

Cross platform web app development

Tomás Pérez - [@tomasperezv](#)
José M. Pérez - [@jmperezperez](#)



Contents

Contents

Web development in Spotify

- Why and how
- Benefits and learnings

Key web technologies that we use

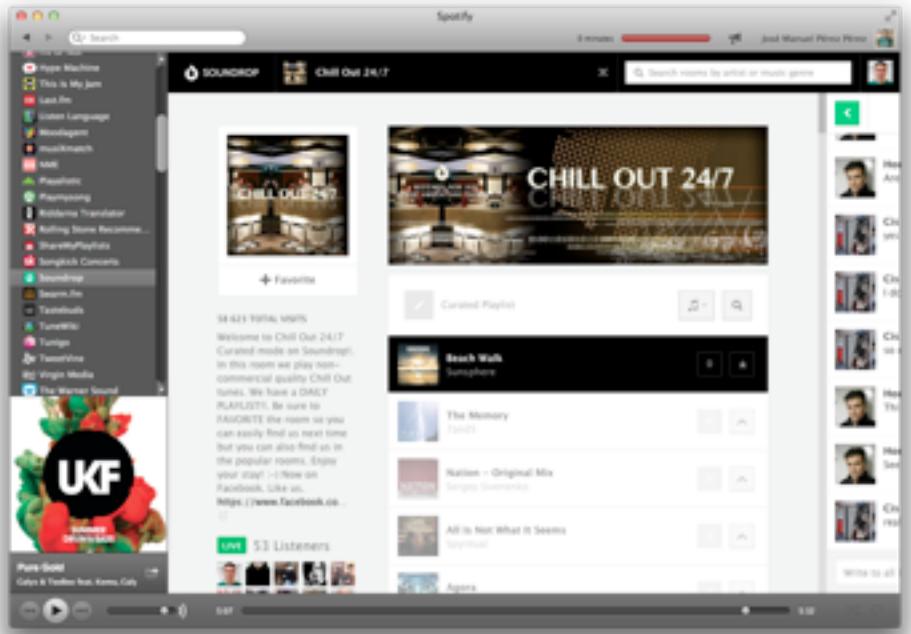
- IFrames and postMessage: security and performance
- Communication between different tabs

