

## **Web Engineering**

# **History and Rendering Approaches**

Adrian Herzog

# A very short history of the web

- 1969: First data transmission in ARPANET (predecessor of today's internet)
- 1971: First specification for FTP (still ARPANET, no TCP/IP yet)
- 1980: Usenet is established (the original internet forum), SMTP protocol for e-mails is under development
- 1989/1990: Tim Berners-Lee builds the *first version of the World Wide Web* at CERN
  - He defines the basic concepts HTTP, HTML and URL
  - He writes the first web server and web browser
  - First web address: <https://info.cern.ch/>

# Publication of the web idea

1991: Announcing the World Wide Web in a newsgroup

The WorldWideWeb (WWW) project aims to allow links to be made to any information anywhere. The address format includes an access method (=namespace), and for most name spaces a hostname and some sort of path.

We also have code for a hypertext server. You can use this to make files available (like anonymous FTP but faster because it only uses one connection). You can also hack it to take a hypertext address and generate a virtual hypertext document from any other data you have - database, live data etc. It's just a question of generating plain text or SGML (ugh! but standard) mark-up on the fly. The browsers then parse it on the fly.

“hack it”  
→ dynamic  
websites

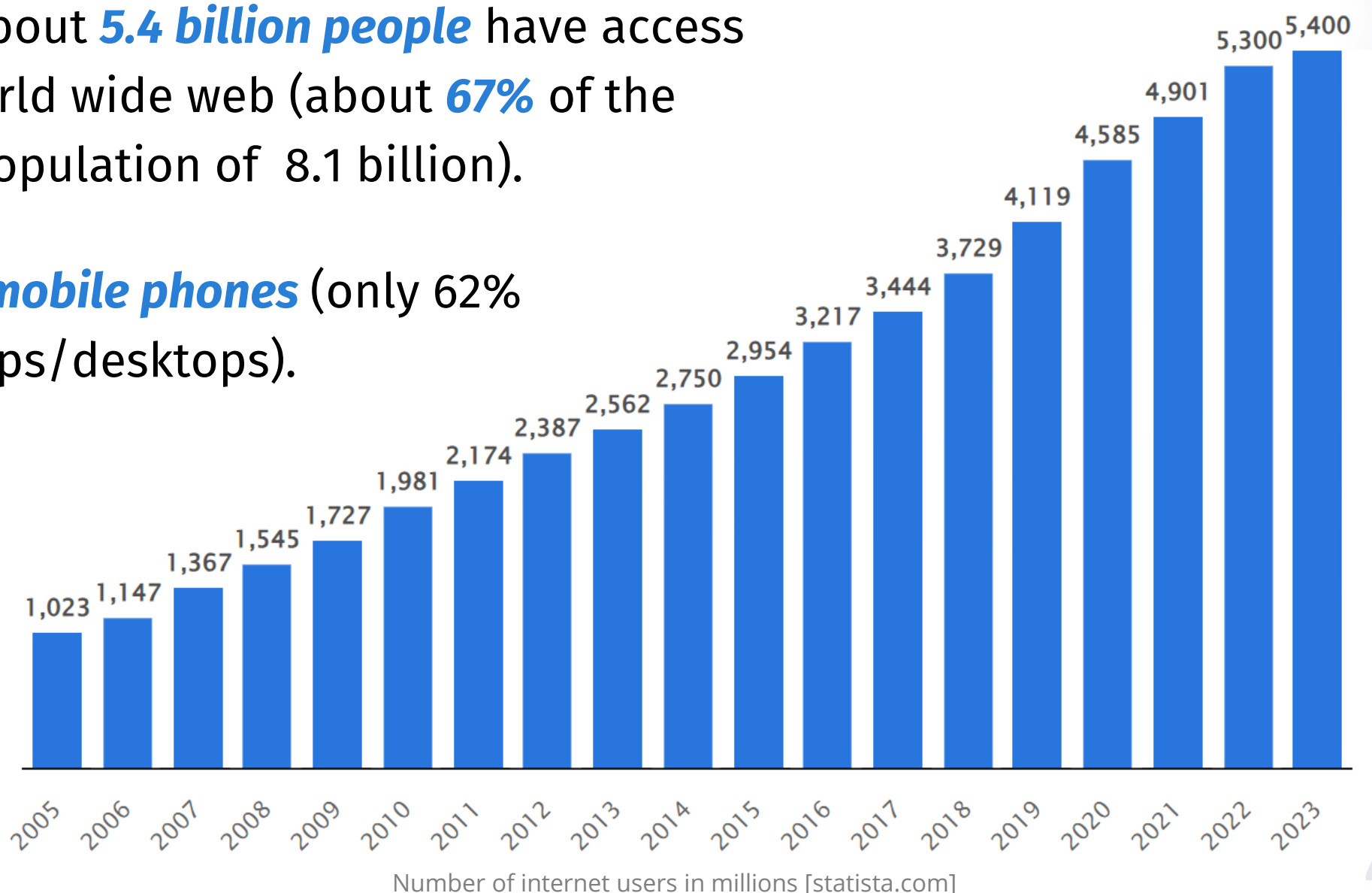
# Today

In 2024 about **5.4 billion people** have access to the world wide web (about **67%** of the world's population of 8.1 billion).

[statista.com]

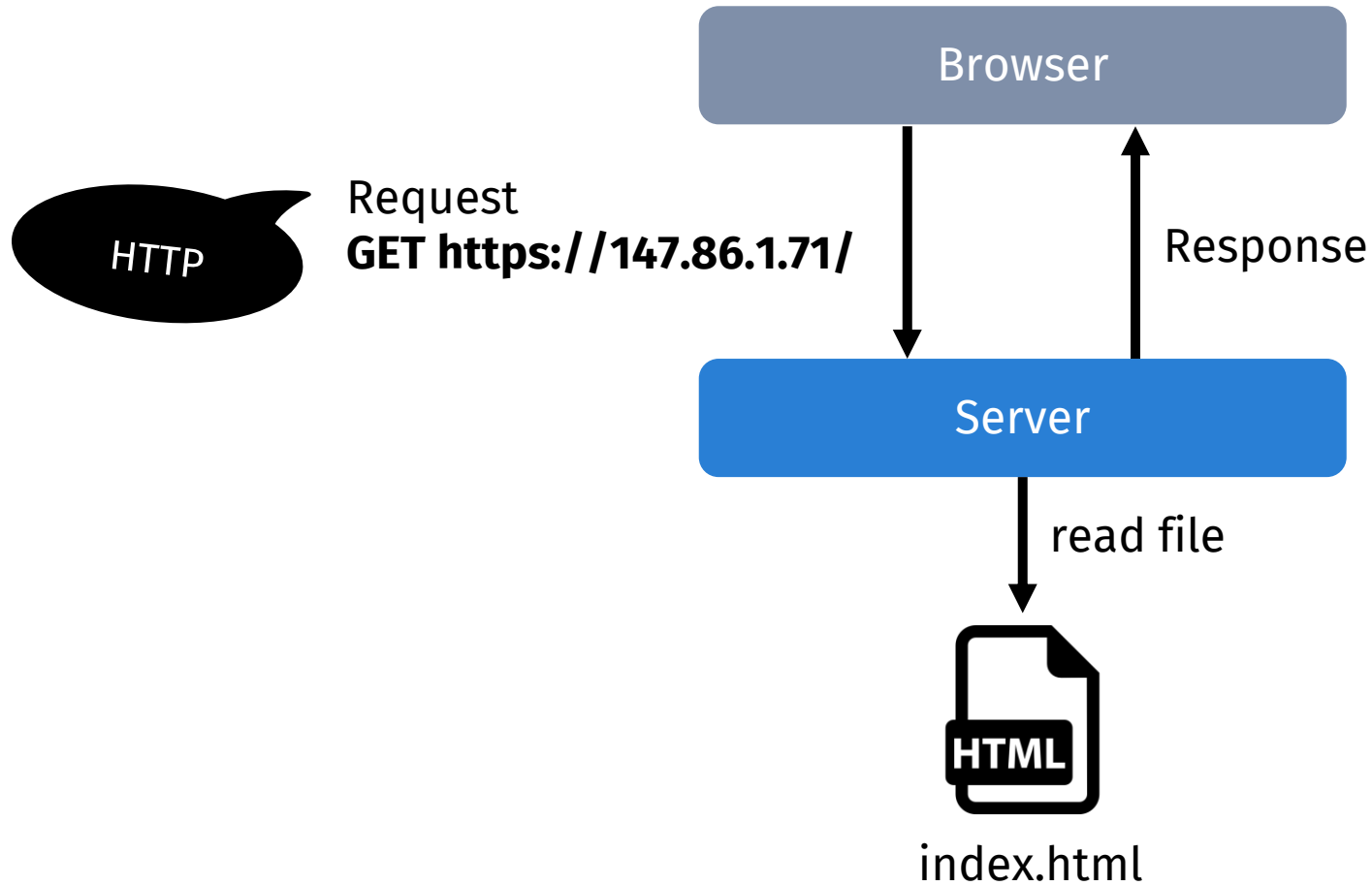
**95% use mobile phones** (only 62% use laptops/desktops).

