14 Mapas a partir de la web y de los móviles

Michael Peterson, EE.UU.

Resumen

La cartografía siempre ha sido dependiente de un conjunto de herramientas, tanto para medir el mundo, como para elaborar mapas. Desde el año 2005, se ha desarrollado para hacer mapas un nuevo conjunto de herramientas en línea o basadas en la nube, llamadas Interfaces de Programación de Aplicaciones (*Application Programming Interface,* (API)). Esas herramientas cuentan con el beneficio adicional de estar disponibles de forma automática, para cualquier persona con una conexión a Internet. En este capítulo se examina esta nueva era de la cartografía, y se muestra cómo elaborar y distribuir mapas utilizando herramientas de cartografía gratuitas.

14.1 Introducción

Es difícil exagerar la importancia de los mapas como una forma de comunicación acerca del mundo. Los mapas nos ayudan a entender tanto nuestro entorno, como el espacio más allá de nuestra percepción directa. Los mapas influyen en nuestra forma de pensar sobre el mundo, y en nuestra forma de actuar dentro de él. Nos conectan con nuestro entorno. Cada uno de nosotros es un cartógrafo, en el sentido de que todos hacemos mapas mentales. A veces, incluso tenemos que dibujar esos mapas a otras personas, con el fin de ayudar a explicar cómo encontrar un determinado lugar.

La elaboración de los mapas y el análisis de la información subyacente se han convertido en una ciencia y constituyen valiosas habilidades de gran utilidad en muchos tipos diferentes de trabajos. Muchas de las herramientas para la cartografía se encuentran ahora en la nube –un sofisticado sistema de *hardware* y *software* accesible a través de Internet–. Las herramientas de cartografía basadas en la nube permiten formas muy avanzadas de cartografía. Otra ventaja de esos mapas en línea es que pueden ponerse fácilmente a disposición de los demás usuarios.

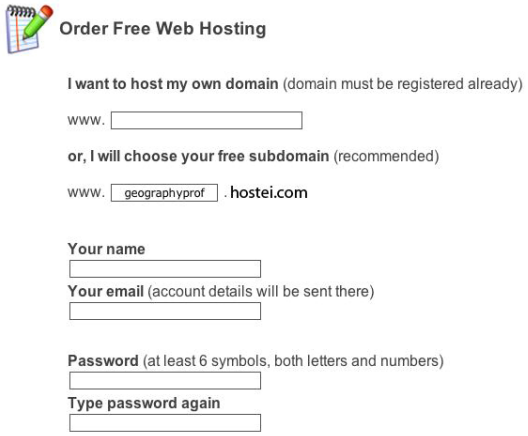


Figura 14.1a. Página de registro para http://000webhost.com. *La información de la cuenta se muestra a la derecha. La elección de un subdominio es gratis. Se cobra por especificar un dominio con un nombre específico, como* http://www.peterson.com, *pero hacer que el sitio elija un dominio gratuito, como* http://geographyprof.hostei.com, *es gratis.*
(© 2014 First Class Web Hosting).

Hacer mapas en la nube requiere el uso de un servidor. Si bien es posible transformar casi cualquier ordenador en un servidor, es más fácil y más seguro utilizar un servicio de alojamiento en línea (*online hosting service*, en inglés). Servicios como Amazon Web Services y Microsoft Azure implementan escalabilidad (Nota del Traductor: capacidad de manejar un crecimiento continuo del trabajo de manera fluida) en el sentido de que pueden ser actualizados para servir a millones de usuarios de manera simultánea. Esos servicios cobran por espacio en disco y por ciclos de uso de procesamiento en el servidor.

» Account Information	
Domain	geographyprof.hostei.com
Username	a8040697
Password	******
Disk Usage	0.2 / 1500.0 MB
Bandwidth	100000 MB (100GB)
Home Root	/home/a8040697
Server Name	**server33.000webhost.com**
IP Address	64.120.177.162
Apache ver.	2.2.13 (Unix)
PHP version	5.2.*
MySQL ver.	5.0.81-community
Activated On	2011-03-10 16:23
Status	Active

Figura 14.1b. Información de la cuenta.

