Experiencias y desafíos de implementar clientes de OGC API en Javascript

Iván Sánchez Ortega


IDERA
Infraestructura de Datos Espaciales de la República Argentina
¡Gracias Male!
1983 - Malena Libman - 2021

Experiencias y desafíos de implementar clientes de OGC API en Javascript

Iván Sánchez Ortega

Soy miembro de número de OSGeo

pero

no represento a OSGeo

y

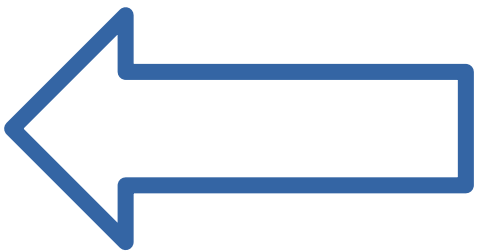
mis opiniones **no** son necesariamente las opiniones oficiales de OSGeo

Iván, 2019:

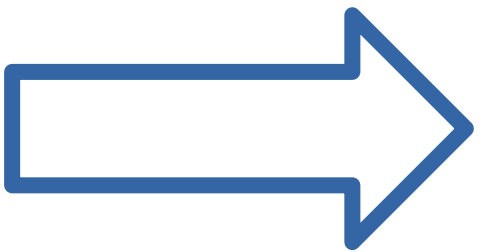
«Lo abierto y lo cerrado que uno es capaz de concebir depende de lo abierto y lo cerrado que uno haya visto y experimentado en su vida»

