

# Introduction to the course



# Prerequisites

- OOP + Java: **essential!** (Must have passed Linguaggi di Programmazione!)
- Network fundamentals
- Database
- No need to already know HTML/HTTP etc.

# Topics

- HTTP + HTML + CSS
- Server side programming: PHP – Java, parameter passing, state problem...
- Data access (DB)
- Javascript – EcmaScript – Typescript
- Patterns: MV\* + ...
- AJAX/XML/JSON
- JS frameworks/libraries: JQuery, (brief remarks on React, Angular...)
- General considerations (accessibility, security...)

# We'll use Piazza



# Exam

- Similar to “Linguaggi di Programmazione – mod.1”
- 1<sup>st</sup> part: output prediction of 8 brief code fragments + 8 true/false questions (40 minutes)
- 2<sup>nd</sup> part: small project development (4 hours)



# What is the difference between the Web and the Internet?

# What is the difference between the Web and the Internet?

- "The Internet is a global system of interconnected computer networks that interchange data by packet switching using the standardized Internet Protocol Suite (TCP/IP)."
- Thus, the Internet is a network of networks, defined by the TPC/IP standards (such as FTP, Telnet, SMTP...).



# What is the web

The Wold Wide Web is an **information system** where documents and other web resources are identified by **Uniform Resource Locators** (URLs, such as `https://www.example.com/`), which may be interlinked by **hypertext**, and are accessible over the Internet.

The resources of the WWW are

- described via the **Hypertext Markup Language** (HTML)
- transferred via the **Hypertext Transfer Protocol** (HTTP)
- accessed by users by a software app called a **web browser**
- published by a software app called a **web server**.

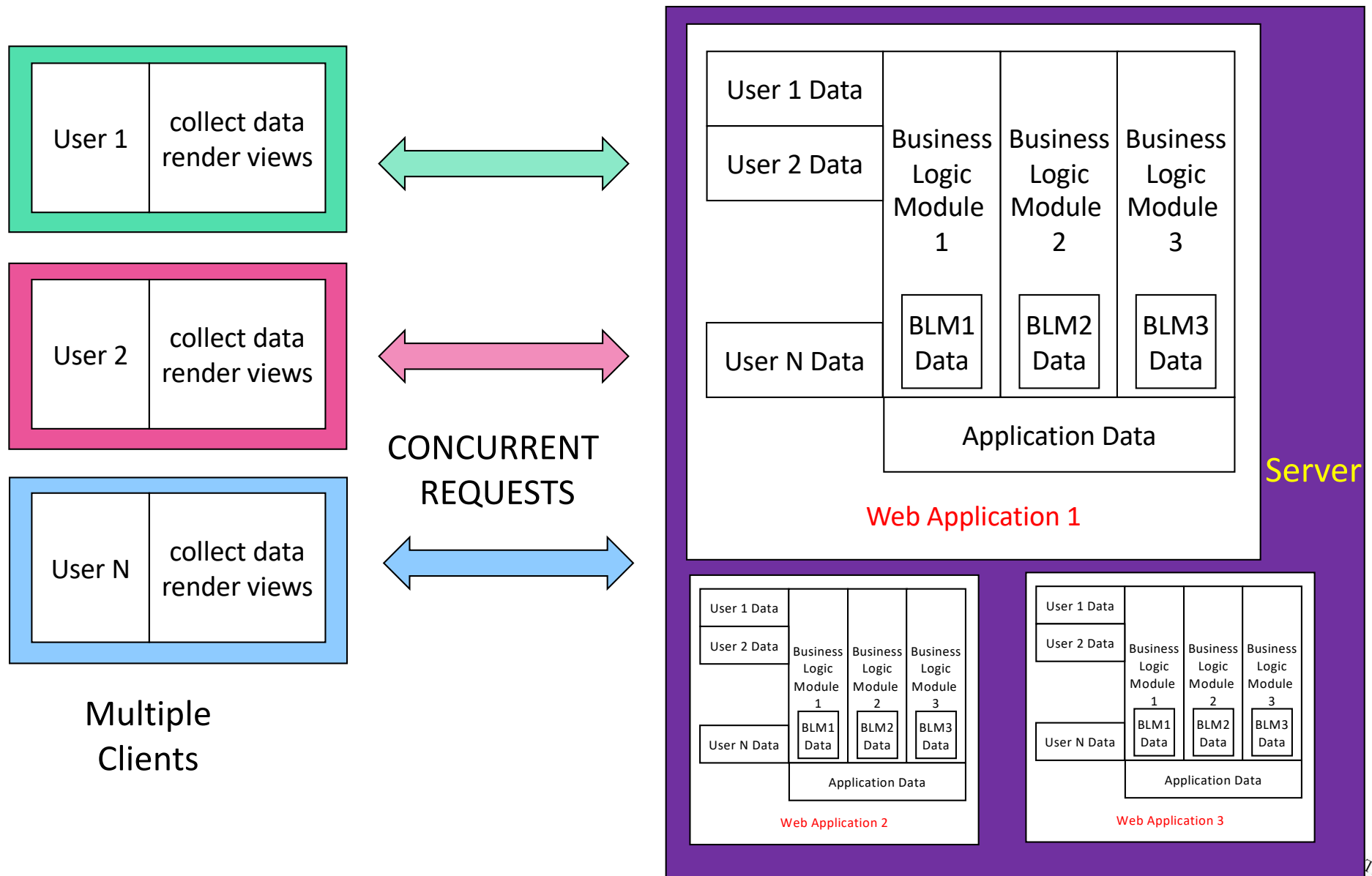
# History of the web

- 1989-1990 – **Tim Berners-Lee** invents World Wide Web at CERN.



- On 30 April 1993, CERN put the World Wide Web software in the public domain. Later, CERN made a release available with an open license, a more sure way to maximize its dissemination.
- TB-L moved from CERN to the Massachusetts Institute of Technology. in 1994 to found the **World Wide Web Consortium (W3C)**, an international community devoted to developing open web standards.

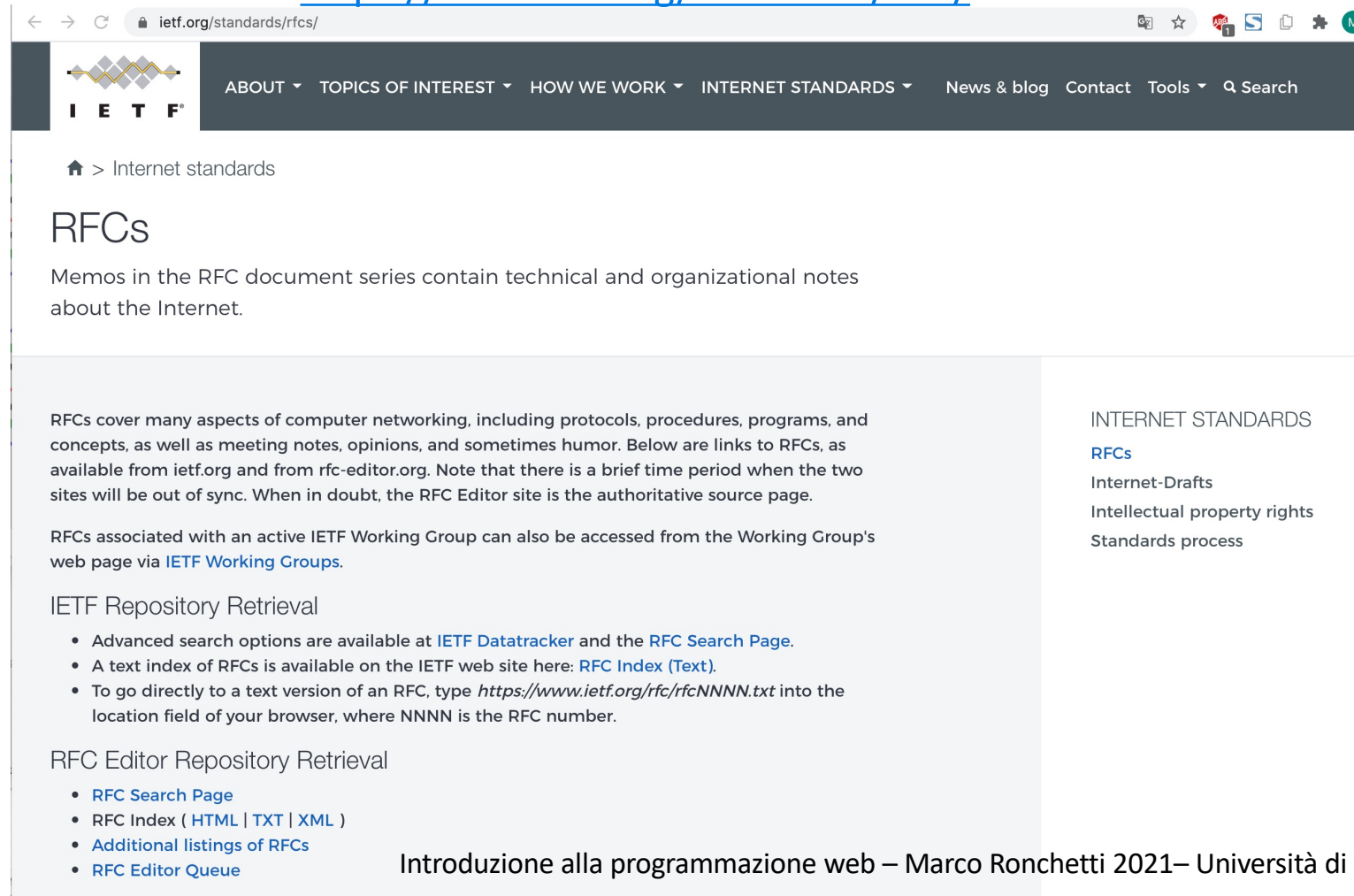
# Why is web programming complex?



# Important references: IETF

The **Internet Engineering Task Force** (IETF) is a large open international community of network designers, operators, vendors, and researchers concerned with **the evolution of the Internet architecture and the smooth operation of the Internet.**

<https://www.ietf.org/standards/rfcs/>



The screenshot shows the IETF RFCs page. At the top is a navigation bar with the IETF logo and links for ABOUT, TOPICS OF INTEREST, HOW WE WORK, INTERNET STANDARDS, News & blog, Contact, Tools, and a search bar. Below the navigation bar is a breadcrumb trail: Home > Internet standards. The main heading is "RFCs", followed by a paragraph: "Memos in the RFC document series contain technical and organizational notes about the Internet." Below this is a large text block explaining that RFCs cover many aspects of computer networking, including protocols, procedures, programs, and concepts, as well as meeting notes, opinions, and sometimes humor. It also mentions that RFCs associated with an active IETF Working Group can be accessed from the Working Group's web page via IETF Working Groups. To the right of this text is a sidebar titled "INTERNET STANDARDS" with links for RFCs, Internet-Drafts, Intellectual property rights, and Standards process. Below the main text block are two sections: "IETF Repository Retrieval" and "RFC Editor Repository Retrieval", each with a list of links for advanced search, text index, and direct access to RFCs.

Internet standards

## RFCs

Memos in the RFC document series contain technical and organizational notes about the Internet.

RFCs cover many aspects of computer networking, including protocols, procedures, programs, and concepts, as well as meeting notes, opinions, and sometimes humor. Below are links to RFCs, as available from [ietf.org](http://ietf.org) and from [rfc-editor.org](http://rfc-editor.org). Note that there is a brief time period when the two sites will be out of sync. When in doubt, the RFC Editor site is the authoritative source page.

RFCs associated with an active IETF Working Group can also be accessed from the Working Group's web page via [IETF Working Groups](#).

### IETF Repository Retrieval

- Advanced search options are available at [IETF Datatracker](#) and the [RFC Search Page](#).
- A text index of RFCs is available on the IETF web site here: [RFC Index \(Text\)](#).
- To go directly to a text version of an RFC, type <https://www.ietf.org/rfc/rfcNNNN.txt> into the location field of your browser, where NNNN is the RFC number.