Summary

Questions

Spotify <3 web

Spotify <3 web



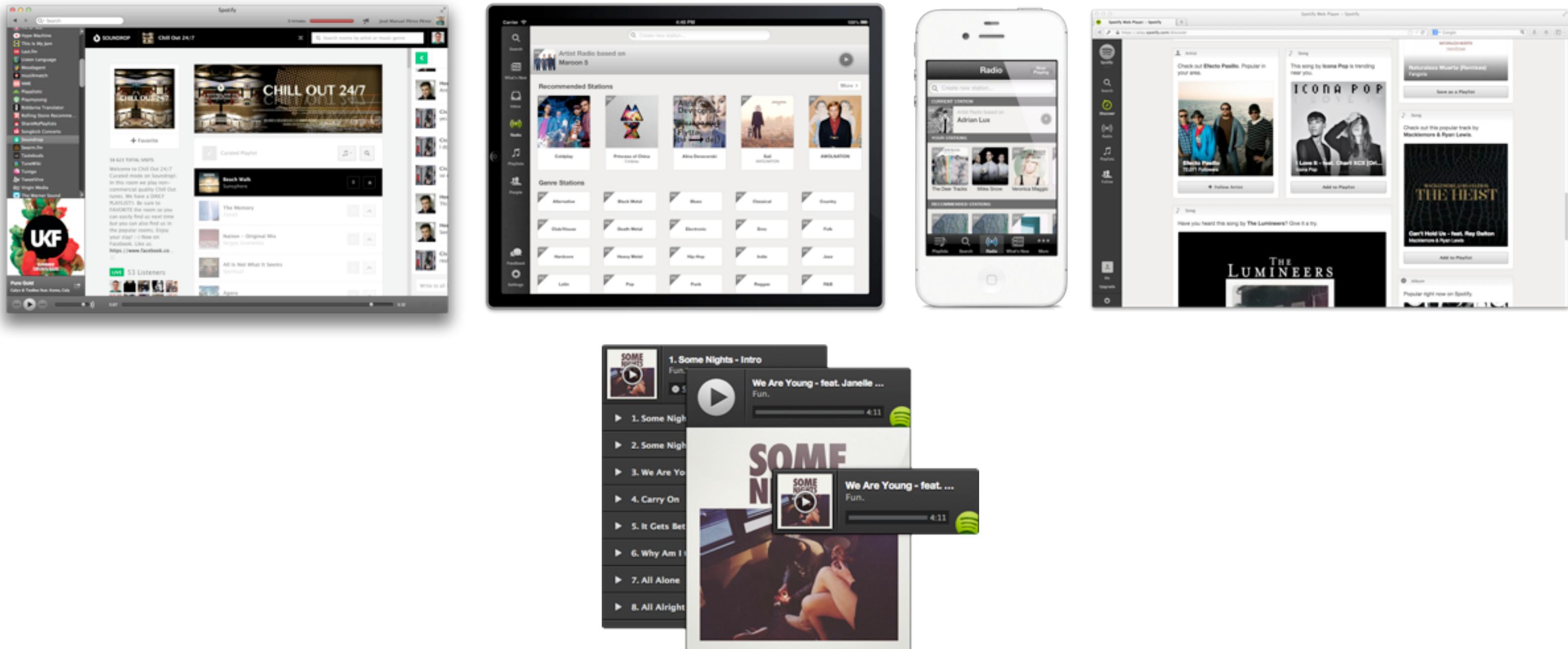
Spotify <3 web



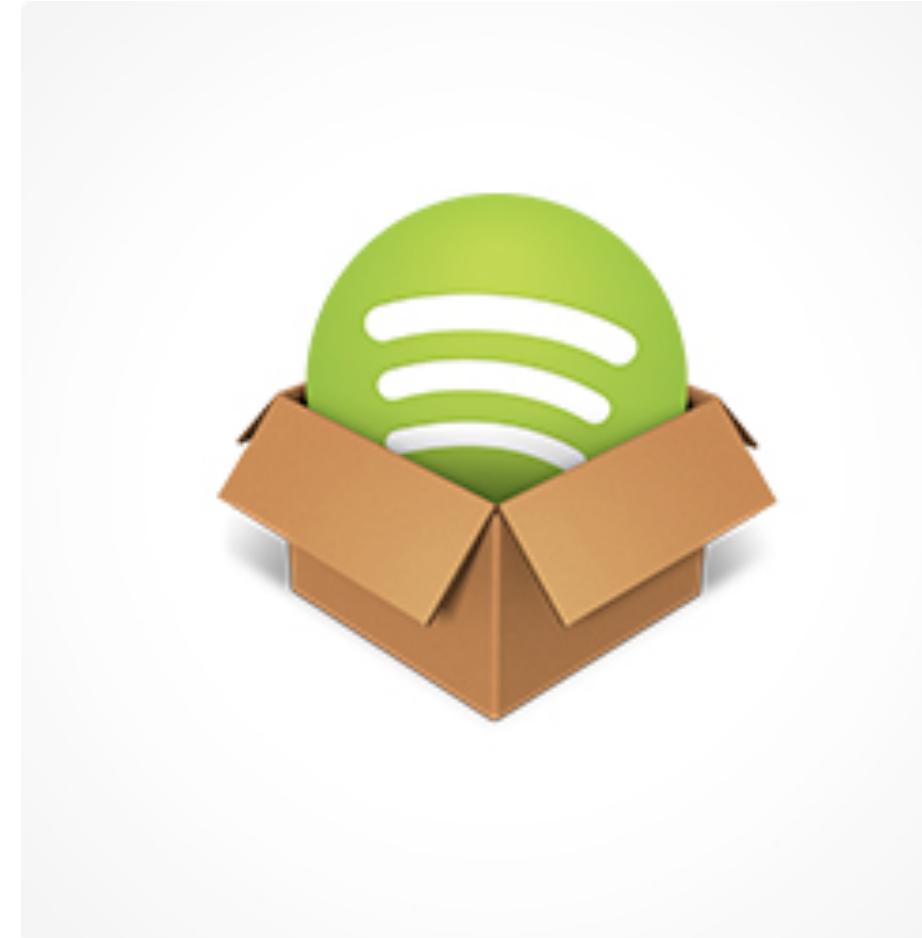
Spotify <3 web



Spotify <3 web



Beginnings of Spotify



Giving web a try

Giving web a try

- Using CEF to move features to web

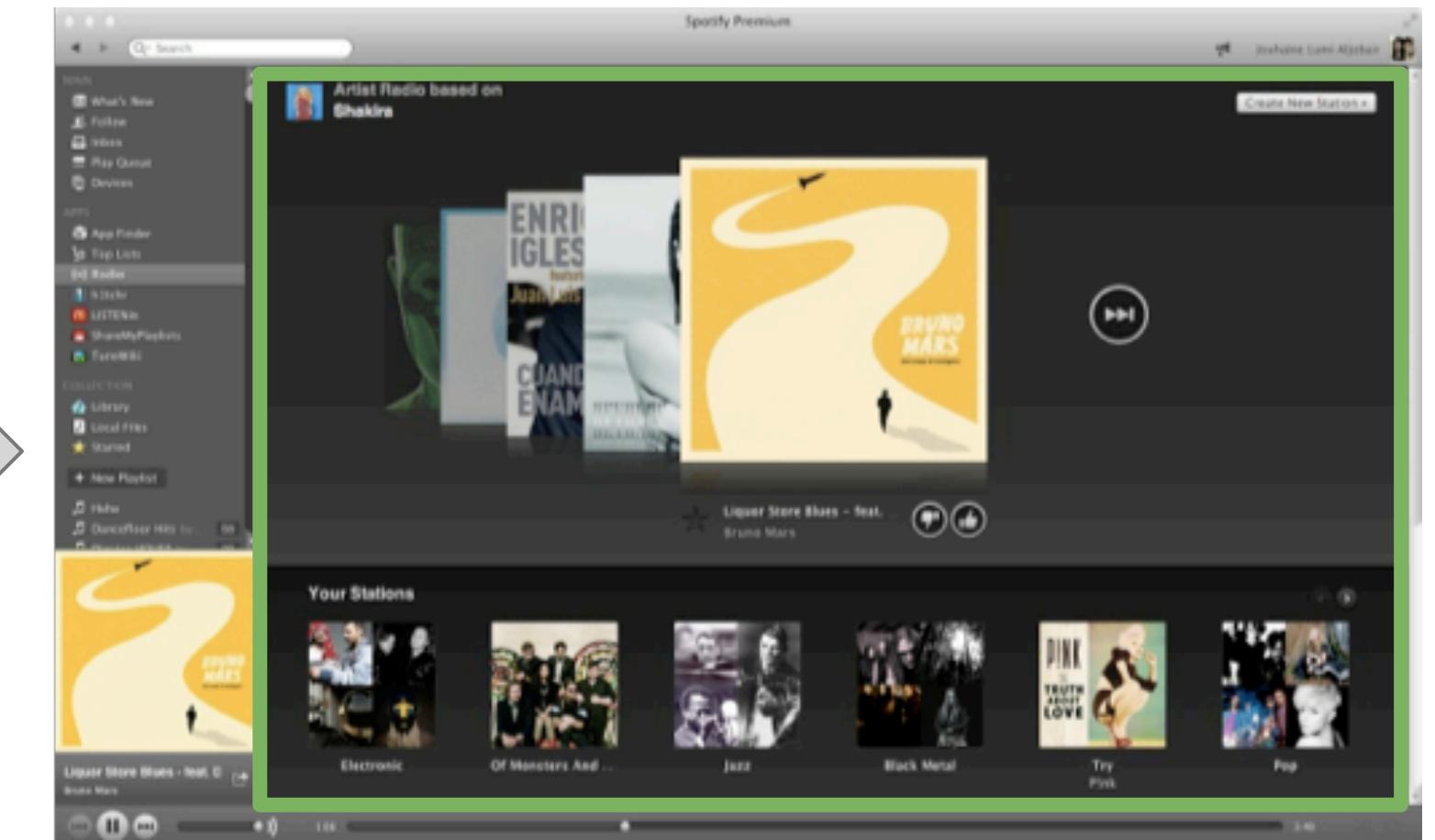
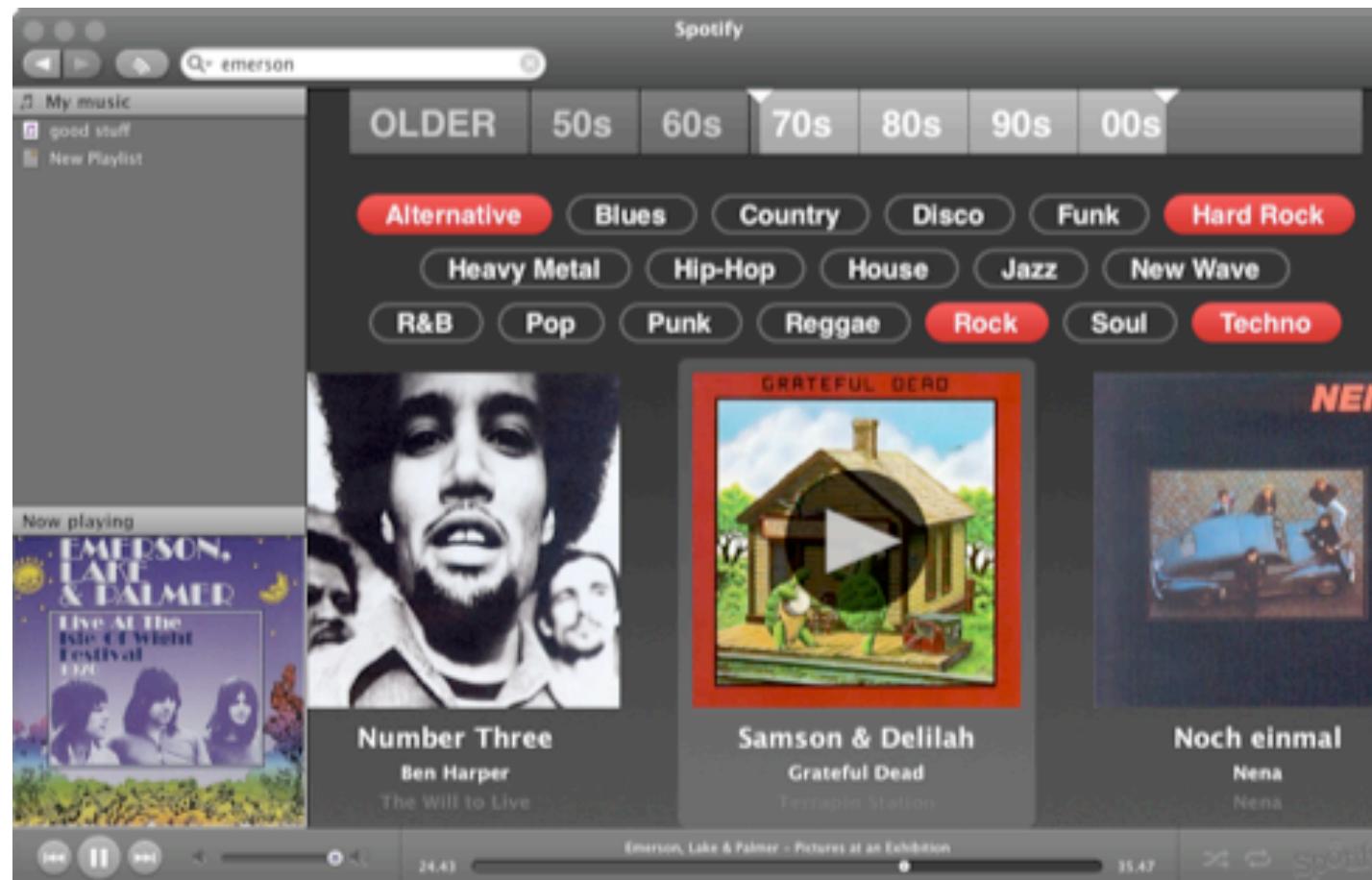
Giving web a try

- Using CEF to move features to web
- Write features as webapps

Giving web a try

- Using CEF to move features to web
- Write features as webapps
- Sandboxed environment

Giving web a try (2)



What is a Spotify app?

What is a Spotify app?

- HTML

What is a Spotify app?

- HTML
- CSS

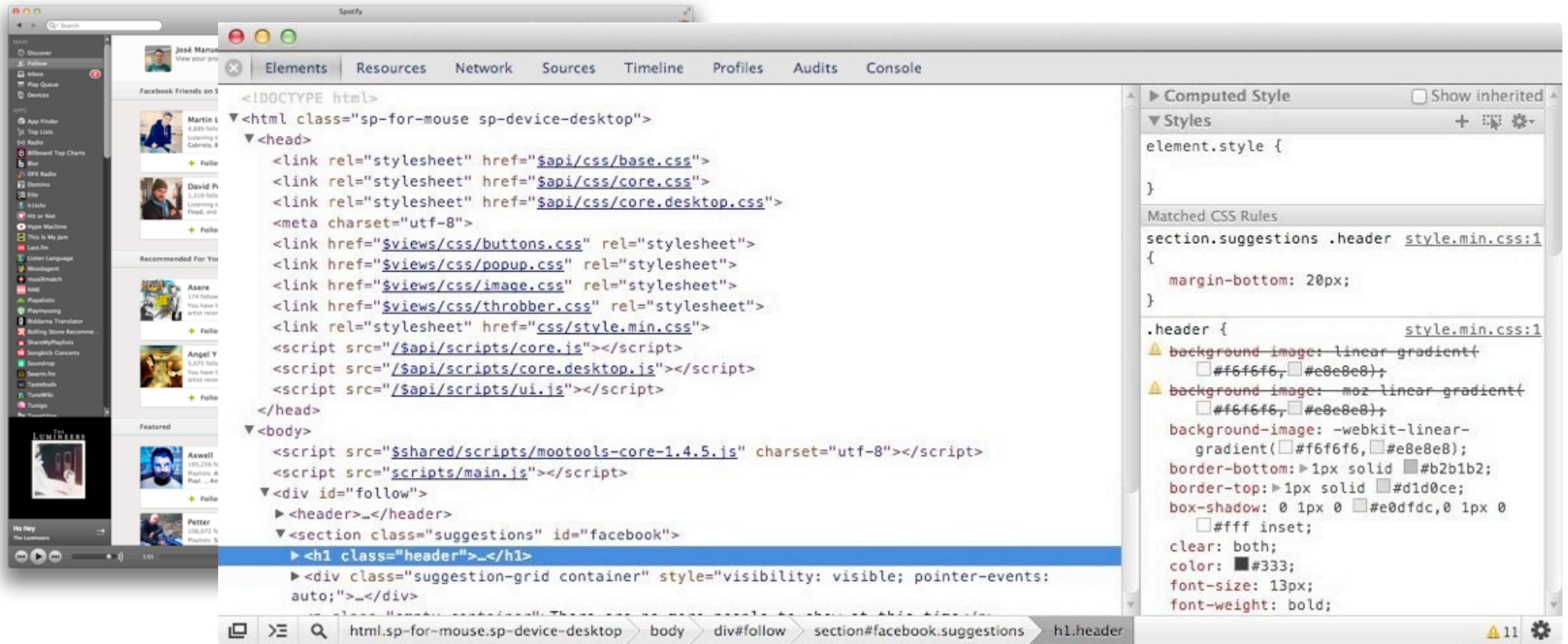
What is a Spotify app?

- HTML
- CSS
- JS

What is a Spotify app?

