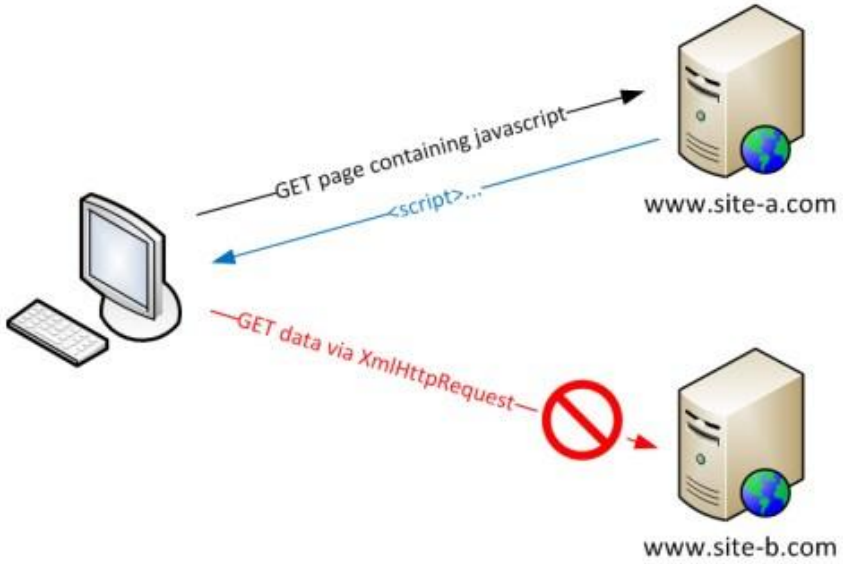
JSONP & CORS

Ferran Agulló López
Antoni Casas i Muñoz
Rubén González López
Joan Marc Pastor Mackenzie

