

Controllare la WP REST API tramite JavaScript e React

Luigi Maselli

Fullstack JavaScript Developer

<https://grigio.org>

<https://corso-javascript.it>

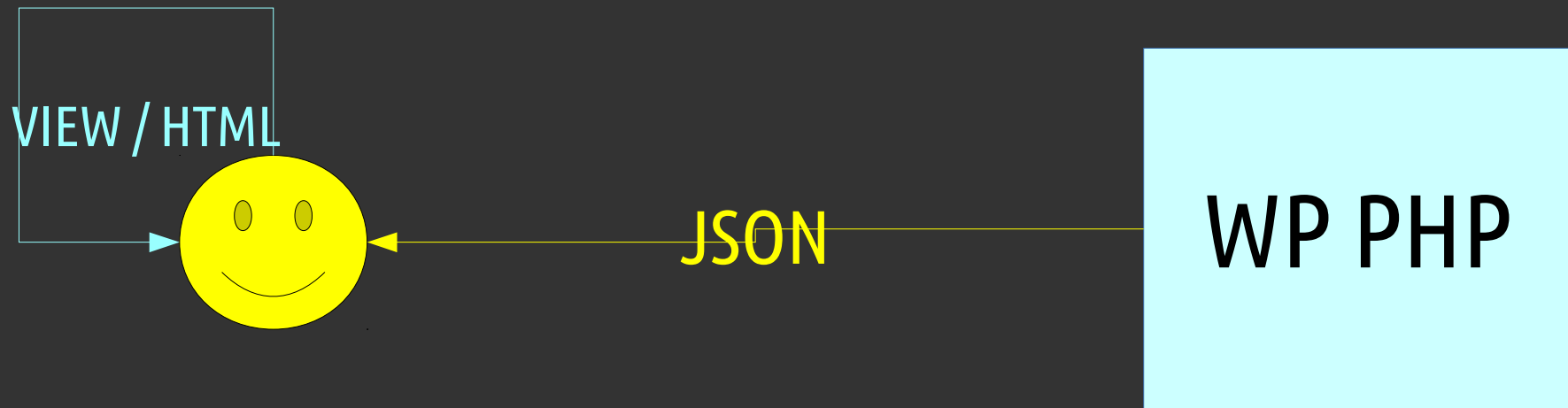
Argomenti

- Cos'è un'API (WP admin Calypso)
- Cos'è il 'nuovo' JavaScript (e React)
- Demo: Frontend alternativo con JS / React

Cos'è WordPress?



Cos'è WP Calypso Admin UI + REST API ?



Perché cambiare?

- **Interoperabilità:** dati (JSON) WP mostrati in app native
- **Performance:** la maggior parte della view non cambia
- **Separazione responsabilità:** Frontend(s), Backend, dati, presentazione, meno complessità e app che 'fa tutto'



Linguaggio di programmazione



Libreria



lombardoshop

Framework / CMS



SaaS

Perché NON cambiare?

- **Solo blog**: mi basta WP, al max ritocco il tema responsive
- **Poco budget / tempo**
- **Vendo un prodotto non un servizio**

Cos'è un'API?

- Accessible Programmable Interface
- Una 'cosa' che fa "fare cose" agli sviluppatori
- es. 'accedere ai post'

WP API (PHP)

- es. `get_posts`
- Doc https://developer.wordpress.org/reference/functions/get_posts/
- Accessibile da PHP (dall'interno dell'applicazione)

WP REST (o JSON, HTTP) API

- /posts (url accessibile dall'esterno dell'applicazione)
- Demo <http://demo.wp-api.org/wp-json/wp/v2/posts>
- Abilitare il plugin sul proprio WordPress
- Doc <http://v2.wp-api.org/reference/posts/>

