

# Pemrograman Web Internet, WWW, dan HTTP

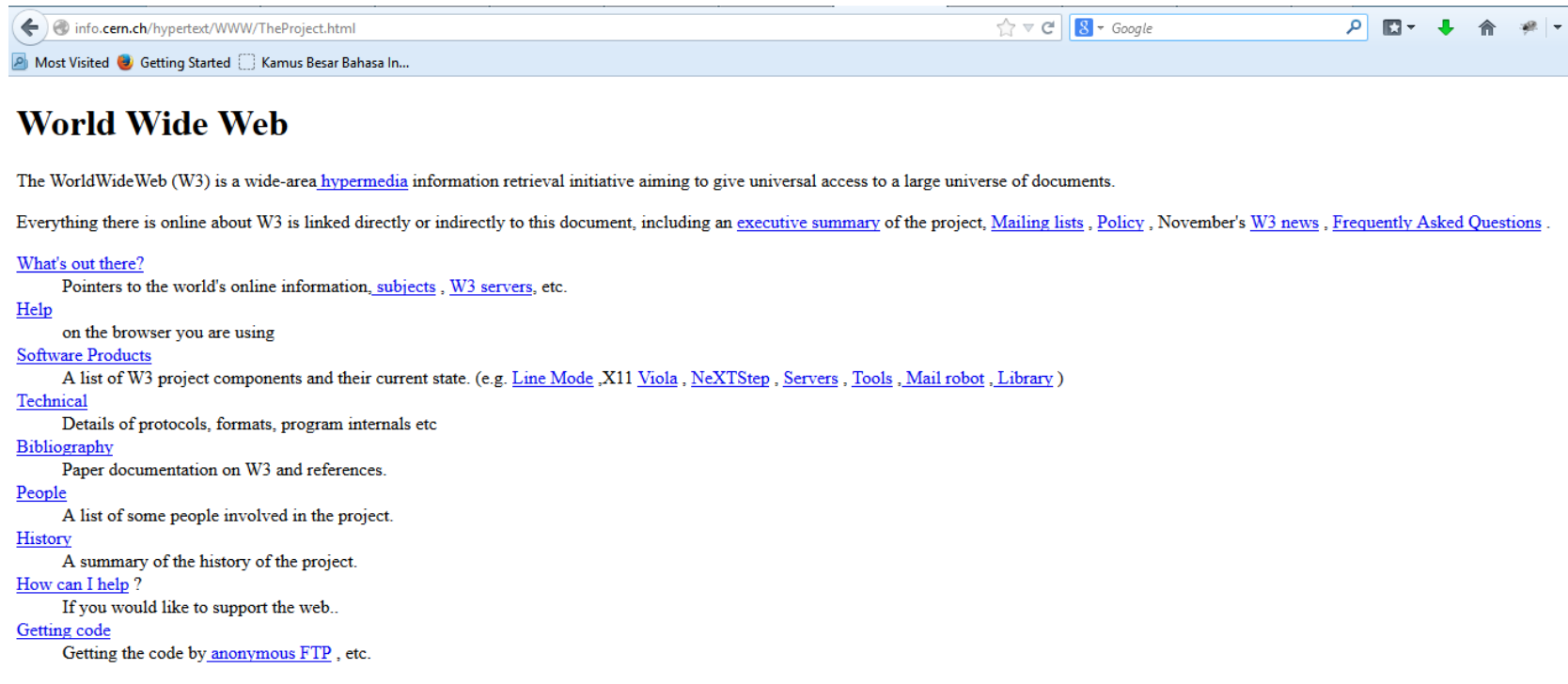
Pertemuan ke-1

# Outline

- Pengertian Internet & WWW
- Sejarah Internet & WWW
- Perkembangan teknologi web
- Komputasi client-server
- Protokol dalam Internet
- Protokol HTTP (*request dan response*)

