

## Integración de la recomendación WMS-C en los estándares preexistentes en la IDE.

*Juan Antonio López Alonso<sup>(1)</sup>, Mónica Citores Fernández<sup>(1)</sup>*

<sup>(1)</sup> Centro de Observación y Teledetección Espacial S.A. (COTESA). Área de Sistemas de Información, P. Tecnológico Boecillo, Edificio Solar, 47151 Boecillo Valladolid, España, [juanlopez@grupotec.es](mailto:juanlopez@grupotec.es)

### RESUMEN

*La experiencia en el uso de los servicios de mapas basados en la especificación Web Map Service (WMS) del Open Geospatial Consortium (OGC) ha demostrado que es necesario utilizar cachés de teselas para lograr un rendimiento aceptable en aplicaciones de difusión masiva, sin embargo no hay ningún mecanismo estándar para que los clientes de mapas aprovechen, a partir de la información proporcionada por el servidor de mapas, la disponibilidad de esta caché. A la espera de que la nueva recomendación WMTS se implante suficientemente, el mecanismo más extendido es la recomendación de perfil WMS-C de OsGeo.*

*Para conseguir que la definición de mapas que contienen servicios WMS-C sea lo más automática posible, se ha ampliado el servidor Geoserver para soportar un modelo de mapas de acuerdo con la recomendación WMC con algunas extensiones ad-hoc. La extensión desarrollada para Geoserver amplía su API REST para incluir soporte de WMC.*

*De esta forma, cuando se registra una nueva configuración de mapa, mediante un documento WMC, en el que ciertas capas están cacheadas se procede automáticamente a la activación del cacheado mediante la extensión GeoWebCache.*

*Para la utilización de las nuevas capacidades proporcionadas a Geoserver, se ha desarrollado un cliente de mapas que identifica la existencia de capas cacheadas y procede a utilizar, según convenga, los servicios cacheados y los servicios WMS tradicionales*

**Palabras clave:** WMS-C, REST, GeoServer, SIG.

## ABSTRACT

*Experience in the use of map services based on the Web Map Service specifications (WMS) of the Open Geospatial Consortium (OGC) has shown that it is necessary to use tessera caches to achieve acceptable levels of performance in mass-dissemination applications; however, there is no standard mechanism for clients to make use of the availability of this cache based on the information provided by the map server. Until the new WMTS recommendation is sufficiently implemented, the most common mechanism is the OsGeo WMS-C profile recommendation.*

*To make the map definition in the WMS-C services as automatic as possible, the Geoserver has been extended to support a map model that complies with the WMC recommendation with certain ad hoc extensions.*

*The extension developed for Geoserver increases its API REST to include WMC support.*

*Accordingly, when a new map configuration is registered using a WMC document on which certain layers are cached, the caching is automatically activated using the GeoWebCache extension.*

*For the use of the new capacities provided to Geoserver, a map client has been developed to identify the existence of cached layers and use the cached services and traditional WMS services as appropriate.*

**Key Works:** *WMS-C, REST, GeoServer, SIG.*

## INTRODUCCIÓN

Con la experiencia en el uso de los servidores de mapas se ha llegado a la conclusión de que existen dos necesidades para que estos sistemas sean completamente operativos y proporcionen un rendimiento aceptable. Dichas necesidades son:

- Integración del concepto de mapa en los servidores de capas.
- Uso del sistema de cachés de teselas [1-3].

La necesidad de implementar el concepto de mapa en un servidor WMS se debe a que no está contemplada en las interfaces de servicios OGC ninguna representación de agregación ordenada de información geográfica ni del estado de visualización y estilos que los clientes necesitan para mostrar la información al usuario [4]. La problemática general de definir una estructura de visualización está abordada en la recomendación WMC (Web Map Context) del [Open Geospatial Consortium](#) [5]. Se ha utilizado esta recomendación como base para incorporar el concepto de mapa a un servidor de capas como es Geoserver. La descripción del contexto de mapa se ha extendido para servir de elementos de configuración y de información integrada de mapas para las aplicaciones cliente.

Aunque existen técnicas y recomendaciones para acelerar el funcionamiento de los servidores de mapas, como las mostradas en [6], la experiencia continuada en el uso de los servicios de mapas basados en la especificación Web Map Service (WMS) del Open Geospatial Consortium (OGC) [7] ha demostrado que para que estos servicios

sean realmente utilizables bajo un elevado volumen de accesos es *necesario utilizar* cachés de teselas [8] y [11].