### RFC Editor Repository Retrieval

- [RFC Search Page](#)
- RFC Index ( [HTML](#) | [TXT](#) | [XML](#) )
- [Additional listings of RFCs](#)
- [RFC Editor Queue](#)

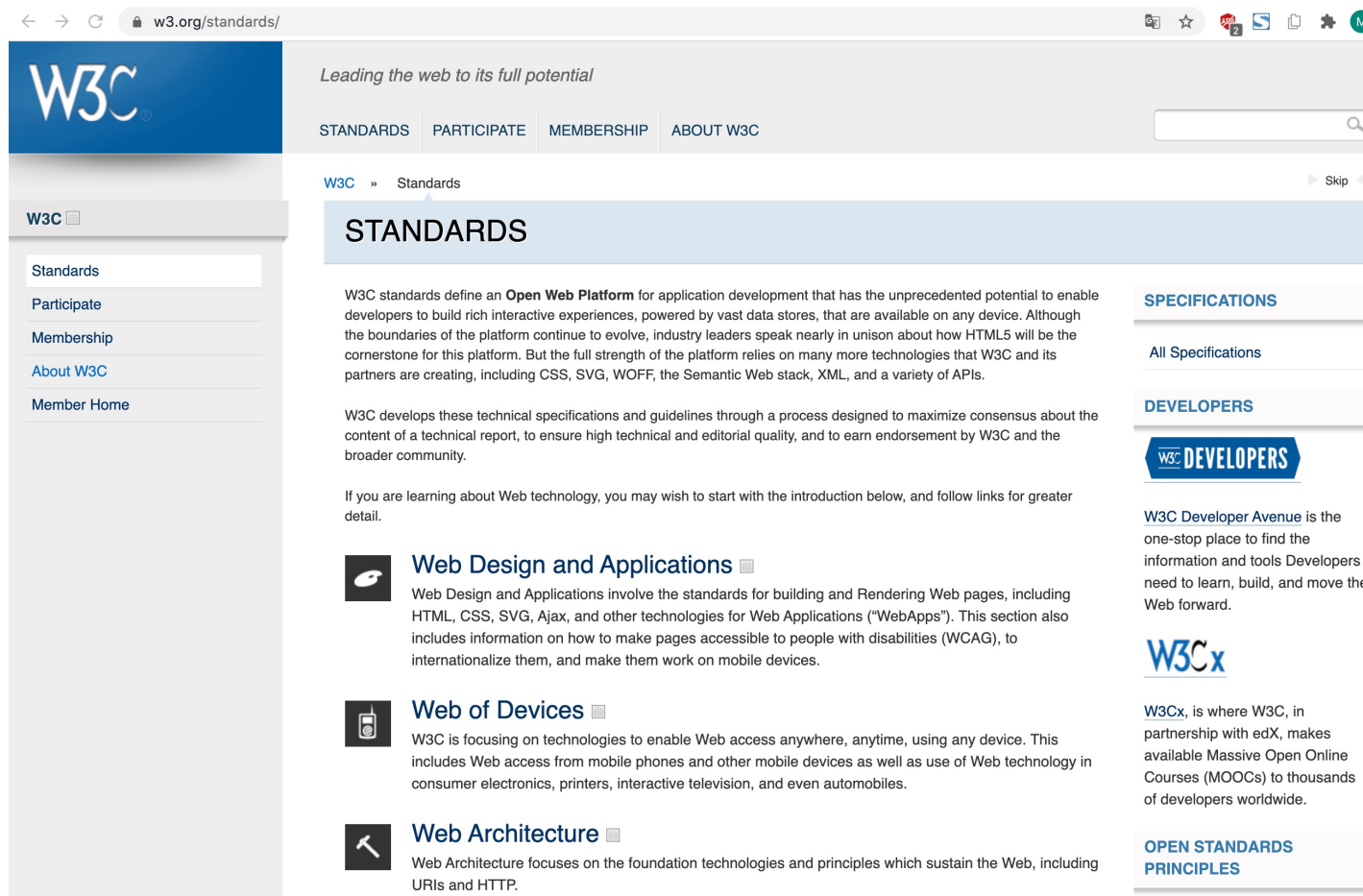
INTERNET STANDARDS

- [RFCs](#)
- Internet-Drafts
- Intellectual property rights
- Standards process

# Important references: W3C

The **World Wide Web Consortium** (W3C) is an international community where Member organizations, a full-time staff, and the public work together to develop **Web standards**.

<https://www.w3.org/>



The screenshot shows the W3C website's 'STANDARDS' page. The header features the W3C logo and the tagline 'Leading the web to its full potential'. Navigation links include 'STANDARDS', 'PARTICIPATE', 'MEMBERSHIP', and 'ABOUT W3C'. A search bar is located on the right. The main content area is titled 'STANDARDS' and contains a paragraph about W3C standards defining an 'Open Web Platform'. Below this, there are three sections: 'Web Design and Applications', 'Web of Devices', and 'Web Architecture', each with a brief description. On the right side, there are sections for 'SPECIFICATIONS', 'DEVELOPERS', and 'OPEN STANDARDS PRINCIPLES'. The left sidebar contains links for 'Standards', 'Participate', 'Membership', 'About W3C', and 'Member Home'.



# Important references: W3C Standards



## Web Design and Applications

Web Design and Applications involve the standards for building and Rendering Web pages, including HTML, CSS, SVG, Ajax, and other technologies for Web Applications ("WebApps"). This section also includes information on how to make pages accessible to people with disabilities (WCAG), to internationalize them, and make them work on mobile devices.



## Web of Devices

W3C is focusing on technologies to enable Web access anywhere, anytime, using any device. This includes Web access from mobile phones and other mobile devices as well as use of Web technology in consumer electronics, printers, interactive television, and even automobiles.



## Web Architecture

Web Architecture focuses on the foundation technologies and principles which sustain the Web, including URIs and HTTP.



## Semantic Web

In addition to the classic "Web of documents" W3C is helping to build a technology stack to support a "Web of data," the sort of data you find in databases. The ultimate goal of the Web of data is to enable computers to do more useful work and to develop systems that can support trusted interactions over the network. The term "Semantic Web" refers to W3C's vision of the Web of linked data. Semantic Web technologies enable people to create data stores on the Web, build vocabularies, and write rules for handling data. Linked data are empowered by technologies such as RDF, SPARQL, OWL, and SKOS.



## XML Technology

XML Technologies including XML, XML Namespaces, XML Schema, XSLT, Efficient XML Interchange (EXI), and other related standards.



## Web of Services

Web of Services refers to message-based design frequently found on the Web and in enterprise software. The Web of Services is based on technologies such as HTTP, XML, SOAP, WSDL, SPARQL, and others.



## Browsers and Authoring Tools

The web's usefulness and growth depends on its universality. We should be able to publish regardless of the software we use. the computer we have. the language we speak. whether we are wired or wireless.



# W3C Standards - 1

## WEB DESIGN AND APPLICATIONS



On this page → [technology topics](#) • [news](#) • [upcoming events and talks](#)

Web Design and Applications involve the standards for building and Rendering Web pages, including HTML, CSS, SVG, device APIs, and other technologies for Web Applications (“WebApps”). This section also includes information on how to make pages accessible to people with disabilities (WCAG), to internationalize them, and make them work on mobile devices.

### HTML & CSS

HTML and CSS are the fundamental technologies for building Web pages: HTML (html and xhtml) for structure, CSS for style and layout, including WebFonts. Find resources for good Web page design as well as helpful tools.

### JavaScript Web APIs

Standard APIs for client-side Web Application development include those for Geolocation, XMLHttpRequest, and mobile widgets. W3C standards for document models (the “DOM”) and technologies such as XBL allow content providers to create interactive documents through scripting.

### Graphics

W3C is the home of the widely deployed PNG raster format, SVG vector format, and the Canvas API. WebCGM is a more specialized format used, for example, in the fields of automotive engineering, aeronautics.

### Audio and Video

Some of the W3C formats that enable authoring audio and video presentations include HTML, SVG, and SMIL (for synchronization). W3C is also working on a timed text format for captioning and other applications.

### Accessibility

W3C’s Web Accessibility Initiative (WAI) has published Web Content Accessibility Guidelines (WCAG) to help authors create content that is accessible to people with disabilities. WAI-ARIA gives authors more tools to create accessible Web Applications by providing additional semantics about widgets and behaviors.

### Internationalization

W3C has a mission to design technology that works across cultures and languages. W3C standards such as HTML and XML are built on Unicode, for instance. In addition, W3C has published guidance for authors related to language tags bi-directional (bidi) text, and more.

### Mobile Web

W3C promotes “One Web” that is available on any device. W3C’s Mobile Web Best Practices help authors understand how to create content that provides a reasonable experience on a wide variety of devices, contexts, and locations.

### Privacy

The Web is a powerful tool for communications and transactions of all sorts. It is important to consider privacy and security implications of the Web as part of technology design. Learn more about tracking and Web App security.

### Math on the Web

Mathematics and formula are used on the Web for business reports, education materials and scientific research. W3C’s MathML enables mathematics to be served, received, and processed on the World Wide Web, just as HTML has enabled this functionality for other types of content.

# W3C Standards - 2

Web Architecture

▶ Skip ◀

## WEB ARCHITECTURE



On this page → [technology topics](#) • [news](#) • [upcoming events and talks](#)

Web Architecture focuses on the foundation technologies and principles which sustain the Web, including URIs and HTTP.

### Architecture Principles

Web Architecture principles help to design technologies by providing guidance and articulating the issues around some specific choices.

### Identifiers

We share things by their names. URL, URI, IRI is the way to name things on the Web and manipulate them. Some additional addressing needs in the Web Services stack motivated some additional layers.

### Protocols

Protocols are the vehicle for exchanging our ideas. HTTP is the core protocol of the Web. W3C is also working on XML Protocols and SOAP in relation to Web Services.

### Meta Formats

XML, the Extensible Markup Language, is used to build new formats at low cost (due to widely available tools to manipulate content in those new formats). RDF and OWL allow people to define vocabularies (“ontologies”) of terms as part of the Semantic Web.

### Protocol and Meta Format Considerations

Documents on the Web are loosely joined pieces by identifiers. It creates a maze of rich interactions between protocols and formats.

### Internationalization

W3C has worked with the community on the internationalization of identifiers (IRIs) and a general character model for the Web.



# W3C Standards - 3

XML Technology

► Skip ◀

## XML TECHNOLOGY



On this page → [technology topics](#) • [news](#) • [upcoming events and talks](#)

XML Technologies including XML, XML Namespaces, XML Schema, XSLT, Efficient XML Interchange (EXI), and other related standards.

### XML Essentials

XML is shouldered by a set of essential technologies such as the infoset and namespaces. They address issues when using XML in specific applications contexts.

### Efficient Interchange

XML standards are omnipresent in enterprise computing, and are part of the foundation of the Web. Because the standards are highly interoperable and affordable, people have wanted to use them in a wide variety of applications. However, in some settings (on devices with low memory or low bandwidth, or where performance is critical) experience has shown that a more efficient form of XML is required.

### Schema

Formal descriptions of vocabularies create flexibility in authoring environments and quality control chains. W3C's XML Schema, SML, and data binding technologies provide the tools for quality control of XML data.

### Security

Manipulating data with XML requires sometimes integrity, authentication and privacy. XML signature, encryption, and xkms can help create a secure environment for XML.

### Transformation

Very frequently one wants to transform XML content into other formats (including other XML formats). XSLT and XPath are very powerful tools for creating different representations of XML content.

### Query

XQuery (supported by XPath) is a query language for XML to extract data, similar to the role of SQL for databases, or SPARQL for the Semantic Web.

### Components

The XML ecosystem is using additional tools to create a richer environment for using and manipulating XML documents. These components include style sheets, xlink xml:id, xinclude, xpointer, xforms, xml fragments, and events.

### Processing

A processing model defines what operations should be performed in what order on an XML document.

### Internationalization

W3C has worked with the community on the internationalization of XML, for instance for specifying the language of XML content.

# W3C Standards - 4

Web of Services

▶ Skip ◀

## WEB OF SERVICES



On this page → [technology topics](#) • [news](#) • [upcoming events and talks](#)

Web of Services refers to message-based design frequently found on the Web and in enterprise software. The Web of Services is based on technologies such as HTTP, XML, SOAP, WSDL, SPARQL, and others.

### Payments

W3C seeks to integrate payments seamlessly into the Open Web Platform. The result will be new business opportunities, an improved user experience for online transactions, reduced fraud, and increased interoperability among traditional solutions and future payment innovations.

### Security

Transferring data from one domain to another domain or between applications needs sometimes a secure transaction and well defined document authentication. XML Encryption and XML Signature are key pieces of the XML security stack.

### Internationalization

Internationalization of Web services concerns service descriptions, communicating language and locale, and internationalization of human-readable messages exchanged by services.

# Introduction

Basic concepts

# Protocol

- **Synonymous of Etiquette**

a code of behavior that delineates expectations for social behavior according to contemporary conventional norms within a society, social class, or group.



**Communications protocol**, a set of rules and regulations that determine how data is transmitted in telecommunications and computer networking

# Protocol

■ *A **protocol** defines:*

- 1. the **format** and*
- 2. the **order** of messages exchanged between two or more communicating entities, as well as*
- 3. the **actions** taken on the transmission and/or receipt of a message or other event.*



# Port

A port is an **endpoint of communication in an operating system.**

A **process** associates its input or output channels, via an Internet **socket**, with a transport protocol, a port number, and an IP address.

socket: {protocol, local address, local port, remote address, remote port}

This process is known as binding.



HTTP on port 80

- HTTP with SSL (HTTPS) on port 443
- FTP on port 21
- SMTP on port 25
- POP on port 110
- SSH on port 22

Mistranslated into Italian as “Porta” (door)

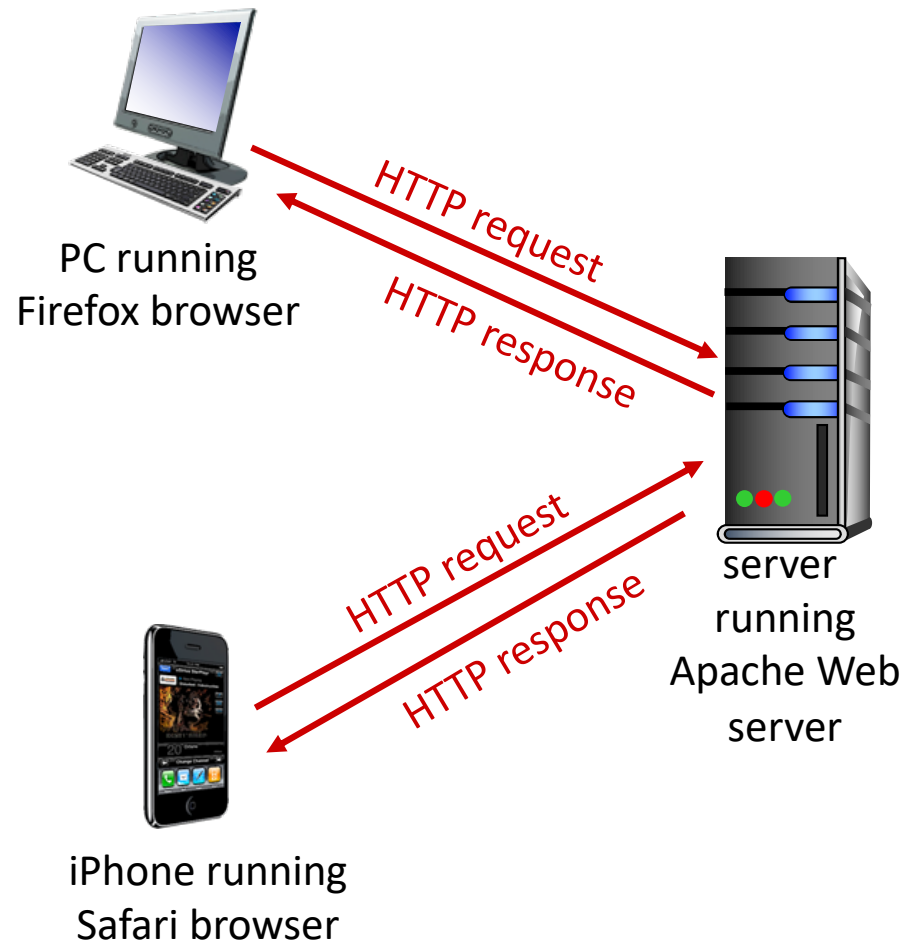
| PID | PORT | IP              | Protocol |
|-----|------|-----------------|----------|
| 84  | 21   | 193.205.196.130 | FTP      |
| 78  | 80   | 193.205.196.130 | HTTP     |
| 321 | 8080 | 193.205.196.130 | HTTP     |
| 541 | 25   | 193.205.196.130 | SMTP     |



# HTTP overview

HTTP: hypertext transfer protocol

- Web's application layer protocol
- client/server model
  - client: browser that requests, receives, (using HTTP protocol) and “displays” Web objects
  - server: Web server sends (using HTTP protocol) objects in response to requests



# HTTP overview (continued)

uses TCP:

- client initiates TCP connection (creates socket) to server, port 80
- server accepts TCP connection from client
- HTTP messages (application-layer protocol messages) exchanged between browser (HTTP client) and Web server (HTTP server)
- TCP connection closed



# HTTP overview (continued)

HTTP is “stateless”

- server maintains no information about past client requests

protocols that maintain “state” are complex:

- past history (state) must be maintained
- if server/client crashes, their views of “state” may be inconsistent, must be reconciled

# URI, URL, URN

- A **web resource**, or simply **resource**, is any identifiable thing, whether digital, physical, or abstract.
- A **Uniform Resource Identifier (URI)** is a compact sequence of characters that identifies an abstract or physical resource. (RFC 3986)
- A **Uniform Resource Locator (URL)** refers to the subset of URI that identify resources via a ***representation of their primary access mechanism*** (e.g., their network "location")
- **Uniform Resource Name (URN)** refers to the subset of URI that are required to **remain globally unique and persistent even when the resource ceases to exist** or becomes unavailable.  
It is intended to serve as persistent, location-independent, resource identifier (RFC 2141).