# The Very First Web

<http://info.cern.ch/hypertext/WWW/TheProject.html>



# Apa itu Web/Internet?



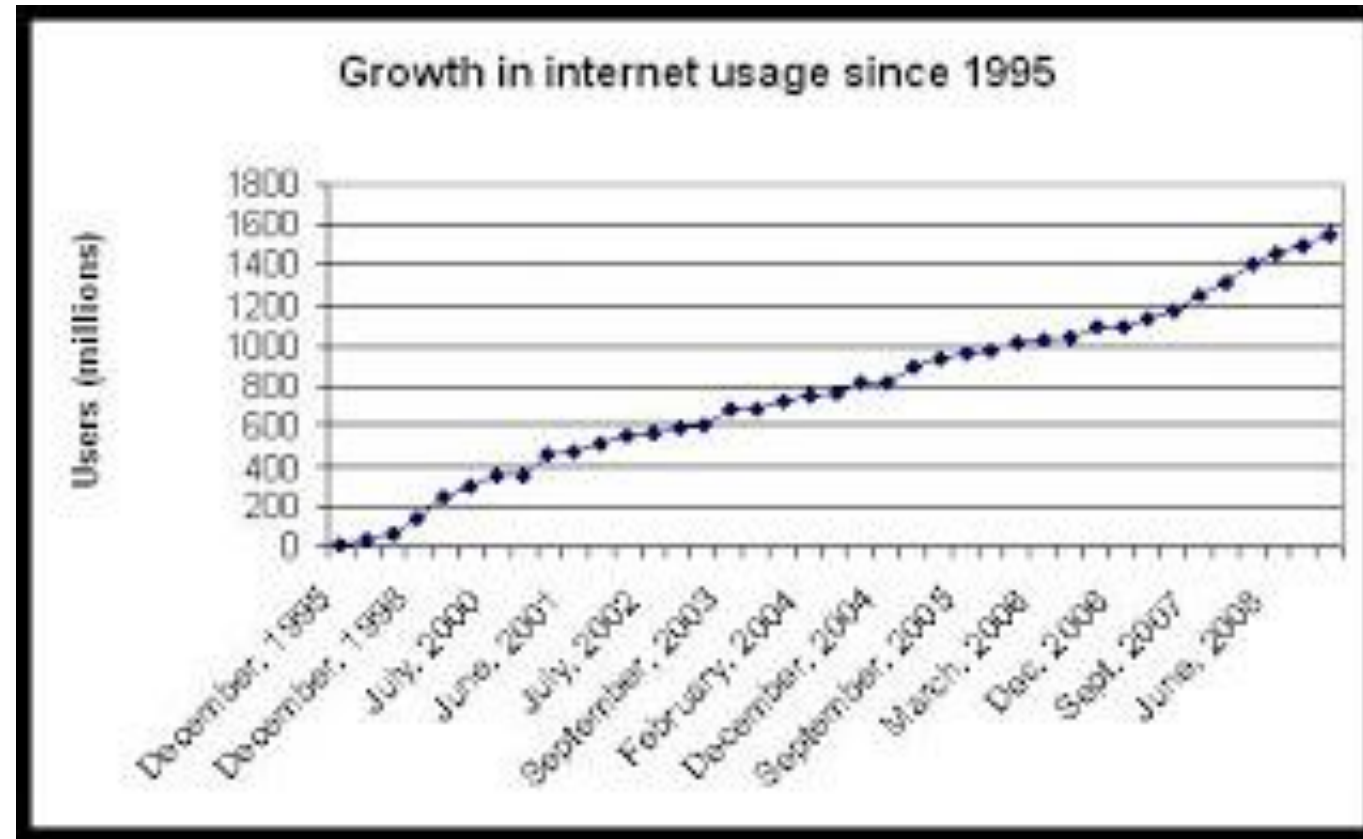
# Sejarah Internet

- 1970 – ARPANET Project
- 1979 – TCP/IP research
- 1980 – IPv4
- 1983 – semua komputer di jaringan ARPANET menggunakan TCP/IP
- 1985 – Proyek National Science Foundation Network (NSFNET)
- 1992 – World Wide Web (WWW) mulai diperkenalkan

# Internet

- Sistem global dari jaringan komputer yang saling terhubung yang menggunakan standar Internet Protocol (TCP/IP) untuk melayani jutaan pengguna di seluruh dunia.
- Sistem informasi global
  - secara logika saling terhubung dengan menggunakan alamat unik yang global berdasarkan Internet Protocol (IP);
  - Mampu mendukung komunikasi menggunakan rangkaian Transmission Control Protocol / Internet Protocol (TCP/IP) dan protokol-protokol lain cocok dengan IP;
  - menyediakan, baik secara publik maupun privat, *high lever services* berdasarkan pada komunikasi dan infrastruktur terkait.

