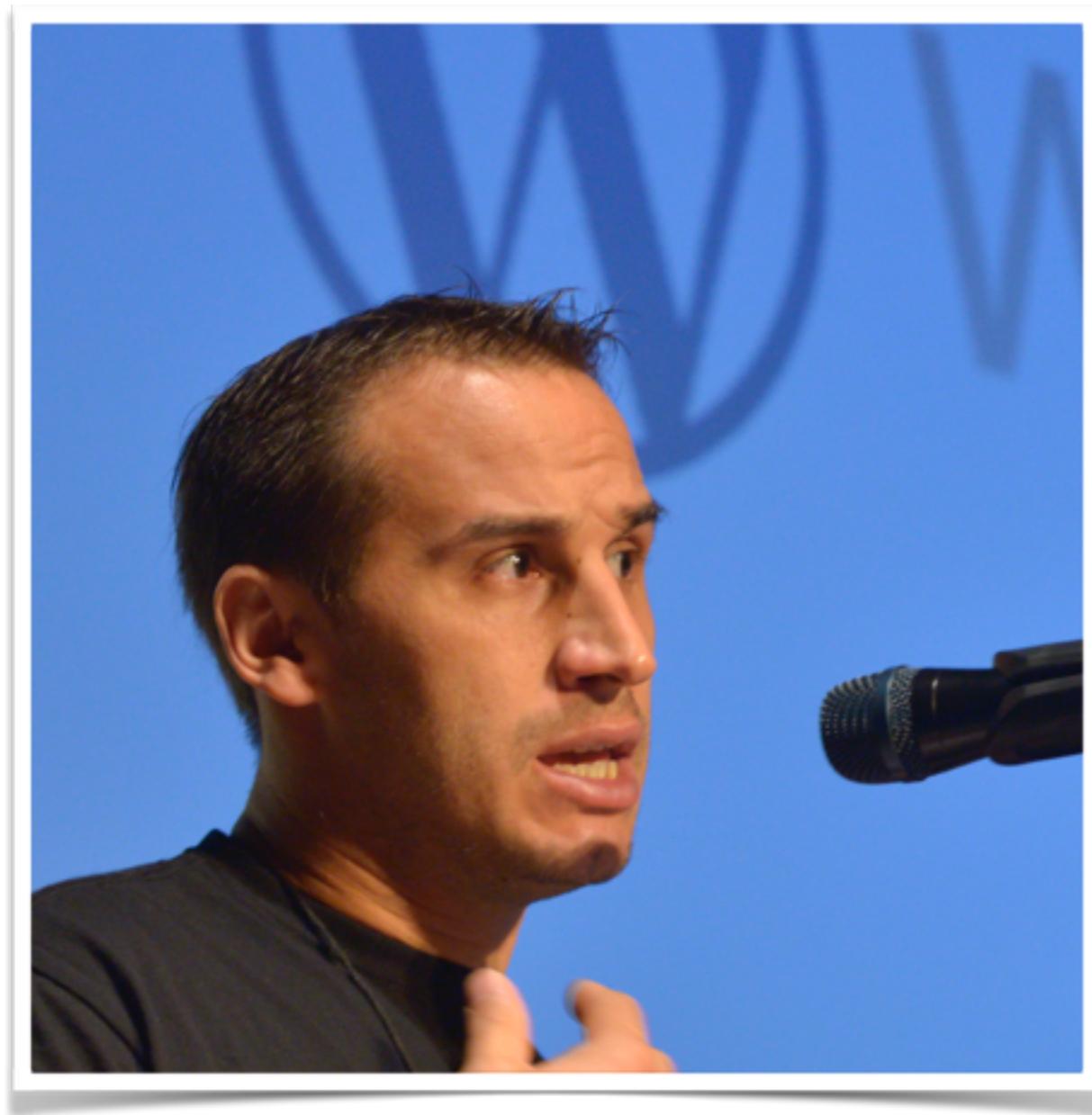


# WordPress

Sin fronteras

# @retrofox

Damian Suarez



- Web Developer
- Automattic
- @wpargentina

# Freelancer

• 2007

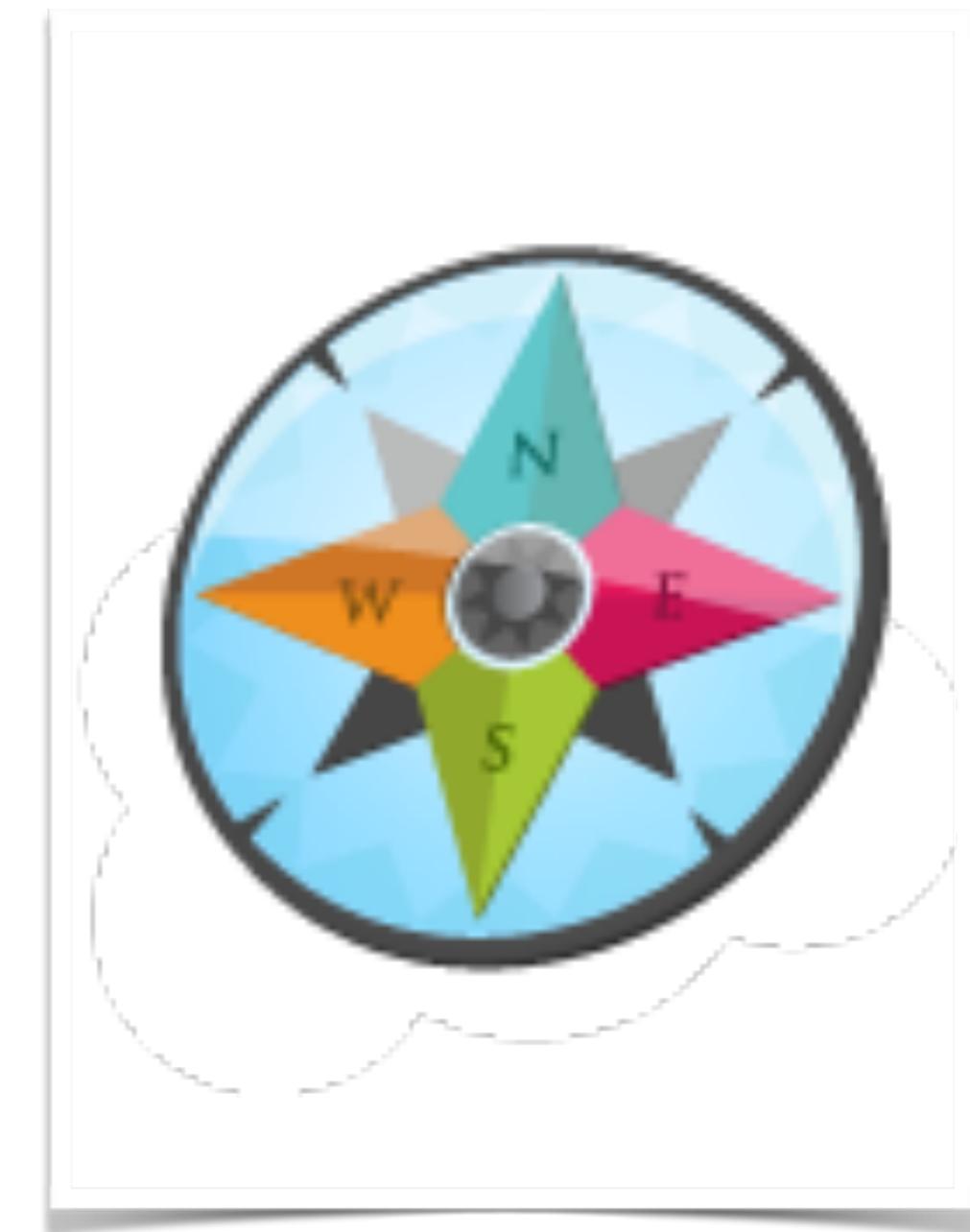
- PHP
- JS
- WordPress



# Rosepad

• 2009

- PHP - Symfony
- JS - MooTools





# LEARNBOOST

www.learnboost.com

- 2011

- NodeJS

- JS - jQuery

The screenshot shows the LearnBoost sign-up interface. At the top, there are navigation links for "Single Apps", "For Schools", and "Blog", along with "Join Now" and "Log In" buttons. A large green "Sign Up" button is prominently displayed. Below it, there are two social media integration options: "with Facebook" and "with Google". A "First name" and "Last name" input field pair is followed by an "Email" input field. A "Password" input field is present, along with a dropdown menu for selecting "I am a" (Teacher, Student, or Parent). A checkbox for "I want to receive learnboost news." is checked, and a link to "Terms" is provided. A "Join" button is located at the bottom right. The background features a blue sky with white clouds and the text "rooms here" and "ed to manageool. Save timearnBoost.".

