

## Servidor de Cartografía Coloma.

*Manuel Clos Crespo<sup>(1)</sup>*

<sup>(1)</sup> Unitat Municipal d'Anàlisi Territorial (UMAT), Ajuntament de Girona, Plaça del Vi, 1, 17004 Girona, [manel.clos@ajgirona.cat](mailto:manel.clos@ajgirona.cat).

### RESUMEN

*En 2009 la Unidad Municipal de Análisis Territorial del Ajuntament de Girona apostó por integrar al sistema de información territorial existente un conjunto de herramientas libres para publicar cartografía. Un año después, más de la mitad de los servicios de cartografía web están funcionando sobre este nuevo sistema. El nuevo sistema, llamado Coloma, se basa en la base de datos PostGIS, la librería de render Mapnik, el servidor WMS Ogcserver, la librería Javascript OpenLayers, el framework Django, el lenguaje de programación Python y el servidor web Apache con el módulo para WSGI. Coloma incluye una herramienta de administración web para facilitar la configuración y la administración del día a día. Aporta pequeñas mejoras a los proyectos por separado y un valor añadido al conjunto, consiguiendo un servidor de cartografía formado por proyectos de éxito, robusto y con muchas posibilidades. Coloma ha demostrado ser fiable y sus componentes siguen mejorando día a día. El hecho de ser software libre nos ha permitido acceder al conocimiento, código y documentación que hay sobre todo este software en Internet. En los próximos meses la UMAT va a liberar el proyecto Coloma para que más gente pueda beneficiarse del trabajo hecho hasta ahora.*

**Palabras clave:** *PostGIS, Mapnik, Ogcserver, OpenLayers, Coloma, UMAT, Ajuntament de Girona.*

### INTRODUCCIÓN

Girona es una ciudad mediana, de casi 100.000 habitantes, con una área urbana funcional de unos 200.000 habitantes. Además es capital de provincia y centro administrativo, de servicios y turístico.

La Unidad Municipal de Análisis Territorial (UMAT) del Ajuntament de Girona, en funcionamiento desde 1993, es la encargada de la producción de cartografía, su gestión y su mantenimiento, así como de los trabajos periódicos y puntuales de análisis del territorio que a menudo son un referente para el conjunto de las áreas municipales en su trabajo o previsión anual.

La UMAT se encarga también del servicio de difusión de cartografía a través del Servidor de Cartografía. Es aquí donde se ha integrado Coloma, proporcionando una alternativa para la publicación de Cartografía, pero sin descartar el sistema actual, ya que Coloma se ha diseñado para convivir con el sistema actual y las actualizaciones futuras, para utilizar en todo momento la herramienta óptima para cada trabajo.

El Servidor de Cartografía empezó a funcionar en el año 2003 con 4 servicios de acceso interno (planeamiento, catastro, ortoimagen y topográfico) y otro de acceso externo, planeamiento, a finales de año. En 2004 eran ya 17 los proyectos de consulta interna, y en 2005 el total de servicios ascendía a 46.

Al acabar el año 2010, el Servidor de Cartografía cuenta con 80 servicios activos, 42 internos y 38 externos. La mayoría de los servicios externos están listados en la página web de la UMAT (<http://www.girona.cat/umat/cat/cartografia.php>). Del total de 80 servicios activos, 48 están funcionando sobre Coloma.

Coloma es una herramienta para servir cartografía a través de estándares OGC. Para la producción de la cartografía y su mantenimiento se siguen utilizando los programas de escritorio de ESRI (ArcInfo - ArcMap), así como ESRI ArcSDE para la base de datos principal donde se realiza la actualización de la cartografía. Próximamente se realizará la migración a ArcGIS 10.

## COLOMA

Existen multitud de componentes de software libre con los que se puede construir un servidor de Cartografía. La UMAT del Ajuntament de Girona ha ido escogiendo entre estos componentes los que mejor se adaptan a su infraestructura. Desde la inclusión a la estructura de difusión de Cartografía del primer componente de software libre, se ha producido una migración progresiva de servicios de cartografía. De esa migración y de la automatización de procesos nace Coloma.