No obstante los clientes de mapas no disponen de mecanismos estándar para que aprovechen esta característica. A la espera de que la nueva recomendación WMTS se implante suficientemente, el mecanismo más extendido es la recomendación de perfil WMS-C de OsGeo [2]. Pero incluir este perfil de acceso implica implementar soluciones a medida y no permite que los clientes aprovechen estas fuentes de teselas de forma automática.

En este proyecto se propone una solución práctica para la integración de los servicios WMS tradicionales y los cachés de teselas ofrecidos mediante la recomendación WMS-C. Para ello se ha trabajado con el servidor de mapas Geoserver y la extensión Geowebcache. Se ha desarrollado una extensión de los servicios del *framework* Geoserver para unificar la gestión y la información de estos recursos. El documento WMC parece un buen soporte para esta funcionalidad ya que permite informar a los clientes de toda la información necesaria para visualizar los mapas respetando el aspecto deseado por el cartógrafo y aprovechando las posibilidades técnicas de los sistemas.

## **ADAPTACIÓN DE GEOSERVER COMO GESTOR DE CARTOGRAFÍA CON EL SOPORTE DE MAPAS**

Geoserver soporta un concepto de agrupación de capas denominado “*layergroup*” para crear un sinónimo para un grupo de capas pero este concepto no es utilizable desde clientes de mapas ya que no incluye información de estilo ni permite mantener la estructura original del grupo. Por esta razón se ha decidido ampliar el catálogo de objetos del diccionario de Geoserver para que utilice el concepto de mapa definido como una combinación ordenada de capas y estilos junto con información de estado y metadatos relevantes. Para esta implementación se ha decidido utilizar como base el WMC con una serie de extensiones del estándar.

### **Modificaciones en la estructura de Geoserver**

Geoserver expone su modelo interno de datos en forma de un diccionario jerárquico donde se pueden localizar las *feature-types*, los estilos, las fuentes de datos, etc. La organización original de Geoserver se puede resumir en la Figura 1 donde capa uno de los niveles contiene:

- Workspace o Namespace: espacio de trabajo/nombres que permite hacer agrupaciones lógicas de los elementos, aquí se almacenan:
  - DataStores : conexión a BBDD, directorio de shapes, etc.. dentro de esto estan:
    - FeatureTypes: son los fenómenos modelados en las distintas fuentes de datos y que se pueden utilizar como origen para definir capas en un mapa.
- Layers: capas que se sirven en el sistema
- Styles: estilos definidos para capa una de las capas
- Layergroups: grupos de capas

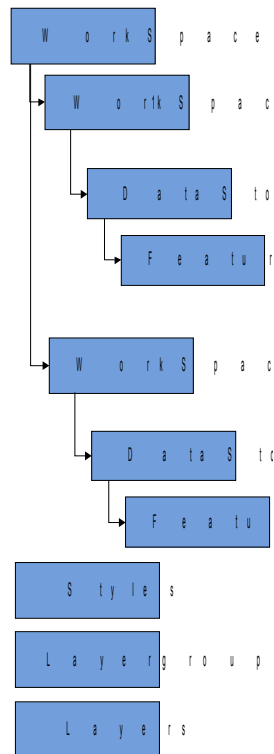


Figura 1: *Catálogo de entidades gestionadas por Geoserver.*

Las modificaciones introducidas en dicha estructura para poder gestionar el concepto de mapa consiste en añadir un nuevo nivel dentro del workspace que contendrá toda la información necesaria para los mapas, con lo que la estructura general quedará de la siguiente manera:

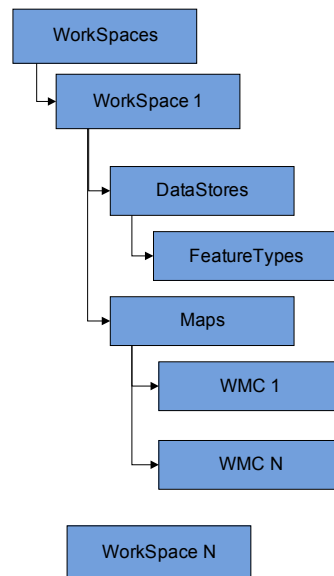


Figura 2: *Modelo resultante de este proyecto donde se incluye en concepto de mapa.*

Cada uno de los WMC contendrá la información necesaria para describir el correspondiente mapa.