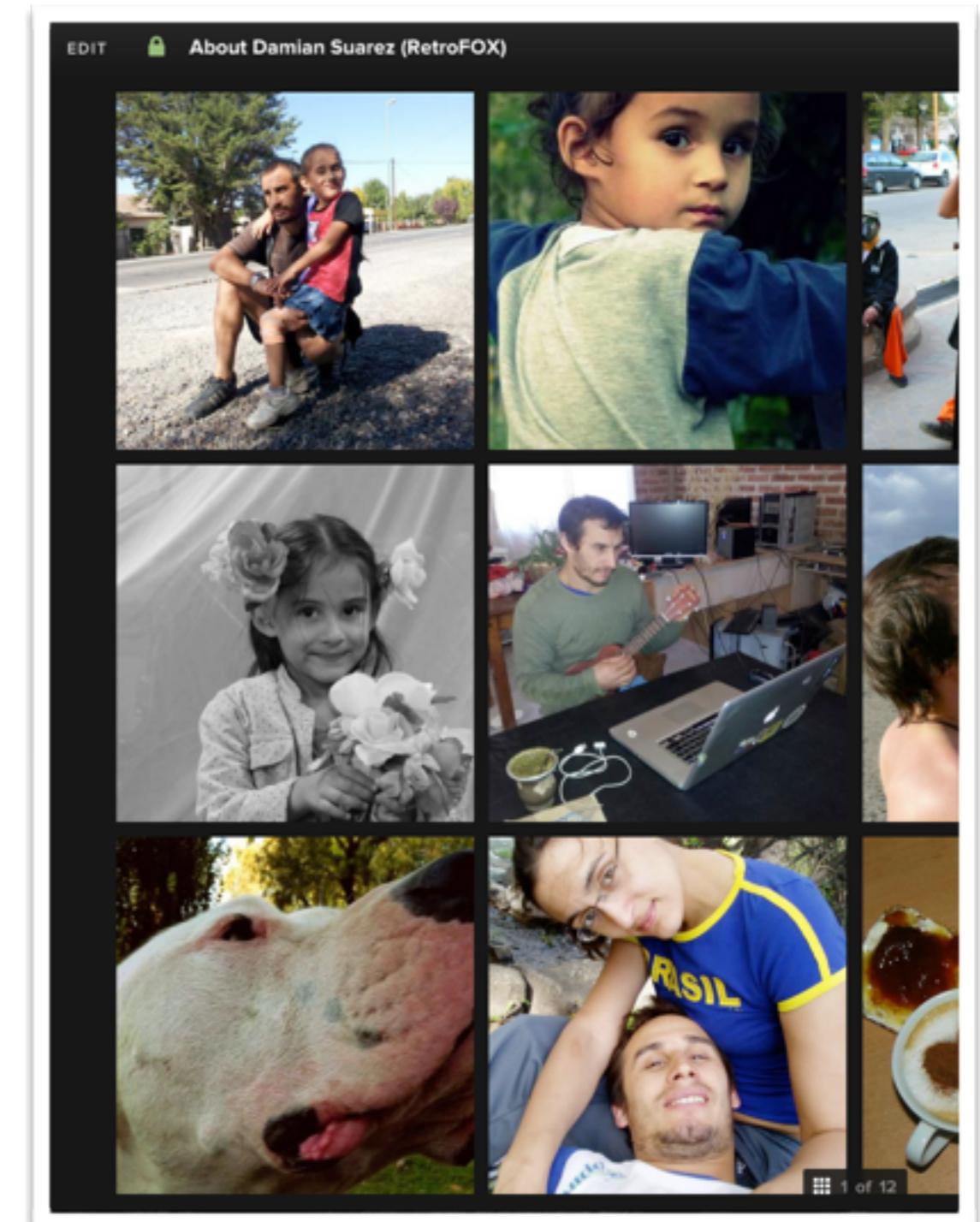
wonderful user  
onder why you've been  
oftware.