Otra opción para el alojamiento de páginas web la constituyen los servicios de alojamiento de páginas web que solo cobran por el almacenamiento en disco. Si bien esos servicios no son tan escalables, sí son más fáciles de usar y ofrecen planes con subdominios gratis de hasta 1.500 MB de espacio en disco. Dos de estos servicios de alojamiento de páginas web gratuitos son `000webhost.com` y `podserver.info`. Las secciones siguientes describen cómo crear un sitio web y ofrecer mapas usando uno de estos servicios de alojamientos en la web gratuitos.

14.2 Servidores en la nube

14.2.1 Haciéndose espacio en la nube

La figura 14.1 muestra la página de registro para el proveedor 000webhost.com y la información de la cuenta resultante. Obsérvese cómo se solicita un subdominio gratis bajo el dominio hostei.com. Mientras que una dirección como http://geographyprof.com podría ser solicitada en la línea superior, eso supondría incurrir en un cargo, ya que representaría un

nuevo dominio. Para obtener una cuenta gratuita, es importante no registrar un dominio propio.

Se necesita una dirección de correo electrónico para inscribirse en el subdominio gratuito. La página Información de Cuenta muestra que la dirección web que le fue asignada por el proveedor de servicios es http://geographyprof.hostei.com (o http://64.120.177.162) y que tiene 1500 MB de almacenamiento gratuito. La información de la cuenta también muestra la disponibilidad del servidor web Apache y otras herramientas en línea incluyendo PHP y MySQL.

En la mayoría de los servicios de alojamiento en la web, se incluye una interfaz gráfica para los servicios que se ofrecen. Eso se conoce como panel de control, o cPanel (véase la figura 14.2). Las herramientas gestionan el correo electrónico, la edición de ficheros, la programación de tareas y la gestión de cuentas. Todas estas herramientas representan los proyectos de código abierto que escribe y mantiene un pequeño batallón de programadores. File Manager es la herramienta más útil para la gestión de ficheros y la creación de páginas web. MySQL y phpMyAdmin se utilizan para la construcción de bases de datos. La mayoría de los servicios de alojamiento de páginas web utilizan un cPanel similar para poder acceder a los recursos del servidor.

La figura 14.3 muestra la ventana del Administrador de ficheros con acceso a las herramientas para subir y crear nuevos ficheros y directorios (subcarpetas). Las herramientas también están disponibles para mover, borrar y renombrar ficheros. Una lista de ficheros muestra el nombre, el tipo y el tamaño de los ficheros, mientras que los campos relativos al propietario, al grupo y a los permisos presentan la configuración de seguridad. Mod Time indica cuándo ha sido modificado el fichero por última vez. Los ficheros se pueden editar directamente desde esta ventana haciendo clic en Editar al final de cada nombre de fichero.

Figura 14.2. Panel de control de alojamiento web estándar, llamado cPanel, que da acceso a diferentes herramientas. File Manager es el programa principal para subir y editar ficheros. (© 2014 First Class Web Hosting).

La carpeta **public_html** es el directorio desde el que se sirven todos los ficheros de la web. Si un fichero html se va a presentar a través de una página web, debe residir en esa carpeta. Por lo general, esa carpeta contiene un fichero llamado `index.htm` (o `index.php`) que es la primera página a la que se accede cuando se hace referencia al sitio. Por ejemplo, si una dirección como http://geographyprof.hostei.com/CloudM_apping/ se introduce en un navegador, el navegador buscará un fichero llamado `index.htm` en un directorio (carpeta) llamado `CloudMapping` que está a su vez ubicado en el directorio `html público`. Eso significa que las dos direcciones siguientes mostrarían el mismo fichero: http://geographyprof.hostei.com/Online_Mapping/ http://geographyprof.hostei.com/Online_Mapping/index.htm

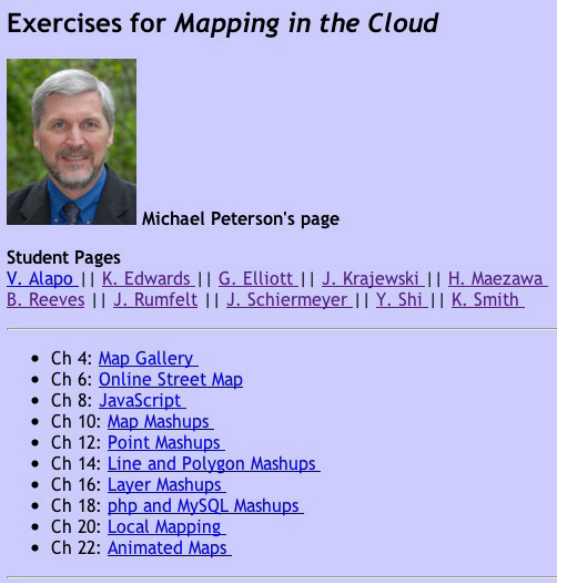
*Figura 14.3. Ventana del Administrador de ficheros de un servicio de alojamiento en línea. Este servicio permite que los ficheros se puedan crear y editar. Todos los ficheros que se sirven a través de la web deben estar en el directorio **public_html**. (© 2014 First Class Web Hosting).*