ISO no concibe
no cobrar por descarga

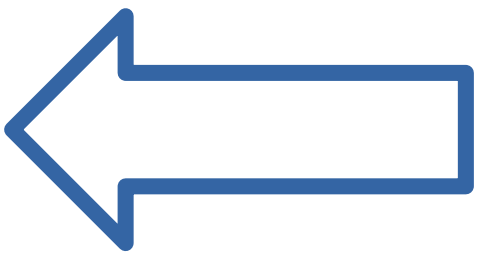
IETF no concibe
membresía cerrada



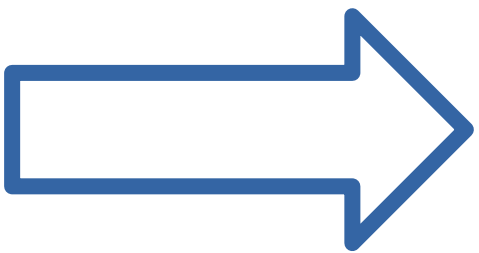
Más cerrado



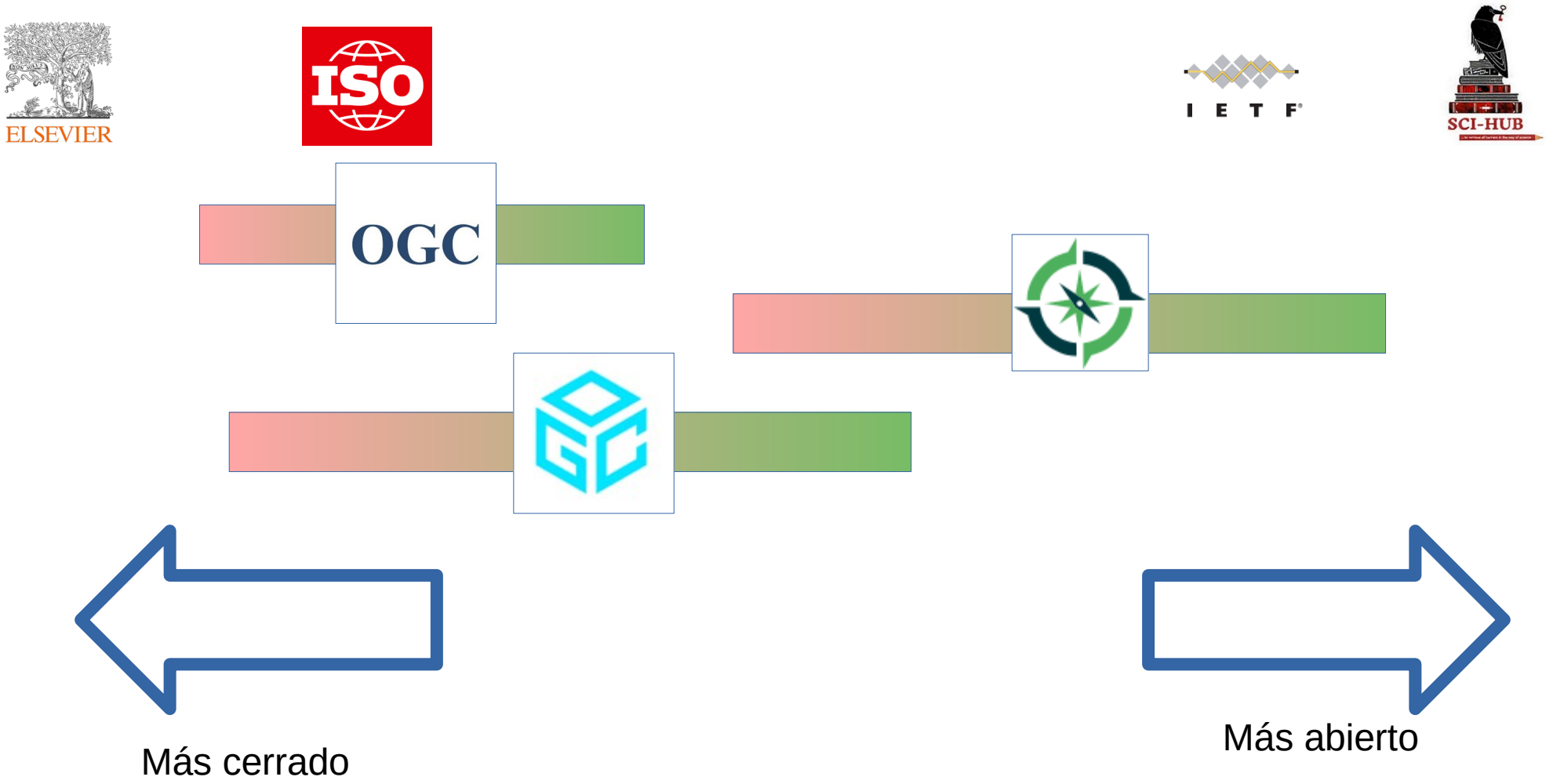
Más abierto

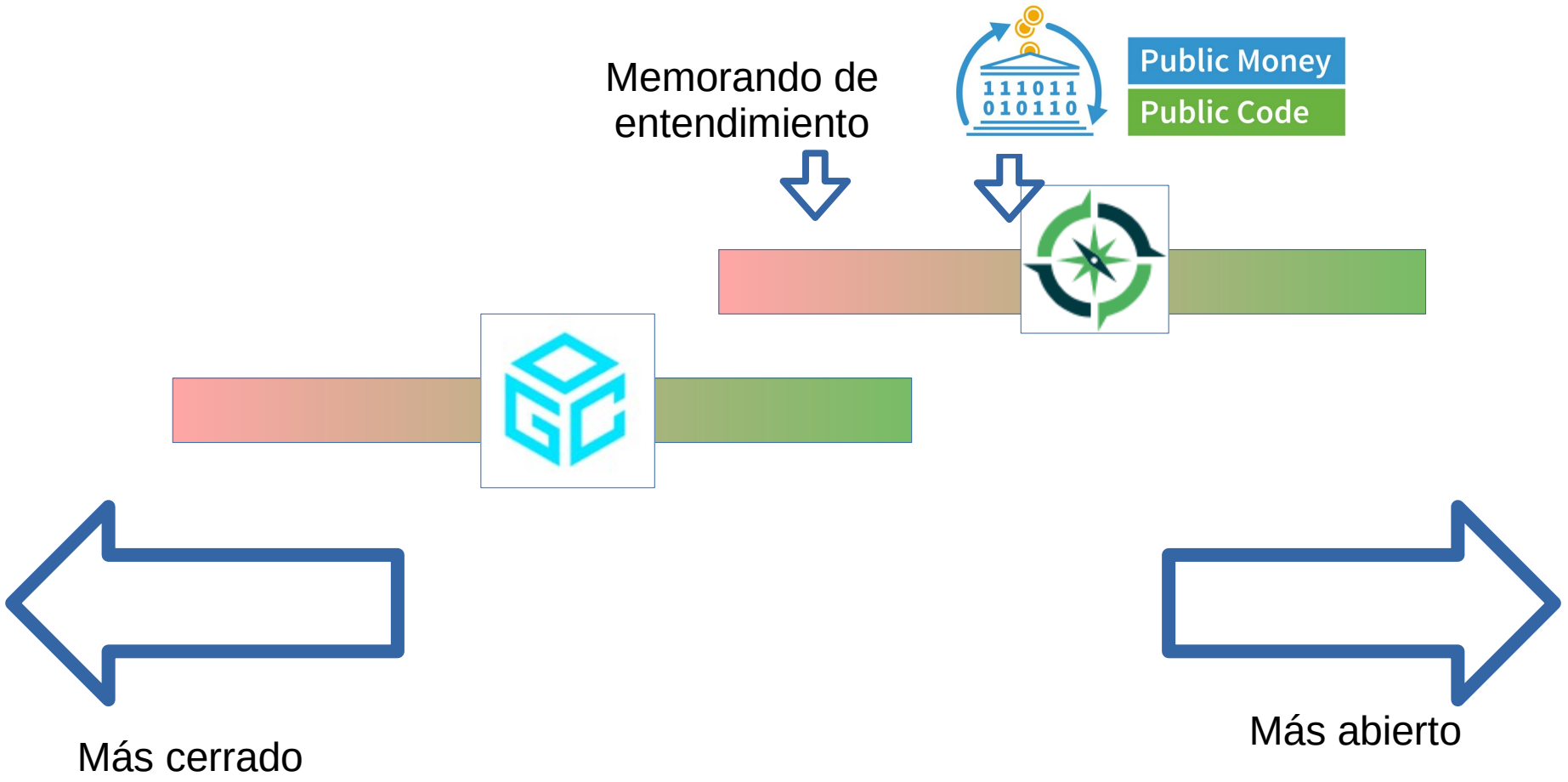


Más cerrado



Más abierto


ELSEVIER
ISO
IETF
SCI-HUB
OGC
Más cerrado
Más abierto


Memorando de entendimiento
111011
010110
Public Money
Public Code
Más cerrado
Más abierto

Public Money
Public Code

«Queremos una legislación que permita que el software desarrollado para el sector público y financiado con recursos públicos esté disponible públicamente bajo una licencia de Software Libre y Código Abierto. Si es dinero público debería ser también código público.»