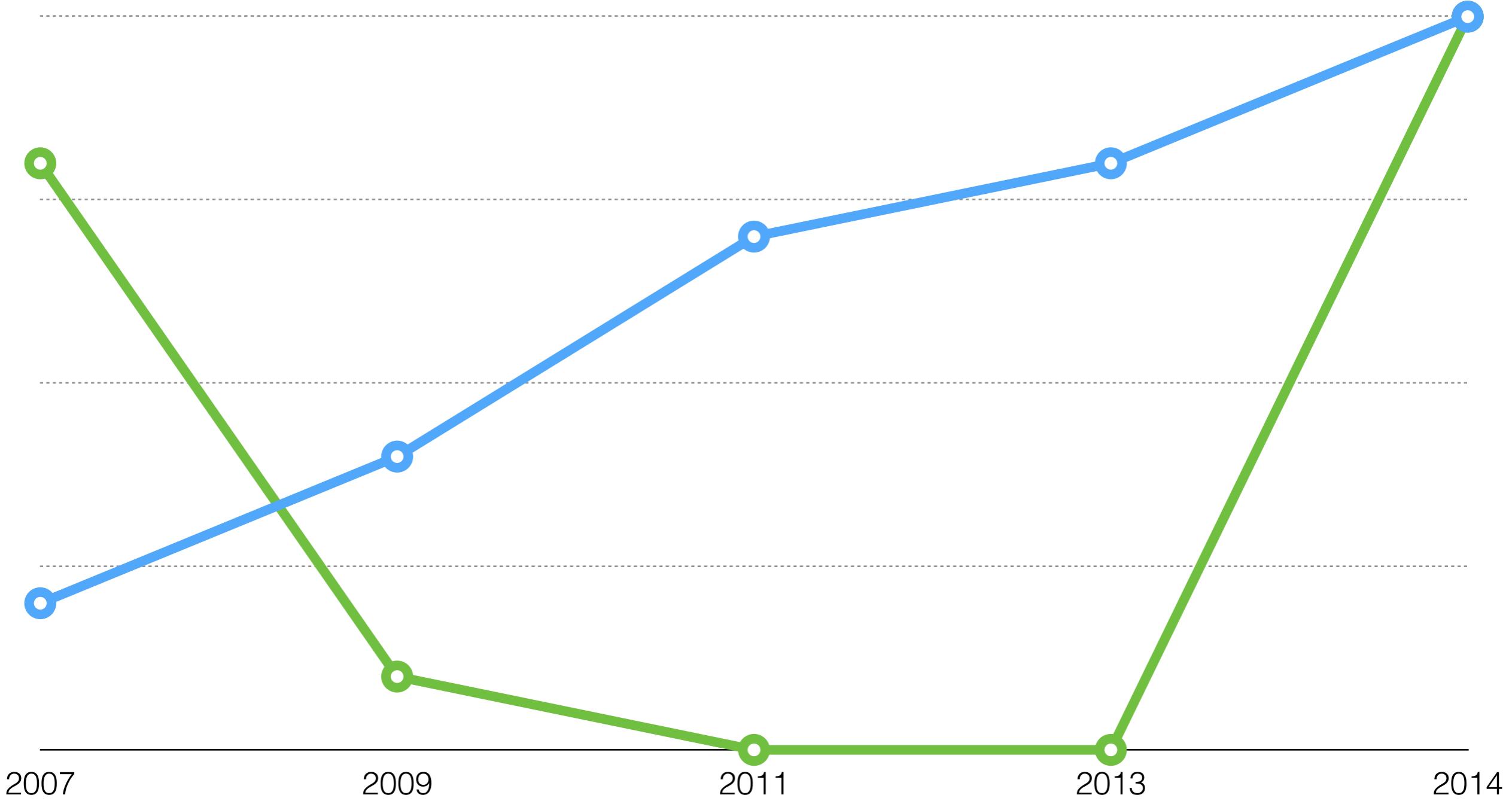
- 2013

- NodeJS

- JS

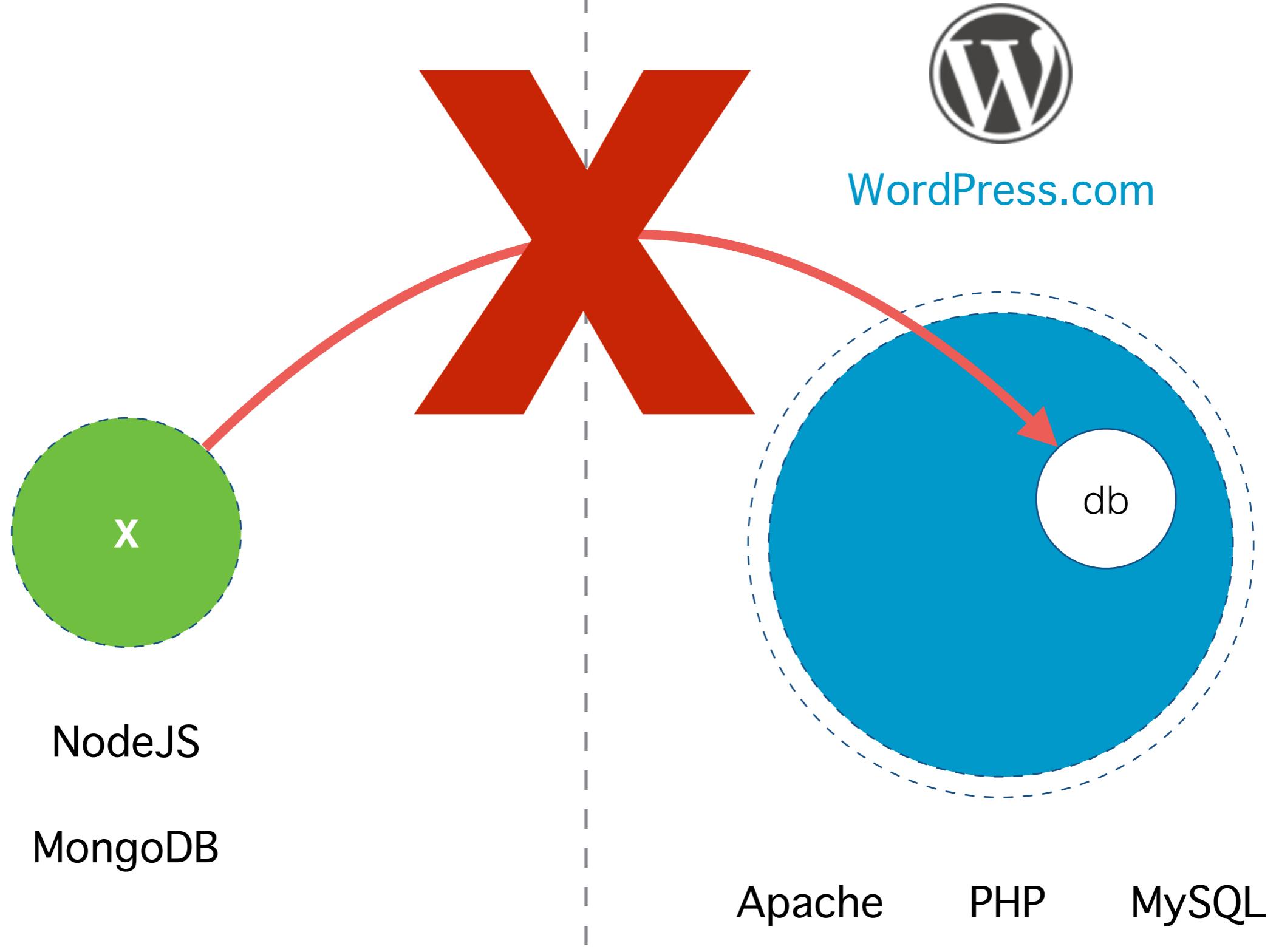
- socket.io





- **Crecimiento de WordPress**
- **Cercanía con WordPress**





# REST API

```
{  
  "ID": 60487750,  
  "name": "WordCamp Buenos Aires",  
  "description": "Sitio web oficial del wordCamp de la ciudad de Buenos Aires",  
  "URL": "http://buenosaires.wordcamp.org",  
  "jetpack": false,  
  "post_count": 43,  
  "subscribers_count": 3,  
  "lang": "es",  
  "logo": {  
    "id": 0,  
    "sizes": [],  
    "url": ""  
  },  
  "visible": true  
  "is_private": false  
  "is_following": false  
  "options": {  
    "timezone": "",  
    "gmt_offset": 0,  
    "videopress_enabled": false,  
    "upgraded_filetypes_enabled": false,  
    "login_url": "https://retrofocs.wordpress.com/wp-login.php",  
    "admin_url": "https://retrofocs.wordpress.com/wp-admin/"  
  },  
  "meta": {  
    "links": {  
      "self": "https://public-api.wordpress.com/rest/v1.1/sites/60487750",  
      "help": "https://public-api.wordpress.com/rest/v1.1/sites/60487750/help",  
      "posts": "https://public-api.wordpress.com/rest/v1.1/sites/60487750/posts/",  
      "comments": "https://public-api.wordpress.com/rest/v1.1/sites/60487750/comments/"  
    }  
  }  
}
```

# REST-API

<http://developer.wordpress.com/>

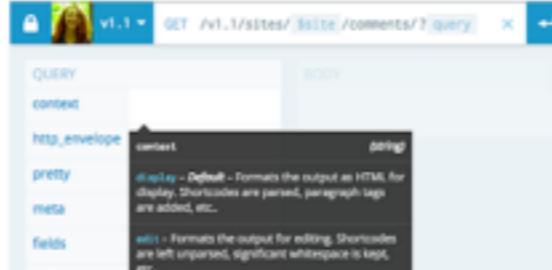
# Howdy, Developers

Build something amazing and make a difference  
for millions of WordPress users.

GET STARTED

```
GET /v1.1/sites/$site/posts/ Posts: Get a list of matching posts.  
GET /v1.1/sites/$site/posts/$post_ID Posts: Get a single post (by ID).  
POST /v1.1/sites/$site/posts/$post_ID Posts: Edit a post.  
GET /v1.1/sites/$site/posts/slug:$post_slug Posts: Get a single post (by slug).  
POST /v1.1/sites/$site/posts/new Posts: Create a post.  
POST /v1.1/sites/$site/posts/$post_ID/delete Posts: Delete a post. Note:  