# RFC (Request for comment)

Memos in the RFC document series contain technical and organizational notes about the Internet.

RFCs cover many aspects of computer networking, including protocols, procedures, programs, and concepts, as well as meeting notes, opinions, and sometimes humour.

<https://ietf.org/standards/rfcs/>

Internet Engineering Task Force (IETF)  
Request for Comments: 8141  
Obsoletes: [2141](#), [3406](#)  
Category: Standards Track  
ISSN: 2070-1721

P. Saint-Andre  
Filament  
J. Klensin  
April 2017

## Uniform Resource Names (URNs)

### Abstract

A Uniform Resource Name (URN) is a Uniform Resource Identifier (URI) that is assigned under the "urn" URI scheme and a particular URN namespace, with the intent that the URN will be a persistent, location-independent resource identifier. With regard to URN syntax, this document defines the canonical syntax for URNs (in a way that is consistent with URI syntax), specifies methods for determining URN-equivalence, and discusses URI conformance. With regard to URN namespaces, this document specifies a method for defining a URN namespace and associating it with a namespace identifier, and it describes procedures for registering namespace identifiers with the Internet Assigned Numbers Authority (IANA). This document obsoletes both RFCs 2141 and 3406.

# URI, URL, URN

- Both URL and URN are URI.
- A URN identifies a resource
- A URL provides a method for finding it.
- A URN may be compared to a person's name,
- A URL may be compared to their street address.
- A URN can be associated to many URLs

# URN + URL example

- The ISBN system (namespace) uniquely identifies books.
- **urn:isbn:0-486-27557-4** cites unambiguously a *specific edition* of Shakespeare's play Romeo and Juliet.
- A typical URL for this book might look like the file path **file:///home/username/RomeoAndJuliet.pdf**

The screenshot shows the Amazon Italy interface for the book 'Romeo and Juliet' by William Shakespeare. The top navigation bar includes the Amazon Prime logo, a search bar with the ISBN '0-486-27557-4', and various promotional banners. Below the navigation bar, a search result summary indicates '1 risultato per "isbn:0-486-27557-4"'. The main content area is divided into three sections: a left sidebar with navigation options like 'Amazon Prime', 'Categoria', and 'Media recensioni clienti'; a central image of the book cover; and a right section with product details. The book cover is titled 'ROMEO & JULIET' and features a classic illustration of the two characters. The product details section includes the title 'Romeo and Juliet', the author 'di William Shakespeare', the publication date '26 lug. 1993', a star rating of 4.5 stars from 60 reviews, the price '2,29€', and the Amazon Prime logo. Below the price, it lists 'Ulteriori opzioni di acquisto' for 2,00 € and mentions '16 offerte prodotti nuovi e usati'. The 'Formato Kindle' section shows the price '0,00€' with the 'kindleunlimited' logo, and notes that it is 'Gratuito con l'iscrizione a Kindle Unlimited' or 'Oppure 0,00 € per acquistare'.

amazon.it prime

Tutte le categorie ▼ isbn:0-486-27557-4

Amazon Hub Locker - a... Povo 38123

Acquista di nuovo Amazon.it di RONCH... Offerte di Natale Occasioni a prezzi bassi

Amazon.it Offerte Usato e ricondizionato Outlet Made in Italy Novità Bestseller Amazon Prime App di Amazon

1 risultato per "isbn:0-486-27557-4"

Amazon Prime

☐ prime

Categoria

Kindle Store

Libri

Media recensioni clienti

★★★★★ e più

★★★★☆ e più

★★★☆☆ e più

★★☆☆☆ e più

Spedizione Internazionale

☐ Ammissibili di spedizione internazionale

William Shakespeare

ROMEO & JULIET

POVER THRIFT EDITIONS

Romeo and Juliet

di William Shakespeare | 26 lug. 1993

★★★★★ ~ 60

Copertina flessibile

2,29€

prime

Ulteriori opzioni di acquisto

2,00 € (16 offerte prodotti nuovi e usati)

Formato Kindle

0,00€ kindleunlimited

Gratuito con l'iscrizione a Kindle Unlimited

Oppure 0,00 € per acquistare

# URI Schemes

http:

https:

ftp:

mailto:<address>[?<header1>=<value1>[&<header2>=<value2>]]

geo:<lat>,<lon>

fax:<phone number>

file:[//host]/path

bitcoin:<address>[?[amount=<size>]]...

skype:<username | phonenumber>...

[https://en.wikipedia.org/wiki/List\\_of\\_URI\\_schemes](https://en.wikipedia.org/wiki/List_of_URI_schemes)

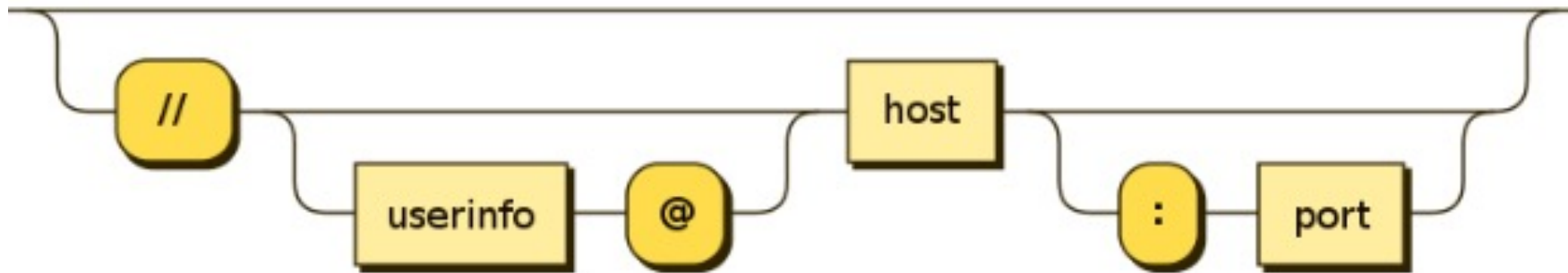
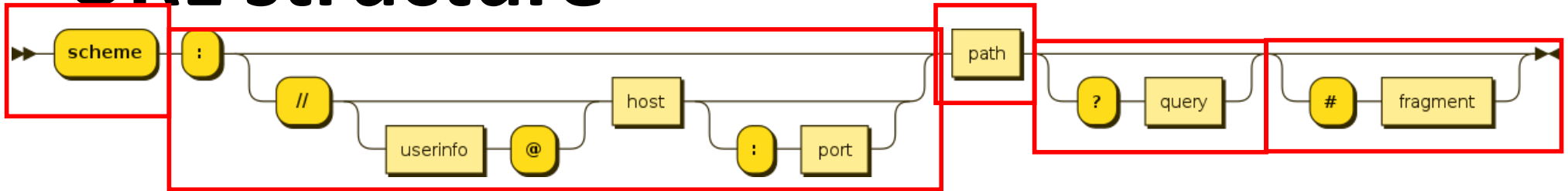
# URI structure

URI = scheme:[//**authority**]path[?query][#fragment]

**authority** = [userinfo@]host[:port]



# URL structure

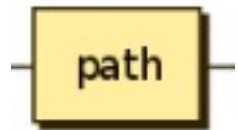
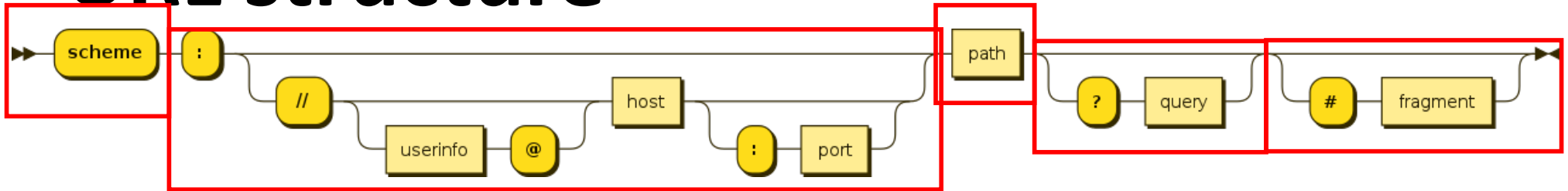


**https://en.wikipedia.org/wiki/URL**

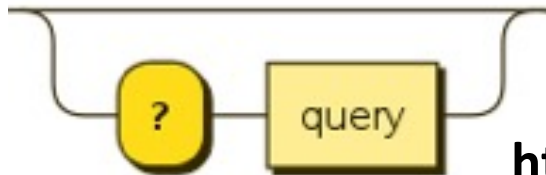
Watch for phishing!

<http://www.sitosicuro.it|search=hello@www.phishing.com/>

# URL structure



<https://en.wikipedia.org/wiki/URL>



<https://en.wikipedia.org/w/index.php?title=URL&action=edit>



[https://en.wikipedia.org/wiki/URL#Protocol-relative\\_URLs](https://en.wikipedia.org/wiki/URL#Protocol-relative_URLs)

# MIME type

URLs give us info about name and location, but what about their content type?

File extensions are bad...

(see Unix magic numbers

<https://www.geeksforgeeks.org/working-with-magic-numbers-in-linux/> )

## Metadata?

# MIME type

Multipurpose Internet Mail Extensions (RFC 2045,2046)

`MEDIA_TYPE/SUBTYPE`

**text** -> text/plain, text/html, text/richtext ...

**image** -> image/jpeg, image/png, image/svg+xml...

**audio** -> audio/basic, audio/ogg, audio/x-wav...

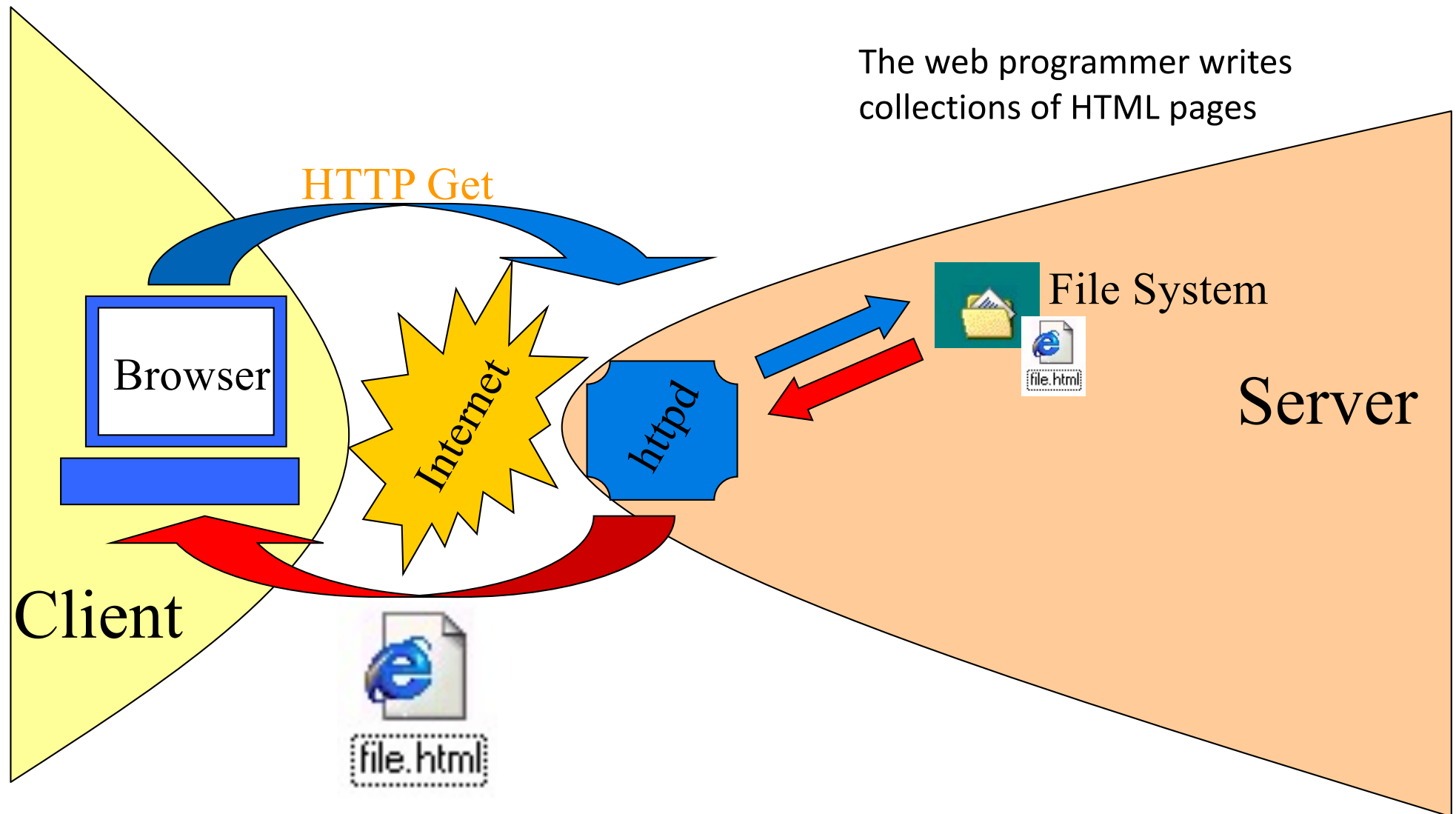
**video** -> video/mp4, video/ogg...