# Perkembangan Pengguna Internet



# Cara Kerja Internet

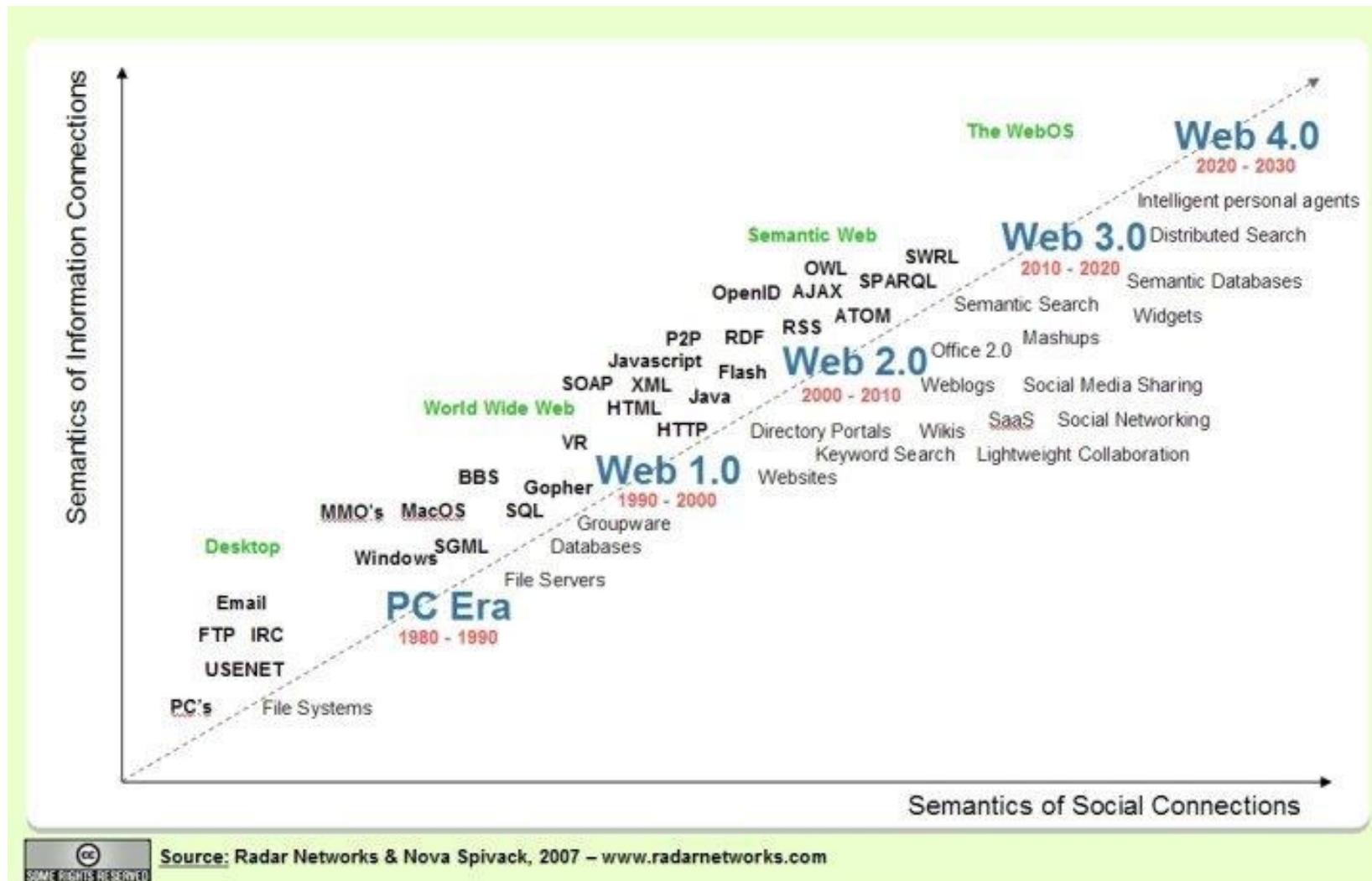
---

**How does the internet work ?**





# Perkembangan Teknologi Web



# Web 1.0 vs. Web 2.0



# Web 3.0



# Perbedaan Web 1.0 dan Web 2.0

## Web 1.0

1. *Brochure web*
2. Owning content
3. About business