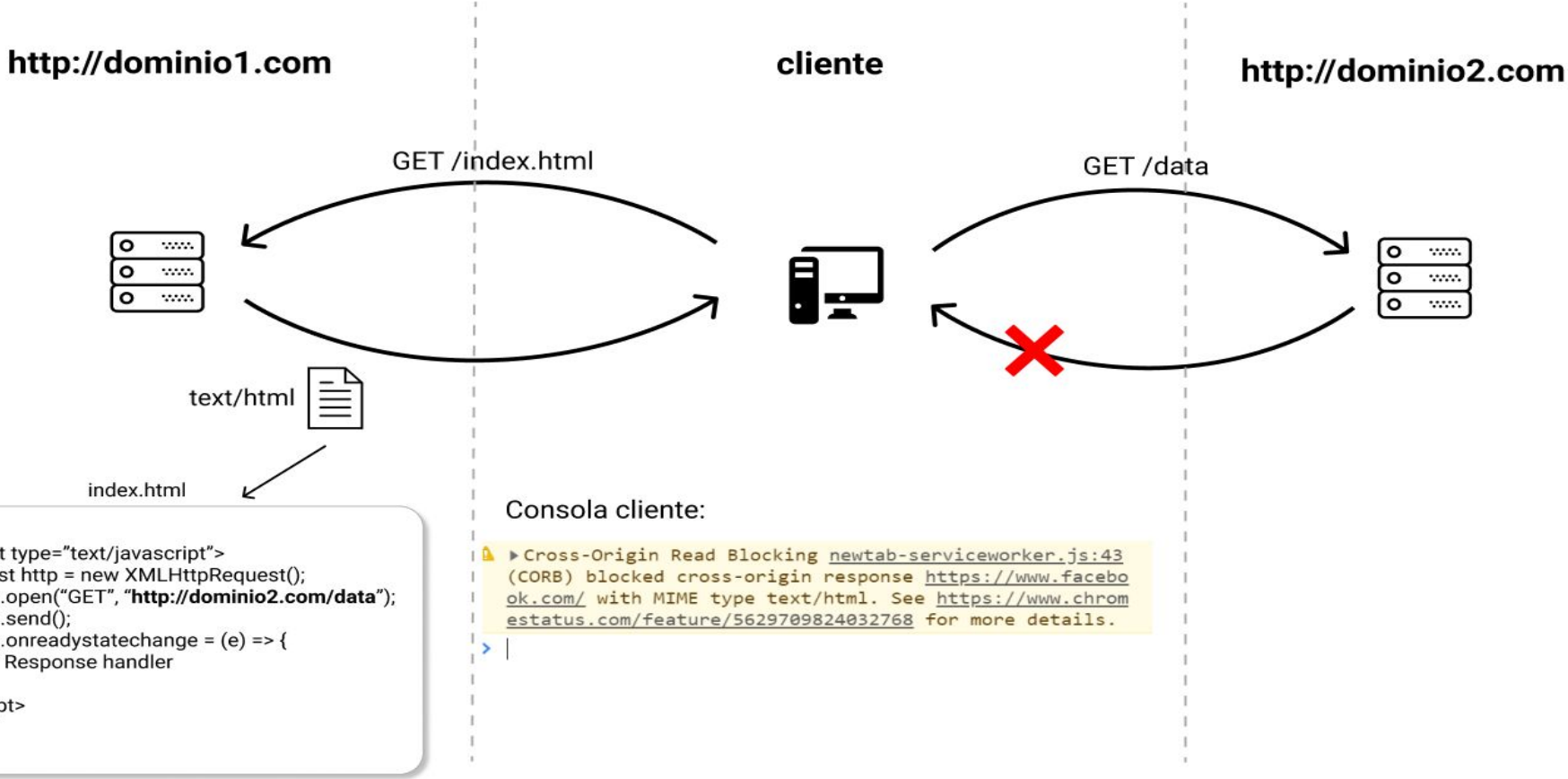
Introducción a CORS y JSONP

- Solución a la comunicación cross-domain
- Diferentes soluciones a un mismo problema

http://www.site-a.com/index.html	
https://www.site-a.com/index.html	Protocolo diferente
http://www.site-a.com**:1453**/index.html	Puerto diferente
http://www.**subdomain**.site-a.com/index.html	Subdominio diferente
http://www.**site-b**.com/index.html	Hostname diferente



Ejemplo: Same-Origin Policy



Evolución histórica

JSONP

Nace el 5 de Diciembre del 2005 por especificación de Bob Ippolito

Se mantiene idéntico en especificación y diseño

CORS

Nace el 13 de Junio del 2005 como nota en W3C sobre cross-domain sharing

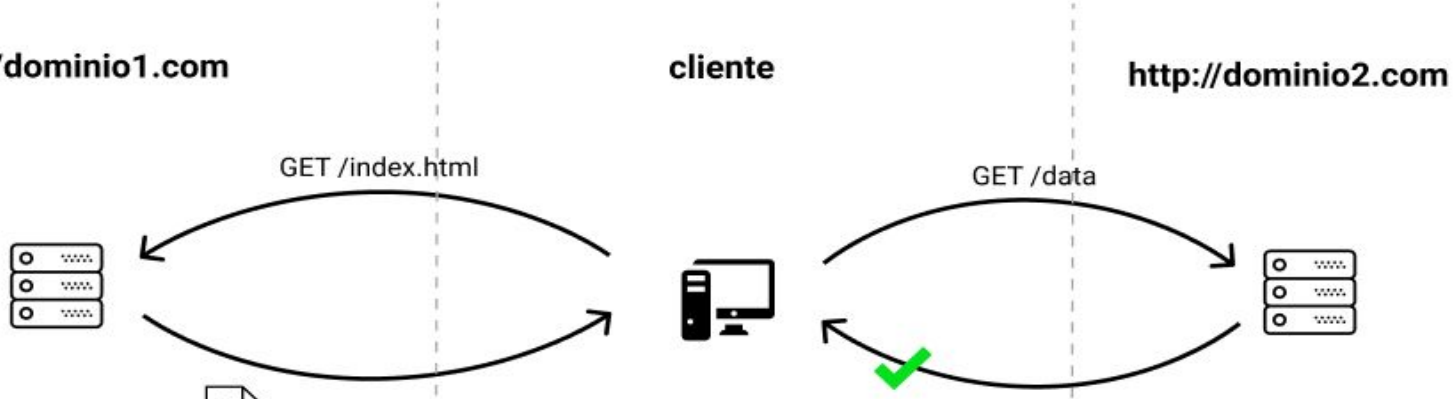
Formalización al nombre de CORS el 17 de Marzo de 2009 en un “Working Draft” de W3C