Los componentes que forman Coloma son los siguientes:

- Ubuntu, sistema operativo tipo Debian, Linux
- **Postgresql + PostGIS**, base de datos de cartografía (almacén de datos), procesador de datos, indexación, fácil acceso desde Mapnik y Python.
- **Mapnik**, herramienta de renderización
- **OGCServer**, servidor WMS, Mapnik, Python, fácil de extender
- **TileCache**, servidor WMSC, funciona, Python
- **Apache y mod\_wsgi**, servidor web y soporte para Python
- **Django**, base de la parte de servidor del visor web
- **OpenLayers**, base del visor web, WMS
- **Python**, lenguaje ágil y potente

## Un poco de historia

### ArcIMS

La arquitectura de partida es un servidor ArcIMS accediendo a los datos que se encuentran en la base de datos ArcSDE. El número creciente de servicios de cartografía hace que alcancemos el límite de lo que ArcIMS es capaz de soportar. Es necesario evolucionar.

### **ArcGIS y OpenLayers**

Aparece en escena ArcGIS Server, en concreto la versión 9.3.0. Las pruebas previas a la migración nos desvelan un nuevo reto más allá de empezar a trabajar con ArcGIS Server: ArcGIS Server no soporta la librería java ArcIMS jconnect en la que está basado la segunda versión del visor web de la UMAT y sobre el cual ya funcionan más de la mitad de los servicios de cartografía. La migración a un nuevo visor parece insalvable.

Gracias a la posibilidad de ArcGIS Server de publicar las capas de los servicios a través de WMS, y después de realizar las pruebas correspondientes, incorporamos el primer componente de software libre: OpenLayers. El nuevo visor está basado en OpenLayers, lo que nos permite dar un salto cualitativo: reusar un proyecto de éxito como OpenLayers y a la vez empezar a consumir los servicios a través de un protocolo estándar OGC.

### **TileCache**

El paso lógico una vez se consumen los servicios a través de WMS es crear una caché de teselas. Los puntos fuertes que nos decantan hacia TileCache por encima de la cache de ArcGIS Server son:

- Implantación sencilla, basada en un script cgi, Python y un fichero de configuración plano
- Más flexibilidad, formatos, opciones, generación al vuelo...
- Sencillez para consumir las teselas desde OpenLayers
- Los buenos resultados de TileCache en las pruebas: rápido, estable y bajo consumo de recursos

### **Mapnik, PostGIS, OGCServer**

ArcGIS Server 9.3.0 tiene todavía aspectos que mejorar y bugs que solucionar. Esto nos hace plantearnos escoger otra manera de publicar los datos. Mapnik impresiona por la calidad del renderizado (antialias, transparencia...), ofrece conexión a PostGIS, y además dispone de un servidor WMS escrito en Python con soporte para las operaciones básicas necesarias (GetCapabilities, GetMap, GetFeatureInfo).

El punto débil de Mapnik es que no cuenta con un editor gráfico para la edición del mapa que se publica (QGIS + QuantumGIS no son una opción). Se pierde la "publicación directa" con la que contábamos hasta ahora: Arcmap -> ArcGIS Server. Una de las ideas para mejorar la integración de Coloma y ArcGIS es generar la descripción del mapa directamente desde ArcMap gracias a ArcPy de ArcGIS 10.

### **ArcSDE**

El reto de integrar Mapnik no consiste en cómo publicar los datos, sino en cómo mantenerlos actualizados. Los servicios basados en ArcIMS o ArcGIS Server acceden siempre a la última versión de los datos. Mapnik no puede acceder a los datos guardados en ArcSDE, lo que significa que hay que mantener una copia de los datos y actualizar esos datos periódicamente.

El precio de servir la cartografía directamente desde la base de datos principal es el de tener las capas bloqueadas. Si en un "Dataset" se está utilizando alguna capa, no es posible modificar la estructura de ninguna de las capas del "Dataset" (por ejemplo añadir un campo a una capa), tampoco crear un índice, etc...

La base de datos desde la cual Coloma sirve la cartografía es una copia parcial de la base de datos de producción de cartografía (ArcSDE). Eso significa menos bloqueos en las capas que hay que editar, la posibilidad de exportar sólo los campos y filas necesarios para la publicación web (menos datos, mayor velocidad), e incluso una versión interna para los servicios con acceso exclusivamente interno.

