

Introduzione al protocollo HTTP

Corso Backend System Integrator
Modulo **Programmazione PHP**

Docente: Dott. Enrico Zimuel

in collaborazione con:



REGIONE
PIEMONTE

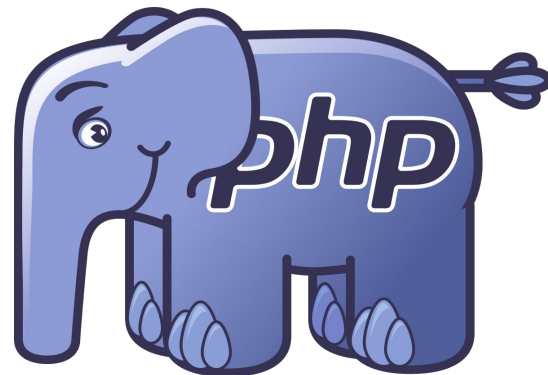
per una crescita intelligente,
sostenibile ed inclusiva

www.regione.piemonte.it/europa2020

INIZIATIVA CO-FINANZIATA CON FSE

Programma

- Protocollo HTTP
- Messaggi HTTP
- Richiesta
- Metodi, URI, Header, Body
- Risposta
- Status code
- HTTP, cloud e Web API
- Formato JSON
- HTTPS
- Cenni su HTTP/2 e HTTP/3



Protocollo HTTP

- L'**HyperText Transfer Protocol (HTTP)** è un protocollo a livello applicativo usato come principale sistema per la trasmissione d'informazioni sul web, in un'architettura tipica client-server



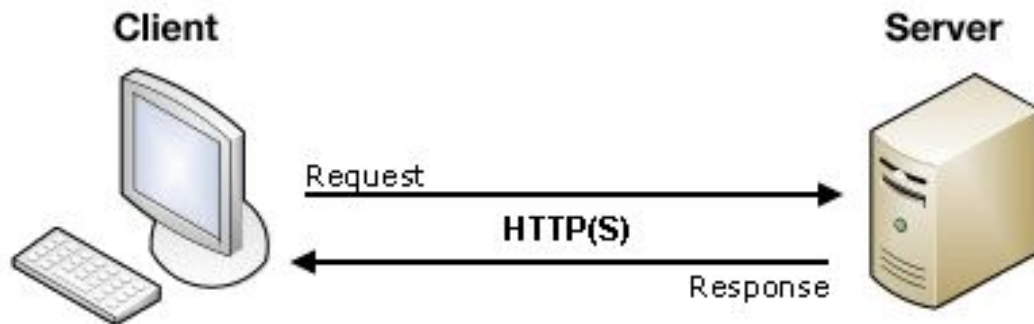
W3C

- Le specifiche del protocollo sono gestite dal [World Wide Web Consortium](https://www.w3.org/) (W3C)

