

PROBLEMAS Y SOLUCIONES EN EL DESARROLLO DE LA IDE EXTREMADURA

Alberto Aparicio Ríos

Gobierno de Extremadura

alberto.aparicio@juntaextremadura.net

Resumen

Se abordarán los problemas encontrados durante el desarrollo del visor de la IDE Extremadura así como de los servicios WMS contenidos en el mismo, aportando las soluciones encontradas para su solventación en los casos pertinentes.

Palabras clave: Visor, WMS, soluciones

1 Introducción

A lo largo de esta ponencia se van a comentar aquellos problemas que se han encontrado a lo largo del desarrollo del visor de la IDE Extremadura así como las soluciones aportadas por los técnicos que la han desarrollada. Estos mismos problemas seguramente hayan afectado a otras organizaciones y esperamos que sean de utilidad las soluciones propuestas

2 Descripción

Uno de los puntos más complicados a la hora de desarrollar un proyecto como la IDE Extremadura ^[1] ha sido plasmar de forma gráfica todo aquello que la teoría de servicios WMS, INSPIRE y demás normativas nos exigen a las organizaciones productoras de datos.

Tras varios años intentando cumplir con todas ellas quizá sería bueno poner en común todos aquellos problemas que nos hemos ido encontrando

así como las soluciones propuestas para que sirvan de ayuda a la gente que se inicia en este mundo.

Desde nuestro punto de vista, quizá lo más importante de una IDE, además de su interoperabilidad, la estandarización... sea la facilidad de uso para el usuario. Por muchos servicios que ofrezcamos, si el usuario final no utiliza la IDE, no hemos logrado nada. Es por ello que, tras encuestar a personal del sector y usuarios finales de varios tipos, lo que más preocupa a la gente es la velocidad de uso. A veces no es operativo para quien necesita información el hecho de tener que esperar varios minutos a que se cargue el mapa, o a que se refresque la página, o no poder utilizar un servicio WMS porque está caído. Para evitar estos problemas lo primero que abordamos fue la estructura de los servidores del Gobierno de Extremadura. Una vez hecho esto intentamos acelerar al máximo los servicios y el visor, para aprovechar el rendimiento que estas máquinas nos ofrecían. Por último se están añadiendo utilidades que hacen el trabajo del usuario externo mucho más sencillas.

2.1 Optimización de los servidores

En primer lugar instalamos dos máquinas con Linux, trabajando directamente con la consola, en lugar de utilizar el entorno gráfico, para poder aprovechar al máximo sus capacidades. Estudiando los rendimientos de la anterior versión en Windows nos dimos cuenta de que, tanto en velocidad de respuesta como en disponibilidad, no tenían ni punto de comparación.

El software utilizado para servicios de mapas es principalmente Mapserver^[2]. Además de ser gratuito ofrece un gran rendimiento y bastantes formas de personalización que posteriormente se explicarán.

Posteriormente se creó un balanceador por software que actúa de frontal ante estos dos equipos. Con ello conseguimos que las peticiones de mapas se repartan entre dos máquinas acelerando al doble las respuestas de cara al usuario. También esto nos sirve para reducir los tiempos de inactividad del servicio. Es cierto que a veces puede caerse una máquina, pero entraría en funcionamiento la otra. Además podemos llevar a cabo el mantenimiento de un servicio mientras el balanceador dirige todas las peticiones al otro, y a la inversa posteriormente, haciendo que la disponibilidad sea cercana al 100%.

2.2 Optimización de servicios WMS

Como ya se ha comentado, el software de publicación de servicios WMS elegido ha sido Mapserver 6.0, el cual nos ofrece servicios WMS acordes al estándar OGC WMS 1.3.0 ^[3].

Este software nos permite trabajar muy rápidamente con una gran variedad de fuentes de datos, como pueden ser shapes, tiff o postgis. Para un óptimo rendimiento partimos del código fuente y lo recompilamos para la máquina que estamos utilizando. Incluso si lo instalamos como fast cgi podremos mantener abiertas las conexiones con los archivos y con las bases de datos, consiguiendo un rendimiento incluso mayor.