https://publiccode.eu/es

</politica>

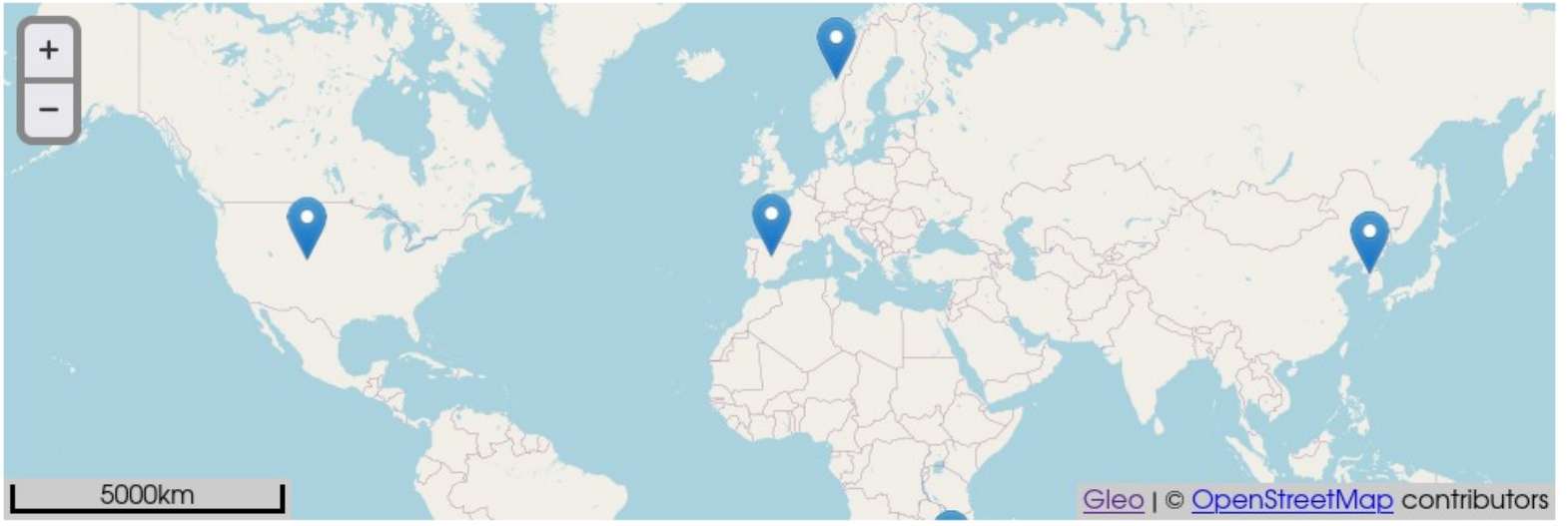
<script type=’text/javascript’>

Iván, 2019-2020:

«No entiendo la arquitectura WebGL de OpenLayers ni de MapboxGL/MapLibreGL, así que voy a reinventar la rueda»

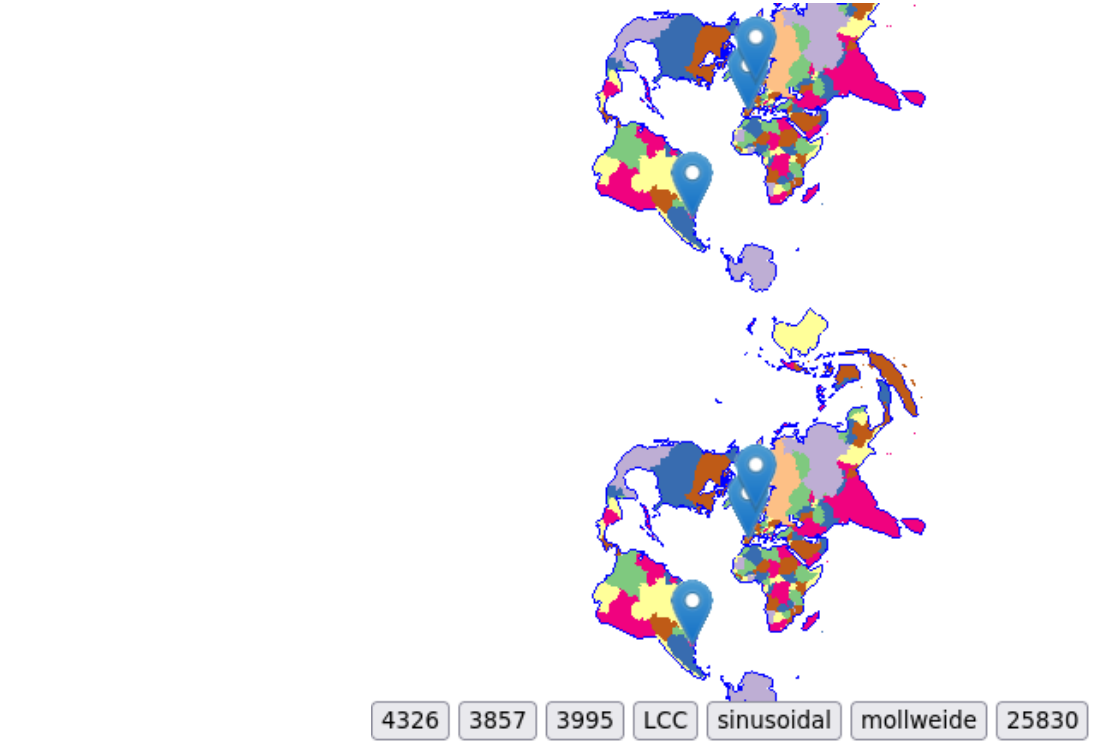
gleo

Reinventing WebGL maps



https://gitlab.com/IvanSanchez/gleo

¡Hora de demos en vivo!