- HTML
- CSS
- JS
- manifest.json

What is a Spotify app? – Developer tools

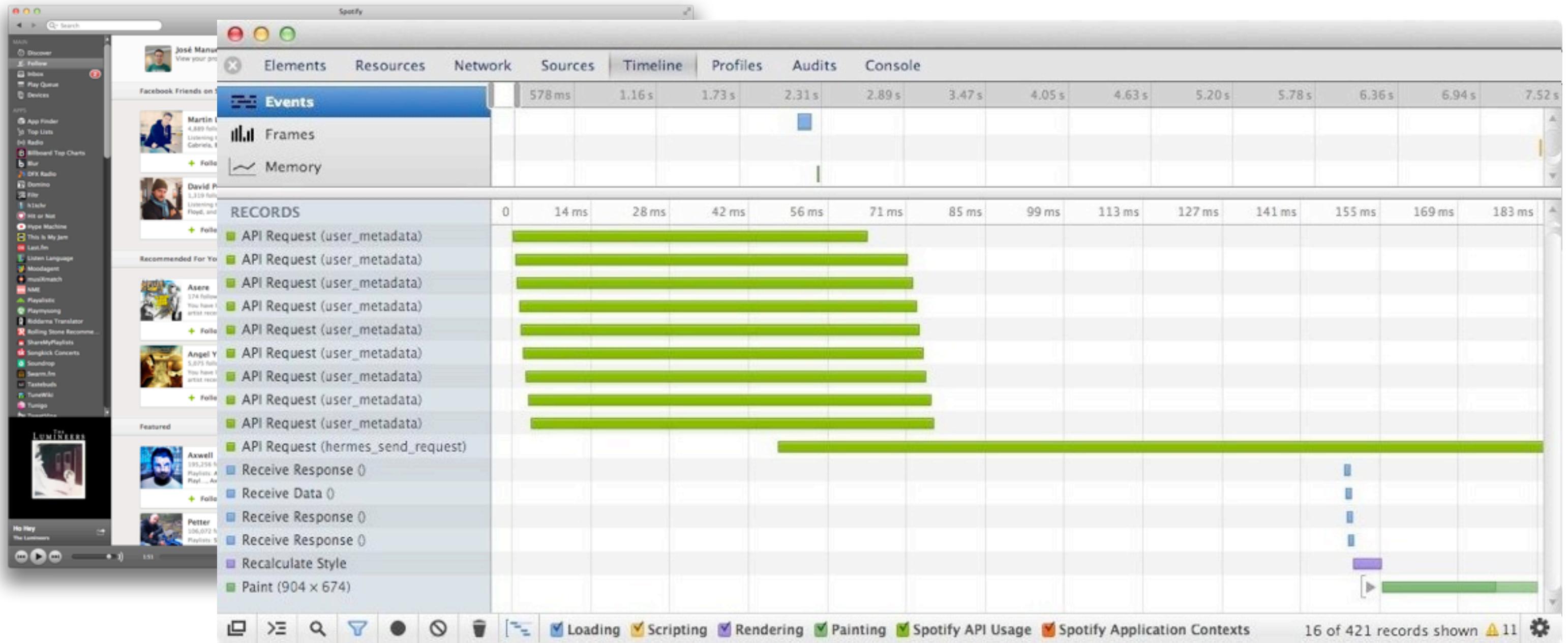


What is a Spotify app? – Developer tools

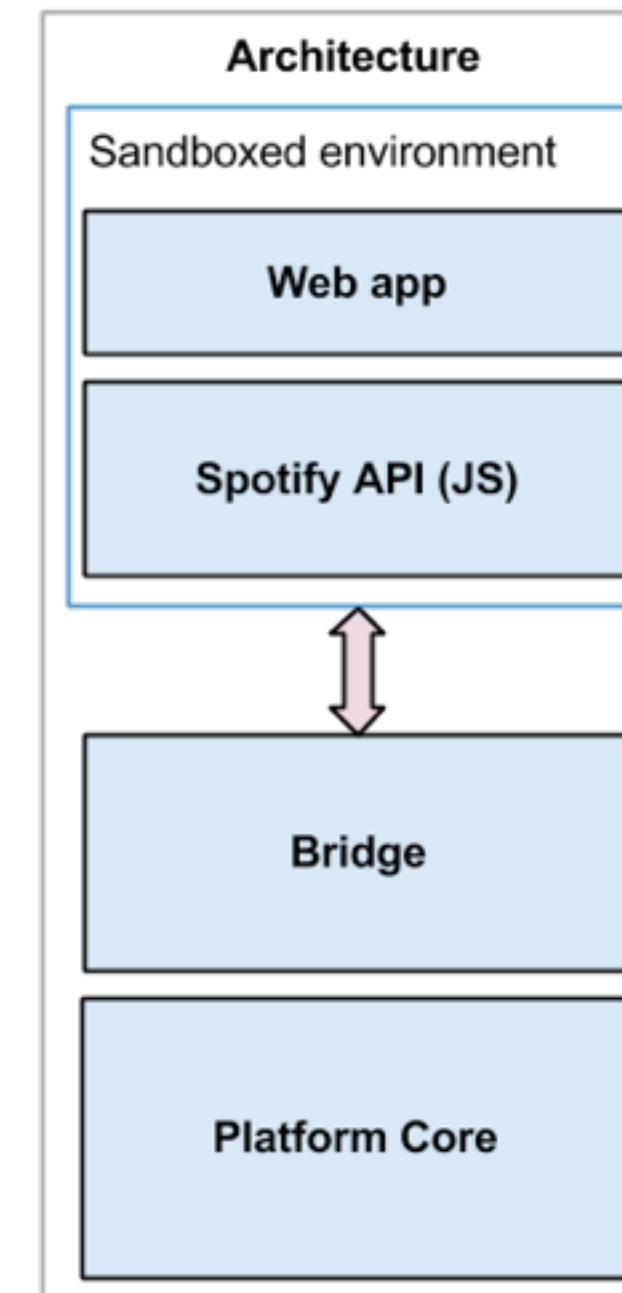
The screenshot shows the Spotify desktop application interface with developer tools overlaid. The main window displays a user's profile and a 'Facebook Friends' section. On the left, a sidebar lists various apps and features. The developer tools are open, showing the Network tab with a list of network requests. The table includes columns for Name, Path, Method, Status, Type, Initiator, Size, Time, and Timeline.

Name	Path	Method	Status	Type	Initiator	Size	Time	Timeline
			Text			Content	Latency	
369108_656177956_174597	profile.ak.fcdn.net/hprofile-;	GET	200 OK	image/j...	image.js:23	8.4 KB	494 ms	
203445_819155496_431317	profile.ak.fcdn.net/hprofile-;	GET	200 OK	image/j...	image.js:23	14.2 KB	492 ms	
274768_639185200_682545	profile.ak.fcdn.net/hprofile-;	GET	200 OK	image/j...	image.js:23	5.3 KB	489 ms	
plus.png	/img	GET	200 OK	image/...	suggestiongrid.js	1.0 KB	342 ms	
button-icons.png	/\$views/img	GET	200 OK	image/...	suggestiongrid.js	3.2 KB	359 ms	
button.png	/\$views/img	GET	200 OK	image/...	suggestiongrid.js	3.3 KB	413 ms	
e9bba4f0021380f46ba6084:	profile-images.scdn.co/artists	GET	200 OK	application/x-	image.js:23	6.0 KB	929 ms	
54d6c9986b6f7e72696f7ad:	profile-images.scdn.co/artists	GET	200 OK	application/x-	image.js:23	6.2 KB	1.07 s	
01bb1178d040554f670f2e6	profile-images.scdn.co/artists	GET	200 OK	application/x-	image.js:23	7.3 KB	763 ms	

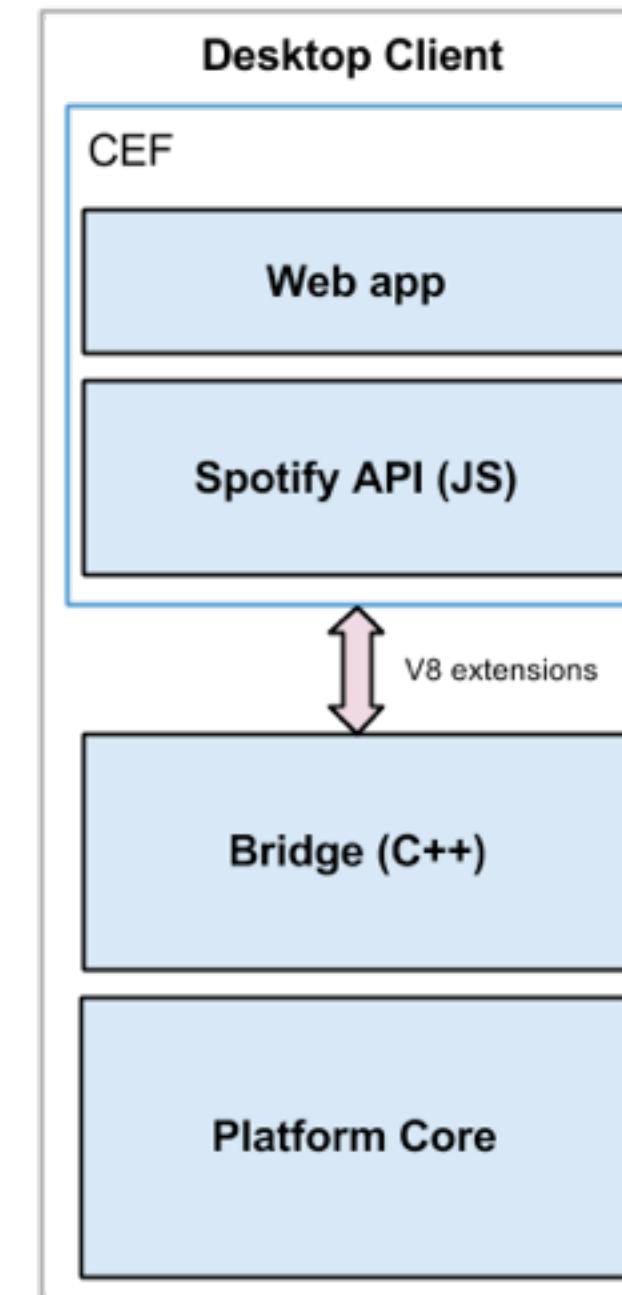
What is a Spotify app? – Developer tools



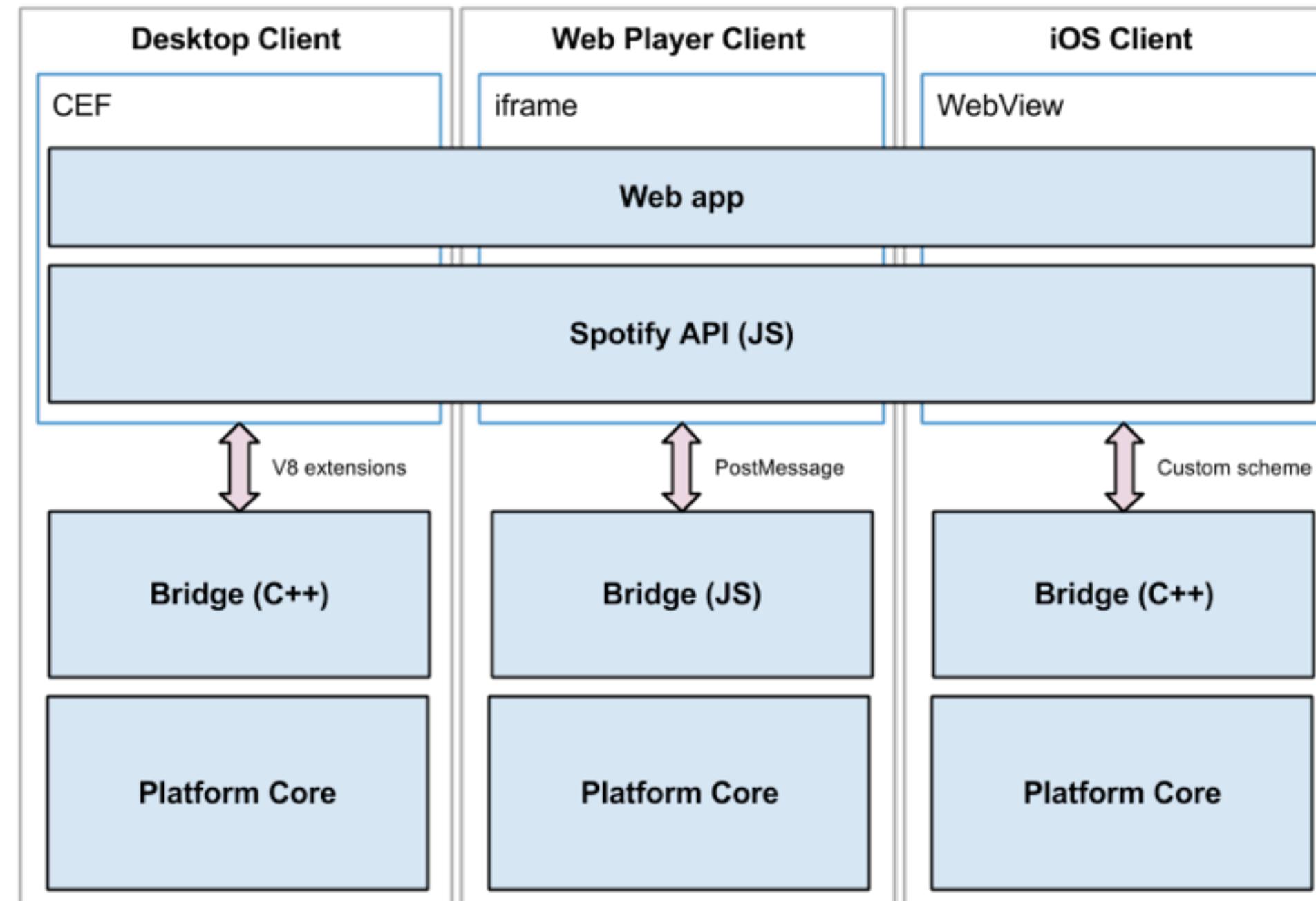
Architecture of the Spotify Apps Platform – overview



Architecture of the Spotify Apps Platform – desktop



Architecture of the Platform – all clients



Multiple platforms



The Songkick website interface shows a concert for Bob Dylan at the Stockholm Waterfront on Sunday, October 13, 2013. The page features a large image of Bob Dylan. Below the image, it says '66 upcoming concerts by Bob Dylan'. A list of upcoming shows is provided, including:

Date	Venue	Time	Artist
27 Jun	Cruzan Amphitheatre	6:10	Bob Dylan
28 Jun	Multiflora Credit Union Amphitheatre at the Fonda State Fairgrounds	7:30	Bob Dylan
30 Jun	Aaron's Amphitheatre at Lakewood	7:30	Bob Dylan
1 Jul	The Lawn at Riverfront Park	7:30	Bob Dylan

At the bottom, there are buttons for 'Create your personalized concert calendar now' and 'Start tracking concerts'.