Normalmente, este fichero index.htm sirve como punto de entrada de la página web y tendrá enlaces a todos los demás ficheros en el directorio.

El fichero `index.htm` tendrá una estructura relativamente simple –es decir, un título seguido de enlaces a todas las asignaciones de tareas–. Ese fichero puede tener una imagen o foto del titular de la web y enlaces a las páginas web de otros usuarios, tal como se muestra en la figura 14.4. El código correspondiente muestra cómo se inserta una imagen con la etiqueta (*tag*):

<img src = filename>

Los enlaces a las páginas de los otros usuarios están separados por dos líneas verticales («||»). El código para el fichero de índices completo puede obtenerse buscando «Cartografía de Peterson en la Nube» («Peterson Mapping in the Cloud»).



14.2.2 HTML

El HTML (HyperText Markup Language) constituye la piedra angular de la web. Es el lenguaje que hace posible presentar información a través de páginas web. También es un contenedor de lenguajes de «script» como JavaScript y PHP. Es necesario un poco de conocimiento sobre HTML para la presentación de mapas a través de Internet.

HTML consta de etiquetas (*tags*) que definen el diseño de la página (Willard, 2009). Incluye códigos de texto simples, rodeados de los delimitadores «<» y «>», que van a especificar el aspecto que tendrá el documento en el navegador. Otras etiquetas HTML también crean enlaces (*links*) a documentos, o pueden mostrar un fichero gráfico. Se puede usar para entrar en el fichero un editor de texto normal (Bloc de notas –Notepad– en Windows o TextEdit con Macintosh, mediante los ajustes apropiados). A diferencia de los procesadores de texto, esos programas están preparados para la introducción de texto sin formato. Una vez creados los ficheros, se pueden abrir con un navegador tal como Explorer, Firefox o Chrome.

```
<html>
<head></head>
<body bgcolor="#CCCCFF">
<h2> Exercises for <i>Mapping in the
Cloud</i> </h2>
<img src=peterson.jpg height=150><b>
Michael Peterson's page
</b><p>
<p>
<b> Student Pages </b><br>
<a href=http://victoriaA.site88.net> V.
Alapo </a> ||
<a href=http://mapsarefuntoo.web44.net> K.
Edwards </a> ||
<br> <hr>
<ul>
<li>
Ch 4: <a href=http://maps.unomaha.edu/
onlinemapping/ code04.zip>
Map Gallery </a><br>
<li>
Ch 6: <a href=code06.zip> Online Street
Map</a><br>
<li>
Ch 8: <a href=http://maps.unomaha.edu/
onlinemapping/ code08.zip>
JavaScript </a><br>
</a><br>
</ul>
<hr>
</body>
</html>
```

Figura 14.4. Muestra un ejemplo de un fichero index.htm, que incluye una foto, vínculos (links) a todos los otros usuarios y enlaces a las asignaciones de tareas. El código muestra cómo presentar una imagen con el tag (etiqueta) img, enlaces a las páginas de los otros estudiantes y a las asignaciones de tareas.

Todos los ficheros HTML comienzan con la etiqueta **«<html>»**, y terminan con el mismo código precedido de una barra (por ejemplo, **«</ html>»**). La barra en la parte delantera del código indica que la codificación HTML está finalizada. Técnicamente, todas las etiquetas HTML tienen un principio y un final, con la etiqueta final indicada con **«/.»** Por ejemplo, el comando **<h1>** se utiliza para iniciar el texto del encabezado –texto más grande utilizado para títulos– y el código **</ h1>** finaliza el formato de texto de cabecera.