**application** -> application/x-apple-diskimage... ..

**multipart**

See <https://www.freeformatter.com/mime-types-list.html>

## The original web architecture: static pages



**Initial idea: get (static) interlinked documents**

# Client and Server

*from “Computer Networking: A Top-Down Approach”, Kurose-Ross:*

- The computers and other devices connected to the Internet are often referred to as **end systems (hosts)**.
- Hosts are sometimes further divided into two categories: **clients** and **servers**.
- Informally, clients tend to be desktop and mobile PCs, smartphones, and so on, whereas servers tend to be more powerful machines that store and distribute Web pages, stream video, relay e-mail, and so on.

# Servers ?





# Server = Waiter, Client = Customer





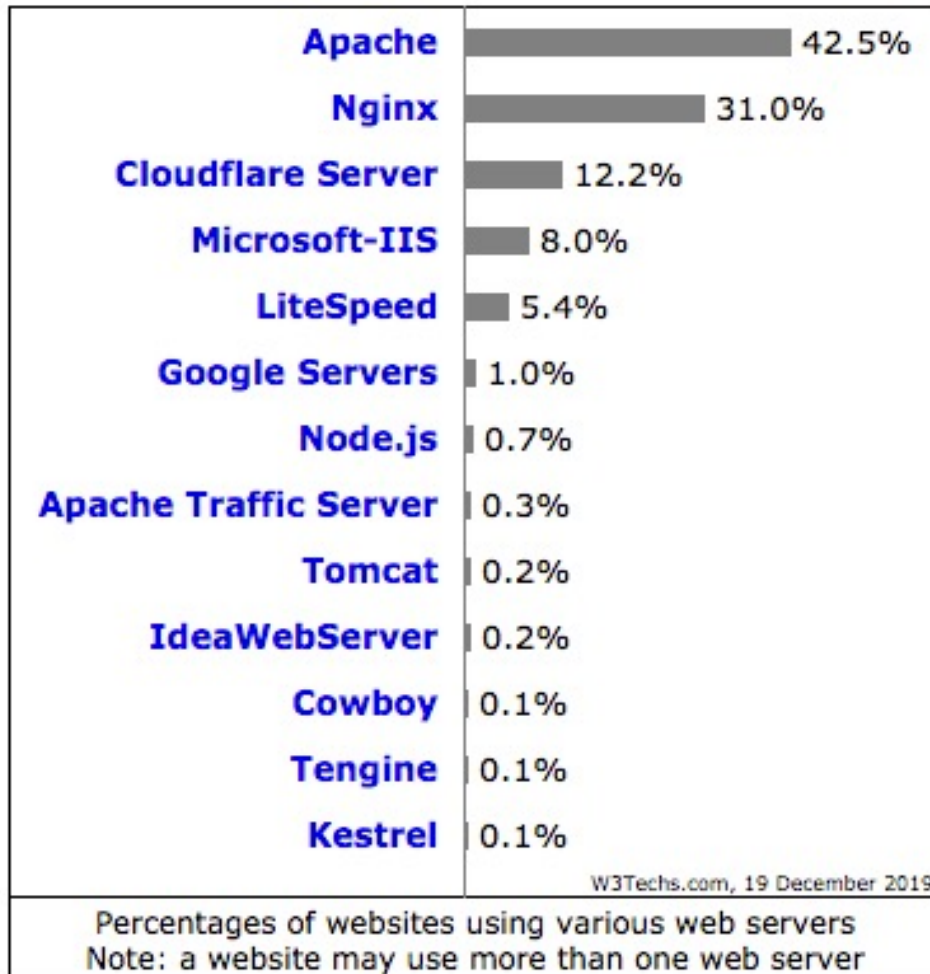
# Client and Server

A better definition:

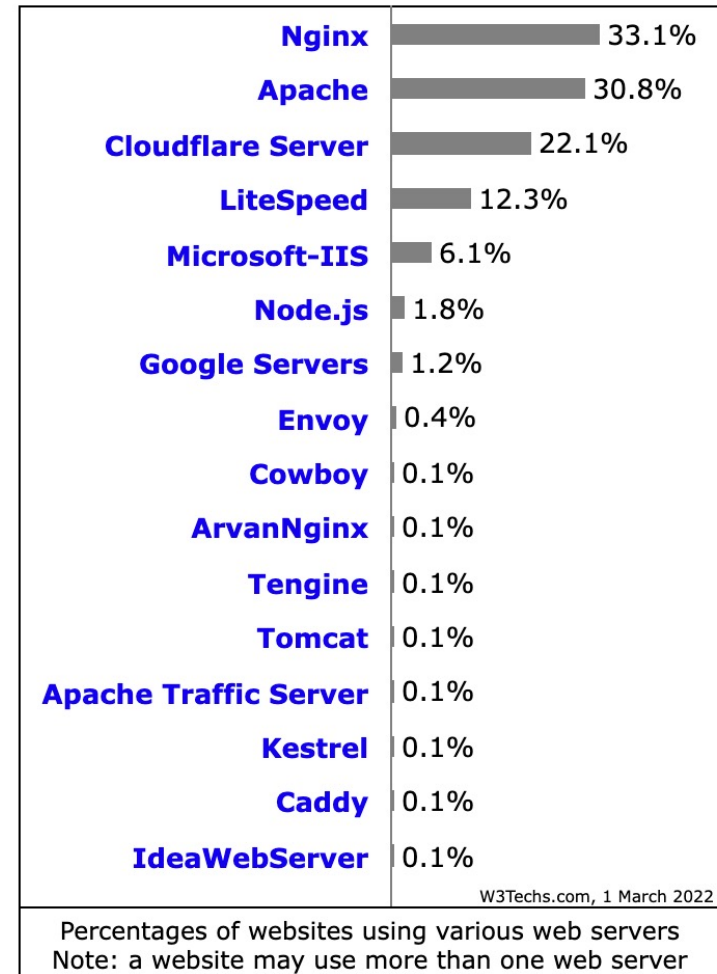
- A Server is a machine that **opens a SocketServer** connection, and **waits** for incoming calls (in order to provide a service)
- A Client is a machine the starts a connection (**opening a Socket to the server**) and requests a service.

**Server and client are SOFTWARE roles,  
not HARDWARE concept!**

# Servers on the web

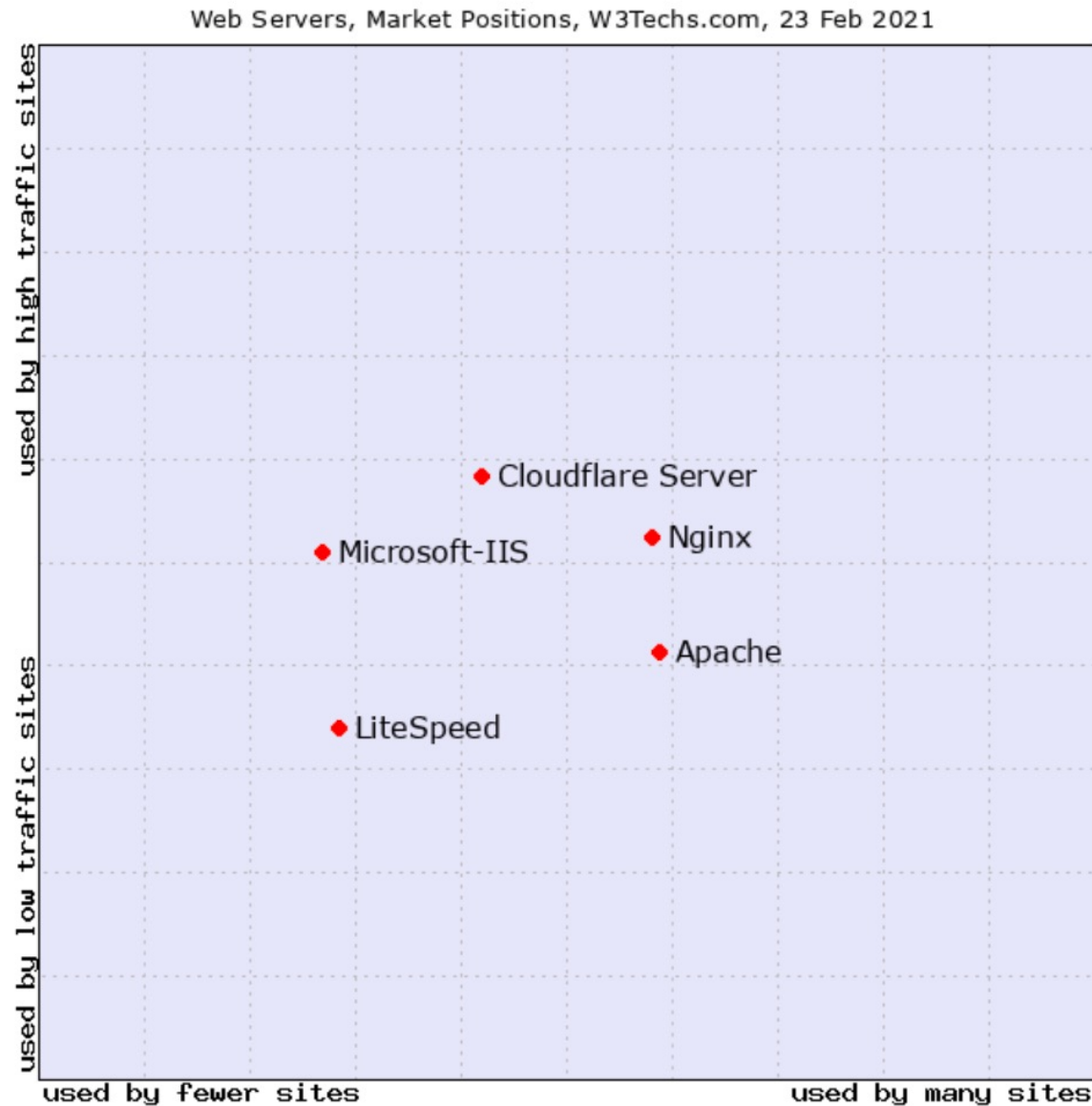


Dec 2019



March 2022

# Servers on the web

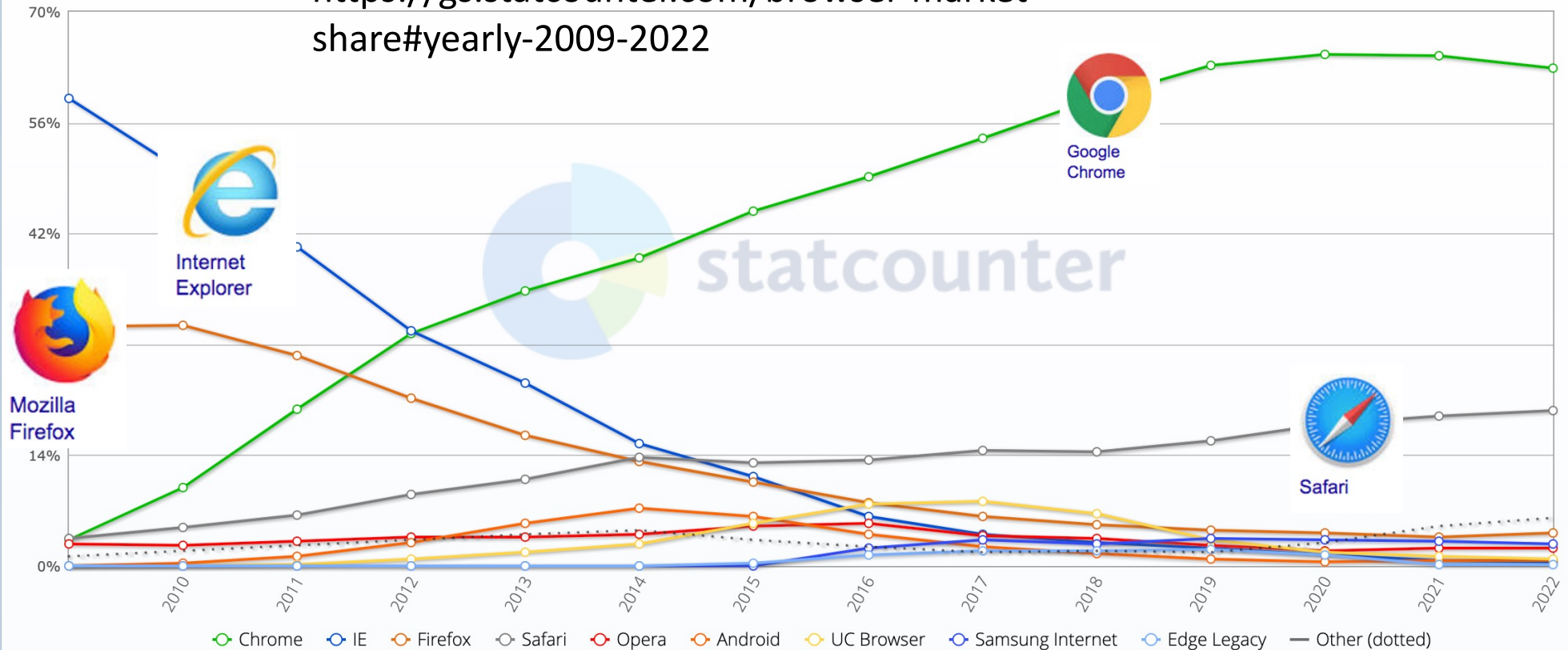


# Clients on the Web: Browsers

Browser Market Share Worldwide  
2009 - 2022

Edit Chart Data

<https://gs.statcounter.com/browser-market-share#yearly-2009-2022>



Save Chart Image (.png)