The Spotify Web Player interface shows a concert for Bob Dylan at the Stockholm Waterfront on Sunday, October 13, 2013. The page features a large image of Bob Dylan. Below the image, it says '66 upcoming concerts by Bob Dylan'. A list of upcoming shows is provided, including:

Date	Venue	Time	Artist
27 Jun	Cruzan Amphitheatre	6:10	Bob Dylan
28 Jun	Multiflora Credit Union Amphitheatre at the Fonda State Fairgrounds	7:30	Bob Dylan
30 Jun	Aaron's Amphitheatre at Lakewood	7:30	Bob Dylan
1 Jul	The Lawn at Riverfront Park	7:30	Bob Dylan

At the bottom, there are buttons for 'Create your personalized concert calendar now' and 'Start tracking concerts'.

Releasing Spotify Apps

Releasing Spotify Apps

- Decoupled system

Releasing Spotify Apps

- Decoupled system
- Provides gradual roll-out, and availability depending on platform and country

Benefits

Benefits

- Feature ownership by teams

Benefits

- Feature ownership by teams
- Fast development pace due to fewer dependencies

Benefits

- Feature ownership by teams
- Fast development pace due to fewer dependencies
- Less hassle to deploy. Makes it easy to tweak a feature after being released

Benefits

- Feature ownership by teams
- Fast development pace due to fewer dependencies
- Less hassle to deploy. Makes it easy to tweak a feature after being released
- 3rd party app development

Only benefits?

Only benefits?

- Performance in views with a lot of data on mobile

Only benefits?

- Performance in views with a lot of data on mobile
- Time to render

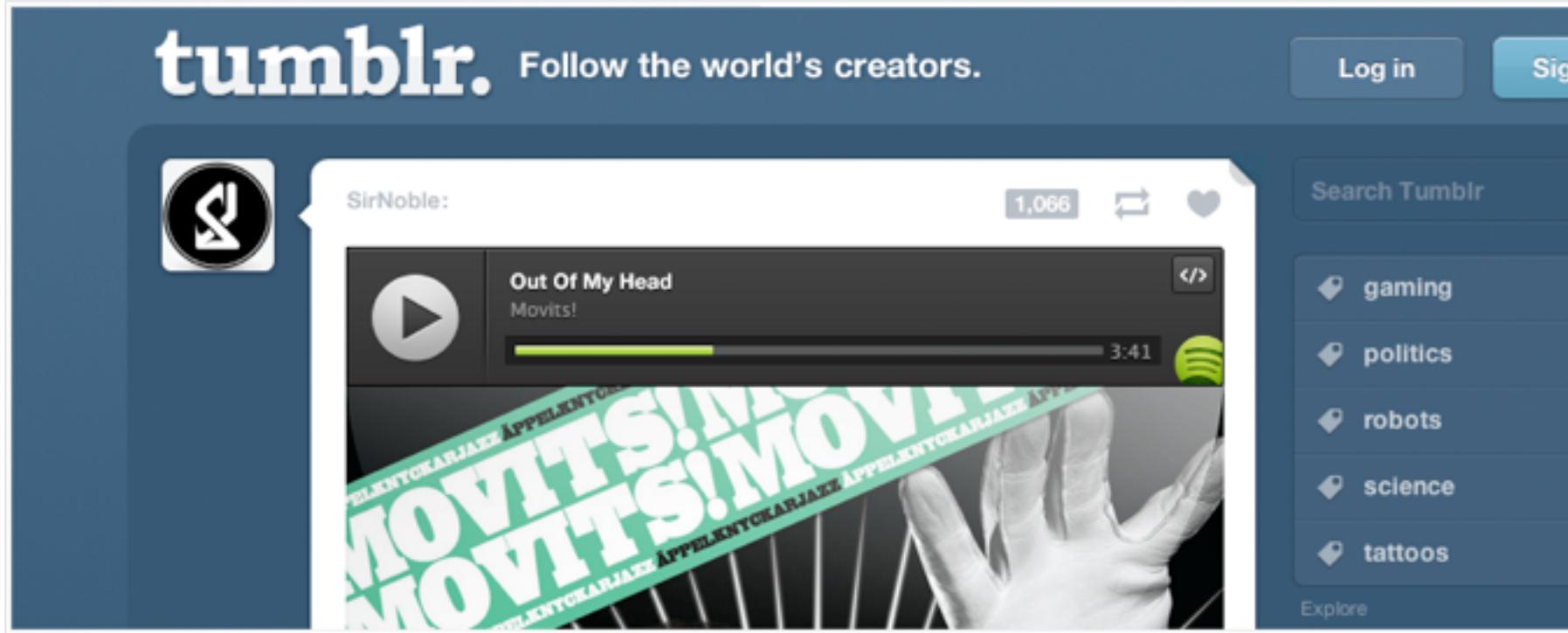
Spotify Embedded Platform

Tomás Pérez

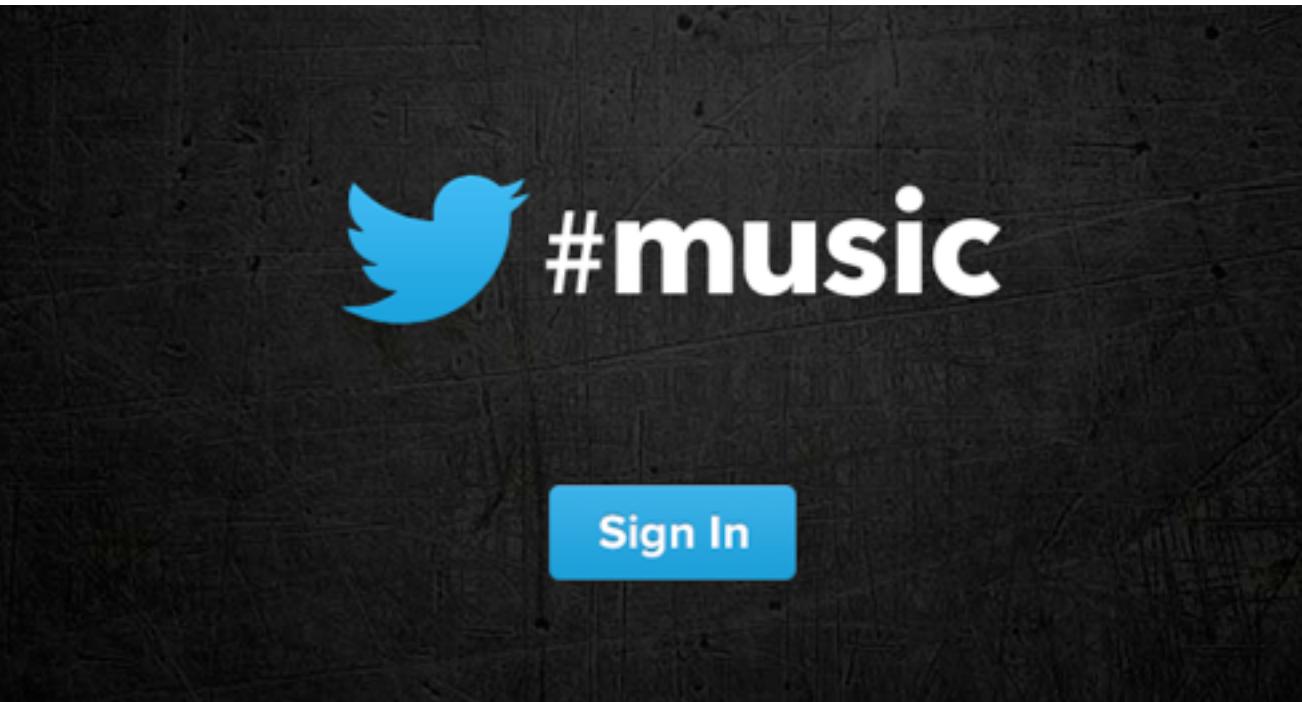
twitter.com/tomasperezv
github.com/tomasperezv



Spotify Embedded Platform



Spotify Embedded Platform



José Manuel Pérez Pérez
Ya no estoy seguro de si
debemos incluir la 16 y la 18.
Creo que con la slide de
Contents del principio deberia
quedar claro.

Content

José Manuel Pérez Pérez
Ya no estoy seguro de si
debemos incluir la 16 y la 18.
Creo que con la slide de
Contents del principio deberia
quedar claro.

Content

Platform architecture

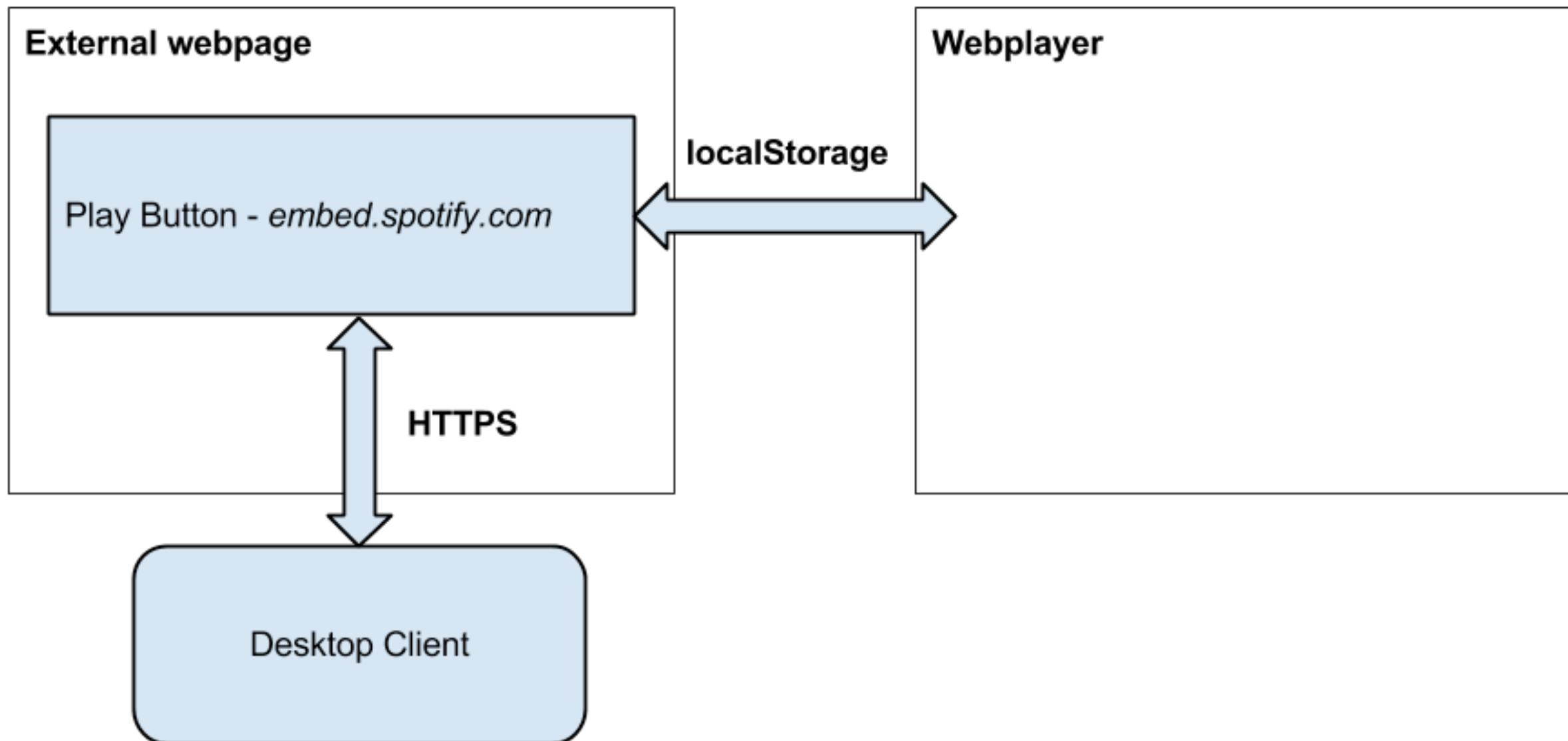
- Iframes and postMessage: security and performance

