

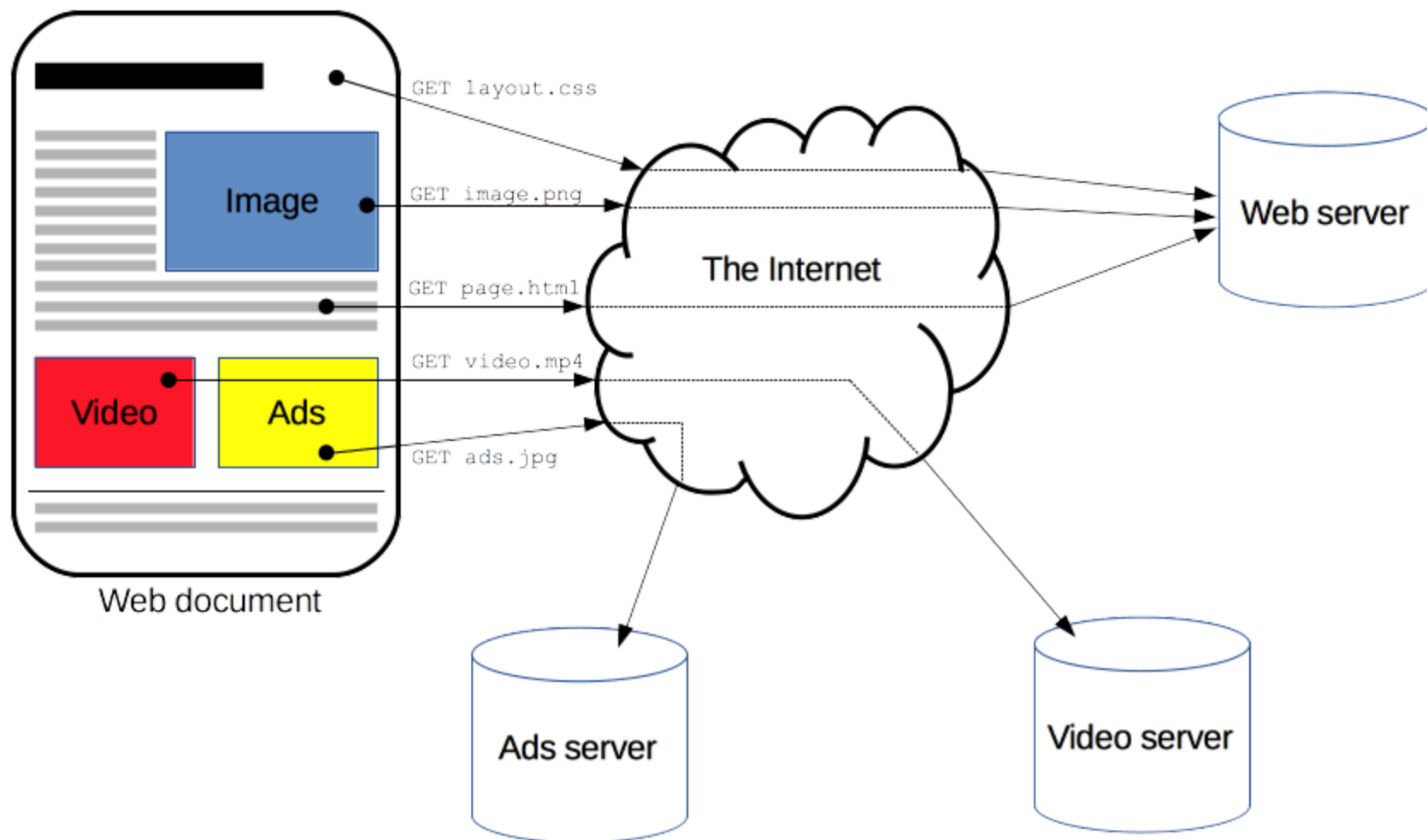
Programowanie Aplikacji Internetowych

API / Komunikacja między serwisami

Plan na dziś

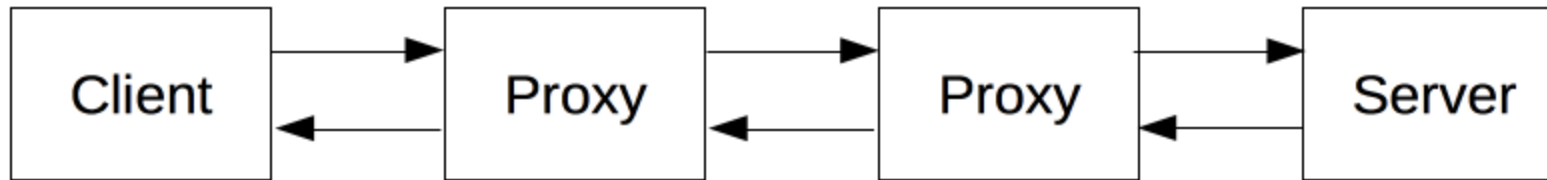
- http
- RPC
- REST
- GraphQL

HTTP



HTTP

Cała infrastruktura przystosowana do pracy z http:



HTTP

Demo:

```
curl -I www.google.com
```

```
curl -I -L google.com
```

HTTP - methods

Methods:

- GET
- POST
- DELETE

HTTP - methods

Demo:

```
http POST https://httpbin.org/post "name"="natalia"
```

HTTP - status code

Status code:

- 5xx: 500, 502
- 4xx: 404, 400, 401
- 3xx: 301, 302
- 2xx: 200, 201, 02

WebSockets

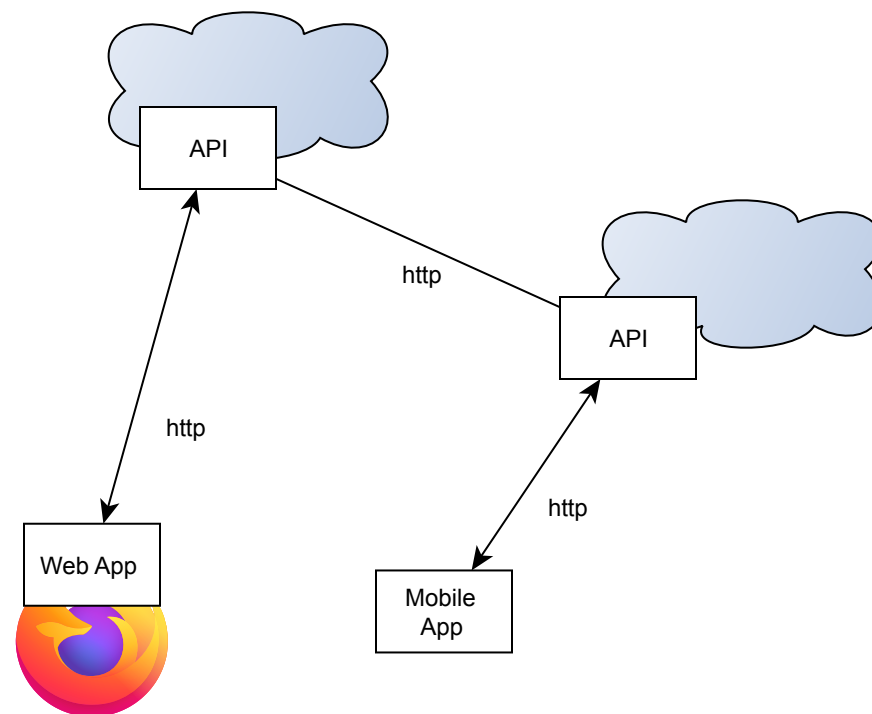
- dwu-stronnej szybkiej komunikacji
- Alternatywa dla *long polling*

Więcej później o websocketach później.

A co z serwisami?

- +/- Wiemy jak działają przeglądarki
- co z serwisami?

A co z serwisami?



Protokoły

Najpopularniejsze:

- (web) RPC
- REST API
- GraphQL

(web) RPC

- RPC (remote procedure call)
- po prostu wywołanie zewnętrznej funkcji

(web) RPC



REST API

- Most popular

REST API

- Most popular
- [json API](#)
- RPC
- REST / almost-REST
- GraphQL

Dziękuję za uwagę

Backup slides

Protokoły

Najpopularniejsze:

- RPC
- REST / almost-REST
- GraphQL

3-tier architecture

- Frontend
- backend
- baza danych

Jak hostować?

- PaaS: [vercel](#), [netify](#), [heroku](#);
- CaaS (AWS EKS, GCP) - container-as-a-service
- XaaS (AWS, GCP):
 - IaaS