Ahora podemos mostrar un mapa con la etiqueta **img**. Todos los nombres de los ficheros a los que se hace referencia como **«img»** debe terminar con GIF, JPG (o JPEG) o PNG, ya que estos son los tipos de fichero comunes para los diferentes navegadores. La opción **img «src»** se utiliza para especificar la dirección URL del fichero. En el siguiente ejemplo, obsérvese el designador de fichero de imagen **«.gif»** al final de la URL. El comando img también incluye un número de opciones que se pueden utilizar para alterar el tamaño de la imagen o cambiar su ubicación en la página. Como **hr**, **img** no tiene una etiqueta de cierre o finalización estándar. En formato XHTML estricto, se escribe:

<img src="http: maps// .unomaha.edu/OnlineMapping/Chapter4/MapExample1.gif " width="500"height="389">

La etiqueta **<embed>** se utiliza para mostrar los ficheros gráficos que no están en los formatos GIF, JPEG o IMG. Los ejemplos incluyen PDF de Adobe, Flash, SVG y QuickTime.

El formato es idéntico a la de la opción **<img>**:
<embed src="http://maps.unomaha.edu/Cloud_Mapping/Chapter4/MapExample4.pdf" width="500" height="389">

Una de las etiquetas más usadas en HTML, es el «ancla» que se utiliza por lo general con una cadena de texto para crear un enlace de hipertexto. También se puede utilizar con la etiqueta **img** para crear un enlace de hiperimagen. El formato del comando de anclaje es el siguiente:

<a href=http://maps.unomaha.edu/Cloud_Mapping/Chapter4/MapExample4.pdf
> Click for PDF file

14.2.3 JavaScript

Por sí mismo, HTML es tan solo un lenguaje de formato de página. Pero en combinación con JavaScript, una página HTML puede ejecutar un código de ordenador (W3Schools.com 2011). Antaño relegada al ámbito de los «empollones», la programación hoy en día se ve como una forma de expresión, como la «amplificación del pensamiento», y una habilidad necesaria. Este concepto de «la codificación como alfabetización» está promoviendo nuevas ideas sobre la importancia de ser capaz de programar. Sitios de programación en línea como la CodeAcademy cada vez son más populares. La Khan Academy ha introducido un conjunto similar de ejercicios gratuitos de programación en línea.

CODE	Result
<html>	x=4
<body>	
<script type="text/javascript">	
var x = 2 * 2	
document.write("x = ", x)	
</script>	
</body>	
</html>	

Figura 14.5. Cálculo realizado con JavaScript dentro de un archivo HTML.

Los siguientes ejemplos proporcionan una breve introducción a JavaScript, y cómo se puede utilizar para otras funciones.

Las funciones son los bloques de construcción fundamentales de JavaScript. Una función es un procedimiento –un conjunto de sentencias que realizan una tarea específica–. Las funciones se definen generalmente en la sección de cabecera de un documento HTML. Esto asegura que todas las funciones se definen antes de mostrar cualquier contenido. El ejemplo de la figura 14.6, define una simple función en la sección de cabecera de un documento HTML.

Code	Result
<head>	The function returned 25.
	All done.
<script LANGUAGE="JavaScript">	
function square(number) {	
return number * number	
}	
</script>	
</head>	
<body>	
<script>	
document.write("The square of 5 is ",	
square(5), ".")	
</script>	
<P>All done.</P>	
</body>	

Figura 14.6. Función que multiplica al cuadrado el número que se le pasa mediante una llamada a la función.

La función **square** (cuadrado) toma un argumento, llamado *number* (número), y la función consiste en una declaración, **return** `number * number`, que indica que debe devolver el argumento de la función multiplicado por sí mismo. La sentencia **return** especifica el valor devuelto por la función.

El fichero externo que contiene las funciones de JavaScript puede estar en el mismo equipo que el fichero HTML, tal y como se ve en la figura 14.7, o puede estar en otro

Code	Result
function square(number) {	A separate file
return number * number	called "common.js"
}	
<head>	The function
	returned 25.
	All done.
<title>Referencing a !le of functions</title>	
<script src="common.js">	
</script>	
</head>	
<body>	
<script>	
document.write("The square of 5 is ",	
square(5), ".")	
</script>	
All done.	
</body>	

Figura 14.7. Función que se coloca en un documento externo, common.js. A la función se le hace referencia a continuación, a partir del fichero HTML con <script src="common.js">.

equipo o servidor. Así es como se distribuye el código *Application Programming Interface* (API). Una referencia a una biblioteca de código API hace posible, a un diseñador de páginas web, el acceso a miles de funciones.