Remote control technology

- Communication between different tabs
- Performance of the communication channel

Performance improvements

Architecture of the Platform



Sandboxed platform using IFrame

Sandboxed platform using IFRAMES

Good

- Separated execution context
- Better security and privacy
- We control the release process

Sandboxed platform using IFrames

Good

- Separated execution context
- Better security and privacy
- We control the release process

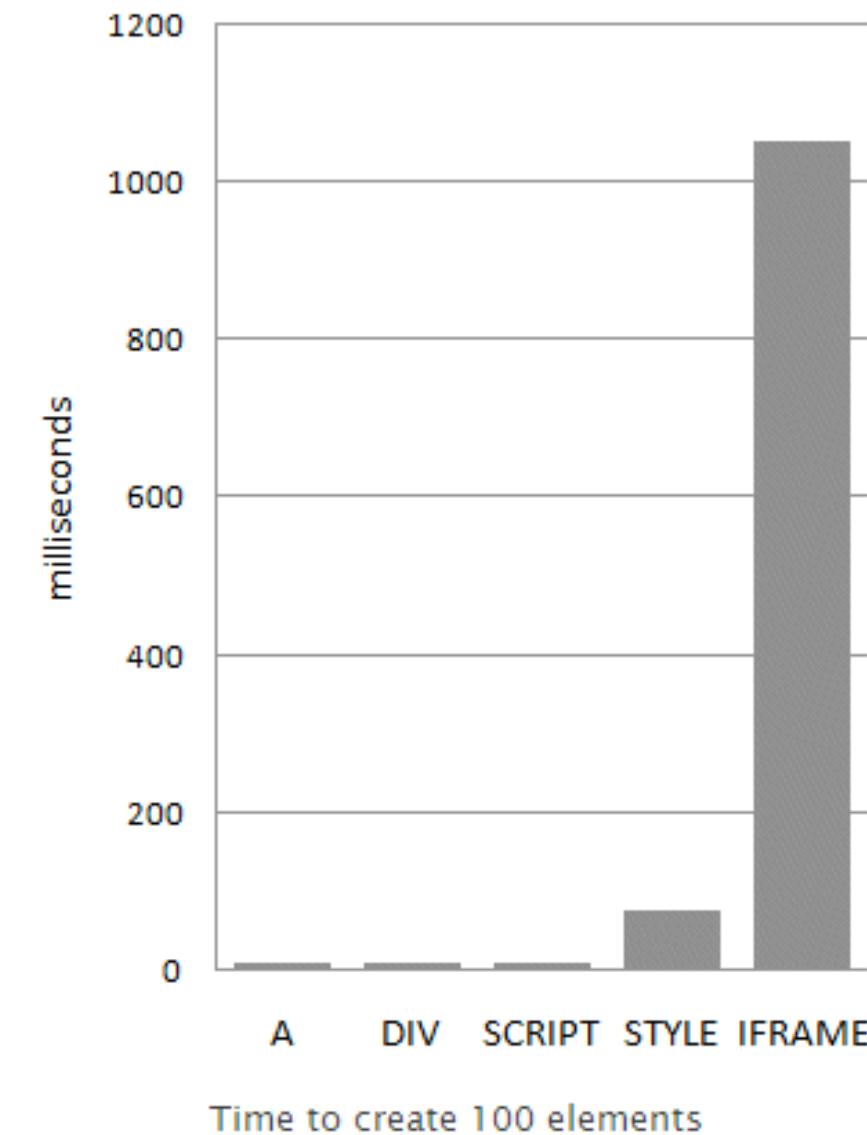
Bad

- More difficult communication
- IFrames are expensive

How expensive are IFrames?

How expensive are IFrames?

A
lot(*):