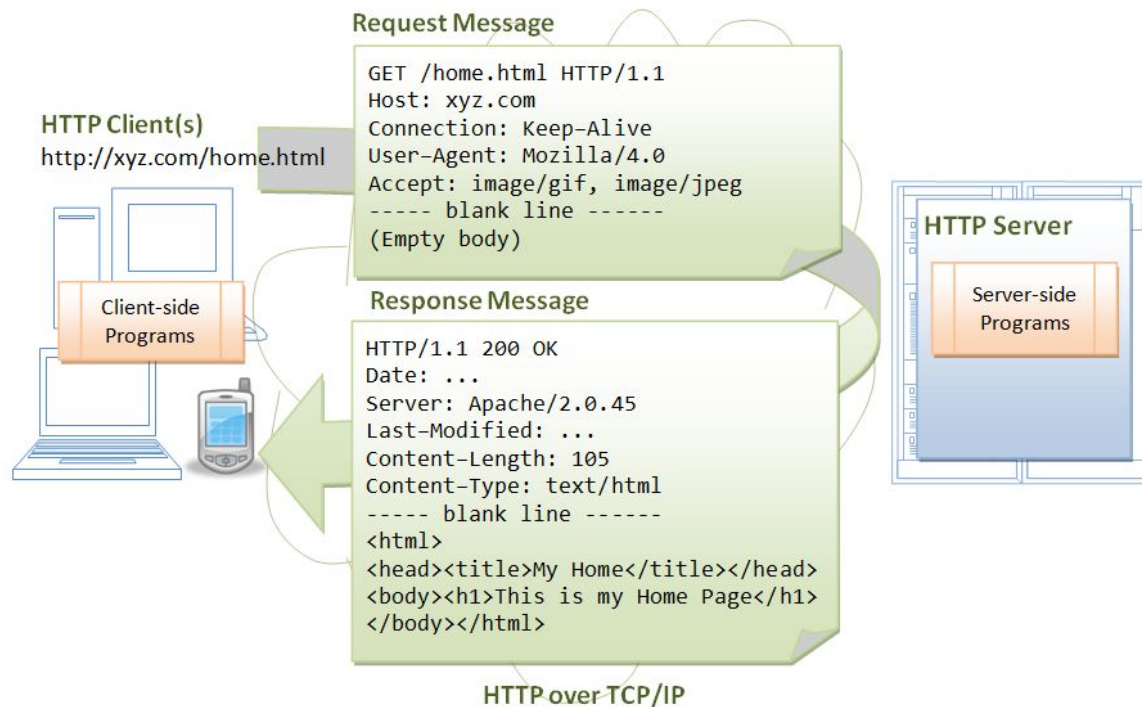


World Wide Web Consortium

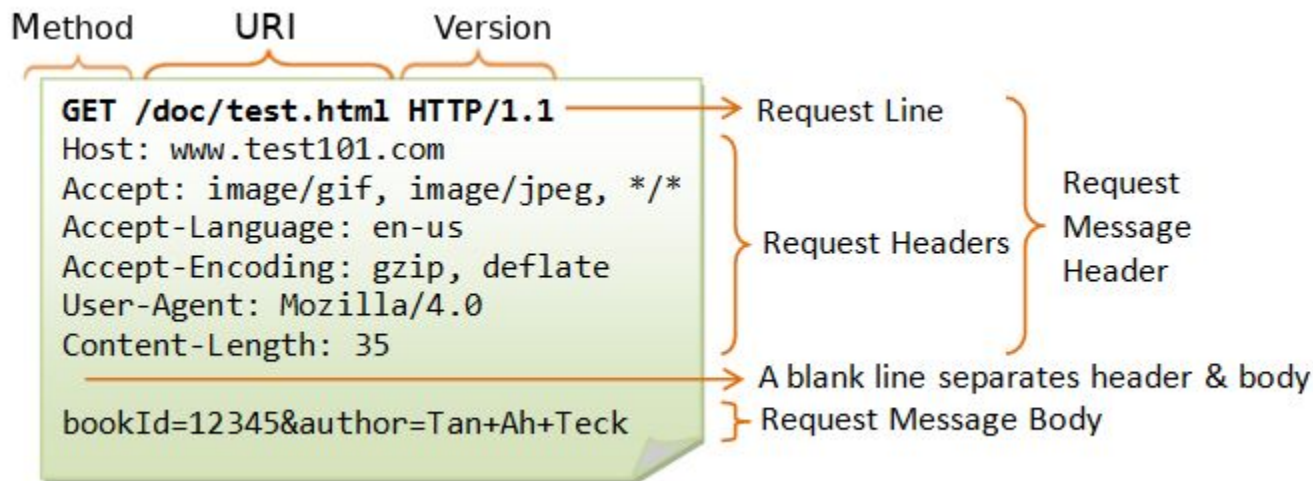
Client-server



Messaggi HTTP



Richiesta HTTP



Elementi della richiesta

- **Method:** GET, HEAD, POST, PUT, PATCH, DELETE, OPTIONS
- **Uniform Resource Identifier (URI):** una sequenza di caratteri che identifica univocamente una risorsa
- **Version:** versione del protocollo HTTP 1.0, 1.1, 2.0, 3.0 (draft)
- **Headers:** coppie di chiave: valore
- **Body:** Corpo del messaggio