POST /v1.1/sites/$site/posts/$post_ID/restore Posts: Restore a post or  
GET /v1.1/sites/$site/posts/$post_ID/likes/ Posts: Get a list of the likes
```

```
1 v {  
2   "version": "1.0",  
3   "provider_name": "Developer Resources",  
4   "provider_url": "Http://developer.wordpress.com",  
5   "author_name": "Justin Shreve",  
6   "author_url": "Https://developer.wordpress.com/",  
7   "title": "oEmbed Provider API",  
8   "type": "link",  
9   "html": "<p>Every post, page and attachment hosted  
10 }
```



## REST API

Our REST API lets you build applications that can manage a user's blogs, fetch stats, and much more.

## oEmbed

Support WordPress.com embeds in your own applications using oEmbed.

## Developer Console

Try before build. Test the REST API using our console to get back live JSON data.

```
1 <script src="wpcom.js"> </script>  
2 <script>  
3   // create a client WPCom instance  
4   var wpcom = WPCom();  
5  
6   // retrieve a listing of the most recent  
7   // 5 posts on "en.blog.wordpress.com"  
8   wpcom  
9     .site('en.blog.wordpress.com')  
10    .postsList({ number: 5 }, function(err, data){  
11      if (err) throw err;  
12  
13      console.log('the newest 5 blog post titles:
```

## wpcom.js

The official JavaScript library for the WordPress.com REST API. It is compatible with Node.js and web browsers.

REST-API

URL

**base + /rest/ + version + endpoint**

base: <https://public-api.wordpress.com>

version: v1.1

endpoint: /sites/\$site

# REST-API

# Endpoints

## Comments

View and manage a post's comments.

### Resource

**GET** /sites/\$site/comments/

**GET** /sites/\$site/posts/\$post\_ID/replies/

**GET** /sites/\$site/comments/\$comment\_ID

**POST** /sites/\$site/comments/\$comment\_ID

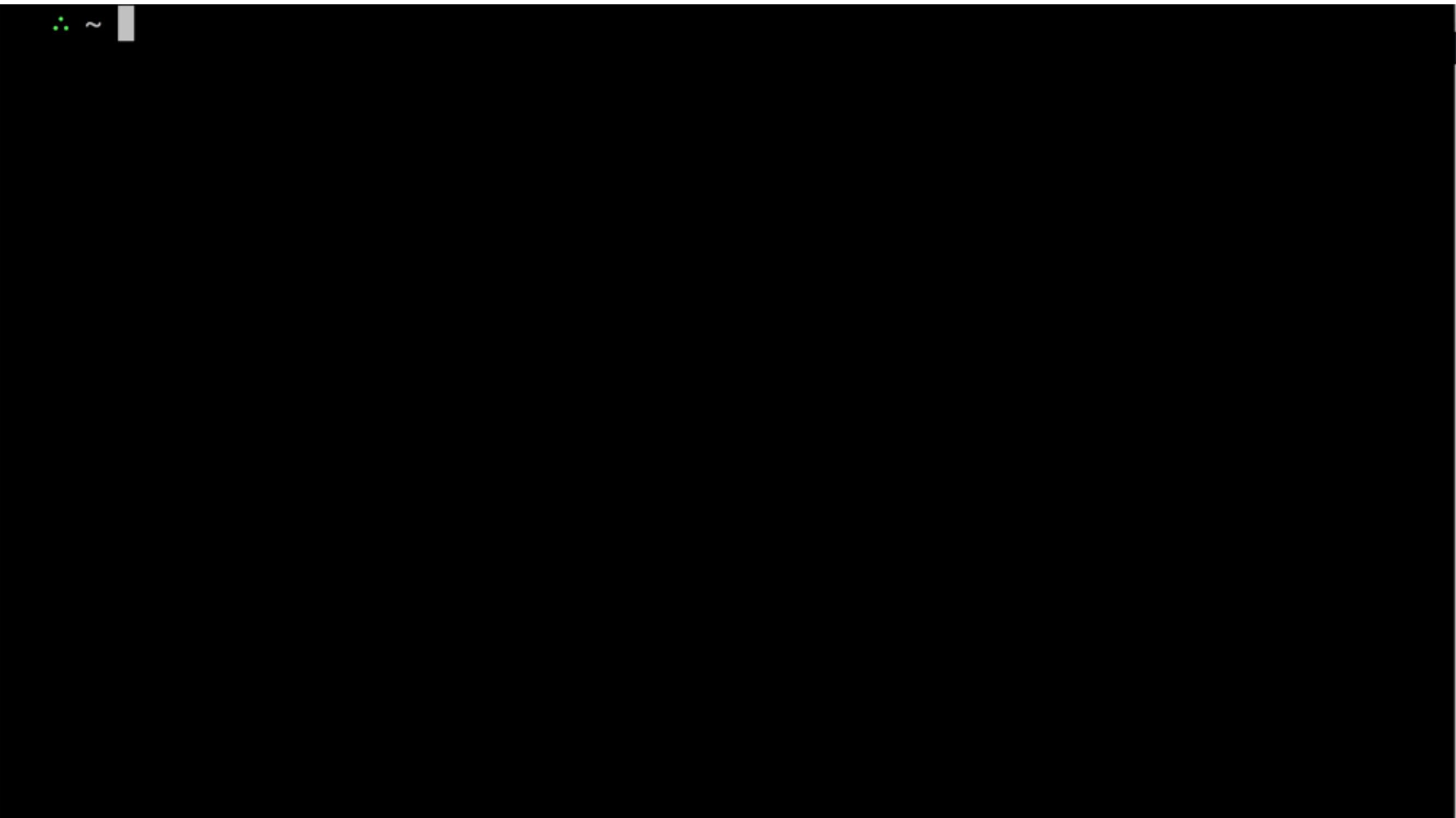
**POST** /sites/\$site/posts/\$post\_ID/replies/new