En lugar de incrustar el código JavaScript directamente en el fichero HTML, ya sea en el cuerpo o en la cabecera, es posible colocar las funciones de JavaScript en un fichero separado. El atributo SRC de la etiqueta **<script>**, especifica el fichero externo, donde se puede encontrar el código JavaScript. La figura 14.8 muestra el fichero externo llamado *common.js,* y cómo se le hace referencia en la sección de cabecera de un documento HTML. El fichero JavaScript externo puede contener múltiples funciones, pero no código HTML.

La figura 14.8 muestra cómo se referencia el código API de **Google Maps**. Para ayudar en la depuración, el código API de Google Maps funciona localmente en un ordenador sin la necesidad de transferir el código a un servidor. Pero, con el fin de que otros puedan ver el mapa, el código debe residir en un servidor.

```
Code
<head>
<title>Google Maps JavaScript API
Example</title>
<script type="text/javascript"
src="http://maps.google.com/maps/api/js?sen
sor=false">
</script>
</head>
```

Figura 14.8. Implementación de una llamada a la API de Google Maps que tiene acceso a un gran número de funciones relacionadas con la cartografía. Aquí, el sensor está ajustado en «false» (falso). El valor del sensor se establece en «true» (verdadero) si se utiliza un dispositivo móvil que pueda proporcionar la posición actual.

14.3 La API de Google Maps

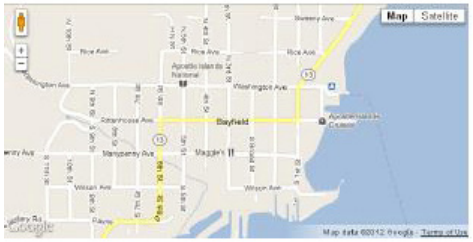
Introducida poco después de Google Maps en el año 2005, la Interfaz de Programación de Aplicaciones de Google Maps (API de Google Maps) consiste en una serie de funciones que controlan la apariencia del mapa, incluyendo su escala y ubicación, y cualquier información añadida en forma de puntos, líneas o áreas y descripciones asociadas. El uso de la API de Google Maps es esencialmente libre, siempre y cuando el sitio no cobre por el acceso. Google pone una limitación en el número de mapas que pueden ser servidos. Un sitio no puede generar más de 25.000 mapas cargados al día, durante 90 días consecutivos. Un mapa cargado es un mapa que se muestra con la API de Google Maps. Una vez cargado, el grado en que un usuario interactúa con un mapa no tiene impacto en el número de mapas cargados. Sería muy difícil para un usuario medio de la API de Google Maps el exceder las 25.000 cargas de mapas. Incluso si un sitio se convirtiese en «viral», tendría que mantener las 25.000 cargas de mapas por día durante 90 días consecutivos antes de alcanzar el límite. Los límites de uso pueden ser colocados en algún lugar de la página, con el fin de que no se pueda superar ese número. Si el sitio superase constantemente los 25.000 mapas al día, Google requeriría que ese sitio se registrase y pagase US $ 0,50 por cada 1.000 mapas vistos más allá de ese límite. Por ejemplo, si su página de mapas de Google sirvió 100.000 mapas al día durante 90 días consecutivos, se le cobraría $ 37,50 (75.000÷1.000 x 0,5) un mes después del período inicial de 90 días.

Los servicios de web especializados de la API de Google Maps tienen límites de uso adicionales, incluyendo:

- Direcciones: proporciona direcciones en forma de texto –limitadas a 2.500 al día–.
- Matriz de Distancias: devuelve distancias y tiempos de viajes –limitado a 100 elementos por consulta y 2.500 al día–.
- Elevación: –elevación en puntos– limitado a 2.500 solicitudes por día, donde cada solicitud devuelve hasta 512 elevaciones.
- Codificación geográfica: convierte una dirección cualquiera de una calle en latitud y longitud –limitado a 2.500 al día–.
- Lugares (*Places*): establecimientos comerciales y otros puntos de interés en torno a un punto –requiere una clave de API y se limita a 1.000 peticiones al día–.

Una clave de API de Google Maps es un código numérico que registra cada sitio (el propio de cada usuario, por ejemplo) en Google. No es necesaria para las aplicaciones normales y solo sería necesaria si los límites de uso se superan o se utiliza el servicio de *Web Places*.