4. Static
5. HTML, portals
6. 45 million global user (1996)
7. Directories (taxonomy)

## Web 2.0

1. *Conversation web*
2. Sharing content
3. About community
4. Dynamic
5. RSS, XML
6. 1 billion + global user (2006)
7. Tagging (folksonomy)

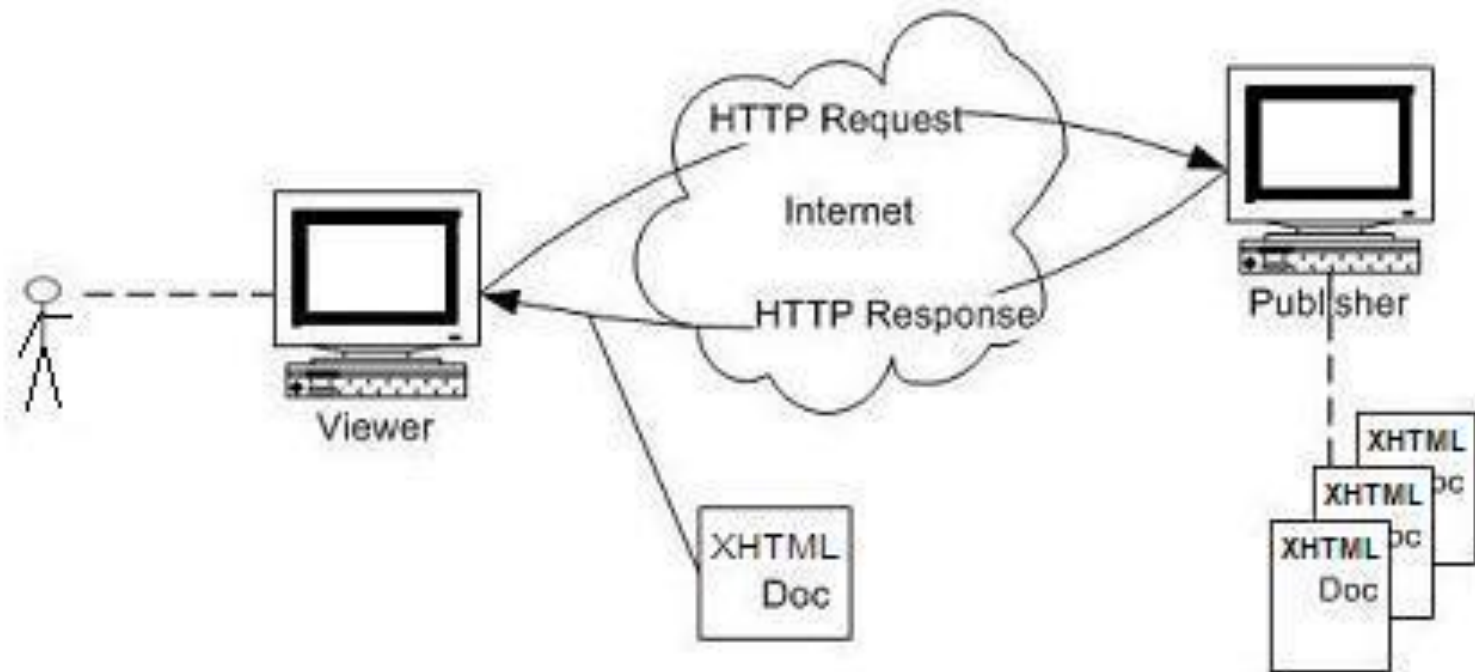
# World Wide Web (WWW)