**POST**  
/sites/\$site/comments/\$comment\_ID/replies/new

**POST** /sites/\$site/comments/\$comment\_ID/delete

# REST-API

<https://public-api.wordpress.com/rest/v1.1/sites/en.blog.wordpress.com/posts/>



# wpcom.js

# www.wpcomjs.com

## JavaScript API for WordPress

Official JavaScript library for the WordPress.com REST API.  
Compatible with Node.js and web browsers.

[SOURCE ON GITHUB](#)

```
1 <script src="wpcom.js"> </script>
2 <script>
3   // create a client WPCOM instance
4   var wpcom = WPCOM();
5
6   // retrieve a listing of the most recent
7   // 5 posts on "en.blog.wordpress.com"
8   wpcom
9     .site('en.blog.wordpress.com')
10    .postsList({ number: 5 }, function(err, data) {
11      if (err) throw err;
12
13      console.log('the newest 5 blog post titles are:\n');
14      data.posts.forEach(function(post, i) {
15        console.log(' %d: "%s"', i+1, post.title);
16      });
17    });
18 </script>
```

# wpcom.js

<https://public-api.wordpress.com/rest/v1.1/sites/en.blog.wordpress.com/posts/>

```
wpcom
  .site("en.blog.wordpress.com")
  .get(function (err, data) {
    // data object
  });
}
```

# wpcom.js

## Site(id, WPCOM)

Create a new `Site` instance giving `id` and `WPCOM` instance.

```
var site = Site('<id>', WPCOM);
```

## Site#get([query, ]fn)

Get site information

```
site.get(function(err, info){  
    // `info` data object  
});
```

## Site#usersList([query, ]fn)

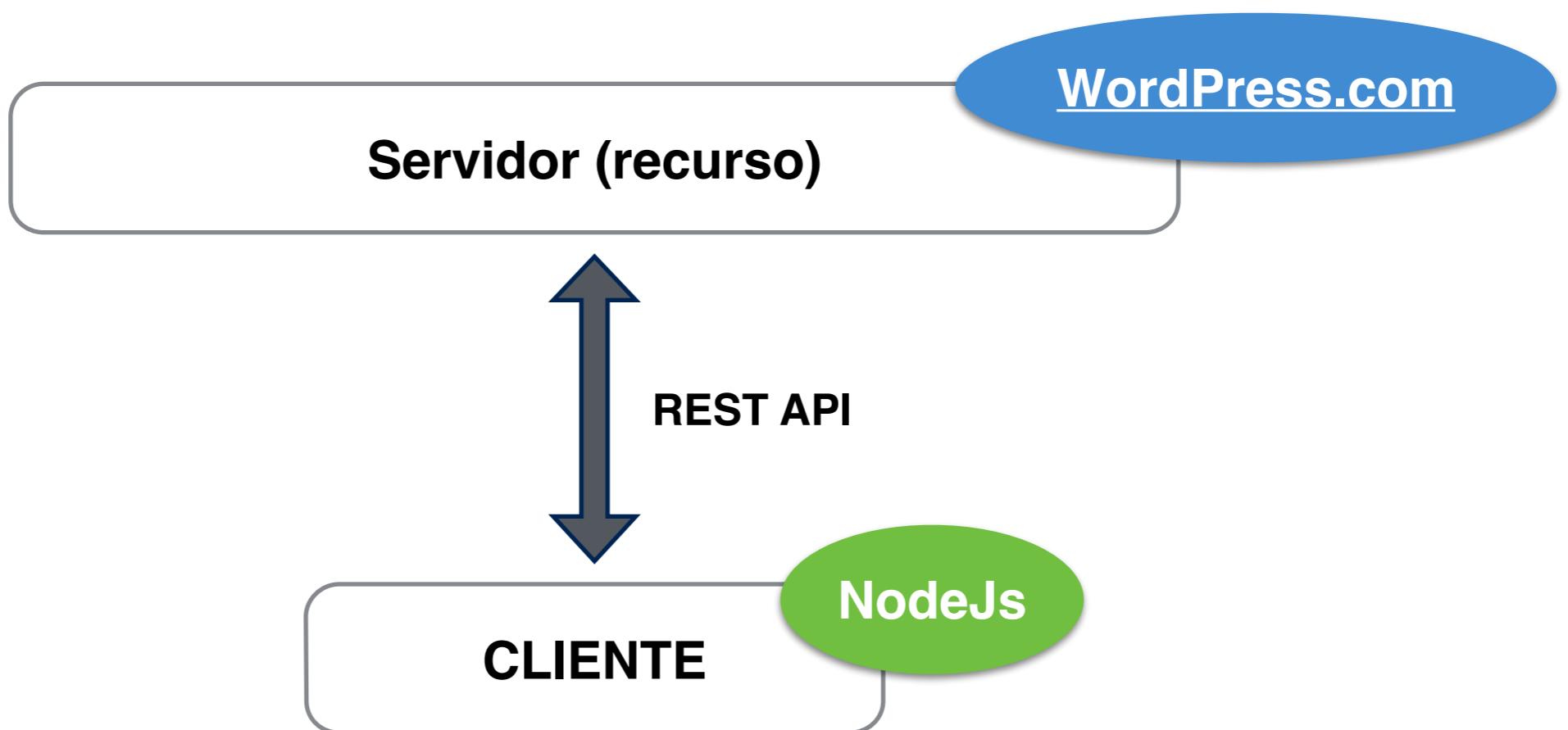
List the users of a site

```
site.usersList(function(err, list){  
    // `list` data object  
});
```

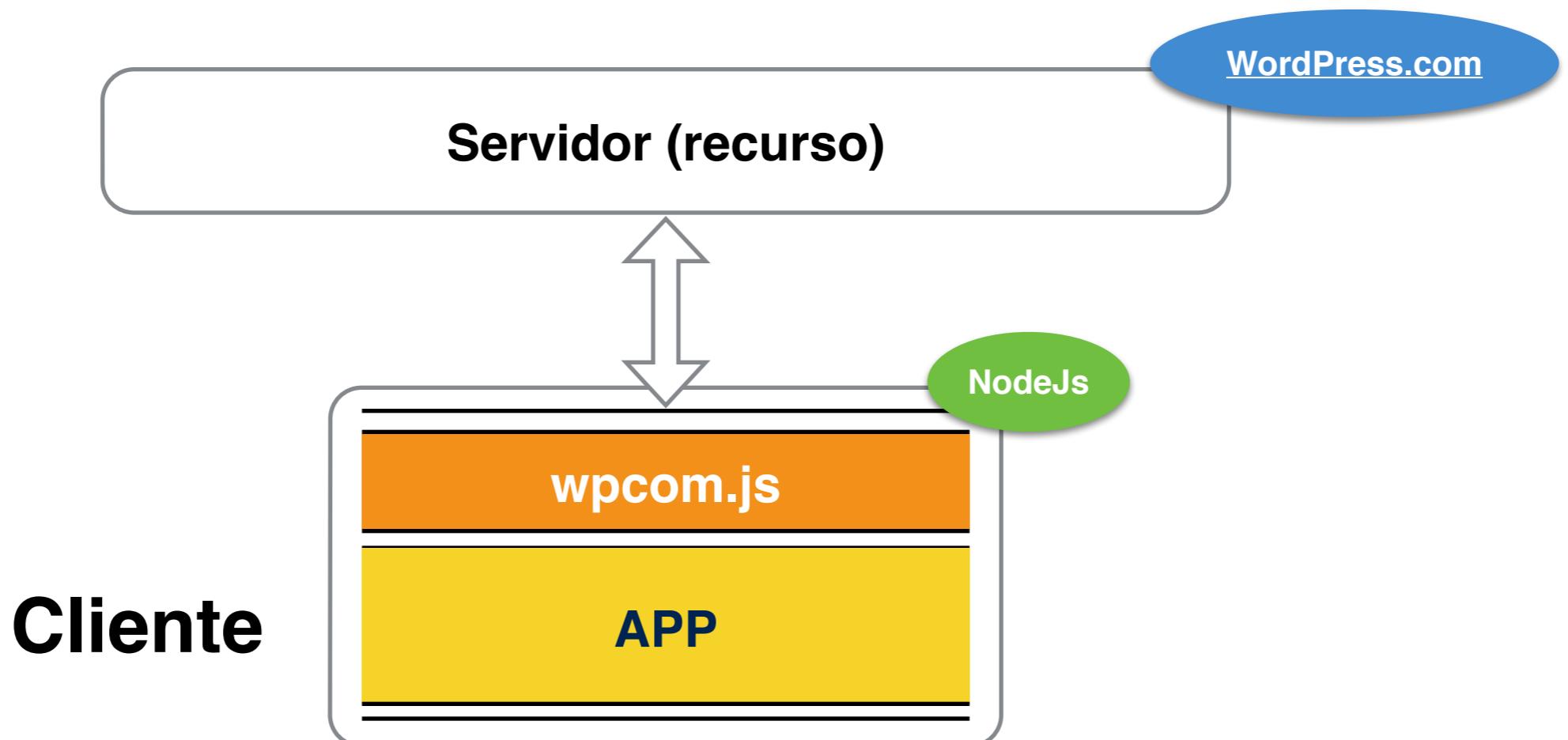
## Site#postsList([query, ]fn)

