

03 - Introduzione al WWW

Cos'è il WWW

Il World Wide Web (WWW) è un sistema distribuito e scalabile su Internet per:

- Presentare a schermo documenti multimediali.
- Utilizzare link ipertestuali per la navigazione.

Si basa su alcuni concetti chiave:

- **Client/Browser:** Visualizzatore di documenti ipertestuali e multimediali (testo, immagini, interfacce grafiche).
 - Non permette l'editing (salvo trucchi).
 - Utilizza plug-in per visualizzare formati speciali.
 - Javascript permette la realizzazione di applicazioni client-side (Rich client).
- **Server:** Meccanismo di accesso a risorse locali (file, database, ecc.).
 - Trasmette documenti via socket TCP, identificati da un identificatore univoco.
 - Può fungere da tramite tra il browser e applicazioni server-side, rendendo il browser l'interfaccia dell'applicazione.

Protocolli Fondamentali del WWW

1. **URI (Uniform Resource Identifier):** Standard per identificare risorse di rete e specificarle all'interno di documenti ipertestuali.
2. **HTTP (Hypertext Transfer Protocol):** Protocollo di comunicazione state-less e client-server per l'accesso a risorse ipertestuali via rete.
3. **HTML (HyperText Markup Language) / XHTML (Extensible HyperText Markup Language):** Linguaggio per la realizzazione di documenti ipertestuali, basato su SGML (e ora XML). Permette di creare connessioni ipertestuali all'interno della struttura del documento.

Evoluzioni del WWW

1. Inclusione di oggetti:

- Immagini in-line (Mosaic).
- Plug-in per oggetti di tipi diversi (Netscape).
- Applet Java.
- ActiveX (protocollo proprietario di Internet Explorer).

2. Client Scripting:

- Javascript (LiveScript) per applicazioni client-side (Netscape).
- Jscript e Vbscript (Internet Explorer).
- Standardizzazione ECMA (sintassi e classi fondamentali).
- Ajax.

3. Stili tipografici:

- CSS (Cascading Style Sheets) per controllare la visualizzazione dei documenti HTML.

4. Gestione delle transazioni:

- Cookies (Netscape, poi standardizzati).

- XMLHttpRequest (Ajax) per una maggiore dinamicità.

5. Siti web dinamici:

- Applicazioni server-side (CGI-BIN) evolvono in linguaggi/ambienti di programmazione: Perl, ASP (Javascript e Visual Basic), PHP, Python, Ruby, ecc.
- Architettura a tre livelli (user-interface, application logic, data storage) con librerie di accesso ai database (ODBC, JDBC, EJB, ActiveModel, ecc.).
- Il browser diventa l'interfaccia principale per applicazioni gestionali distribuite.

6. Strutturazione dei documenti:

- XML per definire linguaggi di markup più adatti ai singoli task, superando i limiti di HTML.

7. Framework di sviluppo:

- Ambienti integrati con API e librerie per semplificare lo sviluppo client-side e server-side.
- Esempi: Django (Python), Rails (Ruby), framework Ajax (Prototype, JQuery, Ext, Dojo), Google Web Toolkit (GWT).

Mode del Passato e del Presente

Trucchi HTML (Obsoleti)

- Uso di immagini di testo, tabelle di layout, spacer, blockquote, tag ``, frame, estensioni browser (es. `<marquee>`) per ottenere effetti grafici.
- **Sostituiti da tecniche standard (es. CSS).**

LAMP

- Stack open source per siti web dinamici: Linux, Apache, MySQL, Perl/PHP/Python.
- Separazione di memorizzazione, logica e distribuzione.

REST (Representational State Transfer)

- Modello di interazione client-server di HTTP.
- Garantisce interoperabilità, scalabilità e uso avanzato di HTTP se applicato correttamente.

Semantic Web

- Attribuzione di semantica ai dati per creare un modello concettuale delle informazioni.
- Ambizioso, forse troppo.

Linked Data

- Modello più modesto di attribuzione di semantica e indirizzabilità ai dati.
- Permette applicazioni sofisticate e connesse anche senza modelli concettuali forti.

Open Linked Data

- Versione "politicamente corretta" del Linked Data, applicata ai dati resi disponibili dalla Pubblica Amministrazione.

Ajax (Asynchronous Javascript And XML)

- Meccanismo per applicazioni web client-side e server-side interattive.
- Minimizza il traffico di rete.

Node.js

- Eseguitibile basato sul motore V8 di Chrome.
- Permette di eseguire Javascript server-side.
- Rivoluzione nella progettazione di applicazioni web server-side.
- npm (Node Package Manager) offre una vasta gamma di estensioni e servizi.

MEAN/MERN Stack

- Stack di tecnologie moderne per il web:
 - MongoDB
 - ExpressJS
 - Angular/React.js
 - NodeJS

Mobile First

- Priorità al supporto per piattaforme mobili (smartphone, tablet) rispetto ai PC tradizionali.

Responsive Web Design

- Adattamento della pagina a diverse dimensioni di schermo e dispositivi.

Single Page Web Sites

- Contenuti e servizi di un sito organizzati in un'unica pagina lunga e scrollabile.

Component-Based Design

- Sviluppo integrato di parti visuali, markup, stile e computazione per ogni componente.
- Componenti autosufficienti (HTML/template, CSS, Javascript).
- Introdotto da AngularJS, poi diffuso con Angular, React, VueJS.

Typescript, CLI, WebPack, Browserify

- **Typescript:** Linguaggio che transpila in Javascript, aggiungendo controlli sui tipi.
- **CLI (Command Line Interface):** Strumenti da riga di comando per configurare e aggiornare progetti web.
- **Webpack:** Web bundler, compilatore di applicazioni web che genera codice eseguibile (compilazione, riorganizzazione, librerie, offuscamento).
- **Browserify:** Permette di importare nel browser librerie pensate per Node e npm.

Progressive Web Applications (PWA)

- Applicazioni web che si comportano come app native su dispositivi mobili (termine usato da Google).

Opinionated vs. Non-Opinionated Frameworks

- Frameworks: librerie che semplificano lo sviluppo web.
- **Opinionated:** Impongono un modello di design specifico (più rigidi).
- **Non-opinionated:** Più flessibili.

Static Site Generators

- Pre-eseguono script server-side e generano siti web "statici".
- Riducono la granularità e migliorano la velocità di caricamento, specialmente per siti complessi basati su componenti.