La automatización de este último punto es el que ha requerido más tiempo y programación. Es también posible gracias a la utilidad `sde2shp` que acompaña a ArcSDE y que hace posible la exportación de datos a un formato ampliamente soportado, haciendo accesibles los datos guardados en ArcSDE y evitando a la vez la complejidad de acceder a una base de datos versionada, como es el caso de la Base de Datos de la UMAT.

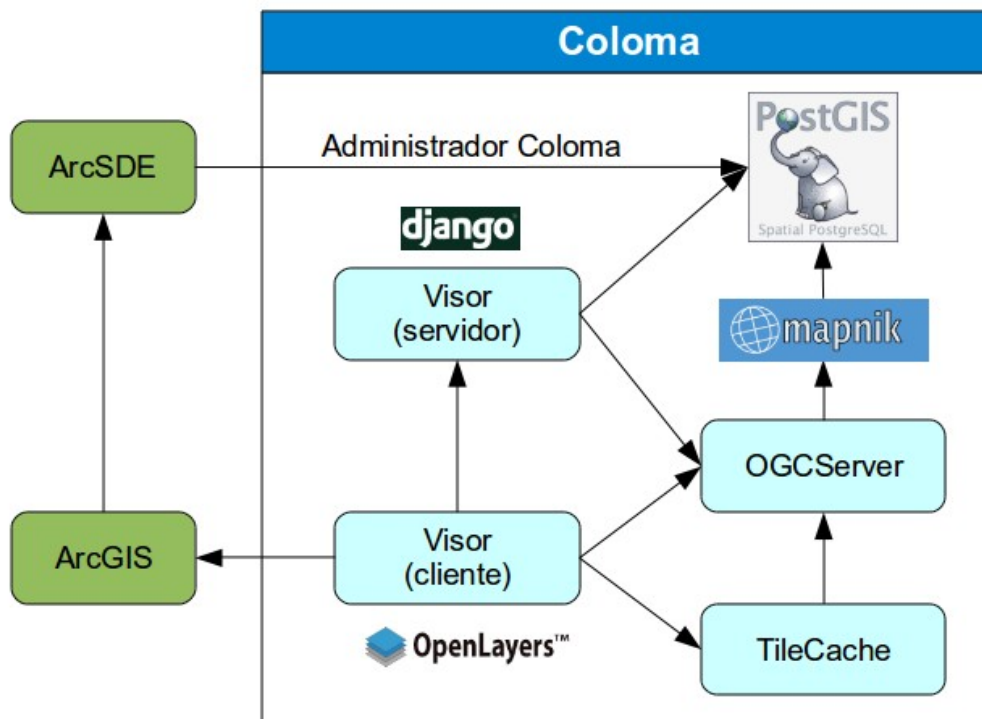


Figura 1: Esquema del funcionamiento de Coloma.

## ¿Qué aporta Coloma?

### Solución probada

Coloma se basa en proyectos de éxito y de acceso libre. En los casos que ha sido necesario modificar algún componente, realizar una mejora o solucionar un fallo, se ha enviado el código o parche al proyecto correspondiente (p.e. Mapnik, OGCServer, TileCache).

### Tecnología

Coloma se maneja desde una API REST, lo que permite una integración más sencilla en sistemas existentes. De hecho el administrador web realiza las acciones a

través de esta API. Esta misma API permite controlar varios servidores Coloma desde una sola interfície web.

Coloma es una aplicación Python lo que permite crear módulos a medida que funcionan dentro de Coloma que tengan en cuenta las peculiaridades del entorno en el que debe funcionar.

#### **Administrador web**

El administrador web facilita la actualización de los datos desde ArcSDE hacia PostGIS, la comprobación del estado de los servidores y la posibilidad de reiniciar los servicios para actualizar la configuración.

Una de las características principales del administrador web es la posibilidad de trabajar con varios servidores y varias bases de datos, facilitando el mantenimiento de servidores y base de datos con diferentes finalidades: servidores internos, servidores de pruebas, bases de datos específicas para servir datos analíticos, etc.

#### **Visor web**

El visor web está formado por una aplicación Django (Python) que permite la programación de funciones adicionales en el servidor, y por el cliente web basado en OpenLayers y JQuery.

El visor web es una aplicación basada en una base común y una configuración específica para cada servicio, permitiendo personalizar las capas que se muestran, las herramientas disponibles, las plantillas html que forman el visor, las opciones de las herramientas de información y selección, etc.