- Definisi
  - Suatu sistem yang saling menghubungkan dokumen-dokumen hypertext yang diakses melalui internet
- World Wide Web Consortium (W3C)
  - Organisasi yang membentuk standard web
    - termasuk Extensible Hypertext Markup Language (XHTML), Cascading Style Sheets (CSS), dan Extensible Markup Language (XML)
  - Diorganisir oleh Laboratory for Computer Science di Massachusetts Institute of Technology (MIT)

# *Protocol*

- TCP (Transmission Control Protocol)
  - Protocol untuk berkomunikasi melalui internet
- IP (Internet Protocol)
  - Protocol untuk inter-komunikasi antar platform-platform yang berbeda
- HTTP (Hypertext Transfer Protocol)
  - Protokol komunikasi antara server dan client yang ditujukan untuk pengiriman file

# Cara Kerja HTTP



By FYIcenter.com

# Cara Kerja HTTP

*“connect a server and transfer a file!”*

- Ketika user meng-klik sebuah hypertext link, user tersebut mengirimkan URL ke browsernya. Melalui URL, browser bisa mengenali server manakah yang harus dihubungi dan file apa sajakah yang harus diminta untuk dikirimkan oleh server.
- Skema:
  1. Client melakukan koneksi ke host
  2. Server menerima koneksi dari client
  3. Client melakukan request file
  4. Server mengirimkan respon (termasuk file maupun bukan, bila kasusnya file tidak ditemukan)



# HTTP *Requests*

- GET
- POST
- PUT
- DELETE

# HTTP *Requests*: GET

- **GET** – request data dari resource yang ditentukan

```
/test/demo_form.asp?name1=value1&name2=value2
```

- Karakteristik:
  - Dapat di-cache
  - Dapat tertinggal di browser history
  - Dapat di-bookmark
  - Seharusnya tidak digunakan untuk mengirimkan data yang sensitif
  - Memiliki batasan panjang string
  - Sebaiknya hanya digunakan untuk *data retrieval* saja

# HTTP *Requests*: POST

- **POST** – mengirimkan data untuk diproses oleh resource yang dituju

```
POST /test/demo_form.asp HTTP/1.1  
Host: w3schools.com  
name1=value1&name2=value2
```

- Catatan:
  - Tidak bisa di-cache
  - Tidak tertinggal di browser history
  - Tidak bisa di-bookmark
  - Tidak memiliki batasan panjang data

# Perbandingan GET vs. POST

|                            | GET   | POST  |
|----------------------------|---|---|
| BACK button/ Reload        | harmless  | data will be re-submitted                   |
| Bookmarked                 | Can be bookmarked   | Cannot be bookmarked                        |
| Cached                     | Can be cached   | Not cached                                  |
| History                    | Parameters remain in browser history  | Parameters are not saved in browser history |
| Restriction on data length | Yes, when sending data, the GET method adds the data to the URL; and the length of a URL is limited (max URL length is 2048 characters) | No restrictions                             |

# Perbandingan GET vs. POST (lanj.)

|                          | GET   | POST   |
|--------------------------|---|--|
| Restriction on data type | Only ASCII characters allowed   | No restrictions, binary data is also allowed   |
| Security                 | GET is less secure compared to POST because data sent is part of the URL.<br>Never use GET when sending passwords or other sensitive information! | POST is a little safer than GET because the parameters are not stored in browser history or in web server logs |
| Visibility               | Data is visible to everyone in the URL  | Data is not displayed  |

# HTTP *Requests*: PUT

- **PUT** – mengunggah sebuah representasi (bisa file maupun resource) ke URI yang dituju
- Jika URI merujuk ke resource yang sudah ada di server, maka PUT akan melakukan rewrite/modify. Bila resource belum ada, maka melakukan create.

# HTTP *Requests*: DELETE

- **DELETE** – menghapus resource tertentu