(*) <http://www.stevesouders.com/blog/2009/06/03/using-iframes-sparingly/>

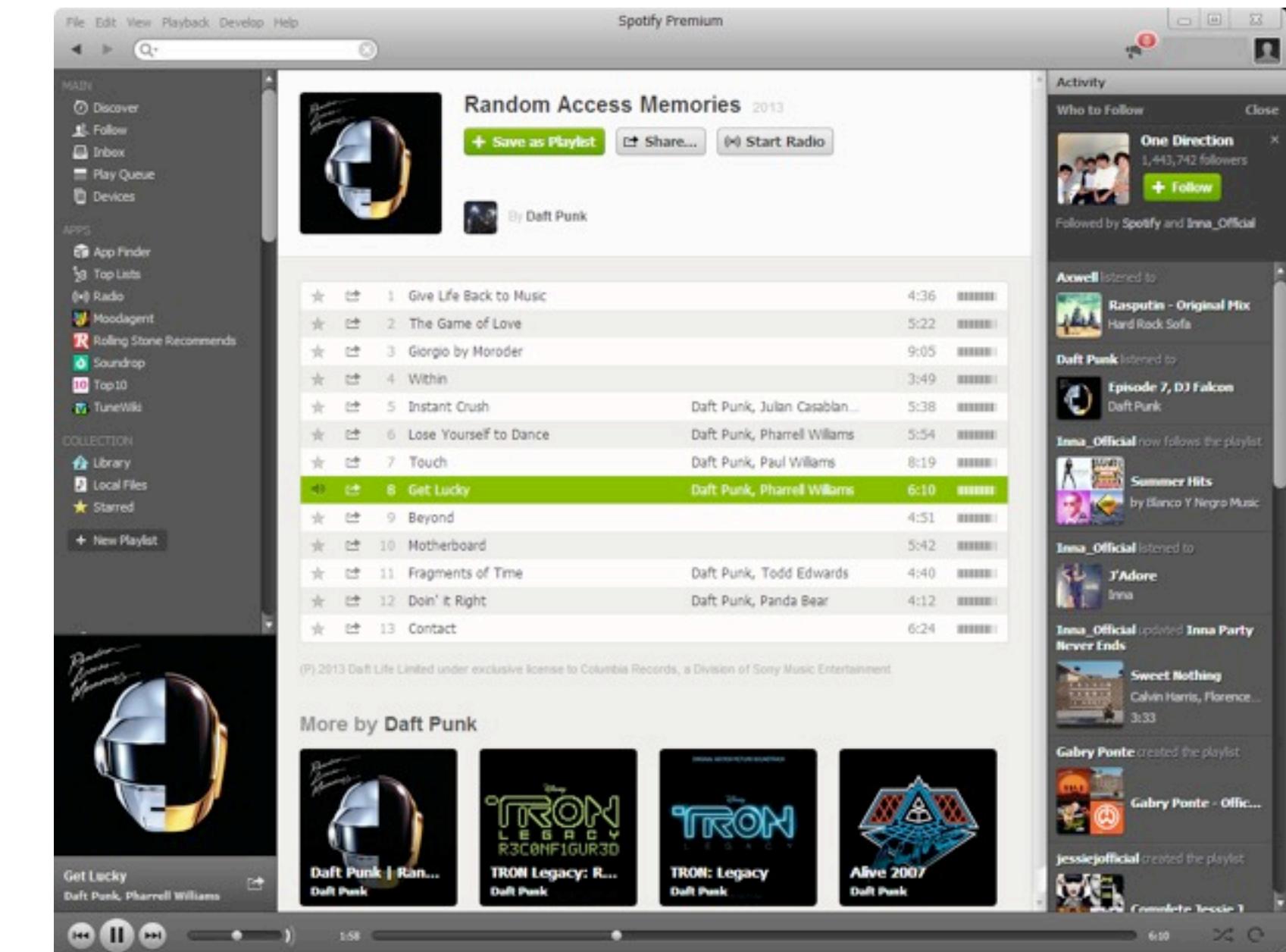
Custom widgets in partner pages

Random Access Memories by Daft Punk

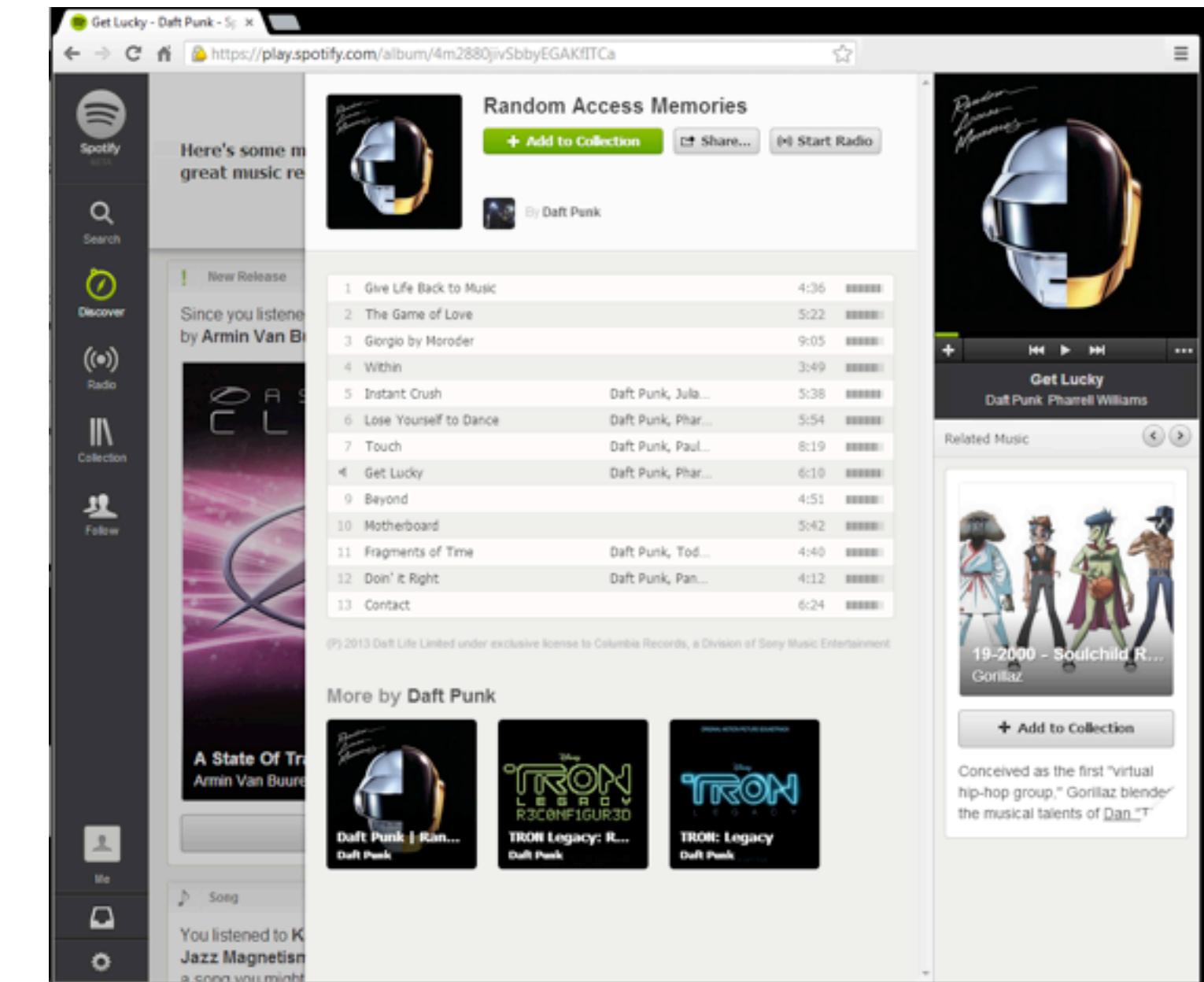
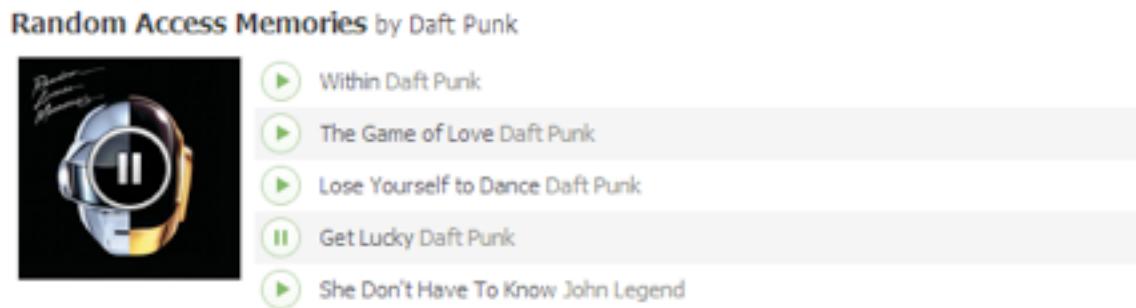


- ▶ Within Daft Punk
- ▶ The Game of Love Daft Punk
- ▶ Lose Yourself to Dance Daft Punk
- ▶ Get Lucky Daft Punk**
- ▶ She Don't Have To Know John Legend

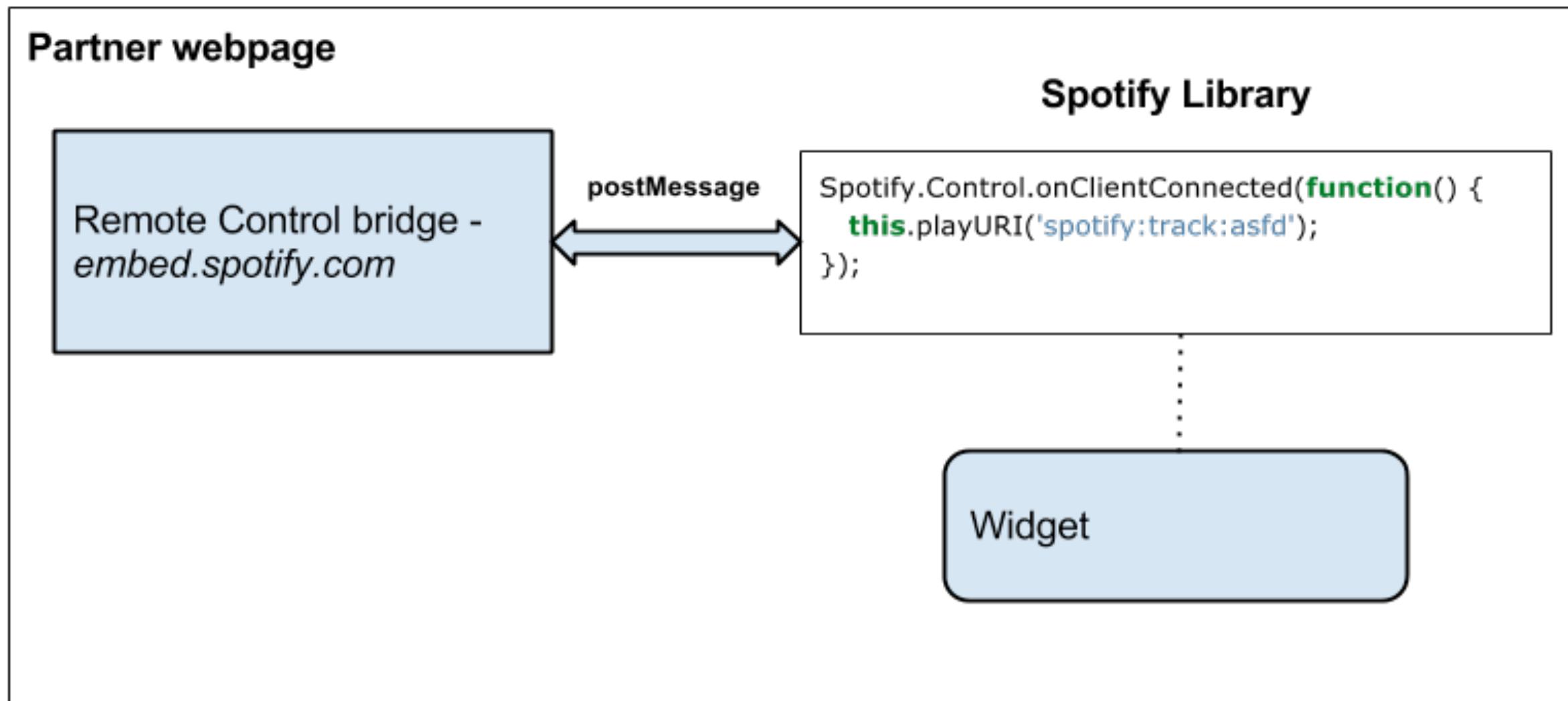
Custom widgets in partner pages



Custom widgets in partner pages



Cross-origin communication using postMessage



Sending: content of the message

```
frame.contentWindow.postMessage(message, ...);
```

Sending: content of the message

```
frame.contentWindow.postMessage(message, ...);
```

msg [in]

Type: **any**

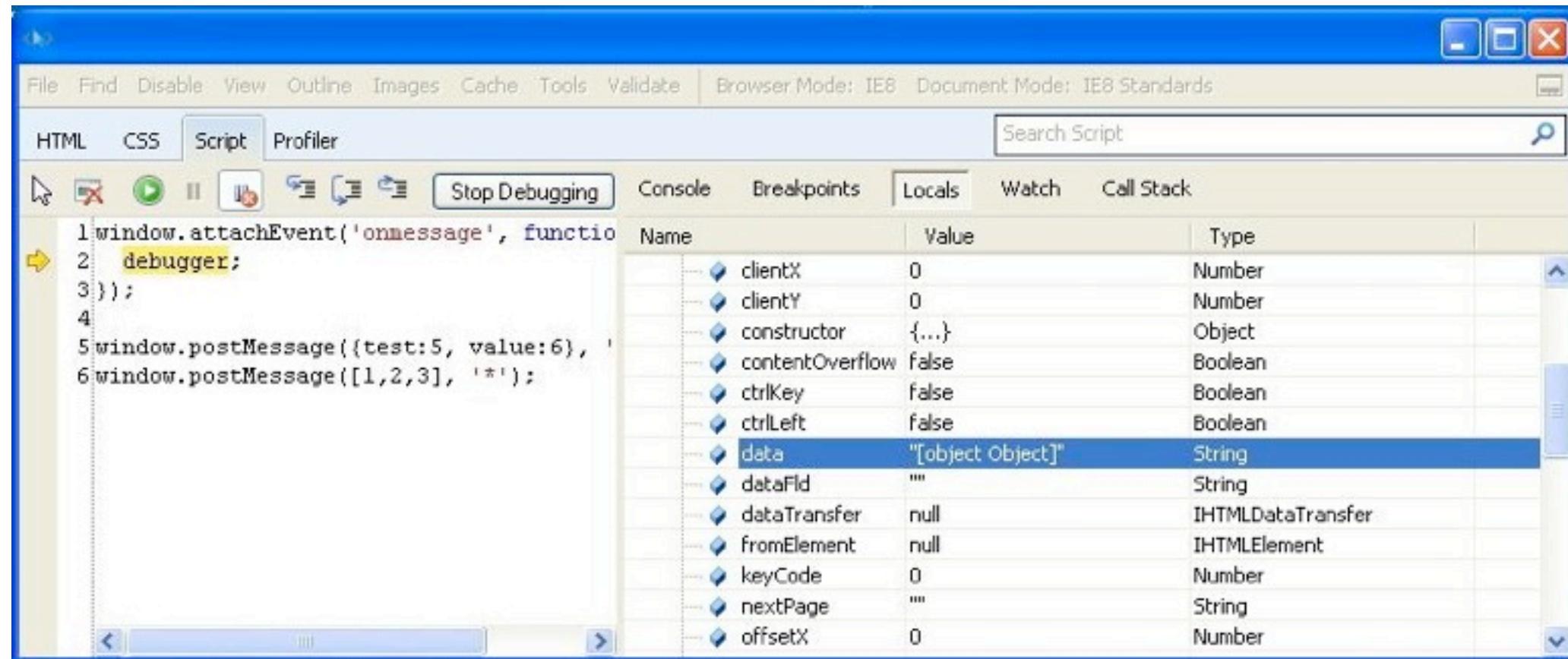
One of the following:

- JavaScript primitive, such as a string
- [**object**](#)
- Array
- ...