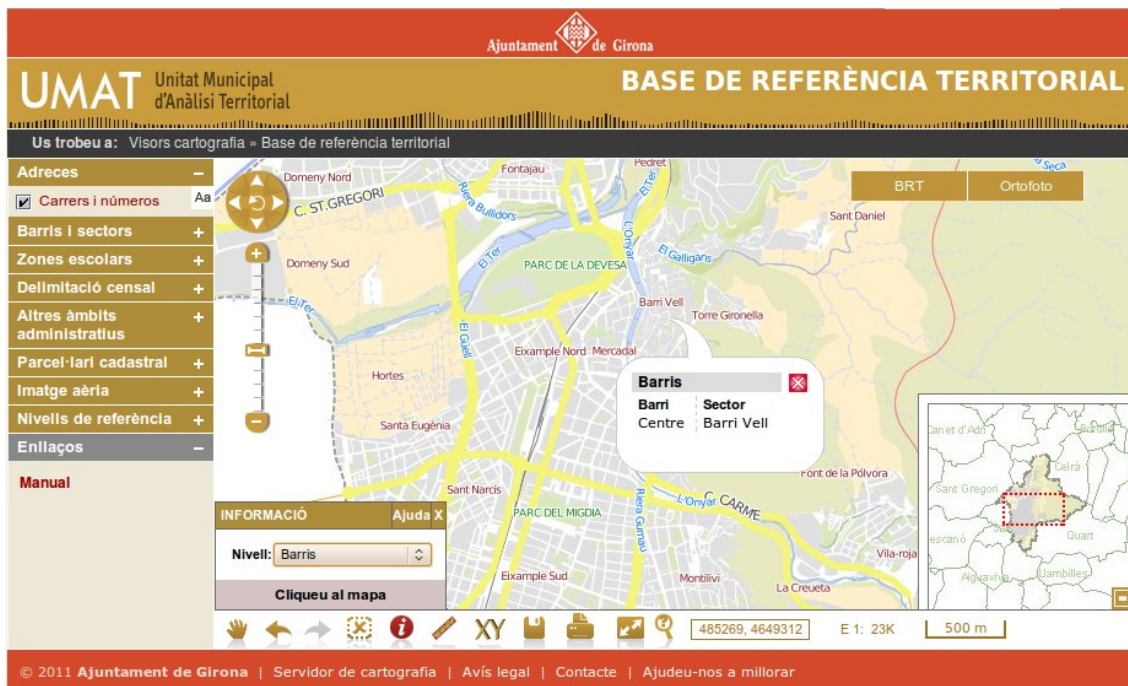


Figura 2: Captura del visor de la UMAT

## CONCLUSIONES

La utilización de Software Libre de calidad nos ha permitido realizar un salto cualitativo en nuestra estructura de publicación de Cartografía Web, haciendo posible el crecimiento del número de servicios con un coste mínimo.

La nueva estructura ha demostrado fiabilidad, estabilidad y bajo consumo de recursos, lo que nos permite invertir más tiempo en la mejora de los servicios y menos en su mantenimiento.

De la misma manera que hemos aprovechado la experiencia depositada en el software libre que utilizamos, esperamos que Coloma pueda ser útil a más gente.

## AGRADECIMIENTOS

A todos los que directa o indirectamente han participado en los proyectos de Software Libre que hacen posible Coloma.

A la UMAT por apostar por este proyecto y a todos sus componentes por su ayuda, recomendaciones y opiniones.

Al departamento de Informática del Ajuntament de Girona por su apoyo y paciencia.

A Roger Colom por el diseño del visor.

## REFERENCIAS

- ◆ Ubuntu, <http://www.ubuntu.com>
- ◆ PostGIS, <http://postgis.refractor.net/>
- ◆ Mapnik, <http://mapnik.org/>
- ◆ JQuery, <http://www.jquery.org/>
- ◆ OpenLayers, <http://openlayers.org/>
- ◆ MetaCarta Inc. (2009), TileCache 2.10, <http://tilecache.org/>
- ◆ Django, <http://www.djangoproject.com/>
- ◆ Piston, <http://bitbucket.org/jespern/django-piston/wiki/Home>