[statista.com]

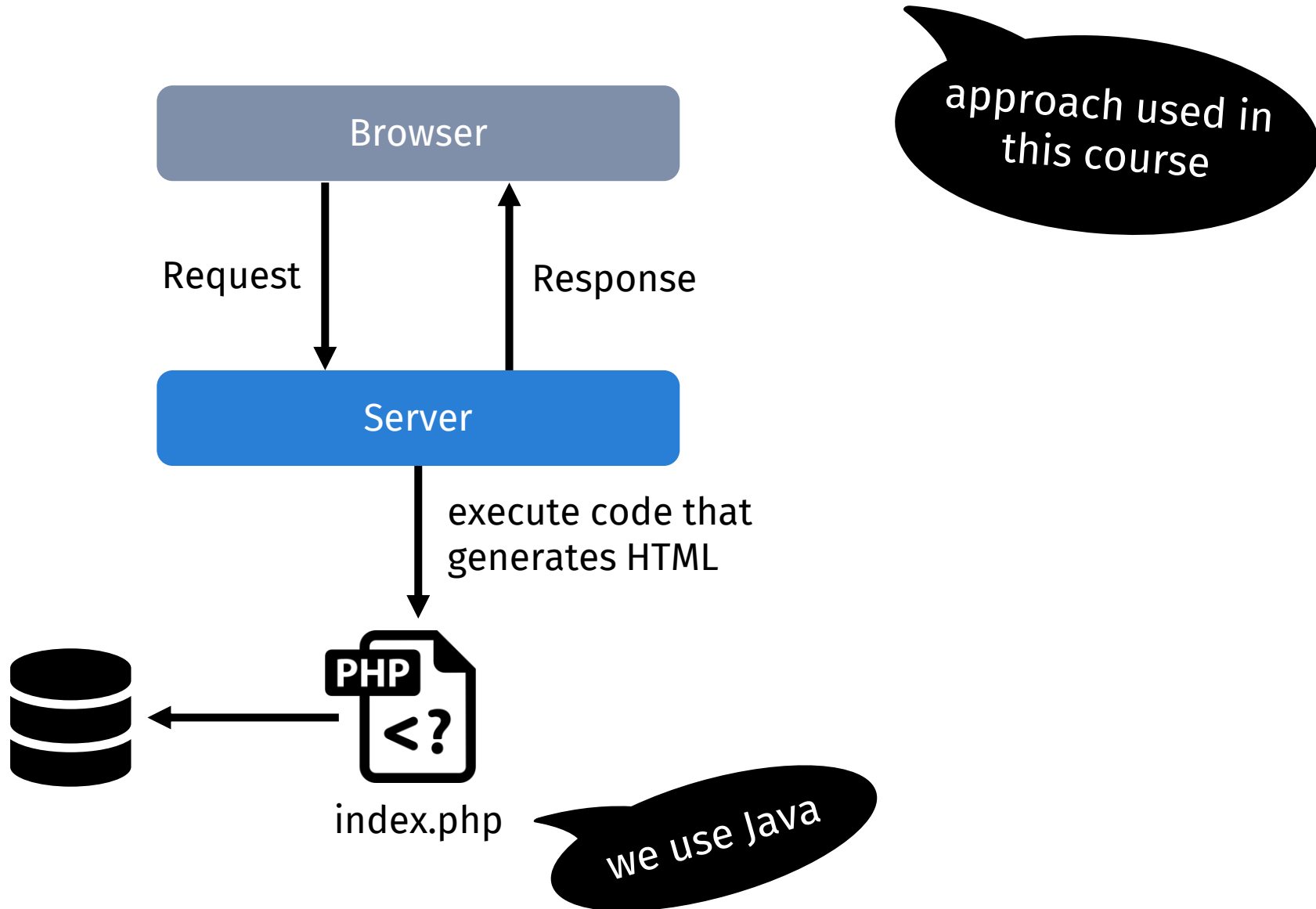




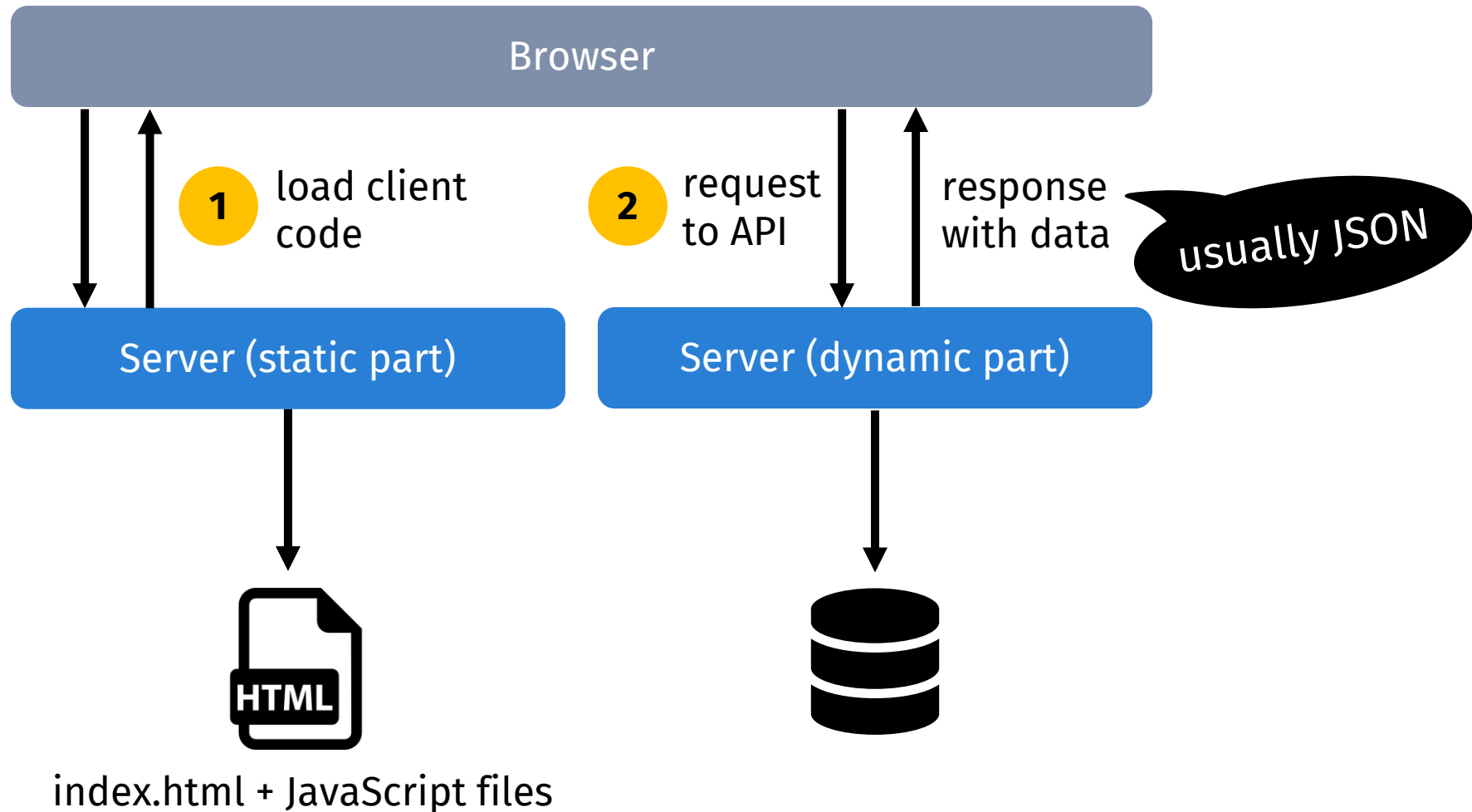
# Static Website



# Server Side Rendering



# Single Page Application





# Further approaches

- Hybrid rendering
  - Use same templating system on frontend and backend
  - Initial page rendered on server, then the client takes over
  - Supported by some SPA frameworks
- Hypermedia-Driven Applications
  - Like server side rendering
  - A small JS library allows replacing just parts of the document with new HTML content from the server
- ...

# What should I use?

it depends

## *Server Side Rendering*

- Easily indexable by search engines
- Faster initial page load
- Lower complexity
- Suitable for content driven applications with low interactivity (news pages, blogs, web shops, etc.)

## *SPA (Client Side Rendering)*

- Rich user experience comparable to a native application possible
- Suitable for highly interactive applications
- Harder to optimize for search engines

# Questions?