https://gitlab.com/IvanSanchez/gleo/-/blob/master/browser-demos/17-proj.html

Iván, Enero 2022:

«Voy a implementar WMS en Gleo»

¡Hora de demos en vivo!

https://gitlab.com/IvanSanchez/gleo/-/blob/master/browser-demos/25-wms-ign.html

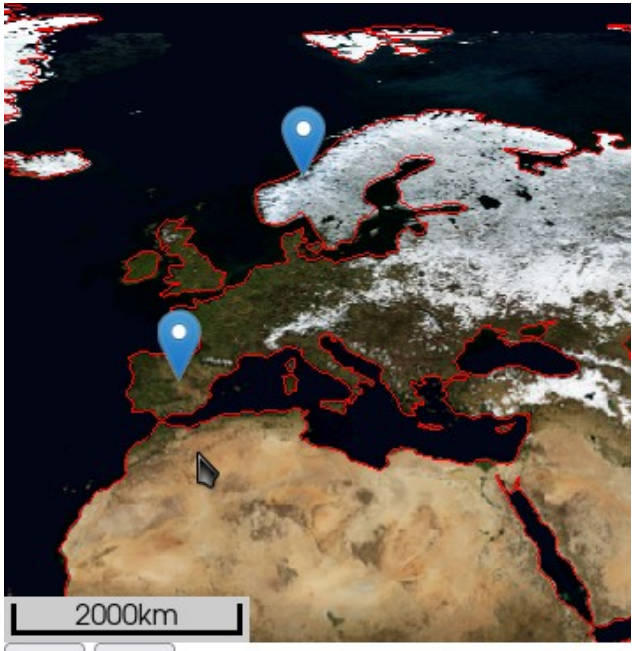
Iván, Enero 2022:

«Voy a implementar WMS en Gleo»

Iván, Febrero 2022:

«Anda, un codesprint de OGC+OSGeo el mes que viene. Voy a probar a implementar OGC API en Gleo»

¡Hora de demos en vivo!



https://gitlab.com/IvanSanchez/gleo/-/blob/master/browser-demos/35-ogcapi-maps-ecere.html

Comparación cliente WMS vs cliente OGCAPI

WMS	OGC API Maps
• 265 líneas de código • Parsear XML (+20 líneas) • Necesario gestionar estilos (+20 líneas) • Atribución con texto plano y URL	• 225 líneas de código • Parsear JSON y seguir documentos • No necesario gestionar estilos • Atribución sin formato (¿plano? ¿HTML? ¿Markdown?)
• 80-100 líneas compartidas • Mismo problema de inversión de ejes (“WMS 1.3.0”, XY vs YX)	

Conversación Iván-Gobe, marzo 2022:

Iván: ¿Por qué los CRS se identifican con una URL en vez de con una cadena corta como “EPSG:4326”?

Gobe: ¿Has trabajado con WMS 1.3?

Iván: Vale.

Idealmente:

Un CRS se identifica por una cadena de PROJ.

Triste realidad:

- Cadena de PROJ
- **Cadena de OGC API (URL)**
- **Orden de ejes (para servicios OGC)**
- Función distancia
- Modo de teselado (X/Y/XY) y período
- Límite (bbox) y escala útiles
- (¿Quizás?) una geometría de antimeridiano

Iván, Abril 2022:

«Anda, un codesprint de OGC+OSGeo
el mes que viene. Voy a probar a
implementar OGC API de Teselas en Gleo»

Expectativa	OGC API Teselas
• Mismos metadatos que OGC API Mapas • Varios CRS por colección, una pirámide por CRS • Teselas de mismo tamaño en toda la pirámide (¡como Leaflet!)	