Get site posts list

# wpcom.js - Nodejs



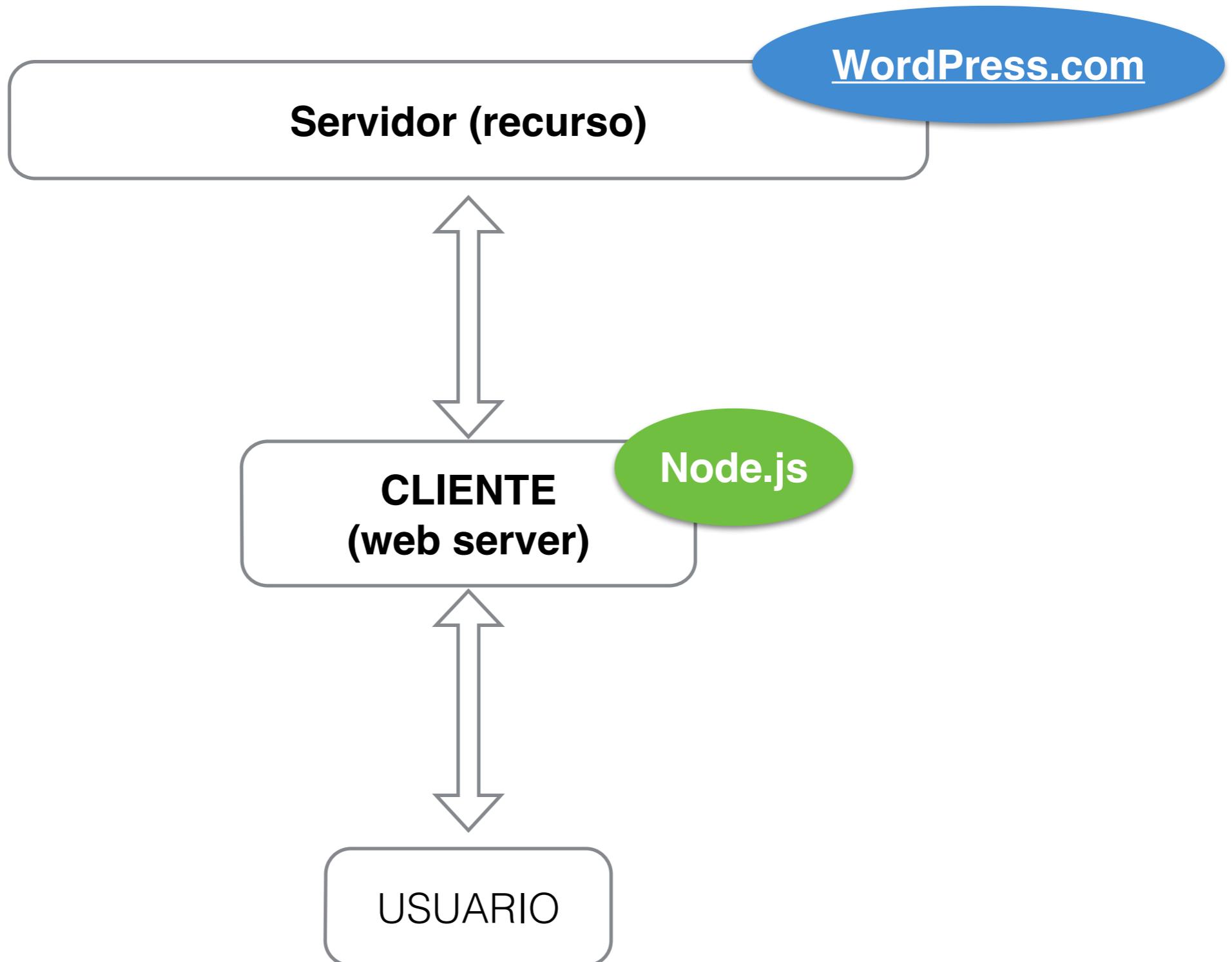
# wpcom.js - Nodejs



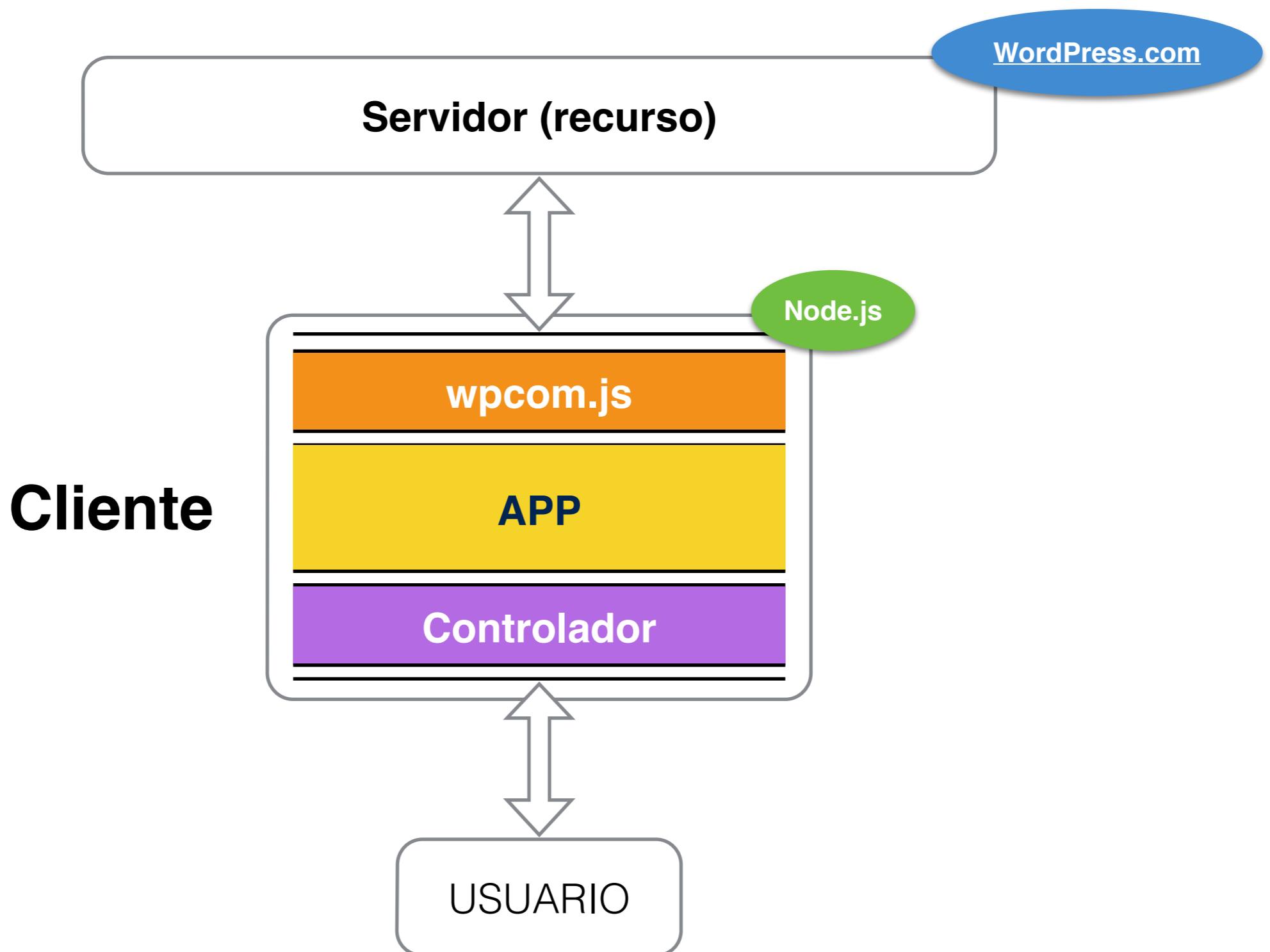
# Ejemplo NodeJs

WC

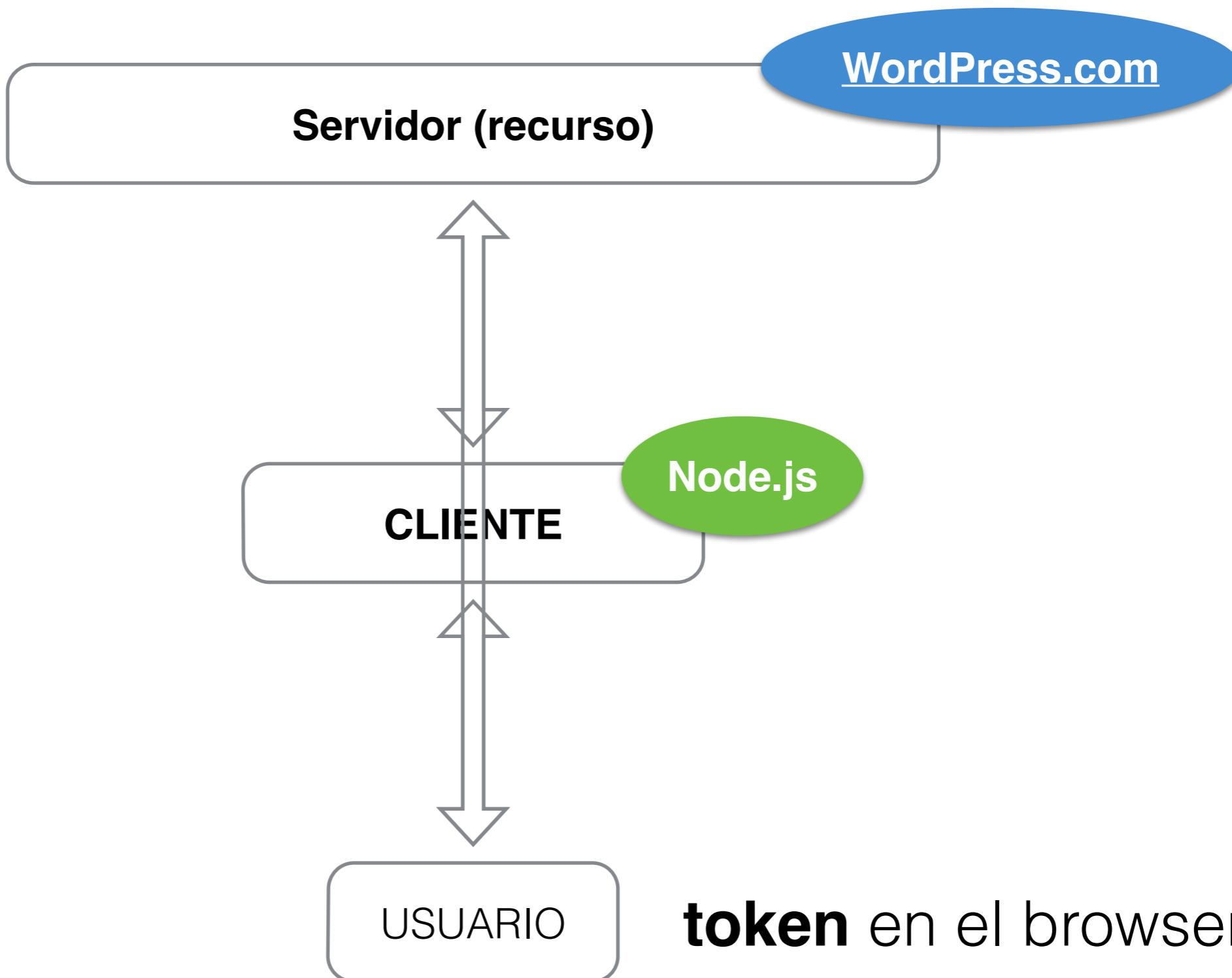
# wpcom.js - Web app



# wpcom.js - Web app

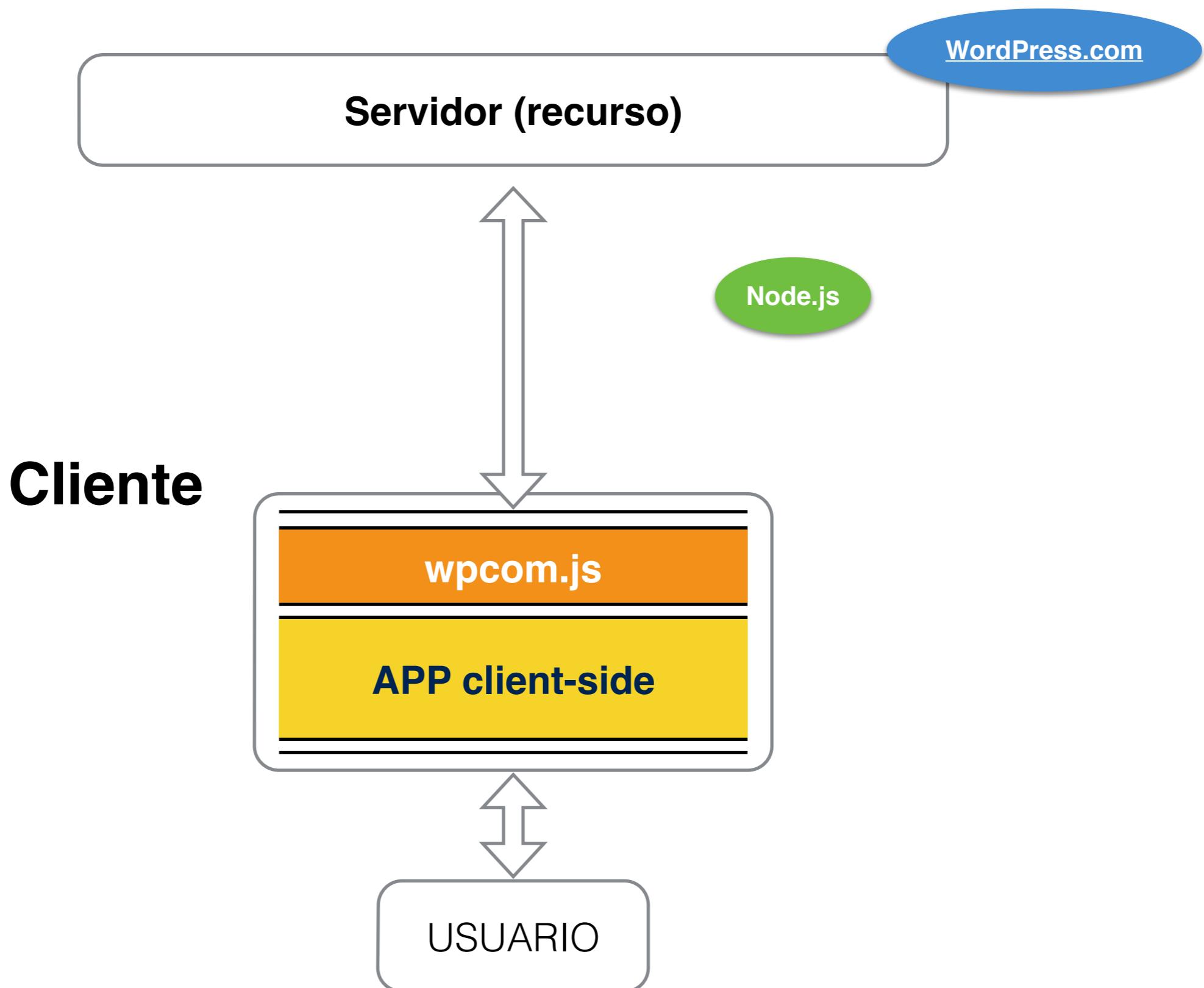


# wpcom.js - Client-side - CORS



**token** en el browser !

# wpcom.js - Client-side - CORS



# wpcom.js - Client-side - CORS

The screenshot shows the Network tab of the Chrome DevTools interface. At the top, there are icons for search, mobile view, Elements, Network, Sources, Timeline, and a gear. Below the tabs, there are filters for 'Preserve log' and 'Preserve log' status. The main list shows one entry: '<top frame>' with a status of 'undefined'. There are navigation arrows ('<', '>', '|') at the bottom of the list.