Sending: content of the message

```
frame.contentWindow.postMessage(message, ...);
```



Security

```
frame.contentWindow  
  .postMessage(..., 'https://embed.spotify.com');
```

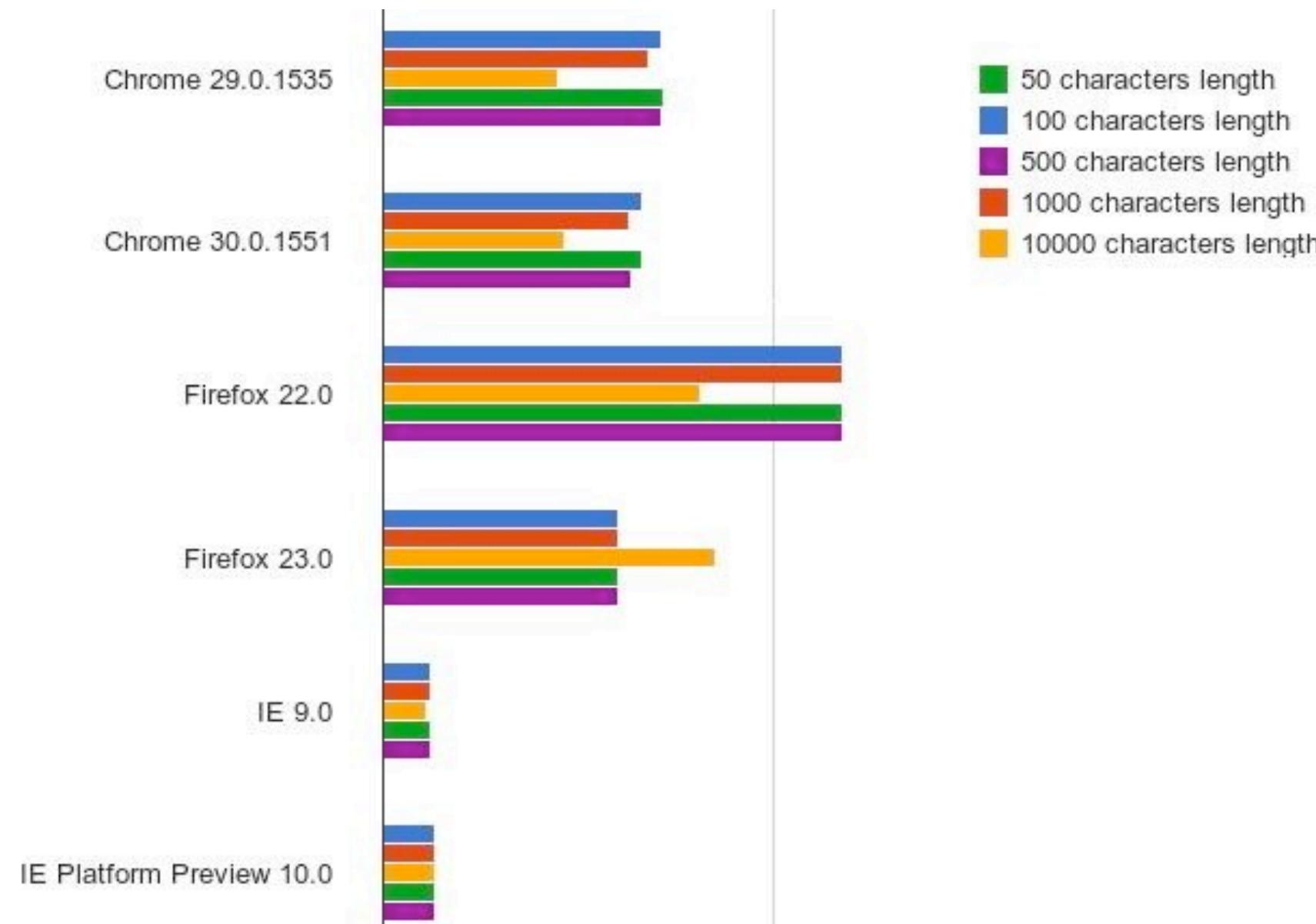
Security

```
frame.contentWindow
  .postMessage(..., 'https://embed.spotify.com');
```

```
window.addEventListener('message', function(messageEvt) {
  if (messageEvt.origin === 'https://valid-domain.com') {
    ...
  }
});
```

postMessage transfer rate

postMessage transfer rate



Remote controlling Spotify Clients

Client Desktop: based on long polling + https

Webplayer: communication with another browser window or tab

Remote controlling Spotify Clients

Client Desktop: based on long polling + https

Webplayer: communication with another browser window or tab

- server side

Remote controlling Spotify Clients

Client Desktop: based on long polling + https

Webplayer: communication with another browser window or tab

- server side
- postMessage

Remote controlling Spotify Clients

Client Desktop: based on long polling + https

Webplayer: communication with another browser window or tab

- server side
- postMessage
- Cookies

Remote controlling Spotify Clients

Client Desktop: based on long polling + https

Webplayer: communication with another browser window or tab

- server side
- postMessage
 - Cookies
 - **localStorage**

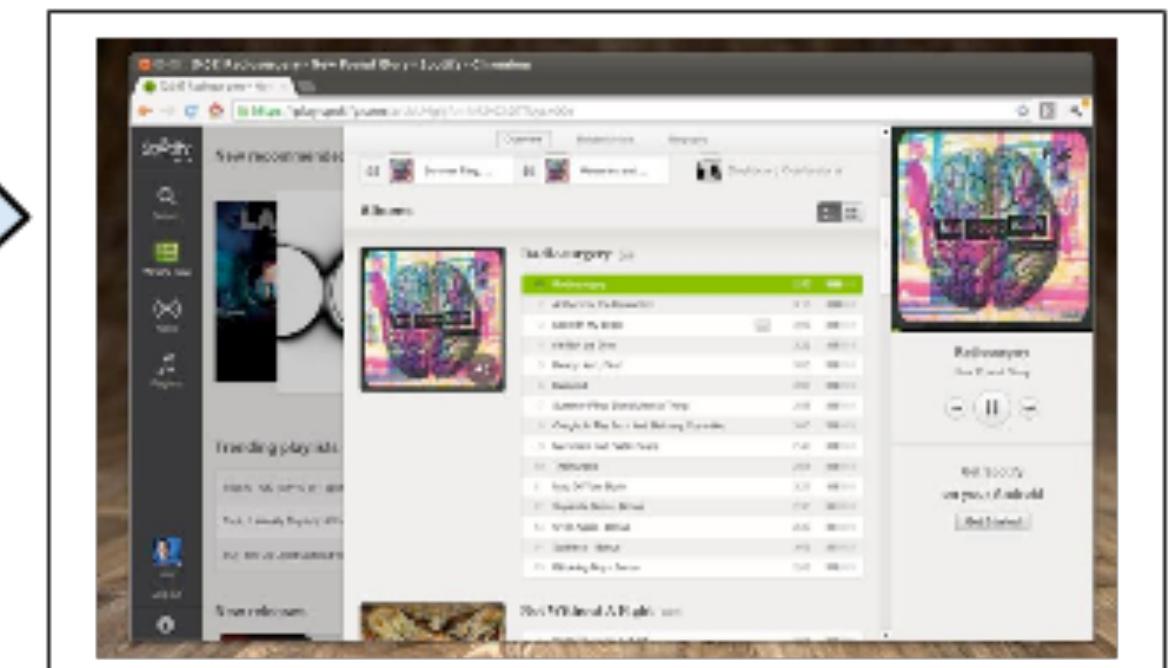
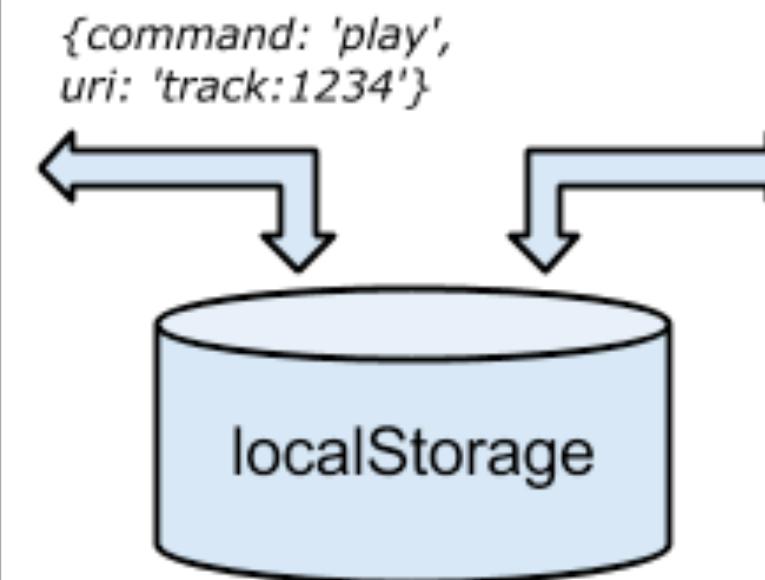
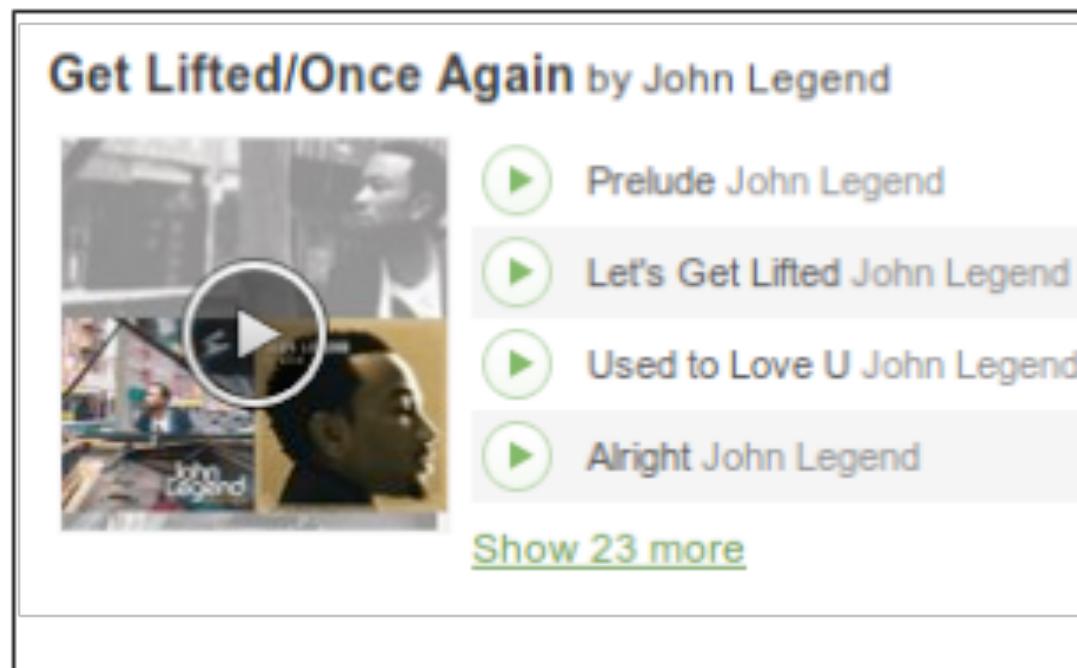
Is localStorage always available?

```
window.localStorage  
Storageconstructor:  
StorageConstructorlength: 0
```