demo.wp-api.org/wp-json/wp/v2/posts

```
[
  - {
    id: 16,
    date: "2015-12-31T14:51:26",
    date_gmt: "2015-12-31T14:51:26",
    - guid: {
      rendered: "http://wp-api-de-elbinsta-n5pu020s131b-1748419096.eu-west-1.elb.amazonaws.com/?p=16"
    },
    modified: "2015-12-31T14:51:26",
    modified_gmt: "2015-12-31T14:51:26",
    slug: "architecto-enim-omnis-repellendus",
    type: "post",
    link: "http://demo.wp-api.org/2015/12/31/architecto-enim-omnis-repellendus/",
    - title: {
      rendered: "Architecto enim omnis repellendus"
    },
    - content: {
      rendered: "<h6>Similique eaque voluptas cum voluptatem. Similique debitis quis sapiente veniam sa molestiae</h6> <p> Magnam omnis necessitatibus veritatis modi. inventore accus <h6>Qui blanditiis et dolorum veniam alias aperiam. Hic quisquam neque ea alias nostrum voluptate
```

Marketing

UI / UX

Frontend

The rest

YOUR APP

Authentication
Authorization
Database
Networking
Security upgrades
...



Luigi Maselli @grigi0 · 25 feb

Visualizza traduzione

"Hi Marina!". How an hacker could contact girls XD #dataleaks #miocuggino #netsec

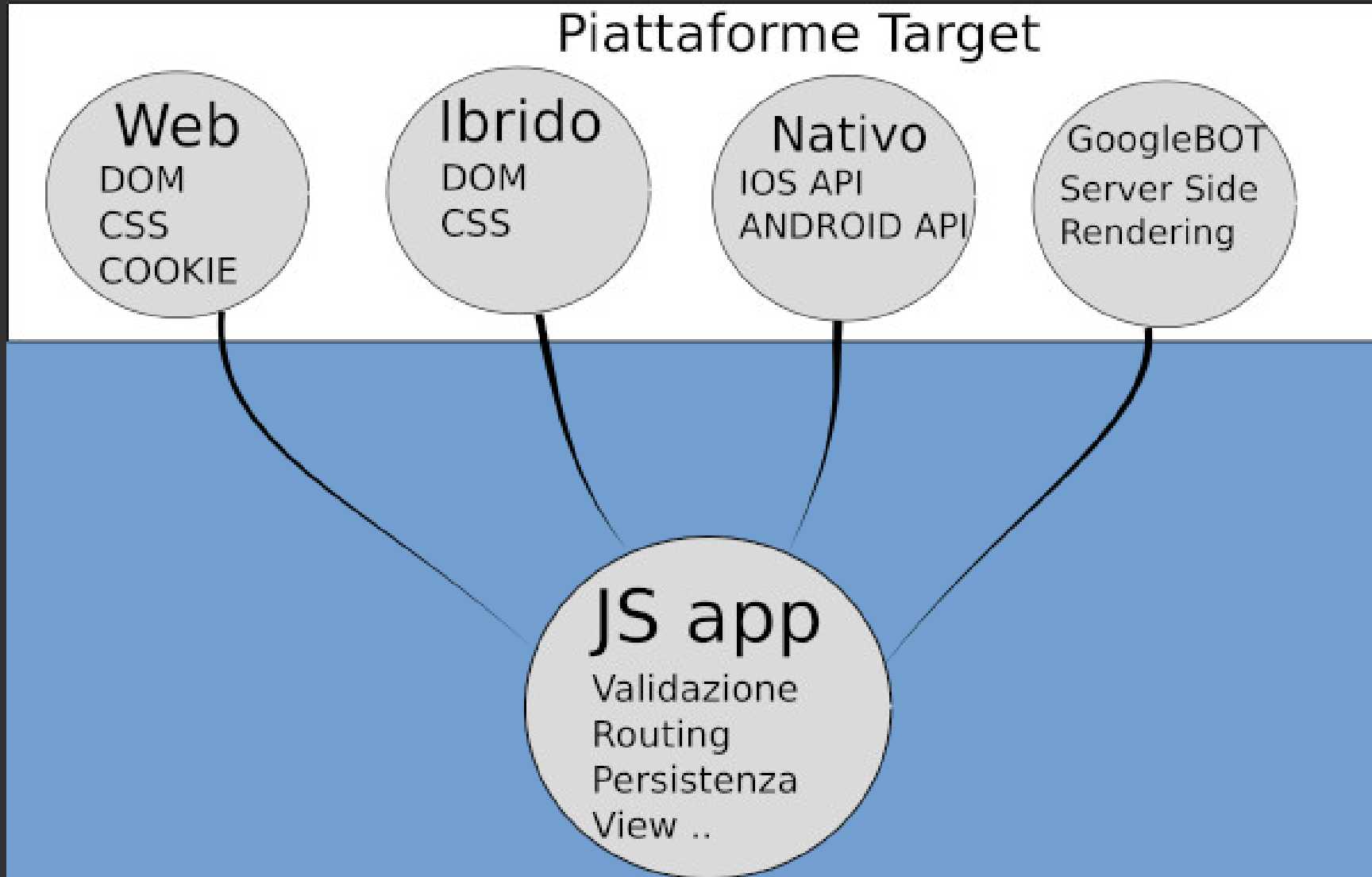
The screenshot shows a dating app interface with a browser developer tool overlay. The app interface includes a filter section on the left with options for 'PROVINCIA' (Milano, Roma, Torino), 'LINGUE', and 'TIPO LAVORO'. The main area displays three user profiles: 'Asia', 'Marina', and 'Ester'. The browser developer tool is open to the 'Network' tab, showing a list of resources. A specific resource is selected, and its response is visible in the 'Response' pane, containing a JSON object with user data:

```
1: {"4298", "username": "marina-milano-7", "email": "b.sorobulina@gmail.com", "height": "178", "hourly_wage": null, "pierci
```