# **EDITAR DATOS ?**

## **OAUTH**

**OAuth (Open Authorization) es un protocolo abierto que permite autorización segura de una API de modo estándar y simple.**

# EDITAR DATOS ?

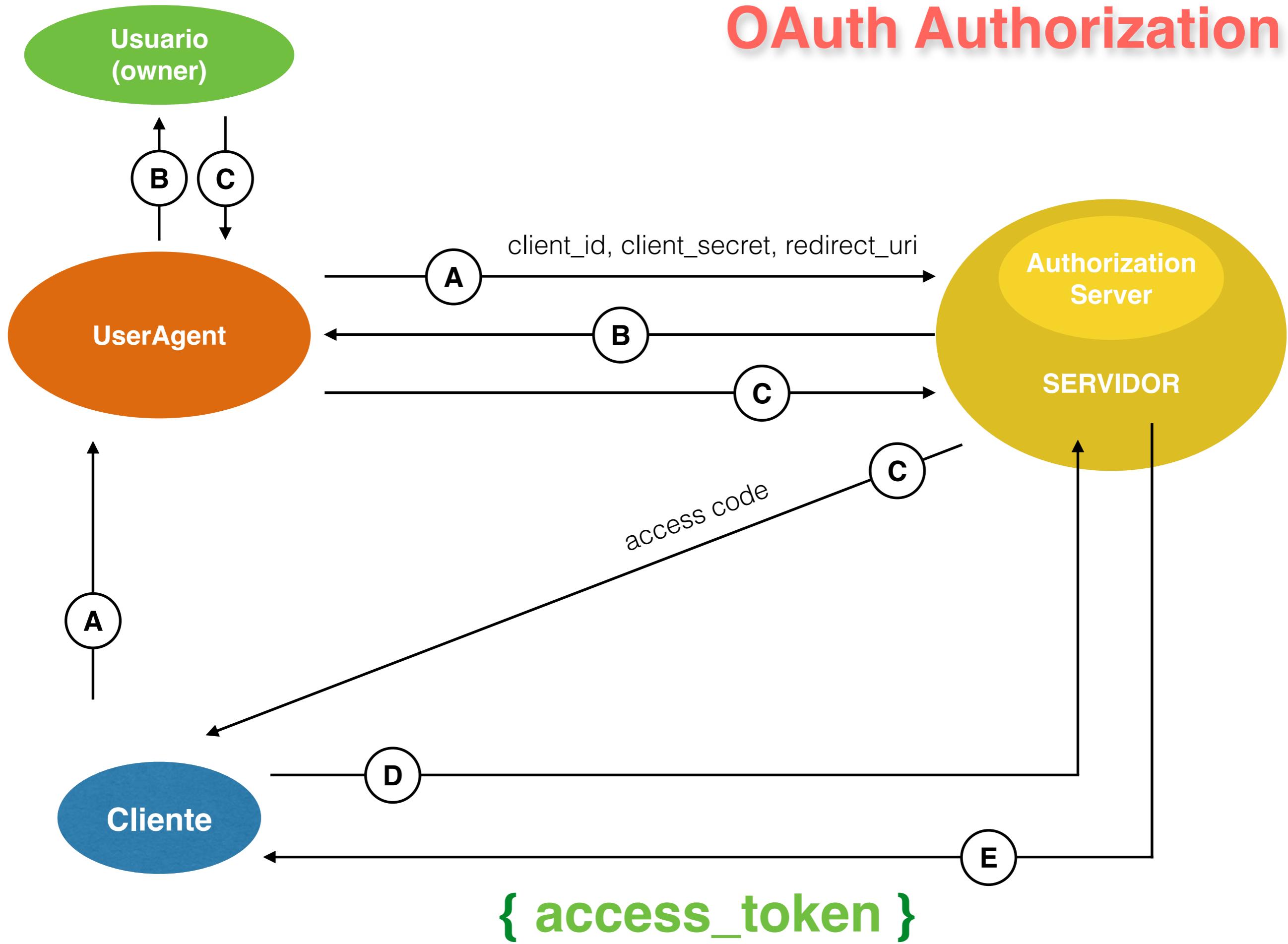


App Description: Buenos Aires WordCamp 2015  
App URL: <http://retrofox.github.io/WCBA2015>

## OAuth Information

<b>Client ID</b>	41012	<a href="#">Reset Key</a>
<b>Client Secret</b>	ZfoI8OhUoH12yTRDIBotLHPnq5qKgM8xAxynHOUHiFiiX3swRqkW18FoFr0No45	<a href="#">Reset Key</a>
<b>Redirect URL</b>	<a href="http://retrofox.github.io/WCBA2015">http://retrofox.github.io/WCBA2015</a>	
<b>Javascript Origins</b>	<a href="http://retrofox.github.io">http://retrofox.github.io</a>	
<b>Type</b>	Web	
<b>Request token URL</b>	<a href="https://public-api.wordpress.com/oauth2/token">https://public-api.wordpress.com/oauth2/token</a>	
<b>Authorize URL</b>	<a href="https://public-api.wordpress.com/oauth2/authorize">https://public-api.wordpress.com/oauth2/authorize</a>	
<b>Authenticate URL</b>	<a href="https://public-api.wordpress.com/oauth2/authenticate">https://public-api.wordpress.com/oauth2/authenticate</a>	

# OAuth Authorization



# node-wpcom-oauth



Automattic / node-wpcom-oauth

Unwatch ▾

108



WordPress.com OAuth2 authorization module for Node.js — Edit

97 commits

3 branches

7 releases

5 contributors



branch: master ▾

[node-wpcom-oauth](#) / +



Merge pull request #11 from Automattic/add/port-setting ...

nylen authored on Apr 14

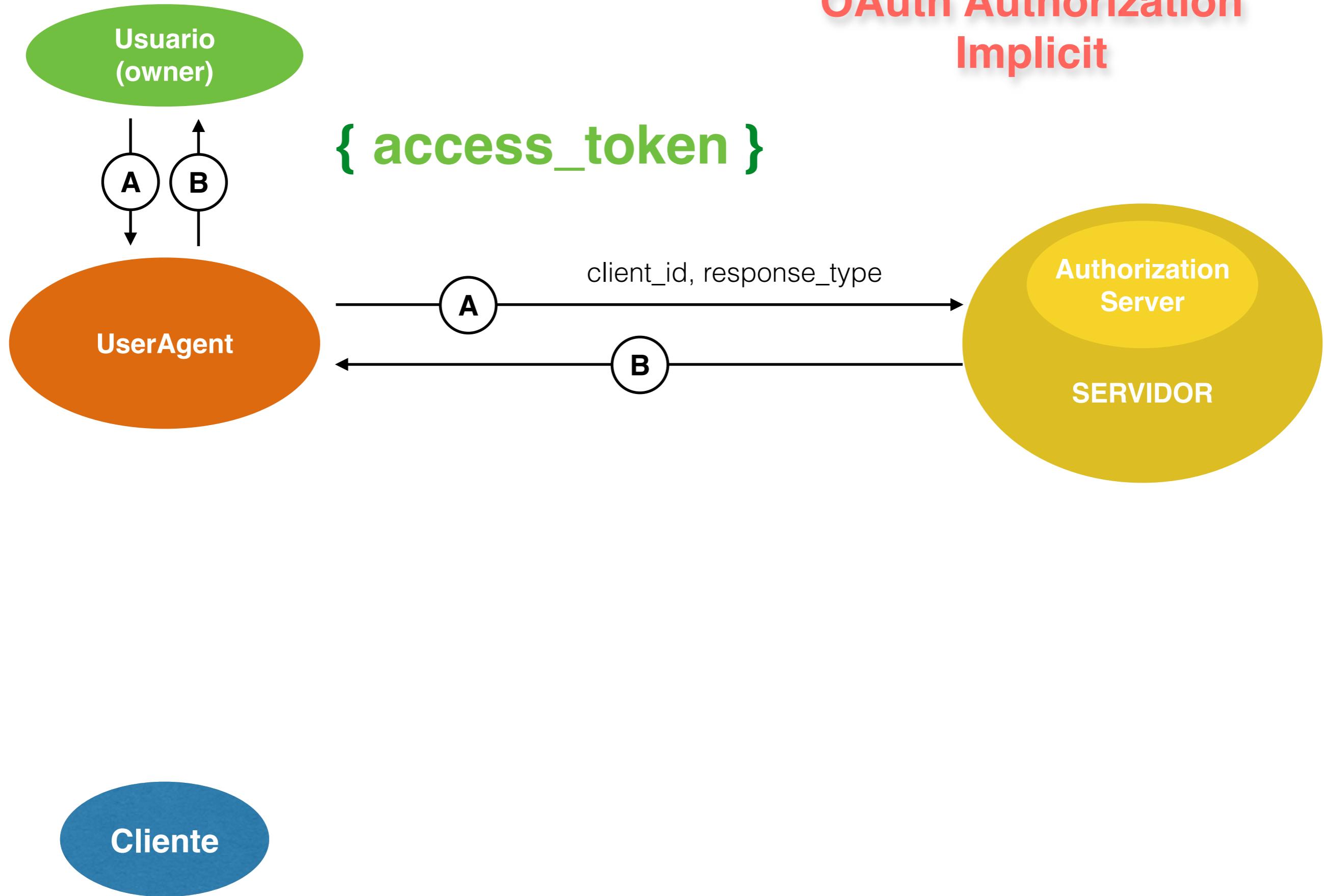
latest commit ee3ccbdc16

example	Fix typo	2 months ago
.gitignore	Example app: Allow changing port, set default to 3001	2 months ago
History.md	Release 0.3.3	2 months ago
LICENSE	add LICENSE file	a year ago
Readme.md	Final fixes to package names and docs crosslink	2 months ago
default.json	default: set grant_type as authorization_code by default	a year ago
index.js	core: reduce setting options	a year ago
package.json	Release 0.3.3	2 months ago

