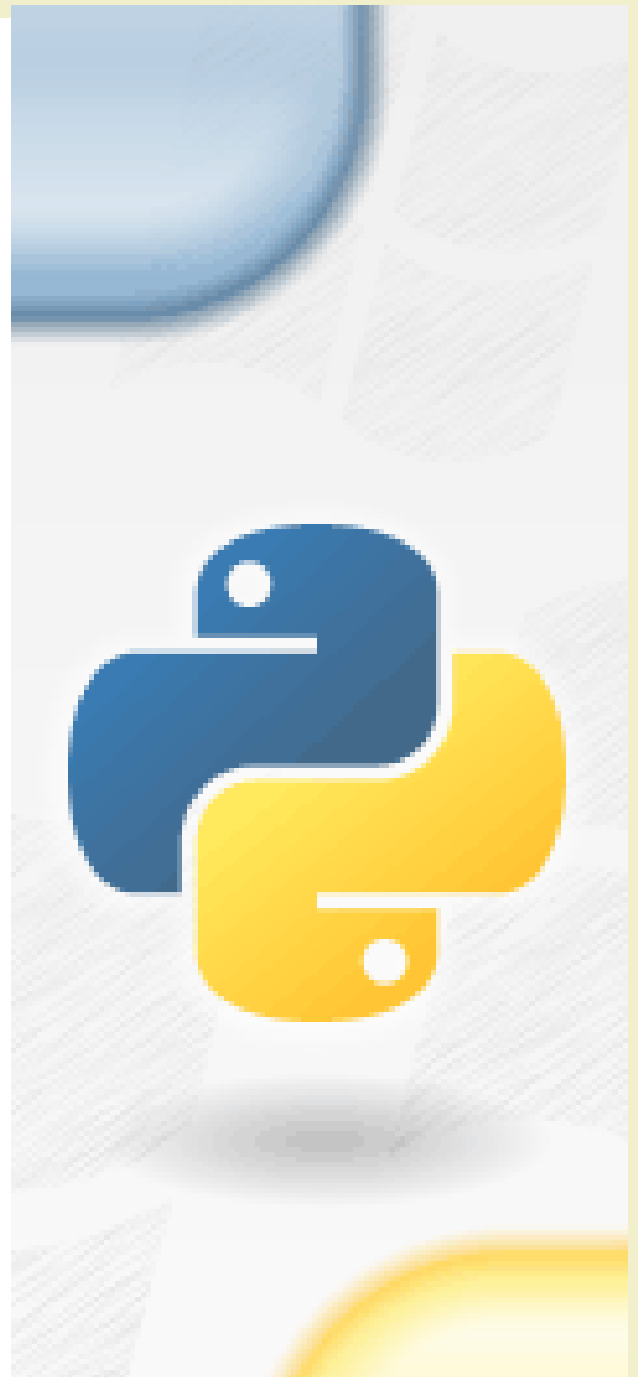- To solve problems and issues that arise among time



`http://www.evolutionoftheweb.com/`

# Questions?

**01QZP AMBIENT INTELLIGENCE**

Luigi De Russis and Fulvio Corno

luigi.derussis@polito.it

fulvio.corno@polito.it

# License

- This work is licensed under the Creative Commons "Attribution-NonCommercial-ShareAlike Unported (CC BY-NC-SA 4.0)" License.
- You are free:
  - to **Share** - to copy, distribute and transmit the work
  - to **Remix** - to adapt the work
- Under the following conditions:
  - **Attribution** - You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
  - **Noncommercial** - You may not use this work for commercial purposes.
  - **Share Alike** - If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.
- To view a copy of this license, visit https://creativecommons.org/licenses/by-nc-sa/4.0/