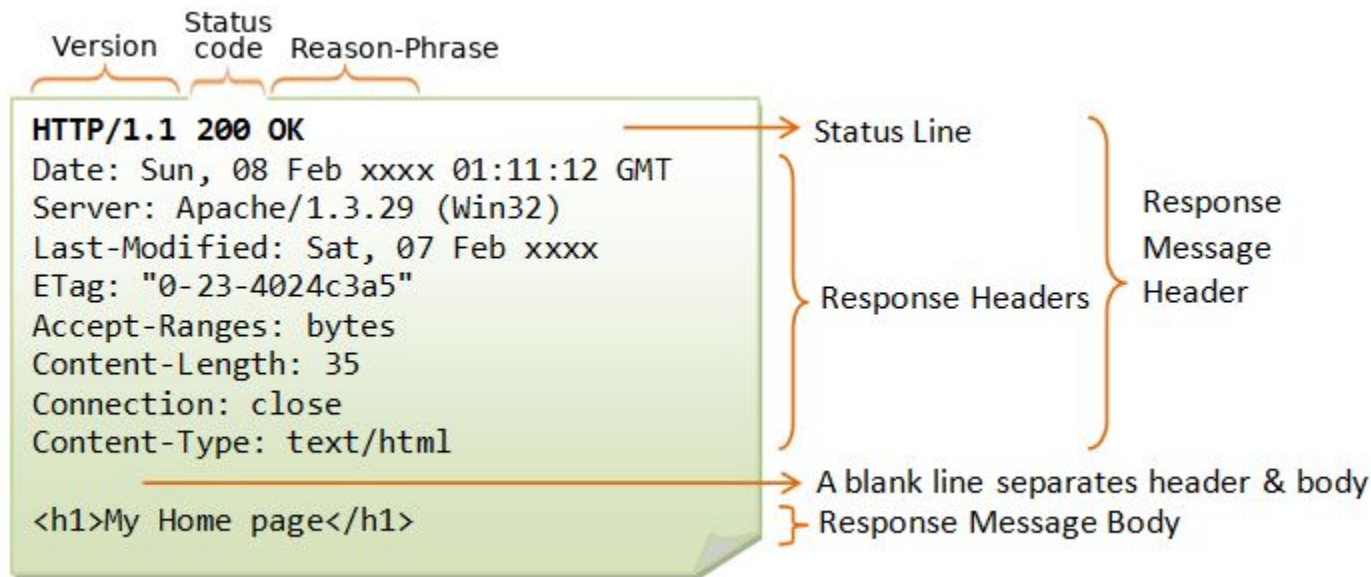
Metodi HTTP

- **GET**: restituisce una risorsa
- **HEAD**: come GET ma senza body nella risposta
- **POST**: creare una nuova risorsa
- **PUT**: sostituire una risorsa
- **PATCH**: modificare una risorsa
- **DELETE**: eliminare una risorsa
- **OPTIONS**: informazioni su una risorsa

Header

- Coppie di **chiave** e **valore**, separati da due punti (:) e disposte su più righe
- Esempi:
 - Accept: text/plain
 - Forwarded: for=192.0.2.43, for=198.51.100.17
 - Authorization: Basic QWxhZGRpbjpvcG==
 - Accept: text/*;q=0.3, text/html;q=0.7

Risposta HTTP



Status code

- **Informational 1xx:** 100 Continue
- **Successful 2xx:** 200 OK, 201 Created, 202 Accepted
- **Redirection 3xx:** 301 Moved Permanently, 307 Temporary Redirect
- **Client Error 4xx:** 401 Unauthorized, 404 Not Found, 405 Method Not Allowed
- **Server Error 5xx:** 500 Internal Server Error, 503 Service Unavailable

Lista completa degli Status-code su [RFC 2616](#)

HTTP e cloud

- La maggior parte dei servizi cloud è erogata tramite servizi HTTP, utilizzando delle web API
- Alcuni cloud vendor offrono anche [SDK](#) per diversi linguaggi di programmazione

Web API

- **Application programming interface (API):** insieme di procedure disponibili al programmatore per l'esecuzione di funzionalità specifiche di una libreria o servizio di terze parti
- **Web API:** API erogate tramite HTTP per l'esecuzione di servizi web

SDK

- **Software Development Kit (SDK)** sono un insieme di strumenti e documenti per lo sviluppo di applicazioni tramite servizi di terze parti

Interscambio di dati

- Nelle web API la maggior parte delle volte i dati vengono scambiati nel formato JSON
- **JavaScript Object Notation (JSON)** è un semplice formato per lo scambio di dati
- Si basa su un sottoinsieme del Linguaggio di programmazione **JavaScript** (ECMA-262, III edizione)
- E' basato su insiemi di coppie nome/valore e su elenchi ordinati di valori (array)

Esempio di JSON

```
{  
  "person" : {  
    "name" : "Alan",  
    "surname" : "Turing",  
    "age" : 42  
  }  
}
```

Esempio di JSON (2)

```
{  
  "contacts" : [  
    {  
      "name" : "Alan",  
      "surname" : "Turing",  
      "age" : 42  
    },  
    {  
      "name" : "John",  
      "surname" : "von Neumann",  
      "age" : 54  
    }  
  ]  
}
```

JSON e Javascript

- E' possibile assegnare un oggetto JSON direttamente a una variabile in Javascript:

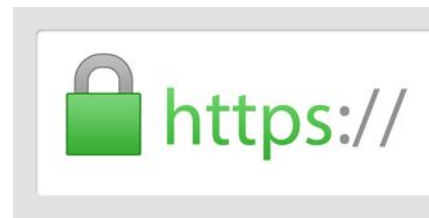
```
var obj = {  
  "person" : {  
    "name" : "Alan",  
    "surname" : "Turing",  
    "age" : 42  
  }  
};  
  
console.log(obj.person.name); // Alan  
console.log(obj.person.surname); // Turing  
console.log(obj.person.age); // 42
```