[Readme.md](#)

**wpcom-oauth**

# OAuth Authorization Implicit



# wpcom-oauth-cors



Release 1.0.0		
	TooTallNate authored on Apr 9	latest commit 9ef1a2b6b6
	Update distribution files too	2 months ago
	Missed a couple of spots	2 months ago
	ignoring npm-debug.log file	7 months ago
	Release 1.0.0	2 months ago
	make: update to new module name	6 months ago
	Match readme spacing from node-wpcom-oauth	2 months ago
	core: remove break line	6 months ago
	Release 1.0.0	2 months ago

README.md

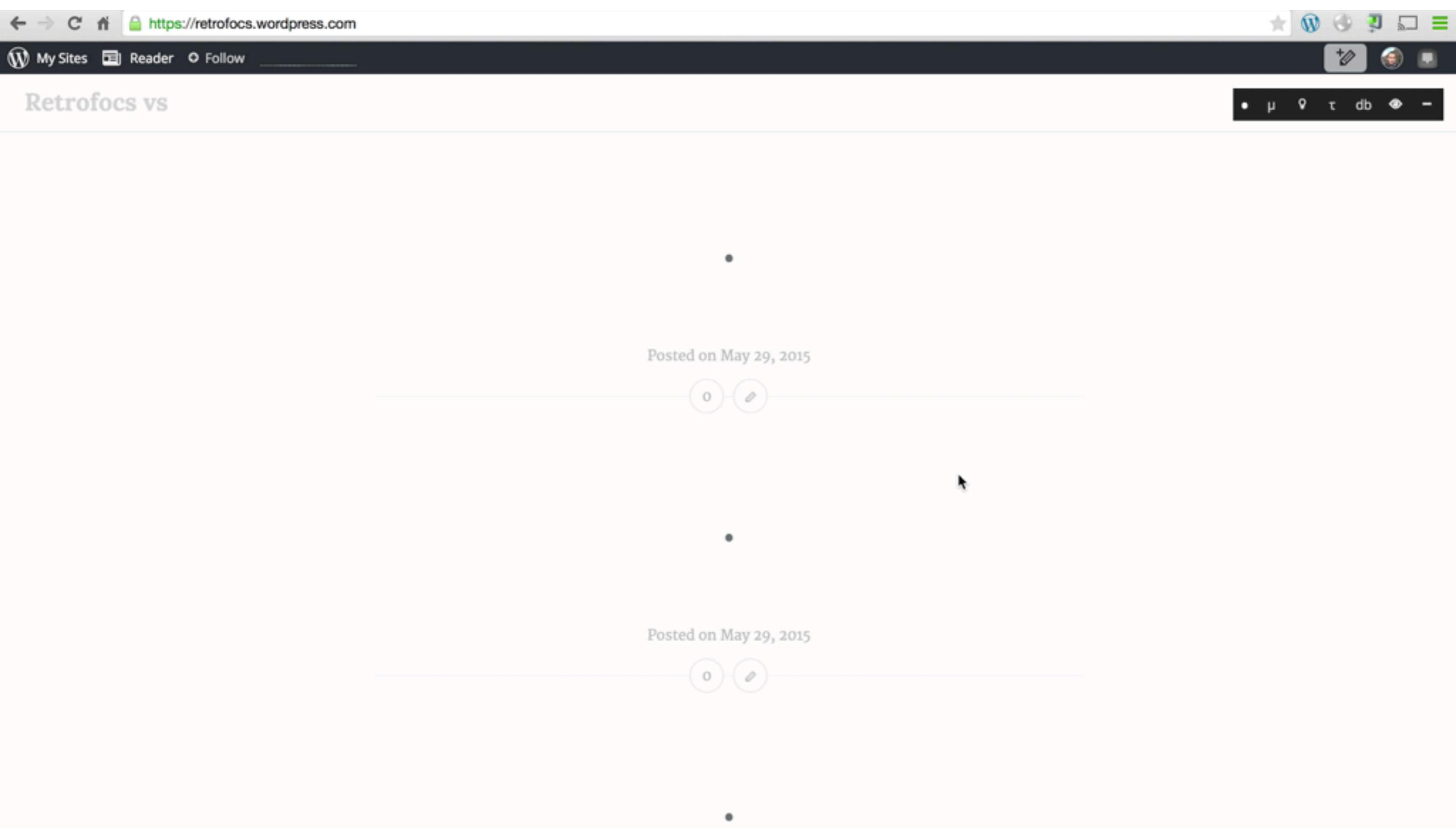
## wpcom-oauth-cors

WordPress.com implicit OAuth2 client-side authorization module.

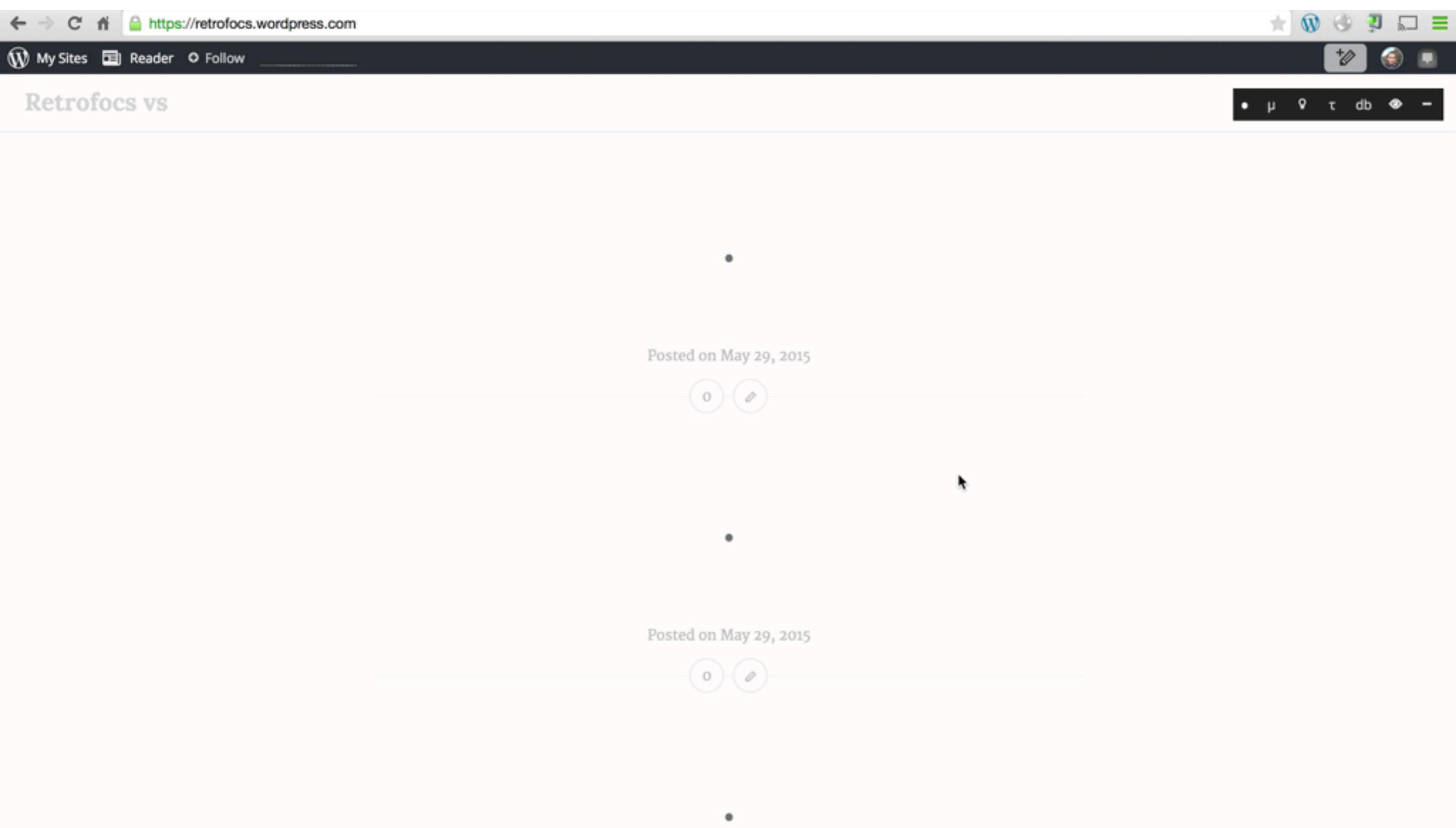
# wpcom-oauth-cors

```
/**  
 * Module dependencies  
 */  
  
var WPCOM = require('wpcom');  
var OAUTH = require('wpcom-oauth-cors');  
var oauth = OAUTH('41012', { scope: "global" });  
  
oauth.get(function(data){  
    // Create wpcom instance given the token  
    var wpcom = WPCOM(data.access_token);  
});
```

# Ejemplo client-side: add/remove post



# Ejemplo client-side: add/remove post



?