Recomendación de W3C el 16 de Enero del 2014

Es declarado obsoleto por W3C el 16 de Agosto del 2017 y pasa a ser mantenido por WHATWG (Web Hypertext Application Technology Working Group) en el Fetch Living Standard

Se mantiene idéntico en diseño

CORS




```
...
<script type="text/javascript">
   const http = new XMLHttpRequest();
   http.open("GET", "http://dominio2.com/data");
   http.send();
   http.onreadystatechange = (e) => {
      // Response handler
   }
</script>
...
```

Request Header needed:

- Origin: "dominio1.com" (puesto por el navegador)

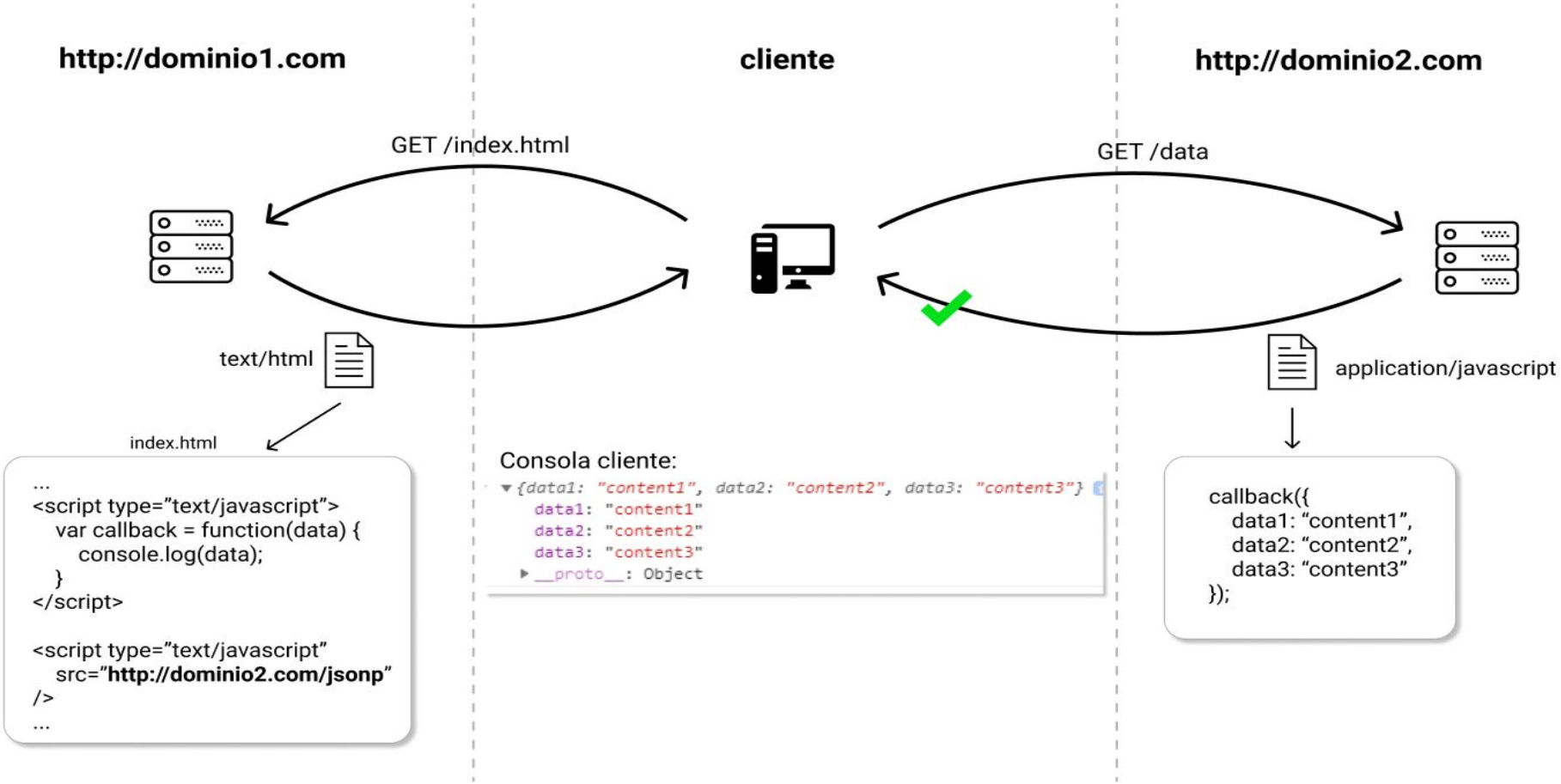
Response Header needed:

- Access-Control-Allow-Origin: "dominio1.com" or "*"

CORS

Simple requests	**Preflighted requests**
GET, HEAD, POST	PUT, DELETE, CONNECT, OPTIONS, TRACE, PATCH
Solo algunos headers	Todo tipo de headers
Solo tres tipos de content-type (text/plain,multipart/form-data,application/x-www-form-urlencoded)	Todos los tipos de content-type

JSONP


http://dominio1.com
cliente
http://dominio2.com
GET /index.html
GET /data
text/html
application/javascript
index.html
...
<script type="text/javascript">
var callback = function(data) {
console.log(data);
}
</script>
<script type="text/javascript"
src="http://dominio2.com/jsonp"
/>
...
Consola cliente:
{data1: "content1", data2: "content2", data3: "content3"}
data1: "content1"
data2: "content2"
data3: "content3"
__proto__: Object
callback({
data1: "content1",
data2: "content2",
data3: "content3"
});

Puntos fuertes

JSONP

No requiere estar implementado en el navegador

CORS

No está limitado a sólo el método GET

Permite al programador web usar la API XMLHttpRequest

Exploradores con soporte de CORS:

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Blackberry Browser	Opera Mobile *	Chrome for Android	Firefox for Android
6-7		2-3		3.1-3.2								
8-9		3.5-60	4-12	4-5.1	10-11.5	3.2-5.1		2.1-4.3				
10	12-17	61-64	13-71	6-11.1	12.1-57	6-11.4		4.4-4.4.4	7	12-12.1		
11	18	65	72	12	58	12.1	all	67	10	46	71	64
		66-67	73-75	12.1-TP		12.2						

Puntos débiles

JSONP

Supone una brecha de seguridad

Es solo read-only (GET)

Ejecuta un script arbitrario

CORS

Algunos navegadores no implementan esta tecnología

El servidor debe aceptar la petición al otro dominio

En determinados escenarios requiere de preflight (no es GET, HEAD o POST)

Se pueden falsear headers

Conclusiones y valoraciones personales

- JSONP está en desuso y todo lo que puede hacer se puede hacer con CORS de forma más segura y fácil.
- JSONP no es nada seguro y tiene varios exploits conocidos.
- Same-origin policy es una buena política para evitar brechas de seguridad entre diferentes dominios.
- Aunque CORS es significativamente más seguro que JSONP, no es completamente seguro

Referencias

https://www.w3.org/TR/2005/NOTE-access-control-20050613/

https://www.w3.org/TR/2009/WD-cors-20090317/

https://www.w3.org/TR/cors/

https://www.w3.org/2017/08/16-webappsec-minutes.html#item03

https://fetch.spec.whatwg.org/

https://bob.ippoli.to/archives/2005/12/05/remote-json-jsonp/

https://web.archive.org/web/20160304044218/http://www.json-p.org/ (Página archivada ya que la original ya no existe)

https://en.wikipedia.org/wiki/JSONP#Security_concerns

https://caniuse.com/#search=cors

Repartición de tareas

Presentación: Rubén González López y Ferran Agulló López

Investigación: Todos

Transparencias:

- Introducción: Antoni Casas i Muñoz
- Evolución histórica: Antoni Casas i Muñoz
- Same-Origin Policy: Joan Marc Pastor Mackenzie
- JSONP y CORS: Joan Marc Pastor Mackenzie
- Puntos fuertes: Rubén González López
- Puntos débiles: Ferran Agulló López
- Conclusiones: Todos