Abbiamo i dati!!!! E ora?

- Script che twitta in automatico ?
- Grafici 'alternativi' sulle stats di utilizzo di WP ?
- ...
- Frontend 'alternativo' con JavaScript ES6 e ReactJS

JavaScript è ovunque!!



E... jQuery ??

Prendere i dati dalla ns App JS

```
70  >> fetch(`http://demo.wp-api.org/wp-json/wp/v2/posts?page=${page}`)
71  >> > .then((response) => {
72  >> >   return response.json()
73  >> > })
74  >> > .then((data) => {
75  >> >   // ...
76  >> >   console.log(data)
77  >> > }).catch((err) => {
78  >> >   console.error(err)
79  >> > })
```

fetch ..

- Funziona in JavaScript Client / Server (NodeJS)
- Polyfill se non supportato nativamente

ReactJS / React Native

- **Virtual DOM**
- **View astratta JSX**: renderizzata in HTML DOM o View Nativa)
- **Componenti**
 - blocchi di codice che encapsulano funzionalità e vista
 - possono interagire con uno stato esterno o proprio

Es. componente <Counter />

```
1 import React from "react";
2
3 class Counter extends React.Component {
4   // stato iniziale
5   state = {counter:0};
6
7   increment() {
8     const {counter} = this.state
9     this.setState({counter: counter + 1 })
10  }
11
12  render() {
13    return (
14      <div>
15        Main component, {this.state.counter}
16        <button onClick={() => this.increment()}>+1</button>
17      </div>
18    )
19  }
20 }
```

Es. utilizzo di <Counter />

```
23  ↵
24  import ReactDOM from "react-dom";
25  ↵
26  const reactRoot = document.getElementById("react-root");
27  ReactDOM.render(
28    <div>
29      <Counter /><Counter />
30    </div>
31    , reactRoot);
32  ↵
```

Nel caso di WordPress..

Componente `<PostList>` che detiene lo stato e passa ai componenti figli le azioni che permettono di variare lo stato dell'app

Codice completo

github.com/corso-javascript/react-wp-demo

```
>> <PostList  
>>   items={items}  
>>   isLoading={isLoading}  
>>   onItemClick={(ev) => {this.onItemClick(ev) }}  
>>   onFetchMore={(ev) => {this.onFetchMore(ev) }}  
>> />
```

Recap

- Le API di WordPress permettono una più facile integrazione in progetti esterni e NON-PHP/WP (es. WP Calypso)
- JavaScript > Web / HTML

Grazie!

Se siete interessati ad imparare il 'nuovo' JavaScript ES6..

corso-javascript.it

codice **WPTO** o bit.ly/corso-javascript-wpto