El ejemplo de la figura 14.9 muestra los códigos de JavaScript y las llamadas a la API que se necesitan para mostrar un mapa sencillo que se centra en un lugar específico. El nivel de zoom, que puede oscilar entre 0 y 21, se establece en 15 en `myOptions`. El centro se define con un valor específico de latitud y longitud, y se selecciona la opción de hoja de ruta (ROADMAP) para definir el estilo del mapa. Todas las llamadas a la API




```
if(navigator.geolocation) {
browserSupportFlag = true;
navigator.geolocation.getCurrentPosition(funct
ion(position) {
initialLocation = new google.maps.LatLng
(position.coords.latitude,position.coords.longitude);
contentString = "Pt:
"+position.coords.latitude+",
"+position.coords.longitude ;
map.setCenter(initialLocation);
infowindow.setContent(contentString);
infowindow.setPosition(initialLocation);
var marker = new google.maps.Marker({
position: initialLocation,
map: map,
title:"Hello World!"
});
google.maps.event.addListener(marker, 'click',
function() {
infowindow.open(map,marker);
});
}, function() {
handleNoGeolocation(browserSupportFlag);
});
}
}
```

Figura 14.9. Petición básica de un mapa en Google Maps. El centro del mapa se puede cambiar junto con el nivel de zoom y el tipo de mapa. (© 2014 Google).

se hacen en la función **initialize**. Esta función se llama **onload** dentro del cuerpo del fichero HTML.

Se puede hacer un simple cambio en el código mediante la sustitución de nuevos valores de latitud y longitud. La determinación de la latitud y longitud de un punto específico se puede hacer de diferentes maneras:

- En Google Maps con un clic en el botón derecho (control + clic en Mac) y seleccionando «¿Qué hay aquí?» y la longitud del punto aparecerá en la línea superior de la ventana de Google Maps.

- Un clic derecho con MapQuest muestra los valores en una ventana emergente «*pop-up*».
- En Bing Maps, la latitud y longitud se muestran con un clic derecho.
- Para mostrar las coordenadas en el formato de grados decimales con Google Earth, seleccione Herramientas/Opciones y haga clic en la opción grados decimales.
- Por último, existen diversas utilidades en línea. Si se busca «Encontrar latitud y longitud» eso nos llevará a un determinado sitio. La mayoría de esos sitios utilizan Google Maps, incluyendo este ejemplo: http://findlatitudeandlongitude.com.

Otro de los cambios con el Google Maps básico es el tipo o estilo de mapa que se muestra. Google ofrece cuatro tipos de vistas:

- **MapTypeId.ROADMAP** muestra la vista predeterminada de un mapa de carreteras.
- **MapTypeId.SATELLITE** muestra imágenes de satélite de Google Earth.
- **MapTypeId.HYBRID** muestra una mezcla de puntos de vista normales y de satélite.
- **MapTypeId.TERRAIN** muestra un mapa físico basado en información del terreno.

El nivel de zoom inicial también puede modificarse. Un valor de «0» dibujaría un mapa a pequeña escala. A medida que el número de nivel de zoom aumenta, también lo hace la escala del mapa. El valor superior varía para diferentes partes del mundo. En general, suelen estar siempre disponibles 20 niveles de detalle. Algunas partes del mundo tienen más de 20 niveles de zoom.

14.4 *Mash-ups* de puntos, líneas, áreas y capas

Todos los mapas se componen de puntos, líneas y áreas. Además, se pueden combinar varios mapas como si fuesen capas individuales, una función que es la base de los Sistemas de Información Geográfica. Los ejemplos de esta sección muestran cómo se pueden agregar esos elementos al mapa de Google Maps.

14.4.1 Puntos

El marcador de Google, por defecto, es el símbolo de una gota de lluvia al revés, pero hay disponibles un gran número de iconos alternativos. Incluso es posible diseñar símbolos porque son simplemente imágenes de 32x32 píxeles en formato PNG. Los marcadores pueden ser estáticos o interactivos. El tipo más importante de marcador interactivo es aquel en el que se pueda hacer clic.

```
function initialize()
{
var myLatlng = new google.maps.LatLng(41.258531, -96.012593);
var myOptions = {
//this defines the zoom level
zoom: 14,
center: myLatlng,
mapTypeId: google.maps.MapTypeId.ROADMAP
}
var map = new
google.maps.Map(document.getElementById("map_canvas"),
myOptions);
var contentString='Cartography & GIS Laboratory<br>University of
Nebraska at Omaha'
var infowindow = new google.maps.InfoWindow(content:
contentString});
var marker = new google.maps.Marker({
position: myLatlng,
map: map,
title: "Cartography and GIS Laboratory"
});
google.maps.event.addListener(marker, 'click', function(){
infowindow.open(map, marker);
});
}
```

