



XI Jornadas Ibéricas de
Infraestructuras de Datos Espaciais
26 | 30 outubro 2020

El giro hacia la Web de los estándares de servicios OGC

F. Javier ZARAZAGA-SORIA, Paloma ABAD, Laura ALEMANY,
Francisco J. LÓPEZ-PELLICER

Y en el inicio fue la Web

1989

Suppose all the information stored on computers everywhere were linked, I thought. Suppose I could program my computer to create a space in which anything could be linked to anything. All the bits of information in every computer at CERN, and on the planet, would be available to me and to anyone else. There would be a single, global information space





Consorcio internacional

1994

443
miembros

Recomendaciones y estándares
que aseguran el crecimiento de
la World Wide Web a largo
plazo

Consorcio internacional

1994

527
miembros

Creación de estándares abiertos
e interoperables para la
comunidad geoespacial global
en el marco de los SIG y la WWW

¿Por qué no migró OGC a W3C hace muchos años?



- ❑ Orientación a la publicación de datos
 - ❑ “Tenemos que publicar, que es lo que nos exigen”
 - ❑ “Además tenemos el mejor dato”
- ❑ Y esto fue suficiente hasta 2005

¿2005? ¿Por qué?

- ❑ En 2003, Lars y Jens Rasmussen, Noel Gordon y Stephen Ma, cofundaron Where 2 Technologies
- ❑ En octubre de 2004 Google compra Where 2 Technologies
- ❑ En febrero de 2005 nace Google Maps



¿Secreto del éxito de Google Maps?

- ❑ Es de Google
 - ❑ Pero también lo eran Google Answers, Lively, Glass, Google Buzz, Google Wave, Google Health, Google+, ...
- ❑ Orientado a dar una rápida respuesta al consumidor de información
 - ❑ La calidad del dato ya la mejoraremos....

¿Secreto del éxito de Google Maps?



¿Secreto del éxito de Google Maps?

- Es de
- Pe
- W
- Ori
- La



¡¡Queremos compartir el éxito de la WWW!!

- ❑ ¿Qué tenemos que hacer?

- ❑ Ponérselo fácil a los que crean soluciones en la WWW

- ❑ ¡Genial! ¡Hagámoslo!

- ❑ Pero, ¿cómo?

- ❑ Permitamos que el acceso a nuestros servicios se haga de modo análogo a como se hace en los servicios Web más populares

- ❑ Hagámoslo RESTful

Transferencia de Estado Representacional

Representational State Transfer (REST)

- ❑ REST es un estilo de arquitectura software
- ❑ Los servicios Web que siguen este estilo se denominan RESTful web services
- ❑ Principios fundamentales
 - ❑ Cliente-servidor
 - ❑ Sin estado
 - ❑ Cacheable
 - ❑ Interfaz uniforme
 - ❑ Posibilidad de transferencia de código (JavaScript por ejemplo)

Estado actual

- ❑ OGC API in progress (<https://ogcapi.ogc.org/apiroadmap.html>)
 - ❑ Basada en Open API (Swagger Specification) de la Linux Foundation
 - ❑ ¿Y esto es bueno?
 - ❑ Sí, porque no se reinventa todo. Existen herramientas para generar automáticamente documentación de métodos, parámetros y modelos, entre otros

Principios de OGC API

- ❑ No reinventar la rueda
- ❑ Mantenerlo simple e intuitivo
- ❑ Usar formatos conocidos
- ❑ Construir URIs consistentes, con una estructura predecible
- ❑ Usar HTTP bien, es decir sin forzar la semántica de HTTP
- ❑ Seguir las prácticas actuales de seguridad en la Web
- ❑ Usar OpenAPI 3 para describir las API para humanos
- ❑ Soportar W3C CORS
- ❑ Las buenas API deben poderse probar desde el principio

<https://github.com/opengeospatial/OGC-Web-API-Guidelines>

Estado actual

- ❑ OGC API in progress

 - ❑ Basada en Open API (Swagger Specification) de la Linux Foundation

 - ❑ ¿Y esto es bueno?

 - ❑ Sí, porque no se reinventa todo. Existen herramientas para generar automáticamente documentación de métodos, parámetros y modelos, entre otros

- ❑ OGC API Features como primer paso

 - ❑ From GET+POST (KVP) to API REST

- ❑ Nueva acción de trabajo de MIG-T para desarrollar buenas prácticas de INSPIRE basada en el uso de OGC API Features como un servicio de descarga de INSPIRE

Y, ¿ya está?

☐ Pues no

- ☐ WFS u OGC API Features, al fin y al cabo devuelven datos
- ☐ Garbage in – Garbage out
 - ☐ Aplicable a los datos y a su estructura

☐ Luego....

- ☐ Seguimos necesitando tener datos de calidad en contenido y en estructura
- ☐ Necesitamos trabajar en los modelos y en la creación de contenidos de acuerdo a esos modelos
- ☐ ¿Y los metadatos? ¿y sus relaciones con los recursos? ¿y entre recursos?

☐ ¿Y si no lo hacemos?

- ☐ Podremos poner fácilmente “basura” en aplicaciones Web
- ☐ Y eso no será bueno para nosotros a medio/largo plazo



@jazarso
javy@unizar.es