JSON e Javascript (2)

```
var obj = {  
  "contacts" : [  
    {  
      "name" : "Alan",  
      "surname" : "Turing",  
      "age" : 42  
    },  
    {  
      "name" : "John",  
      "surname" : "von Neumann",  
      "age" : 54  
    }  
  ]  
};  
  
console.log(obj.contacts[0].name); // Alan  
console.log(obj.contacts[1].name); // John
```

HTTPS

- L'**HyperText Transfer Protocol over Secure Socket Layer (HTTPS)**, è un protocollo per la comunicazione sicura attraverso una rete di computer utilizzato su Internet
- Utilizza il protocollo HTTP all'interno di una connessione criptata, tramite crittografia asimmetrica, dal **Transport Layer Security (TLS)** o dal suo predecessore, **Secure Sockets Layer (SSL)** fornendo come requisiti chiave
 - un'**autenticazione** del sito web visitato;
 - **protezione** della privacy;
 - **integrità** dei dati scambiati tra le parti comunicanti.



Come funziona HTTPS



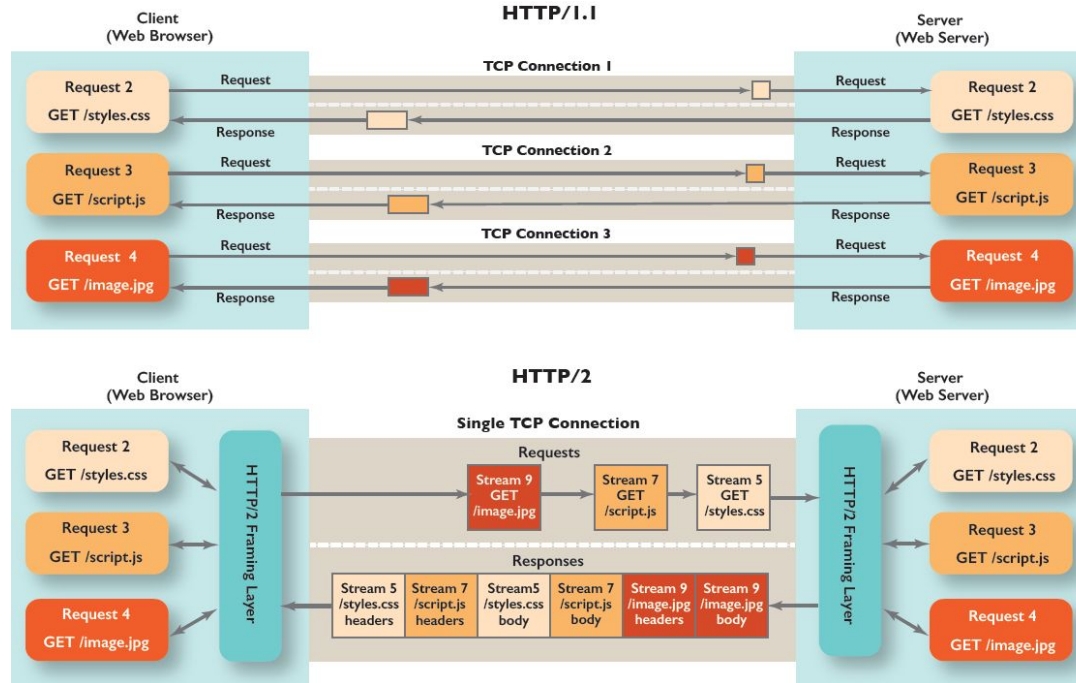
HTTP/2

- HTTP/2 è un nuovo protocollo ideato nel 2015 dall'[IETF HTTP Working Group](#)
- E' basato sull'idea del protocollo [SPDY](#) ideato da Google
- HTTP/2 introduce numerose novità rispetto alla versione 1:
 - **Server push:** la possibilità di inviare risorse senza una specifica richiesta del client
 - **Multiplexing:** l'invio di più richieste utilizzando una singola connessione tra client e server
 - **Protocollo binario:** le richieste e risposte sono codificate in binario, consentendo di migliorare la la sicurezza, la compressione dei dati e la capacità di applicare il multiplexing