## Extensiones del WMC

En el desarrollo se ha tenido muy en cuenta que el cliente de cartografía más utilizado para proyectos en la IDE, Openlayers, tiene una implementación para leer WMC así como algunos campos de la extensión, por lo que se ha partido de esta implementación como base para la integración.

Básicamente, un documento WMC se ajusta a un esquema XML que permite describir la configuración necesaria para que un cliente de visualización de mapas pueda recrear todos los aspectos relevantes de un mapa en una posición concreta.

Un documento WMC hipotético que define un mapa puede ser:

```
<ViewContext id="Callejero_42020" version="1.0.0" xmlns:ol="http://openlayers.org/context"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns="http://www.opengis.net/context"
xsi:schemaLocation="http://www.opengis.net/context
http://schemas.opengis.net/context/1.0.0/context.xsd">
  <General>
    <BoundingBox maxx="4580798.0" maxy="4603585.0" minx="529692.0" miny="544538.0"
SRS="EPSG:23030"/>
    <Abstract>WMC to map: Callejero_42020</Abstract>
    <DescriptionURL format="text/html">
      <OnlineResource xlink:type="simple" xlink:href="http://servidor/aplicacion-web/showMap.do?
idMunicipio=42020_idMap=330"/>
    </DescriptionURL>
    <KeywordList>
      <Keyword>Callejero</Keyword>
      <Keyword>provincias</Keyword>
      <Keyword>municipios</Keyword>
    </KeywordList>
    <LogoURL format="image/jpeg" height="130" width="655">
      <OnlineResource xlink:type="simple" xlink:href="http://servidordemapas/wms?
LAYERS=Callejero_42020&REQUEST=getMap&FORMAT=image/png&TRANSP
ARENT=true&SERVICE=WMS&VERSION=1.1.1&STYLES=&SRS=EPSG:2
3030&BBOX=529692.0,4580798.0,544538.0,4603585.0&WIDTH=150&HEIGHT=
150"/>
    </LogoURL>
    <Title>Callejero_42020</Title>
  </General>
  <LayerList>
    <Layer queryable="1" hidden="0">
      <Server service="WMS" version="1.1.0" title="MapServer Users">
        <OnlineResource xlink:type="simple" xlink:href="http://servidordemapas/wms"/>
      </Server>
      <Abstract>Layer: provincias</Abstract>
      <Name>provincias</Name>
      <Title>provincias</Title>
      <FormatList>
        <Format current="1">image/png</Format>
        <Format>image/gif</Format>
        <Format>image/jpeg</Format>
        <Format>image/wbmp</Format>
      </FormatList>
      <StyleList>
        <Style current="1">
          <Abstract>style: provincias__default_provincias</Abstract>
          <Name>provincias__default_provincias</Name>
          <Title>provincias__default_provincias</Title>
        </Style>
      </StyleList>
    </Layer>
  </LayerList>
</ViewContext>
```

```

<Extension>
  <cached>true</cached>
  <urlCached>http://servidordemapas/gwc/service/wms</urlCached>
  <ol:singleTile>>false</ol:singleTile>
  <ol:transparent>TRUE</ol:transparent>
</Extension>
</Layer>
.....
</LayerList>
<WSpace>ws_42020</WSpace>
</ViewContext>

```

Donde resulta especialmente ilustrativo el uso de la etiqueta “*Extension*” y la forma de identificar los distintos recursos disponibles on-line.

Los documentos WMC permiten introducir nuevos elementos no definidos en la especificación. El mecanismo que se utiliza es una etiqueta del XML reservada para tal fin con el nombre “*extension*”. El esquema permite, al menos, ampliar la información proporcionada para el objeto mapa y para cada una de las capas.

Este tag extension se puede añadir bien sea en el elemento raíz del WMC o dentro del elemento “*layer*”. Se ha elegido introducir el concepto de capa cacheada dentro del elemento “*layer*” para poder particularizar la oferta de servicios de un servicio de mapas. El modelo propuesto ha sido el siguiente:

```

<Extension>
  <cached>true</cached>
  <urlCached>
    http://servidor:puerto/servidordemapas/gwc/service/wms
  </urlCached>
  <ol:singleTile>>false</ol:singleTile>
  <ol:transparent>TRUE</ol:transparent>
</Extension>

```

Donde los parámetros contienen la siguiente información:

- **cached:** indica si dicha capa es cacheada o no. Los posibles valores serán (TRUE/FALSE).
- **urlCached:** indica la URL del WMS-C que cachea dicha capa. Esta etiqueta únicamente es necesaria si la capa es cacheada.
- **ol:singleTile:** indica si la capa debe mostrarse mediante una técnica teselada o no, el prefijo ol se debe a la necesidad de facilitar que Openlayers lo interprete. Los valores posibles son (TRUE/FALSE).
- **ol:transparent:** indica si se desea transparencia para la capa. Los posibles valores son (TRUE/FALSE). Al igual que en el caso anterior este parámetro también lo puede interpretar el visor Openlayers.