Download Data (.csv)

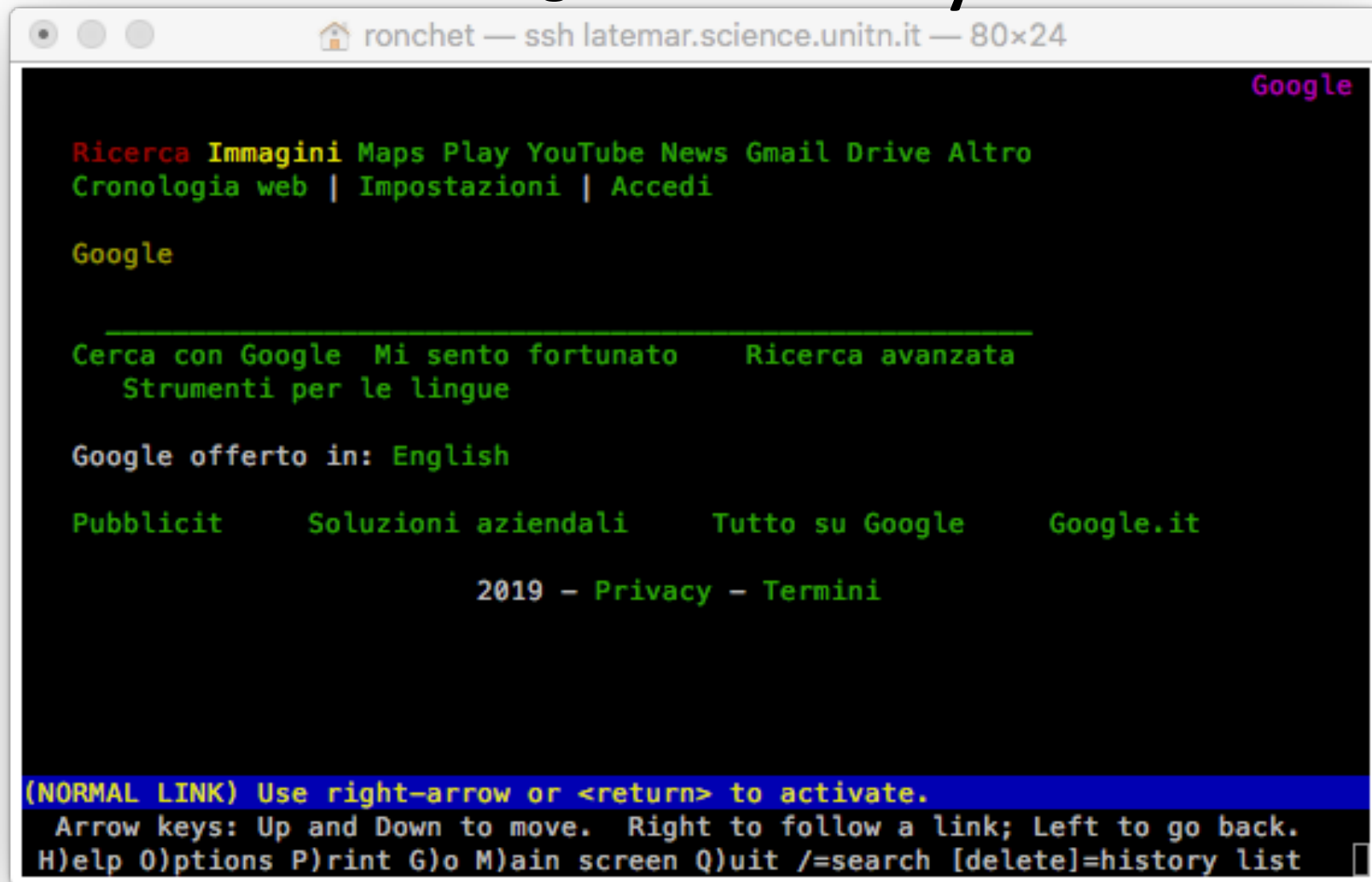
Embed HTML

```
<div id="all-browser-ww-yearly-2009-2022" width="600" height="400" style="display: flex; align-items: center; justify-content: center; border: 1px solid #ccc; padding: 10px;"></div>
```

# Clients on the Web: Browsers

Could also be something else!