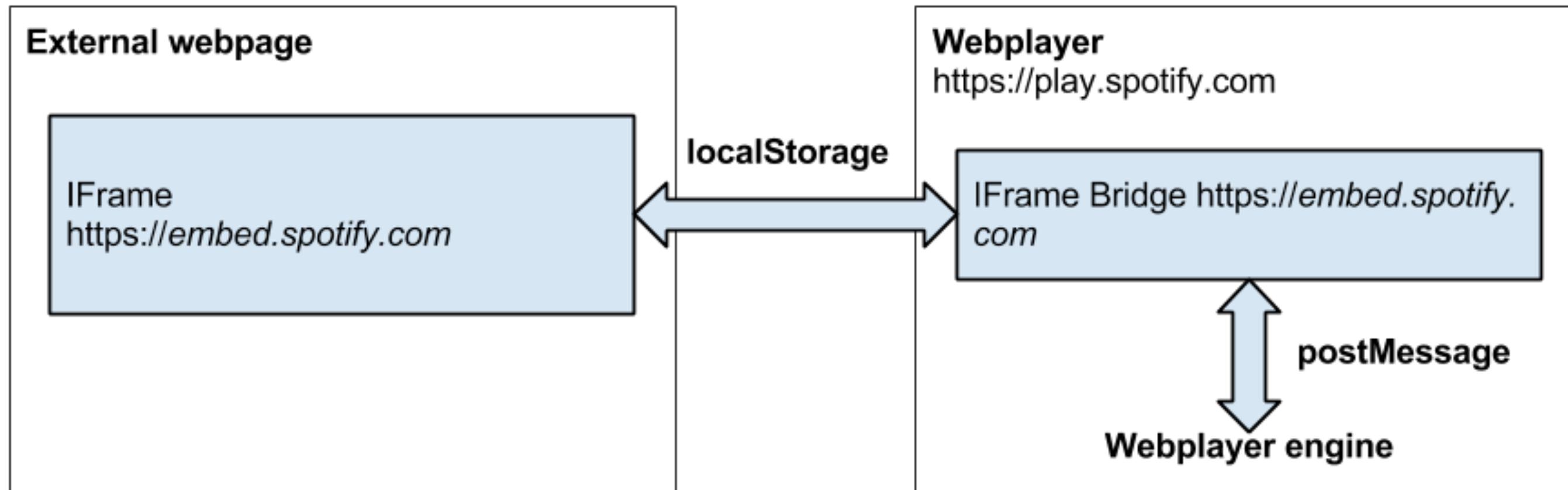
Is localStorage always available?

```
localStorage.setItem('key', 'value')
DOMExceptioncode: 22constructor:
DOMExceptionConstructormessage:
"QUOTA_EXCEEDED_ERR: DOM Exception 22"
```

Communication between different tabs



Communication between different tabs



localStorage locks

localStorage locks

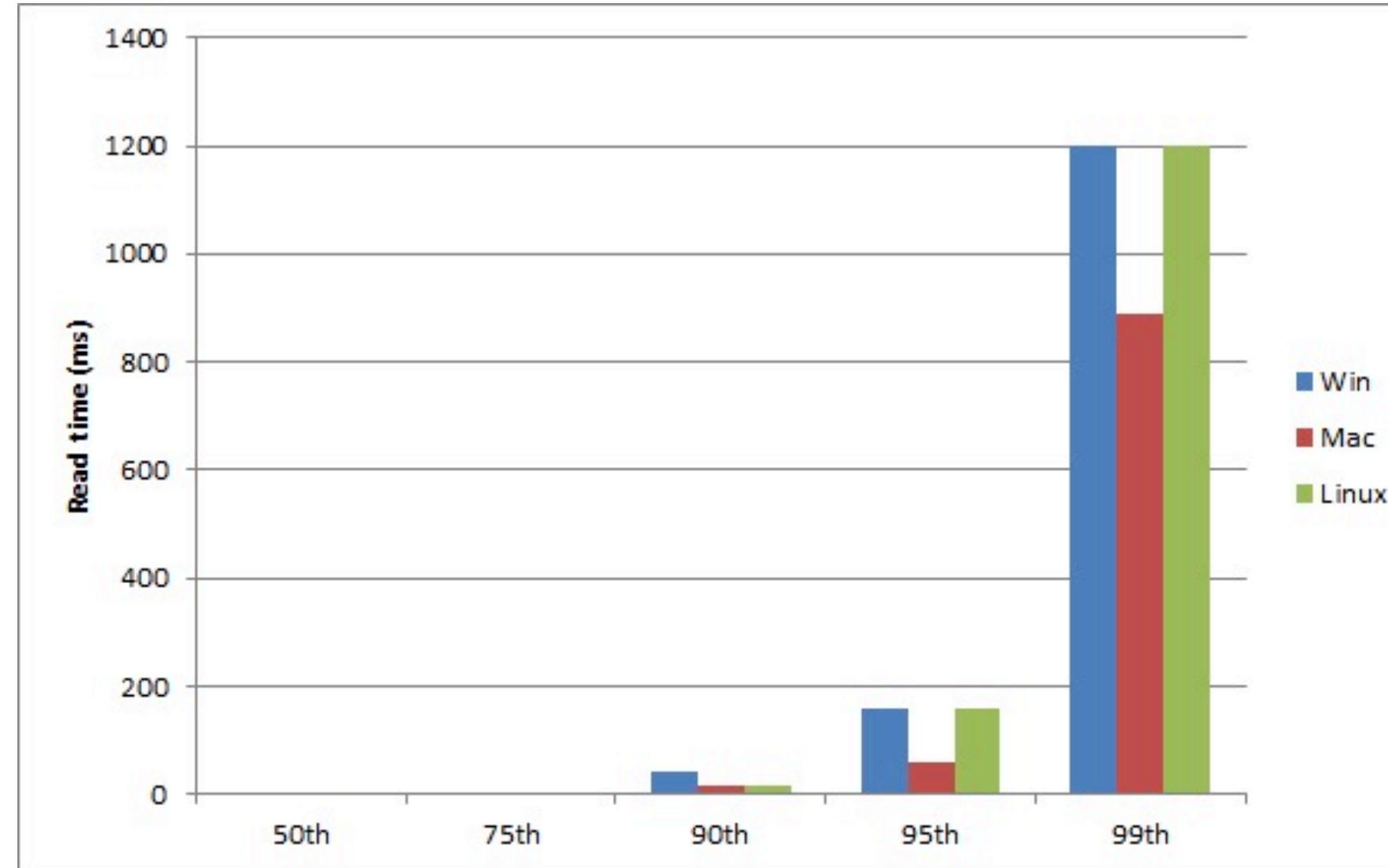
The UI Thread is stopped for some operations

localStorage locks

The UI Thread is stopped for some operations

Firefox and Chrome wait until the first read is requested, then load all the items

localStorage locks



<http://calendar.perfplanet.com/2012/is-localstorage-performance-a-problem/>

localStorage notifications via the storage event

```
window.addEventListener('storage', function() {}, false);
```

localStorage notifications via the storage event

```
window.addEventListener('storage', function() {}, false);
```

The event change is only emitted when the values actually change:
use a timestamp if needed

```
{key: 'command', value: 'play', timestamp: 1373021690574}
```

Can we trigger the event manually?

```
var evt = document.createEvent('StorageEvent');  
evt.initStorageEvent('storage', ...);  
window.dispatchEvent(evt);
```

Can we trigger the event manually?

```
var evt = document.createEvent('StorageEvent');  
evt.initStorageEvent('storage', ...);  
window.dispatchEvent(evt);
```

Yes...although not to communicate between different windows.

localStorage communication strategies

Options:

- A) Create a new item every time, but remove it once it's received.
- B) New item, don't remove once it's received, instead do it regularly.
- C) Reuse always the same item (use timestamp)

Notification delay

Chrome ~2 milliseconds

Notification delay

IE10 ~34 milliseconds
Chrome ~2 milliseconds

Performance improvements

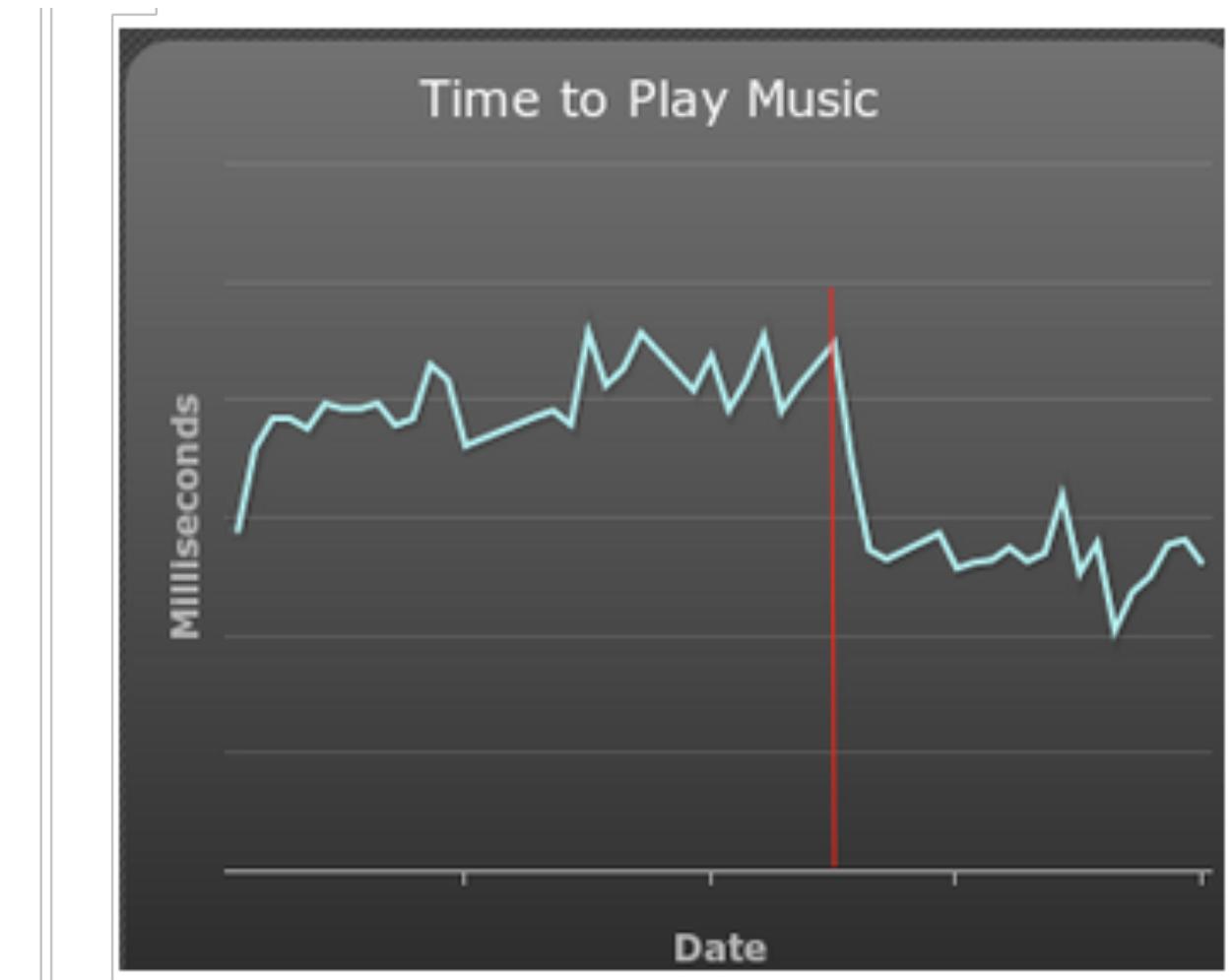
Prefetch / Prerender

- Rendering pages
`<link rel="prerender" href="...">`
- Fetching static resources
`<link rel="prefetch" href="...">`

Performance improvements

Prefetch / Prerender

- Rendering pages
`<link rel="prerender" href="...">`
- Fetching static resources
`<link rel="prefetch" href="...">`



Potential improvements

Batching postMessages

Webworkers

Conclusions

- Web apps have been a good choice to extend our functionality in multiple platforms.
- Web technologies make relatively easy to create complex integrations to improve the user experience.
- It is very important to monitor and analyze carefully the performance.



Questions?

Tomás Pérez

[@tomasperezv](https://twitter.com/tomasperezv)

José M. Pérez

[@jmperezperez](https://twitter.com/jmperezperez)

Slides, code examples and reference

github.com/tomasperezv/spainjs-2013

Thanks!