## Servicios REST

Para proporcionar a Geoserver las características necesarias para esta implementación se han desarrollado nuevos servicios sobre los ya existentes en el servidor de mapas utilizando el interfaz orientado a recursos REST dado que es utilizado de forma nativa por el propio Geoserver. Los servicios web REST son una alternativa más simple a los servicios basados en SOAP y WSDL [12] y [13].

Se han generado nuevos servicios REST necesarios para:

- Crear mapa.
- Obtener información de un mapa
- Obtener lista de mapas existentes

### **Creación de un mapa nuevo**

La filosofía de REST reside en que todo lo que tiene asignado un URL puede considerarse un recurso existente o accesible mediante esa URL. En esta línea, la extensión de Geoserver habilita el concepto de mapa como recurso gestionado. Para crear un mapa nuevo hay que enviar un documento que represente el mapa mediante una petición HTTP estándar de tipo POST al URL que va a representar en lo sucesivo el recurso tipo mapa:

<http://servidor/rest/workspaces/workspaceid/maps>

### **Obtención de información de un mapa**

Para obtener la información de un mapa concreto accedemos al recurso que lo representa en el URL adecuado mediante una sencilla petición HTTP de tipo GET:

<http://servidor/rest/workspaces/workspaceid/maps/map>

La respuesta nos proporcionará datos del mapa igual que otras peticiones GET muestra documentos de otros tipos.

### **Obtención de la lista de mapas existentes**

Para obtener la información de la lista de mapas publicados en Geoserver esta disponible la operación GET

<http://servidor/rest/workspaces/workspace/maps>

La respuesta obtenida nos proporcionará un resumen de los mapas disponibles siguiendo el formato:

```
<maps>
< ViewContext>
  <name>nombre mapa1</name>
  <atom:link rel="alternate"
    href="http://servidor:puerto/rest/workspaces/workspace/maps/map_1"
    type="application/xml"/>
</ ViewContext>
< ViewContext>
  <name>nombre mapa N</name>
```

```
<atom:link rel="alternate"
href="http://servidor:puerto/rest/workspaces/workspace/maps/map_N"
type="application/xml"/>
</ ViewContext>
</maps>
```

## **CACHEADO DE CAPAS**

El WMS-C (GeoWebCache) y la información correspondiente informada en los documentos WMC se configuran automáticamente utilizando la información proporcionada en el cacheado de capas. Los parámetros de la caché se configuran para que cumplan la definición contenida en el WMC. Los clientes que sepan interpretar el apartado de extensión del estándar WMC podrán utilizar esta información cacheada en lugar de las capas equivalentes WMS recogidas en el WMC.

Al cachear una capa se realiza automáticamente la configuración de los siguientes parámetros:

- Perfil de teselado: se definen el bounding box de referencia y los niveles de escala disponibles.
- Sistema de proyección: incluyen el sistema de referencia y la proyección correspondiente.

## **INTEGRACIÓN DE WMS-C EN LA GESTIÓN DE MAPAS**

En el descriptor de un mapa se puede especificar que ciertas capas están disponibles como fuentes de teselas WMS-C. Esta información es de gran importancia tanto para los clientes como para el modelo mismo del catálogo del servicio. Cuando un mapa contiene una combinación de capas y estilos que están disponibles en la caché de teselas, dicha información se incluye en el documento WMC que representa el mapa, dando la oportunidad de aprovechar el almacén de cartografía pre-generada a los clientes que lo soporten.

Esta operación es automática y las operaciones que se realicen sobre el modelo de capas y estilos en el servidor se materializarán de forma coherente y segura.