HTTP/2 server push

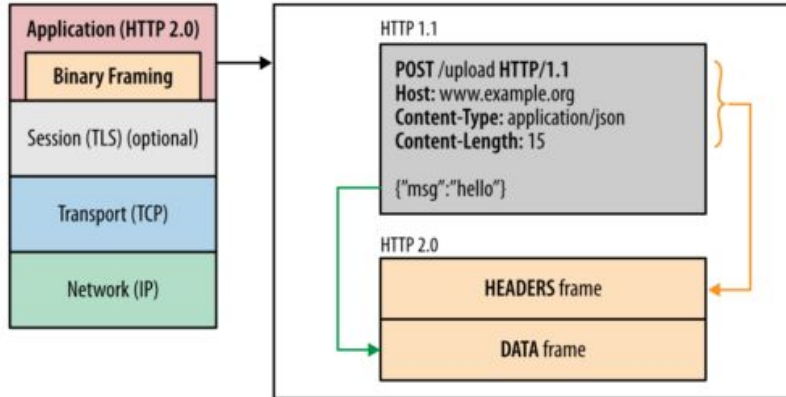
From HTTP/2 IN ACTION by BARRY POLLARD, Copyright 2018.

HTTP/2 multiplexing



Fonte: <https://freecontent.manning.com/mental-model-graphic-how-is-http-1-1-different-from-http-2/>

HTTP/2 binary protocol



HTTP/2 Inside: binary

HTTP/2.0 request:

```
00 00 9D 01 25 00 00 00 01 00 00 00 00 B6 41 8A ...% ... .A.
90 B4 9D 7A A6 35 5E 57 21 E9 82 00 84 B9 58 D3 ...z.S^W!...X.
3F 85 61 09 1A 6D 47 87 53 03 2A 2F 2A 50 8E 9B ?.a..mG.S.*/*P..
D9 AB FA 52 42 CB 40 D2 5F A5 11 21 27 51 88 2D ...RB.@...!'Q.-
48 70 DD F4 5A BE FB 40 05 DE 7A DA D0 7F 66 A2 Kp..Z..@..z...f.
81 B0 DA E0 53 FA D0 32 1A A4 9D 13 FD A9 92 A4 ...S..2.....
96 85 34 0C 8A 6A DC A7 E2 81 04 41 04 4D FF 6A ...4..j.....A.M.j
43 5D 74 17 91 63 CC 64 B0 DB 2E AE CB 8A 7F 59 C]t..c.d.....Y
B1 EF D1 9F E9 4A 0D D4 AA 62 29 3A 9F FB 52 F4 .....J...b)...R.
F6 1E 92 B0 D3 AB 81 71 36 17 97 02 9B 87 28 EC .....q6.....(
33 0D B2 EA EC B9
```

HTTP/1.1 request:

```
GET / HTTP/1.1
Host: demo.nginx.com
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
User-Agent: Chrome/47.0.2518.0
```

HTTP/3

- **HTTP/3** è la terza versione, ancora in fase di definizione, del protocollo Hypertext Transfer Protocol usato per il World Wide Web nonché il successore di HTTP/2
- HTTP/3 è allo stadio di [Internet Draft](#), basato su una precedente bozza di RFC intitolata Hypertext Transfer Protocol (HTTP) over QUIC
- [QUIC](#) è un protocollo ideato da Google nel 2012 e ha lo scopo di essere quasi equivalente a una connessione TCP ma con una latenza notevolmente ridotta

Le novità di HTTP/3

- Le novità principali di HTTP/3 che lo rendono più performante rispetto a HTTP/2 sono:
 - riduzione notevolmente dell'overhead durante l'impostazione della connessione;
 - utilizzo di [UDP](#) al posto del TCP. UDP è un protocollo più veloce ma non offre le stesse garanzie di TCP (non include il recupero delle perdite). Per questo motivo HTTP/3 si occupa di gestire le perdite ma senza interrompere il flusso.

Riferimenti

- Ilya Grigorik, Surma, [Introduction to HTTP/2](#), web.dev
- Barry Pollard, [HTTP/2 in action](#), Manning, 2019
- [Come funziona Internet, Una guida per “policy-makers”](#), European Digital Rights, Traduzione italiana a cura del Centro NEXA su Internet e società, Politecnico di Torino
- J Cole Morrison, [HTTP 1, 2, and 3 in a Nutshell](#)
- Google Webmaster, [Secure your site with HTTPS](#)
- A. Ghedini, R. Lalkaka, [HTTP/3: the past, the present, and the future](#), Cloudflare blog
- M.Amundsen, S.Ruby, L.Richardson, [RESTful Web APIs](#), O'Reilly Media, 2013
- Tom Marrs, [JSON at Work](#), O'Reilly Media, 2017

Grazie dell'attenzione!

Per informazioni:

enrico.zimuel@its-ictpiemonte.it