**Lynx – text browser**



The image shows a terminal window titled "ronchet — ssh latemar.science.unitn.it — 80x24". Inside the terminal, the Lynx text browser displays the Google homepage. The text is rendered in a monospaced font with various colors (red, green, yellow, magenta) for different elements. At the top right, the word "Google" is in magenta. Below it, a navigation bar includes links like "Ricerca", "Immagini", "Maps", "Play", "YouTube", "News", "Gmail", "Drive", and "Altro". A search bar is present with the text "Cerca con Google". Below the search bar, there are links for "Mi sento fortunato", "Ricerca avanzata", and "Strumenti per le lingue". Further down, it says "Google offerto in: English". At the bottom of the main content area, there are links for "Pubblicit", "Soluzioni aziendali", "Tutto su Google", and "Google.it". A footer line reads "2019 - Privacy - Termini". At the very bottom of the terminal, a blue status bar contains the text: "(NORMAL LINK) Use right-arrow or <return> to activate. Arrow keys: Up and Down to move. Right to follow a link; Left to go back. H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list".


```
ronchet — ssh latemar.science.unitn.it — 80x24
Google
Ricerca Immagini Maps Play YouTube News Gmail Drive Altro
Cronologia web | Impostazioni | Accedi
Google
Cerca con Google Mi sento fortunato Ricerca avanzata
Strumenti per le lingue
Google offerto in: English
Pubblicit Soluzioni aziendali Tutto su Google Google.it
2019 - Privacy - Termini
(NORMAL LINK) Use right-arrow or <return> to activate.
Arrow keys: Up and Down to move. Right to follow a link; Left to go back.
H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```



# Clients on the Web: Browsers

Could also be something else!

**WebView in JavaFX**



JavaFX Documentation Home > Adding HTML Content to JavaFX Applications


## Adding HTML Content to JavaFX Applications

This chapter introduces the JavaFX embedded browser, a user interface component that provides a web viewer and full browsing functionality through its API.

The embedded browser component is based on [WebKit](#), an open source web browser engine. It supports Cascading Style Sheets (CSS), JavaScript, Document Object Model (DOM), and HTML5.

The embedded browser enables you to perform the following tasks in your JavaFX applications:

- Render HTML content from local and remote URLs
- Obtain Web history
- Execute JavaScript commands
- Perform upcalls from JavaScript to JavaFX
- Manage web pop-up windows
- Apply effects to the embedded browser



**Release: JavaFX 2.2**  
Last Updated: January 2014  
[Download as PDF](#)

---

[\[+\] Show/Hide Table of Contents](#)

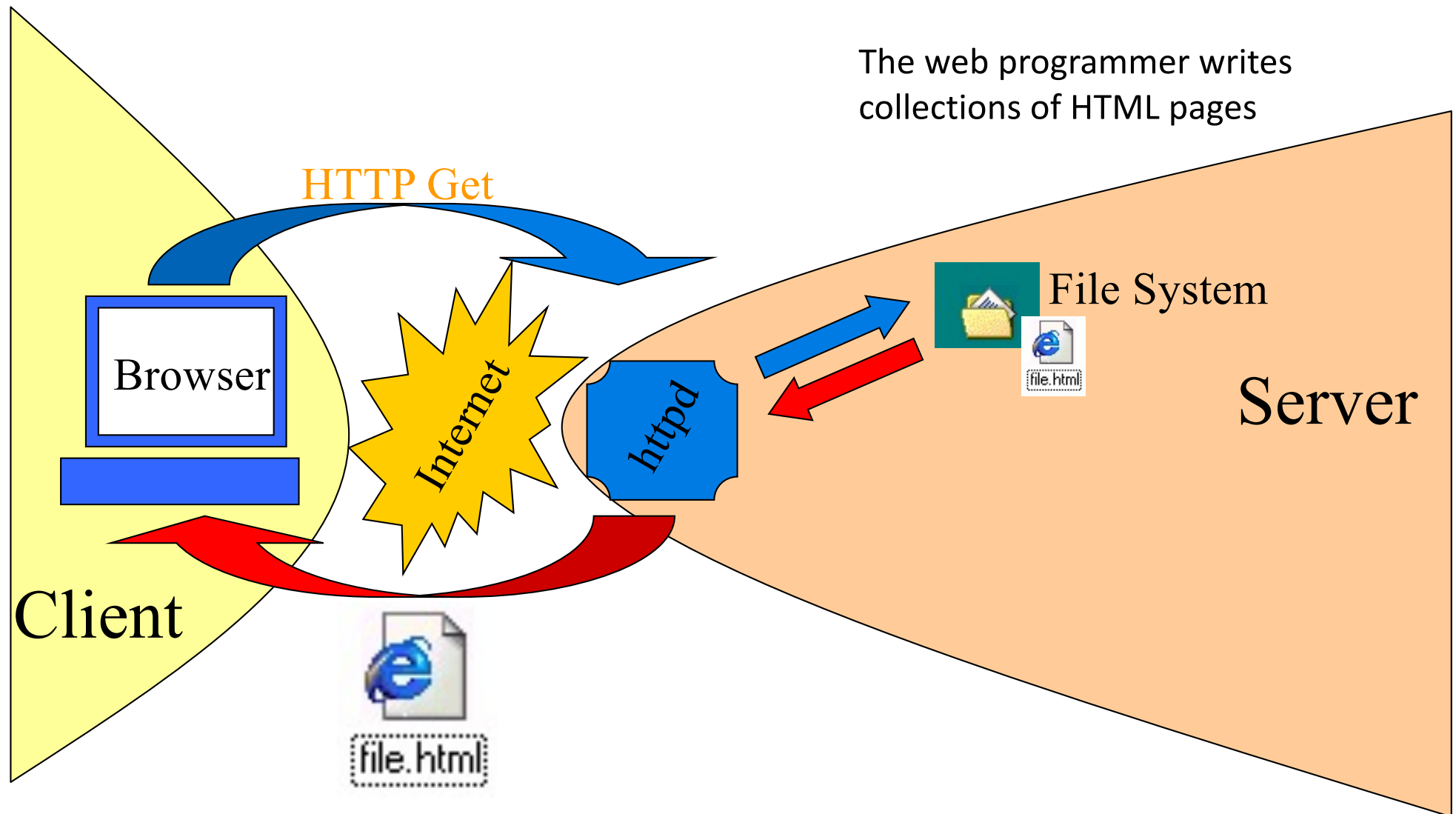
---

**Application Files**  
**View Source Code**

- WebViewSample.java
- BrowserToolbar.css
- help.html
- blog.png
- documentation.png
- partners.png
- product.png
- help.png



## The original web architecture: static pages



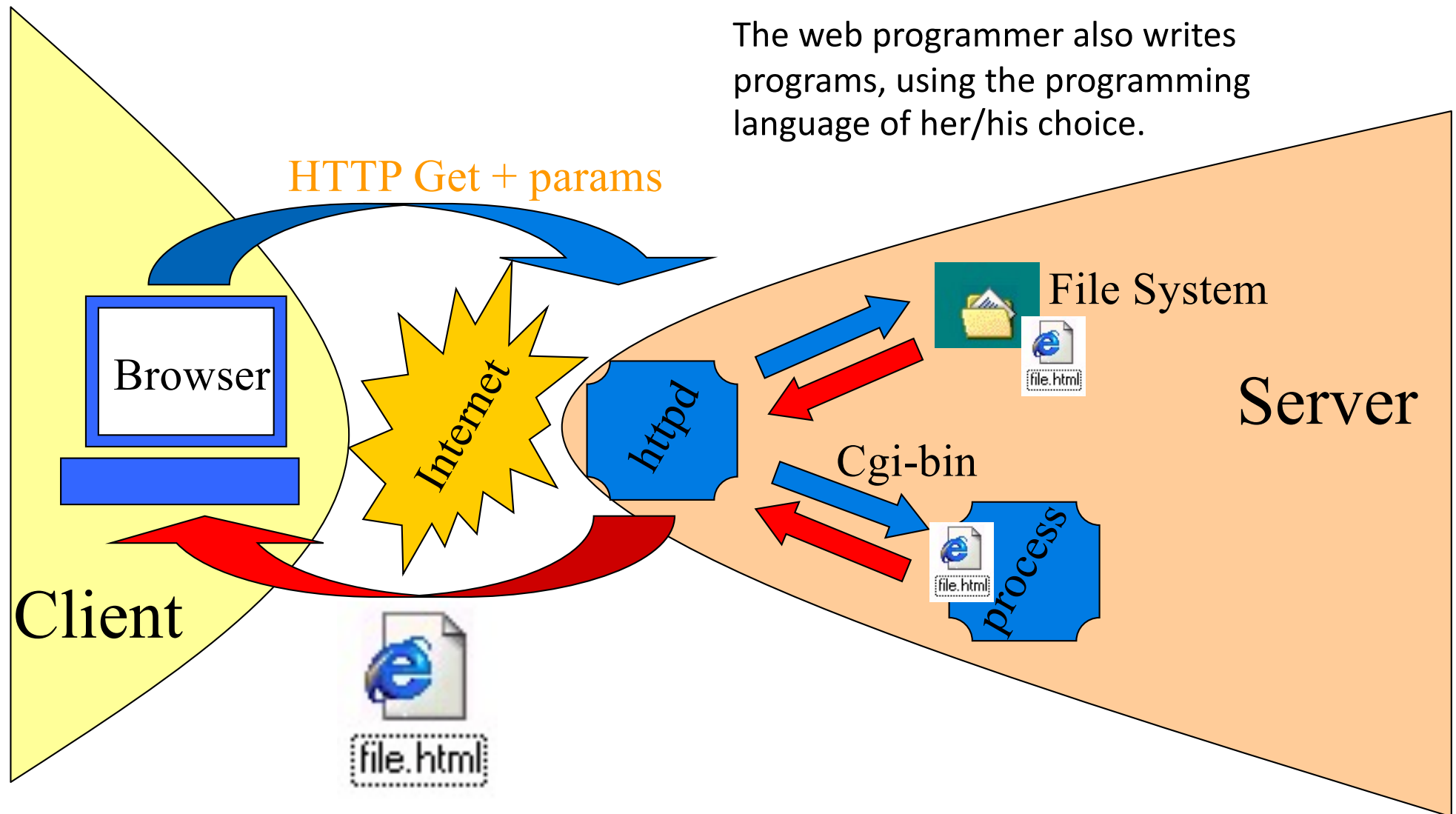
**Initial idea: get (static) interlinked documents**

# The model

- Mapping 1:1 between URLs and (static) resources
- The Web server is nothing but a retriever of the content associated to an URL (typically HTML page, image...)



## The original web architecture: dynamic pages

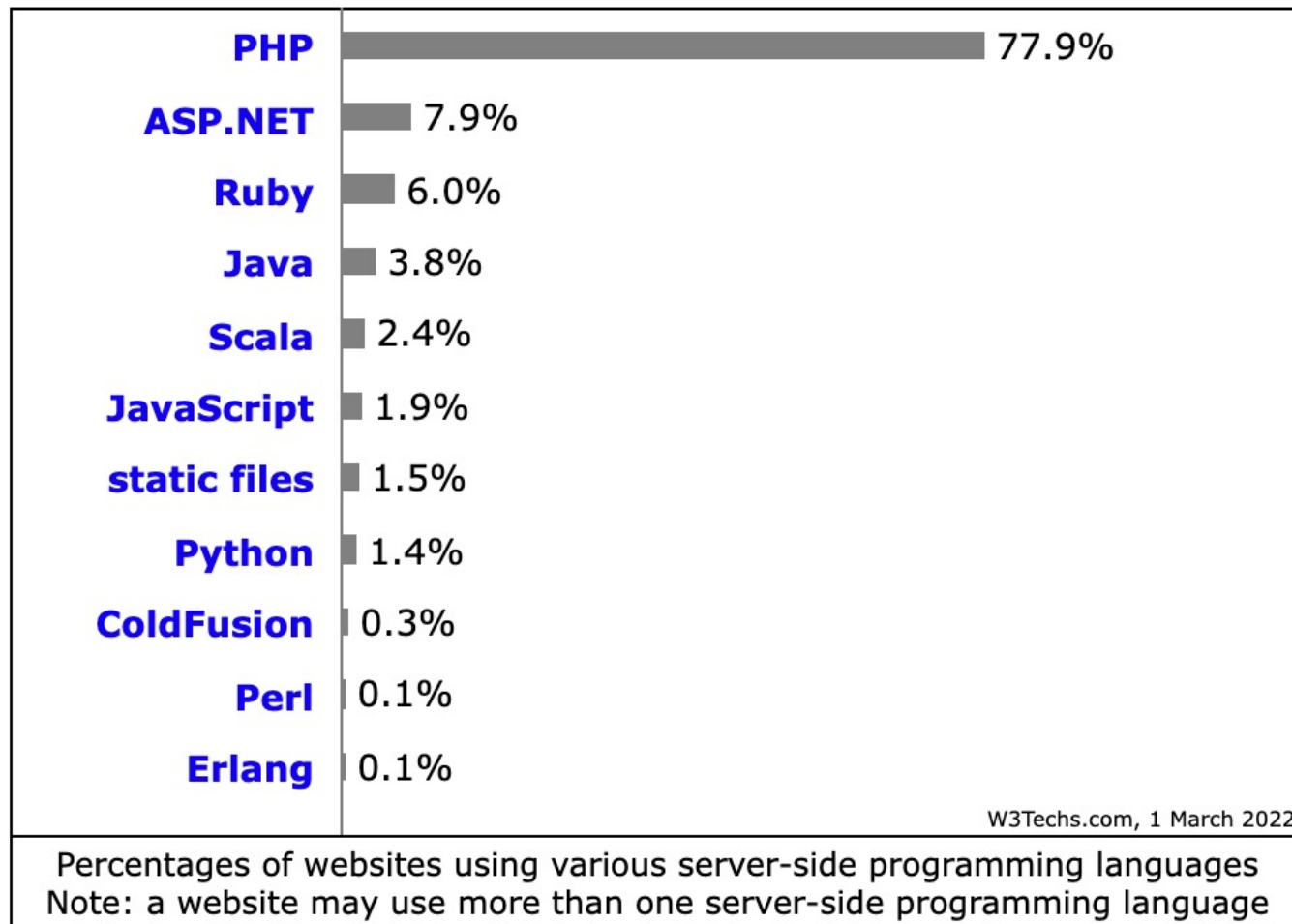


**Evolution 1: dynamically create (interlinked) documents**

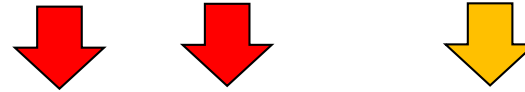
# The model

- Certain URLs are associated with **actions** and carry **parameters** for the action
- The Web server:
  - understands that certain URLs are *dynamic*
  - parses the parameters
  - starts a process (or thread) corresponding to the desired action
  - obtains from the process some data (for human consumption)
  - passed the data to the client

# Languages



# Back-end (Server-side) table in most popular websites



Back-end (Server-side) table in most popular websites

| Websites      | C#  | C   | C++ | D   | Erlang | Go  | Hack | Java | JavaScript | Perl | PHP | Python | Ruby | Scala | XHP |
|---------------|-----|-----|-----|-----|--------|-----|------|------|------------|------|-----|--------|------|-------|-----|
| Google        | No  | Yes | Yes | No  | No     | Yes | No   | Yes  | Yes        | No   | No  | Yes    | No   | No    | No  |
| YouTube       | No  | Yes | Yes | No  | No     | Yes | No   | Yes  | No         | No   | No  | Yes    | No   | No    | No  |
| Facebook      | No  | No  | Yes | Yes | Yes    | Yes | Yes  | Yes  | No         | No   | Yes | Yes    | No   | No    | Yes |
| Yahoo         | No  | Yes | Yes | No  | No     | Yes | No   | Yes  | Yes        | Yes  | Yes | Yes    | Yes  | Yes   | No  |
| Amazon        | No  | No  | Yes | No  | No     | No  | No   | Yes  | No         | Yes  | No  | No     | No   | No    | No  |