*Figura 14.10. Ejemplo de un marcador en el que se puede hacer clic. La variable de texto **contentString** se define en HTML. (© 2014 Google).*

En el ejemplo de la figura 14.10, la variable **contentString** se define con el texto formateado en HTML. Eso se asocia con una variable de **infoWindow** que posteriormente se asocia con **google.maps.event.addListener**. Cuando el usuario hace clic en el marcador, el texto se muestra en una burbuja emergente (*pop-up*). El HTML para esta burbuja podría incorporar una fotografía o incluso un video usando las etiquetas **img** o **embed**.

En los casos en que se debe cartografiar un gran número de puntos, que deben ser actualizados con frecuencia, se suele usar un formato web llamado Really Simple Syndication (RSS). RSS tiene muchas ventajas. Los editores de los canales RSS se benefician mediante la sindicación del contenido de forma automática, mientras que los usuarios se benefician a través de las oportunas actualizaciones de la información cartografiada. Un formato de fichero estandarizado permite publicar la información una sola vez y que se vea utilizando muchos programas diferentes.

KML, Keyhole Markup Language, es un formato que se utilizó para describir el espacio de dos y tres dimensiones, que se desarrolló originalmente para Google Earth. Ahora es un estándar abierto llamado oficialmente OpenGIS® KML Encoding Standard (OGC KML) y es mantenido por el Open Geospatial Consortium (OGC). El formato especifica características tales como marcas de posición, imágenes, polígonos y modelos en 3D. Los lugares siempre se especifican con su latitud y su longitud. Hay un gran número de ficheros KML disponibles a través de Internet.

La función **google.maps.KmlLayer** lee un «*feed*» RSS con formato KML especificado mediante una dirección HTTP. Por lo general, los mapas hechos de esa manera se muestran muy rápidamente. La desventaja es que hay menos control sobre el aspecto del mapa real porque los símbolos están definidos en el fichero KML.

El ejemplo de la figura 14.11 muestra una aplicación de un canal RSS para la representación de terremotos.

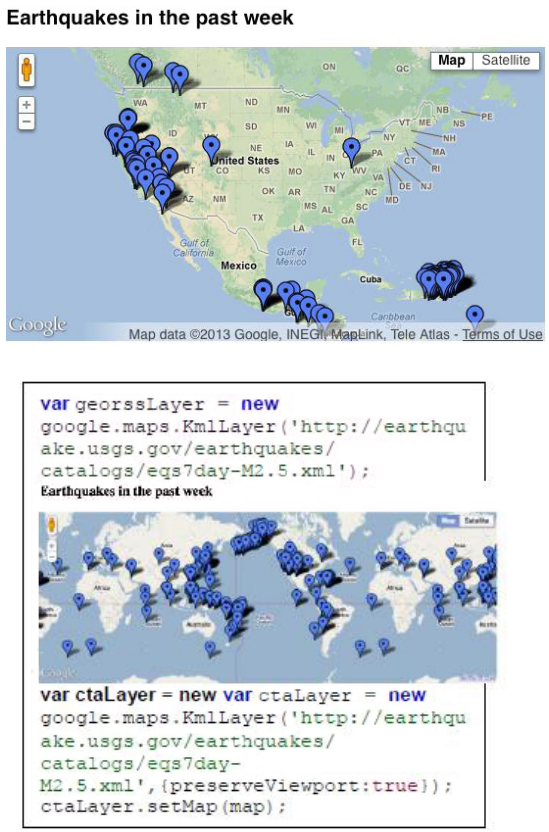


Figura 14.11. Feed RSS definido en formato KML del United States Geological Survey. Cada icono señala el terremoto y, al hacer clic, describe el evento. El mapa de abajo se muestra usando una llamada básica a una capa KML que ignora el centro y el nivel de zoom definidos y, por tanto, duplica la mayor parte del mundo. El mapa de la parte superior se muestra con la opción {preserveViewport: true} y ctaLayer que aplica el centro y el nivel de zoom definidos por el usuario. (© 2014 Google).