Una vez hecho esto, los servicios ofrecidos pueden optimizarse siguiendo varias pautas:

- Proyecciones permitidas: Para realizar las reproyecciones se utiliza la librería Proj.4 ^[4]. Esta nos permite hacer un montón de conversiones entre miles de sistemas de referencia, de los cuales a nosotros solamente nos interesarán unos pocos. Eliminando el resto de referencias a los SRS no utilizados o bien situando las más utilizadas al inicio, conseguimos optimizar los tiempos de carga.
- Creación de índices espaciales en servicios que utilizan imágenes raster: Se trabaja con un shape de fondo que nos indica en cada caso cuál es el raster que se aplica en cada zona del mapa. Esto es útil, por ejemplo, trabajando con ortofotos. La directiva utilizada es TILEINDEX.
- Hay que evitar mostrar capas cuando no sean útiles. Para ello podemos limitar las escalas de visualización utilizando las directivas MINSCALEDENOM y MAXSCALEDENOM, consiguiendo así mostrar aquellas capas que realmente sean útiles a una cierta capa y evitando malgastar recursos cuando no sean necesarias.
- Lo mismo ocurre con la extensión geográfica. De nada nos sirve darle la extensión de España si solamente tenemos datos de la zona de Extremadura. Limitando la zona al máximo conseguiremos evitar peticiones a los servidores si estamos en zonas alejadas de la misma.

- A la hora de crear varias capas dentro de un fichero MAP, lo lógico sería colocar las más usadas al inicio. También dentro de una capa, si se utilizan varios CLASS para distinguir los datos en categorías, el hecho de colocar las más utilizadas al inicio hará que consigamos mejores tiempos de respuesta.
- Cuando utilizamos shapes como fuentes de datos es bueno crear índices espaciales, sobre todo si se trata de ficheros grandes. La utilidad "shptree" que provee Mapserver nos puede ser de gran utilidad.

2.3 Optimización del visor OpenLayers

La librería OpenLayers ^[5] nos permite optimizar la carga de los mapas de ciertas maneras:

- Cuando añadimos una capa necesitamos especificar al menos una url de origen de los datos. En algunos navegadores se limita el número máximo de peticiones simultáneas a un servidor, pudiendo ocurrir que se pida solamente la mitad del área visual de cada vez si tenemos tileadas las peticiones, teniendo que esperar a que lleguen unas para poder pedir las otras. Puede solucionarse añadiendo varias URL's a dicha capa, bien con varios dominios o bien utilizando servidores virtuales a tal efecto.
- También funcionó en nuestro caso la utilización de SINGLETILE a la hora de hacer las peticiones. Se consigue así hacer una sola petición por servidor, de tal forma que se minimiza el nº de peticiones.
- Lo óptimo sería tener un servicio cacheado (WMS-C) ó WMTS utilizando la misma rejilla que el visor está pidiendo, con los mismos niveles de zoom para evitar tener que hacer modificaciones a las imágenes cacheadas antes de devolverlas al cliente. Para ello el parámetro scales de openlayers nos puede ser bastante útil, no teniendo que limitarnos a la subdivisión en escalas que se hacía en versiones previas.

2.4 Servicios para el usuario

Por último, los últimos cambios que se están realizando son para facilitar el trabajo a los usuarios finales de la aplicación:

- Búsqueda por polígono/parcela: A la hora de trabajar, por ejemplo, con expedientes de calificación se hace imprescindible localizar los polígonos y parcelas de los distintos municipios. Es una de las herramientas más utilizadas.
- Consulta gráfica de información: Utilizando las plantillas de Mapserver es muy sencillo mostrar las peticiones GetFeatureInfo de forma visual para que le sea fácil al usuario interpretar el resultado de sus consultas. Permite incluso accesos a metadatos, ficheros asociados a la capa, etc.
- Servicios WMS de utilidad, como son expedientes de calificación, planeamiento urbanístico vigente, planos escaneados de ordenación municipal, atlas de Extremadura...
- Se están incluyendo posibilidades para que el usuario cargue sus propios shapes, kml, gml... para interactuar con la información disponible en la IDE.

2.5 Referencias bibliográficas

- [1] <http://www.ideextremadura.com/Geoportal>
- [2] <http://mapserver.org/>
- [3] <http://www.opengeospatial.org/standards/wms/>
- [4] <http://trac.osgeo.org/proj/>
- [5] <http://openlayers.org>