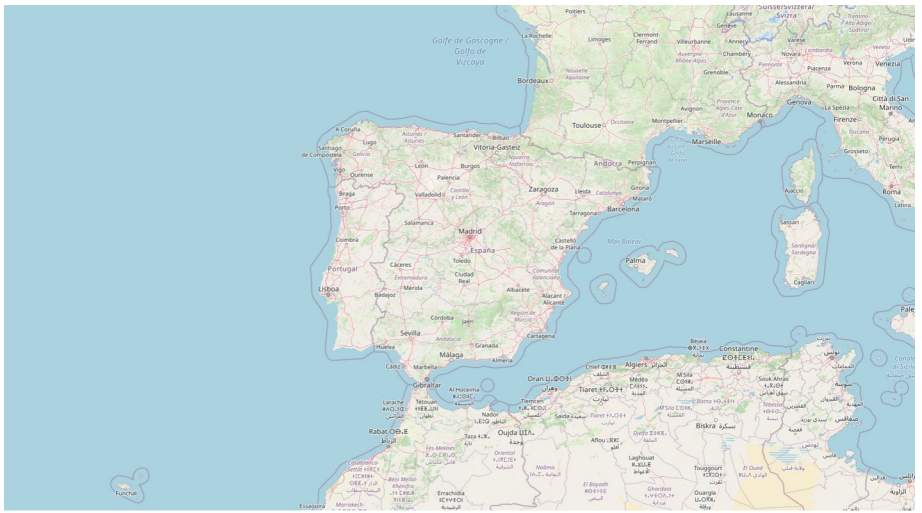
Expectativa	OGC API Teselas
• Mismos metadatos que OGC API Mapas • Varios CRS por colección, una pirámide por CRS • Teselas de mismo tamaño en toda la pirámide (¡como Leaflet!)	• Mismos metadatos que OGC API Mapas • Varios CRS por colección • Varias pirámides por CRS • En una pirámide, tamaño de teselas depende del nivel de la pirámide • En un nivel de pirámide, tamaño de teselas depende de la fila

¡Problemas insalvables!

- ¿Cómo decidir la pirámide correcta para un CRS?
  - ¿Cuándo se da esto? ¿Se usa para estilos? ¿Resoluciones de símbolos? ¿Formatos de imagen? Estilos y formatos van por otros metadatos
- Usar teselas de tamaños distintos dependiendo de la fila sencillamente rompe el almacenaje en memoria de esas teselas
  - O genera artefactos inevitables

¡Hora de demos en vivo!

En pantalla



Texturas internas

https://gitlab.com/IvanSanchez/gleo/-/blob/master/browser-demos/18-tile-textures.html

Yo entiendo por qué se permite esto, pero:

- Mi código debe poder elegir (él sólo) la pirámide para un CRS dado.
- Mi código debe reutilizar texturas en hardware antiguo (o poco potente); tamaños distintos en niveles distintos rompen esto.
- Es sencillamente **imposible** almacenar teselas de tamaño variable según fila sin provocar artefactos

Conclusiones de Iván: 1

El poder participar en eventos de OGC
sin burocracia de por medio es **fantástico,**
y muestra voluntad de OGC de hacer
sus procesos más participativo

Conclusiones de Iván: 2

Comparado con WMS/WFS/etc, las APIs OGC son más digeribles para el desarrollador

SOAP → REST
PDF → swagger
XML → JSON
etc

Conclusiones de Iván: 3

Las APIs quieren cubrir casos de uso complejos y/o específicos.

Por desgracia, esto
complica la implementación básica.

Conclusiones de Iván

- Más fácil participar: *¡bien!*
- Tecnologías no obsoletas: *¡bien!*
- Casos de uso complejos: *¡mal!*

Fin.

Iván Sánchez Ortega
https://ivan.sanchezortega.es