## **REFRESCO DE LA INFORMACIÓN CACHEADA.**

Este punto de acceso al servidor permite notificar a la caché de teselas las modificaciones que se produzcan en la cartografía de forma que la caché sea siempre un reflejo exacto de la cartografía almacenada en el sistema. Para este propósito se ha implementado un servicio REST de tipo POST al que se le debe indicar una layer y el bounding box sobre el que ha sido realizada la modificación de cartografía y éste se encarga de actualizarlo en la Geowebcache. La estructura de la llamada es la siguiente:

<http://servidor/rest/layers/layer/update>



## INTEGRACIÓN CON LOCALGIS

Se han utilizado las nuevas características proporcionadas a Geoserver para mejorar el servicio de mapas proporcionado por el sistema LocalGIS. De esta forma se proporciona a este sistema un nuevo servidor de mapas más estable, que cumple más estándares OGC y sobre todo que proporciona la posibilidad de utilizar cacheado en las capas del sistema.

Dado que LocalGIS es un sistema que permite la modificación dinámica de la información de la cartografía y es necesario que estas modificaciones se plasmen directamente en la información que el gestor de cartografía ofrece a los clientes a través de sus múltiples interfaces de comunicación, es imprescindible la utilización del servicio REST que permite la actualización de la caché en las zonas requeridas.

Para la integración de los dos sistemas ha sido necesario adaptar el sistema de autenticación de forma que se pueda elegir si Geoserver utilizar sus propios ficheros de configuración de usuarios y permisos o la estructura de seguridad y control de acceso de LocalGIS.

## CONCLUSIONES

Tras este proyecto creemos que Geoserver está más cerca de ser un servidor de cartografía porque permite almacenar colecciones preestablecidas de capas, cada una de ellas con su correspondiente estilo y orden establecido.

Una instalación con hardware modesto podrá dar servicios de publicación de altas prestaciones, gracias al aumento de rendimiento de Geoserver, si se eligen adecuadamente las capas que se gestionarán con el sistema de cacheado de teselas. Esto hace que el rendimiento sea el necesario para permitir aplicaciones de acceso masivo como puede ser una explotación normal para un servicio de publicación de información geográfica municipal en Web.

Como líneas futuras se pretende integrar en esta gestión integral otros servicios de la IDE como CWS, WFS, etc.

## REFERENCIAS

- [1] OSGeo, "Tile Map Service Specification," *Tile Map Service Specification - OSGeo Wiki* Available: [http://wiki.osgeo.org/wiki/Tile\\_Map\\_Service\\_Specification](http://wiki.osgeo.org/wiki/Tile_Map_Service_Specification).
- [2] OSGeo, "WMS Tiling Client Recommendation - OSGeo Wiki" Available: [http://wiki.osgeo.org/wiki/WMS\\_Tiling\\_Client\\_Recommendation](http://wiki.osgeo.org/wiki/WMS_Tiling_Client_Recommendation).
- [3] Matt Mills, "NASA World Wind Tile Structure" Available: <http://www.ceteranet.com/nww-tile-struct.pdf>.
- [4] Tom Kralidis, "Web Map Context" Available: <http://www.kralidis.ca/gis/webmapcontext/>.
- [5] OGC, *OpenGIS Web Map Context Implementation Specification*, 2005.
- [6] J.G. Arne Kepp, "Making Maps Fast - How to increase the performance of GeoServer," Nov. 2009.
- [7] OGC, "OpenGIS Web Map Service (WMS) Implementation Specification," *Open Geospatial Consortium* Available: <http://www.opengeospatial.org/standards/wms>.
- [8] J. Masó y N. Julià, *OpenGIS® Tiled WMS Discussion Paper 0.3*, OpenGIS, 2007.
- [9] Lucian Plesea, "The Design, Implementation and operation of the JPL OnEarth WMS Server," *Geospatial Services and Applications for the Internet*, Sample, J.T., Shaw, K., Tu, S., y Abdelguerfi, M., eds., Berlin: Springer, 2008, págs. 93-

109.

- [10] J. Auksztol y T. Przechlewski, "Towards Flexible Geographic Information Infrastructure for e-Government," *Electronic Government*, 2004, págs. 377-380.
- [11] S. Tu, E. Normand, S. Kuchimanchi, V. Bizot, S. Shu, M. Abdelguerfi, J. Ratcliff, y K. Shaw, "Integrating Web services into map image applications," 2004, págs. 44-49 Vol.1.
- [12] R. Lucchi, M. Millot, y C. Elfers, *Resource Oriented Architecture and REST: Assessment of impact and advantages on INSPIRE*, European Commission, 2008.
- [13] R.T. Fielding, "Architectural Styles and the Design of Network-based Software Architectures," California, 2000.