| Wikipedia     | No  | No  | No  | No  | No     | No  | No   | No   | No         | No   | Yes | No     | No   | No    | No  |
| Twitter       | No  | No  | Yes | No  | No     | No  | No   | Yes  | No         | No   | No  | No     | Yes  | Yes   | No  |
| Bing          | Yes | No  | Yes | No  | No     | No  | No   | No   | No         | No   | No  | No     | No   | No    | No  |
| eBay          | No  | No  | No  | No  | No     | No  | No   | Yes  | Yes        | No   | No  | No     | No   | Yes   | No  |
| MSN           | Yes | No  | No  | No  | No     | No  | No   | No   | No         | No   | No  | No     | No   | No    | No  |
| LinkedIn      | No  | No  | No  | No  | No     | No  | No   | Yes  | Yes        | No   | No  | No     | No   | Yes   | No  |
| Pinterest     | No  | No  | No  | No  | Yes    | No  | No   | No   | No         | No   | No  | Yes    | No   | No    | No  |
| WordPress.com | No  | No  | No  | No  | No     | No  | No   | No   | No         | No   | Yes | No     | No   | No    | No  |

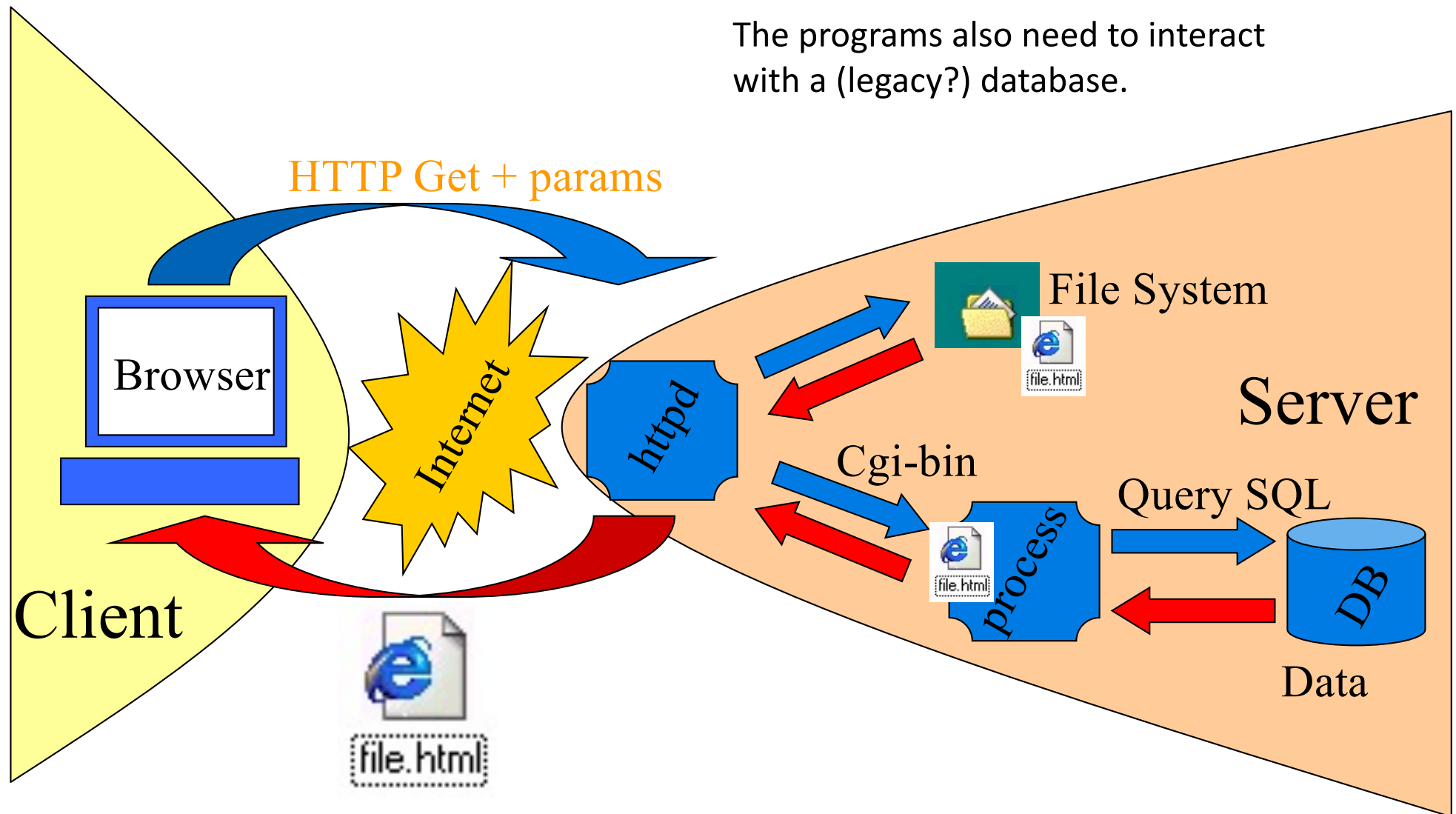
7

8

5

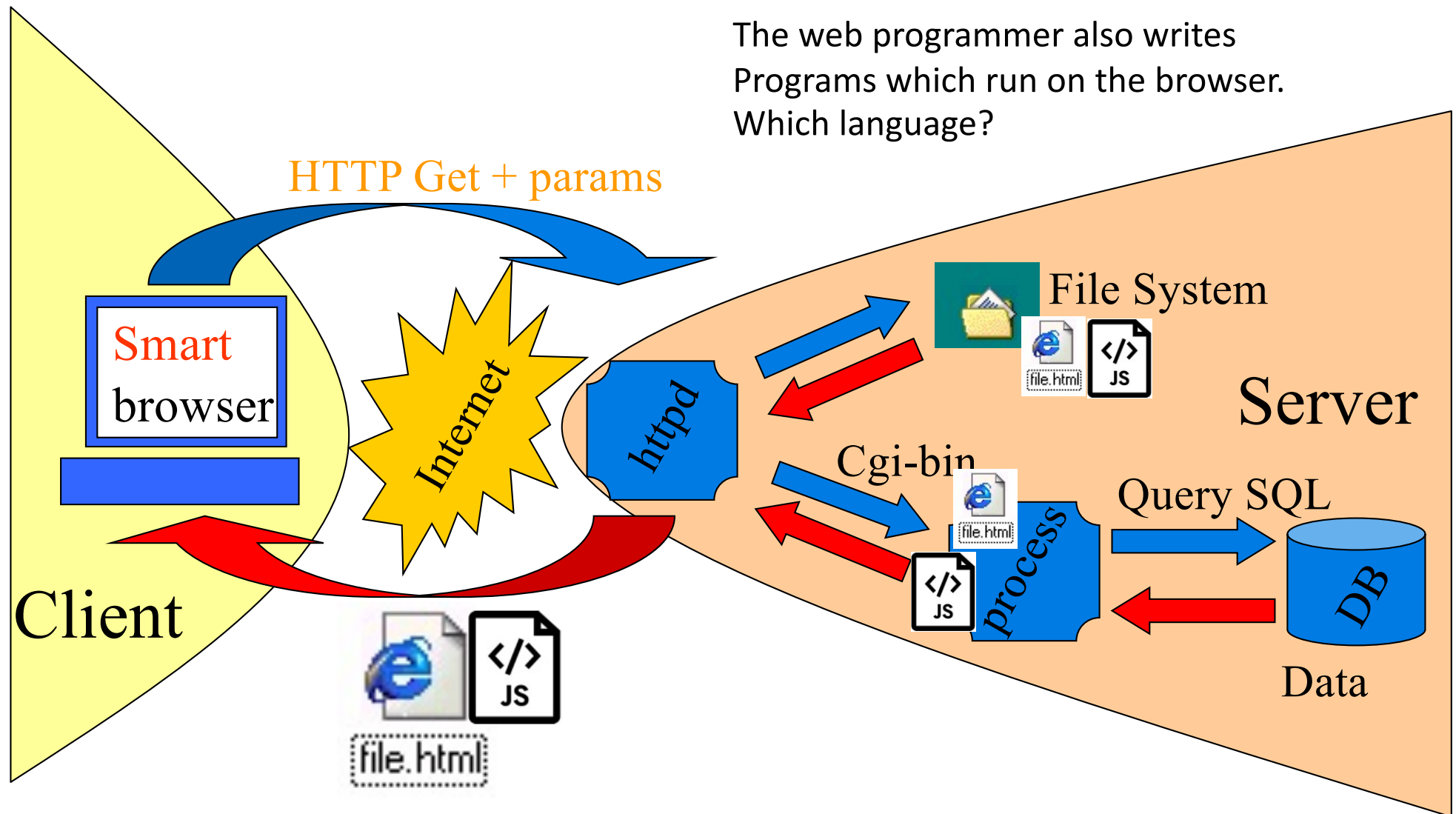
*"Il prof ha la fissa di Java"*

## The original web architecture: dynamic pages with DB



**Evolution 2: dynamically create (interlinked) documents interacting with a persistent data storage**

## The web architecture with smart browser



**Evolution 3: execute code also on client! (How ?)**

# Front-end (client-side) table in most popular websites

| Websites      | Popularity (unique visitors per month) | Front-end              | Notes  |
|---------------|--|------------------------|--|
| Google.com    | 1,600,000,000                          | JavaScript, TypeScript | The most used search engine in the world   |
| Facebook.com  | 1,100,000,000                          | JavaScript             | The most visited social networking site  |
| YouTube.com   | 1,100,000,000                          | JavaScript             | The most visited video sharing site  |
| Yahoo         | 750,000,000                            | JavaScript             |  |
| Amazon.com    | 500,000,000                            | JavaScript             | Popular internet shopping site   |
| Wikipedia.org | 475,000,000                            | JavaScript             | "MediaWiki" is programmed in PHP; free online encyclopedia   |
| Twitter.com   | 290,000,000                            | JavaScript             | Popular social network.  |
| Bing          | 285,000,000                            | JavaScript             | Search engine from Microsoft.  |
| eBay.com      | 285,000,000                            | JavaScript             | Online auction house.  |
| MSN.com       | 280,000,000                            | JavaScript             | An email client, for simple use. Previously known as "messenger", not to be confused with Facebook's messaging platform. |
| Linkedin.com  | 260,000,000                            | JavaScript             | World's largest professional network.  |
| Pinterest     | 250,000,000                            | JavaScript             | Search engine for ideas.   |
| WordPress.com | 240,000,000                            | JavaScript             | Website manager software.  |

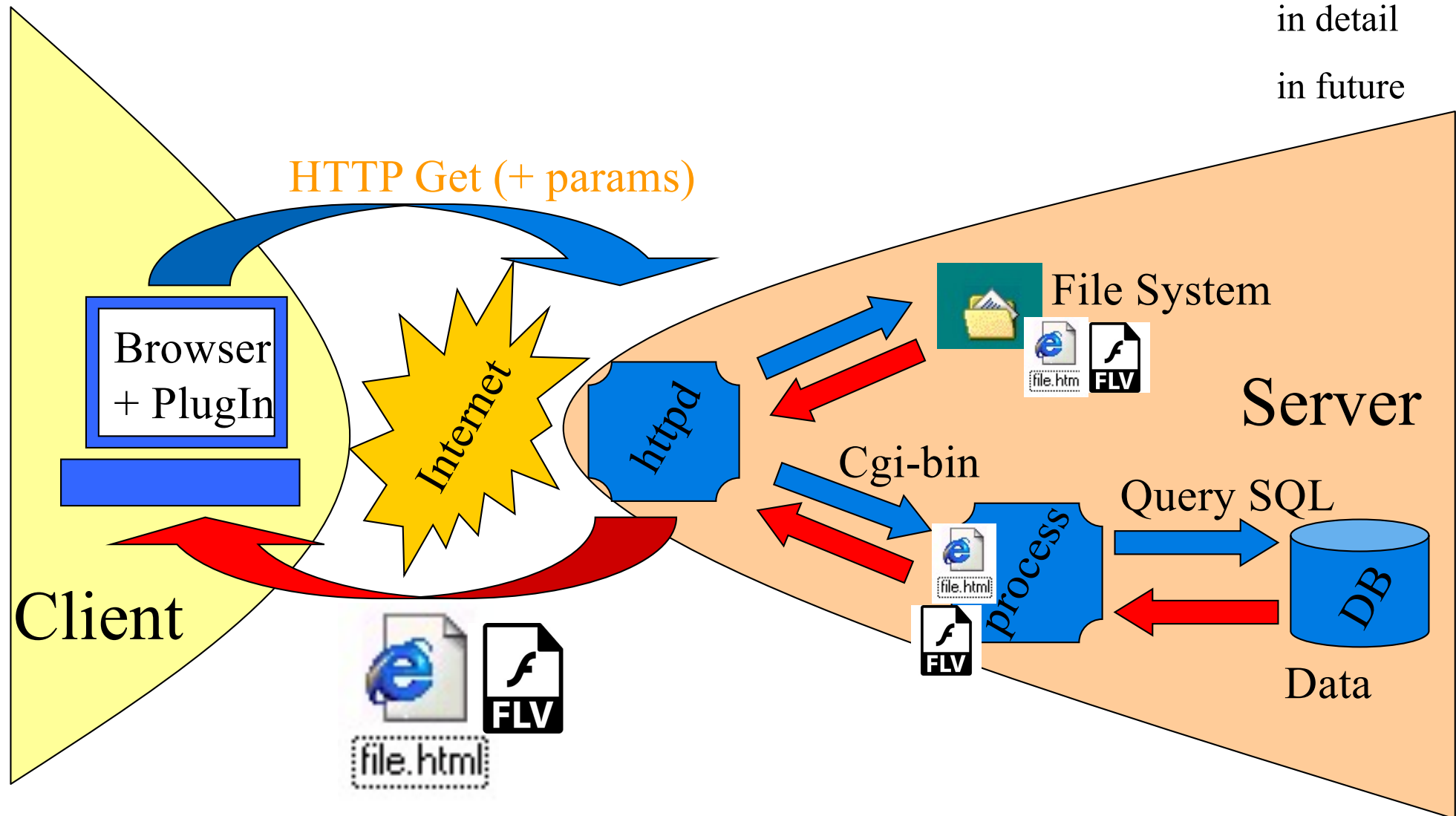


# JavaScript: History

| Name                     | Edition | Date published |              |        |                  |
|--------------------------|---------|----------------|--------------|--------|------------------|
| Mocha                    |         | May-95         |              |        |                  |
| LiveScript               |         | Sep-95         |              |        |                  |
| JavaScript               |         | Dec-95         |              |        |                  |
|                          |         |                | Jscript      | Aug-96 | Microsoft        |
| ECMAScript               | 1       | Jun-97         |              |        |                  |
| ECMAScript               | 2       | Jun-98         | ActionScript | 1998   | Macromedia/Adobe |
| ECMAScript               | 3       | Dec-99         |              |        |                  |
| ECMAScript               | 4       | Abandoned      |              |        |                  |
| ECMAScript               | 5       | Dec-09         |              |        |                  |
| ECMAScript               | 5,1     | Jun-11         |              |        |                  |
| ECMAScript 2015 (ES2015) | 6       | Jun-15         |              |        |                  |
| ECMAScript 2016 (ES2016) | 7       | Jun-16         |              |        |                  |
| ECMAScript 2017 (ES2017) | 8       | Jun-17         |              |        |                  |
| ECMAScript 2018 (ES2018) | 9       | Jun-18         |              |        |                  |
| ECMAScript 2019 (ES2019) | 10      | Jun-19         |              |        |                  |

## The web architecture: plug in

We will  
discuss it  
in detail  
in future



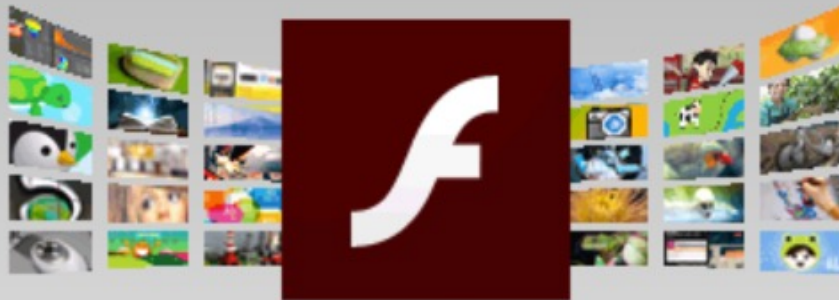
**Evolution 4: augment browser with an ad-hoc engine to be able to execute a (proprietary) language**



# Plug in: Adobe Flash (former Micromedia Flash)

FLV-Video-Media-Content played by the Adobe Flash OCX Plugin.

## Adobe Flash Player



### About:

Adobe® Flash® Player is a lightweight browser plug-in and rich Internet application runtime that delivers consistent and engaging user experiences, stunning audio/video playback, and exciting gameplay.

Installed on more than 1.3 billion systems, Flash Player is the standard for delivering high-impact, rich Web content.

# Plug in: Silverlight



SEARCH WITH BING



[Home](#) [About](#) [Features](#) [Partners](#) [Resources](#) [Forums](#) [Download](#)

Prepare for Silverlight 5 end of support after October 2021. [Learn more >](#)

## Get Silverlight 5

Silverlight is a powerful development tool for creating engaging, interactive user experiences for Web and mobile applications. Silverlight is a free plug-in, powered by the .NET framework and compatible with multiple browsers, devices and operating systems, bringing a new level of interactivity wherever the Web works.

DOWNLOAD NOW



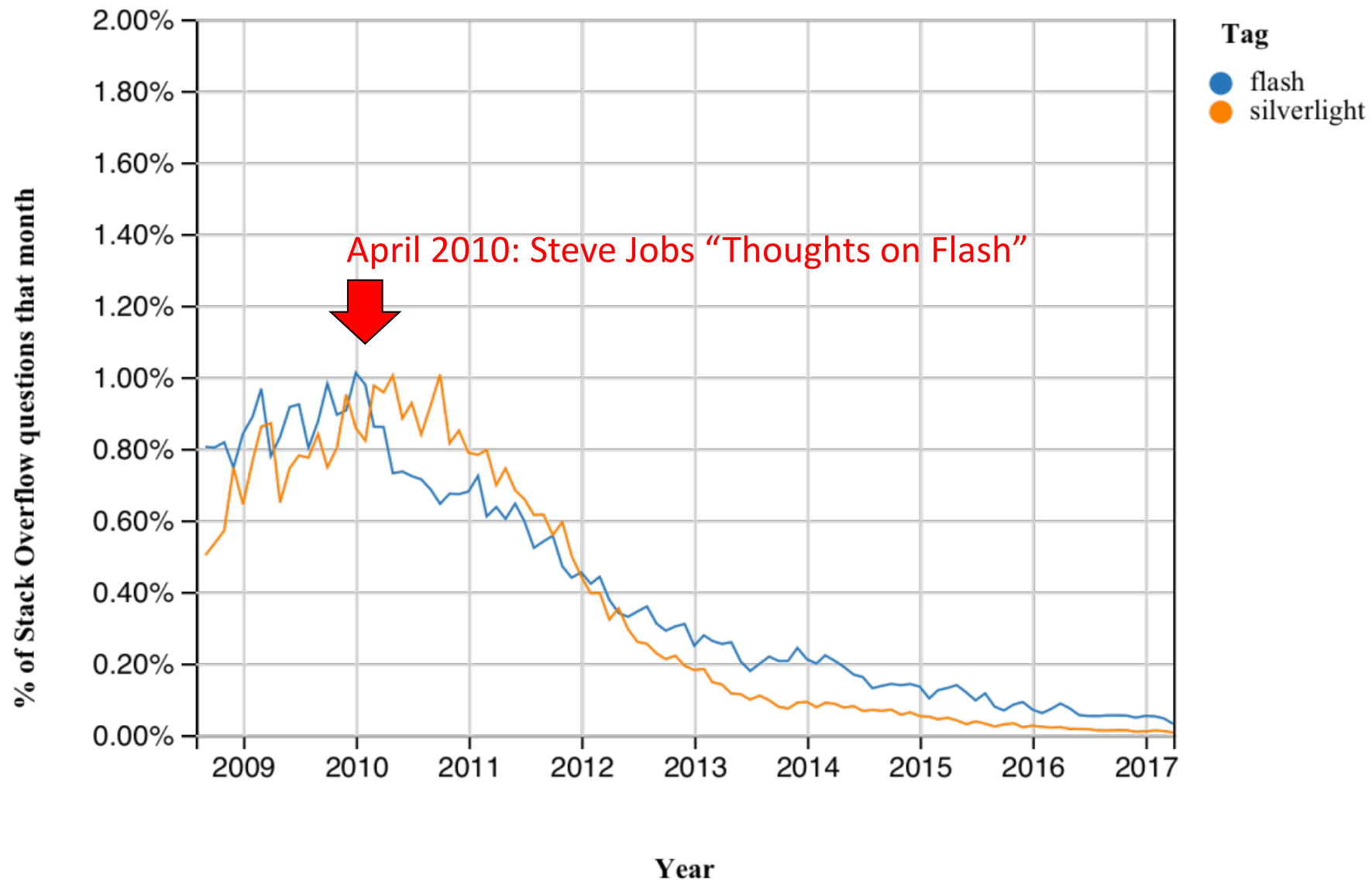
Silverlight 5  
now available

[Download now >](#)

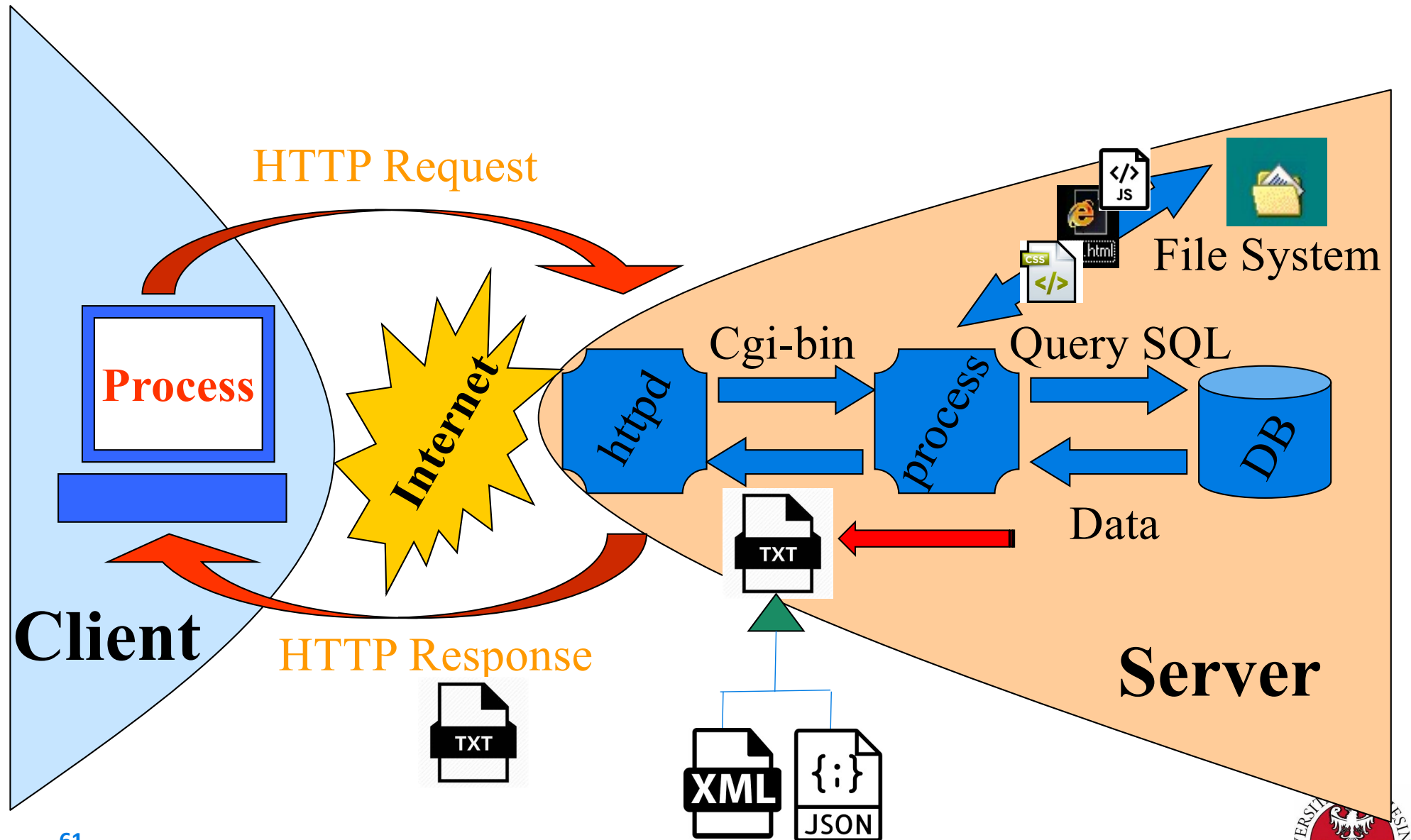


Get the Silverlight 5: A powerful tool for creating interactive web and mobile app

# Flash and Silverlight decline



# The web architecture: web services



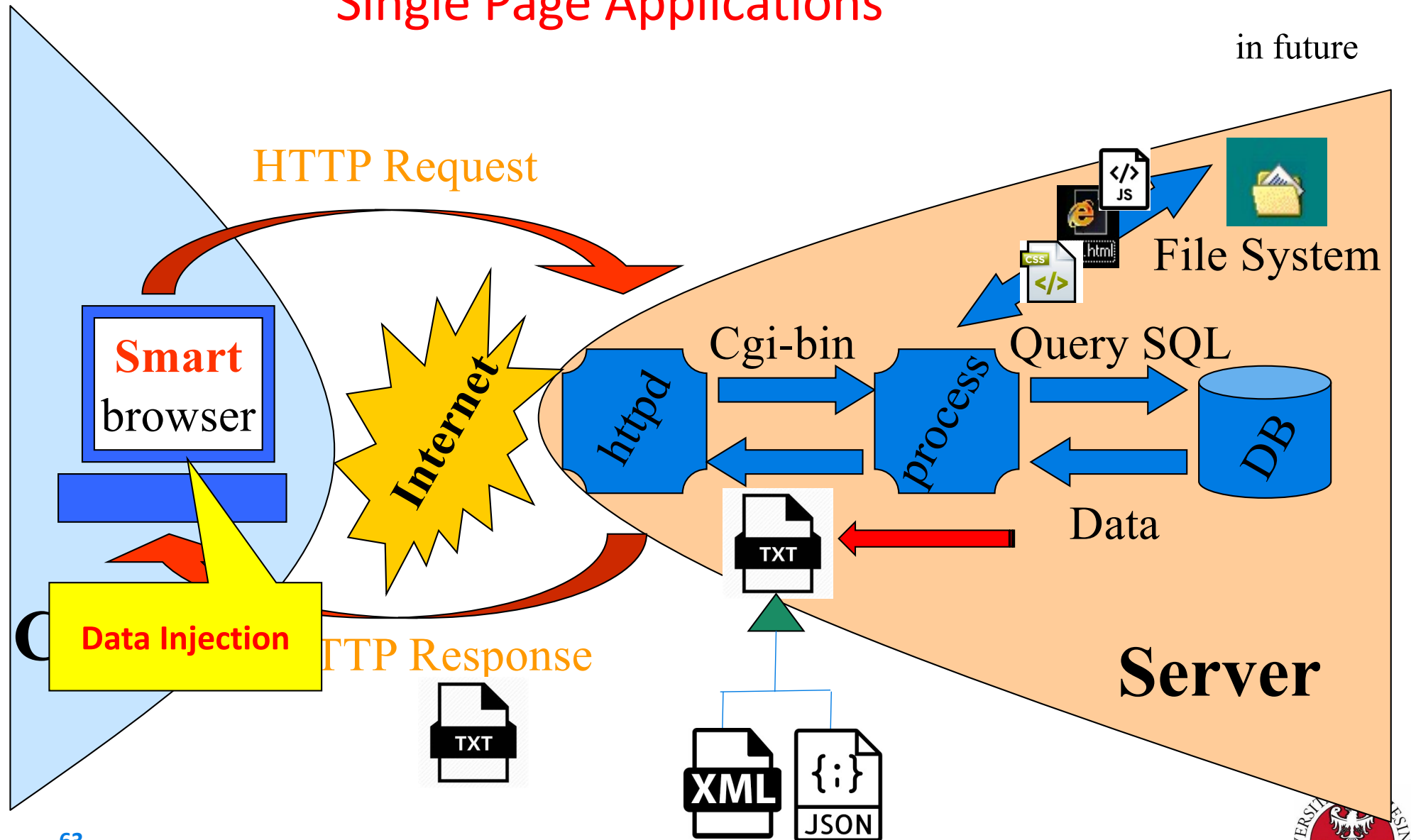
# The model

- A web service is a software system designed to support interoperable machine-to-machine interaction over a network.
- Certain URLs are associated with **actions** and carry **parameters** for the action
- The Web server:
  - understands that certain URLs are *dynamic*
  - parses the parameters
  - starts a process (or thread) corresponding to the desired action
  - obtains from the process some data (**for machine consumption**)
  - passed the data to the client

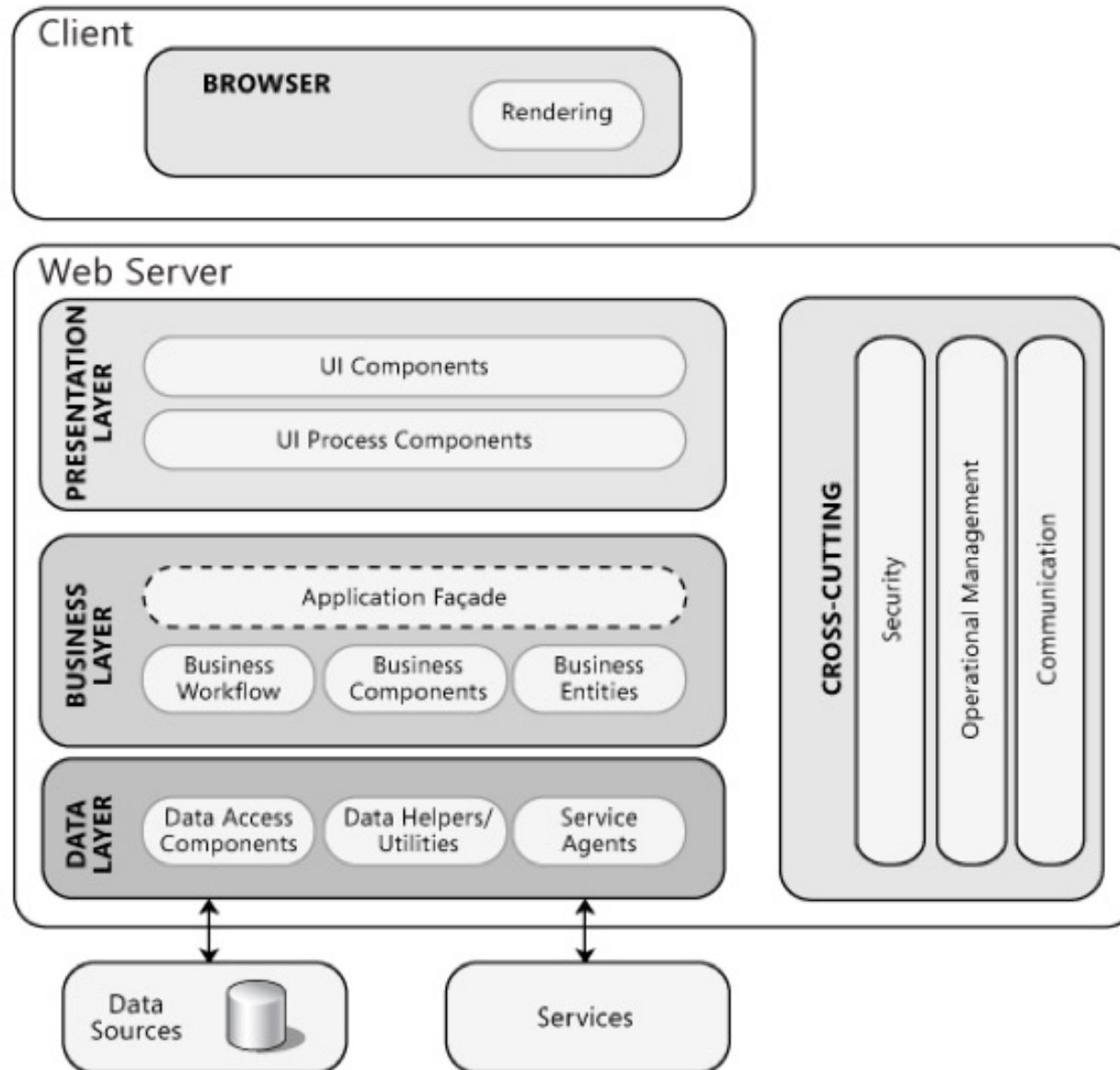


We will  
discuss it  
in detail  
in future

## Variant: Ajax processing and Single Page Applications



# Logical Web Architecture



# Advanced challenges



# Web development challenges

- Web applications support multiple users with (what appears to be) a single server. Developers have to deal with **concurrency and scalability**. Popular web applications must handle 100 – 1000× the workload of traditional applications.
- Web applications must be highly customizable. In the pre-Web GUI world a uniform look and feel was encouraged, but web applications strive for unique appearance and behaviors. This makes it more **difficult to create reusable components**.
- The public accessibility of web applications introduces a variety of security and privacy issues. It is easy for unaware developers to create **security loopholes** such as SQL injection attacks.



# Open issues at today's conferences

Making the Web Human-Centric? New Directions in the Web and AI  
12th International ACM Web Science Conference (July 2020)

Fairness, inclusion and diversity of Web and AI:

the construction of online identities, representation on the Web, access to the Web and technology, making 'smart' fair; Web culture and Web values.

Futures:

- the impact of the Web and new technologies on future society and social transformation,
- work futures and the data economy,
- health futures,
- political futures



# Open issues: Safety, Security & Trust:

- Safeguarding and governance of the Web and/or AI;
- Anonymity, security and trust for Web access;
- False news;
- Data for the public good;
- Crime on the Web;
- Ethical challenges of technologies, data, algorithms, platforms, and people.

# Open issues: Web & Contemporary Society

- Arts, culture and humanities on the Web;
- Web economics,
- Social entrepreneurship and innovation; health and online;
- Knowledge, education, and scholarship on/through the Web;
- Politics & political activism.



# Open issues: Techno-social Web

- Modeling Web data, users
- Detecting, preventing and predicting anomalies in Web data (e.g., fake content, spam, algorithmic and data );
- Analysis and modelling of human vs. (e.g., bots) and their influence on the structure of the Web and responding ;
- Social machines, crowd computing, collective intelligence, and collaborative production on the Web.