Este *feed* KML en particular, se actualiza diariamente y muestra los terremotos en los últimos siete días. Se puede hacer clic en cada marcador y se proporciona más información sobre el terremoto. El código muestra cómo hacer un mapa solo para una parte del mundo.

14.4.2 Líneas

La función Polilínea de los mapas de Google Maps se utiliza para dibujar líneas con la API de Google Maps. En la figura 14.12, la función Polilínea conecta los puntos que se han definido anteriormente. Las opciones para controlar la apariencia de la línea incluyen strokeColor, strokeOpacity, y strokeWeight. Como siempre, también se deben definir un centro y un nivel de zoom adecuados. El centro podría ser el punto medio de la propia línea.

Figura 14.12. Una línea a partir de tres elementos consistentes en cuatro puntos. (© 2014 Google).

La línea geodesica es la distancia más corta entre dos puntos y en un mapa no es una línea recta porque el mapa ha sido proyectado desde la Tierra, que es esférica. En la mayoría de las proyecciones, incluyendo la Mercator que es la utilizada por todos los servicios importantes de mapas en línea, la distancia más corta en el terreno se representa como una curva –aparentemente una distancia más larga entre dos ubicaciones–. Un círculo máximo se define como la distancia más corta entre dos puntos que divide la Tierra en dos hemisferios iguales. Es compatible con la API de Google Maps a través de la opción polilínea *geodesic : true* (véase la figura 14.13).

```
var flightPlanCoordinates = [
new google.maps.LatLng(37.772323, -
122.214897),
new google.maps.LatLng(21.291982, -
157.821856),
new google.maps.LatLng(-18.142599, 178.431),
new google.maps.LatLng(-27.46758, 153.027892)
];
var flightPath = new google.maps.Polyline({
path: flightPlanCoordinates,
strokeColor: "#FF0000",
strokeOpacity: 1.0,
strokeWeight: 3
```

*Figura 14.13. Opción polilínea **geodesic : true** que conecta dos puntos a través de un círculo máximo, la distancia más corta entre dos puntos de la esfera. Aparece como una línea más larga en este mapa, debido a la proyección. (© 2014 Google).*

14.4.3 Áreas

Un polígono puede ser visto como una línea que se cierra sobre sí misma. Se compone de una serie de puntos, de los que el último es siempre el mismo que el primero. Los dos atributos adicionales que necesitan ser definidos para **google.maps.Polygon** son el sombreado y la opacidad del área interior.

La figura 14.14 muestra el triángulo de las Bermudas en el océano Atlántico. Se definen cuatro puntos para representar el triángulo. Estos puntos se cargan en una matriz denominada **triangleCoords**. Esta matriz se pasa entonces a **google.maps.Polygon**. Los parámetros incluyen **strokeColor, strokeOpacity, strokeWeight, fillColor** y **fillOpacity.**

```
<script type="text/javascript">
function initialize() {
var myLatLng=new
google.maps.LatLng(24.886436490787712,-
70.2685546875);
var myOptions = {
zoom: 5,
center: myLatLng,
mapTypeId: google.maps.MapTypeId.TERRAIN
};
var map = new
google.maps.Map(document.getElementById("map_c
anvas"),
myOptions);
var triangleCoords = {
new google.maps.LatLng(25.774252, -80.190262),
new google.maps.LatLng(18.466465, -66.118292),
new google.maps.LatLng(32.321384, -64.75737),
new google.maps.LatLng(25.774252, -80.190262)
};
var bermudaTriangle = new
google.maps.Polygon({
paths: triangleCoords,
strokeColor: "#FF0000",
strokeOpacity: 0.8,
strokeWeight: 2,
fillColor: "#FF0000",
fillOpacity: 0.35
});
bermudaTriangle.setMap(map);
}
</script>
```

Figura 14.14. La función Google Polygon dibuja una forma cerrada. Las opciones incluyen strokeColor, strokeOpacity, strokeWeight, fillColor y fillOpacity. (© 2014 Google).

14.4.4 Capas

Hasta el momento, tenemos puntos, líneas y áreas superpuestas que han sido definidos como vectores de latitud y longitud. Ahora, superponemos una imagen ráster, una imagen que podría ser una fotografía aérea, imagen de satélite o un mapa escaneado. La ventaja de la superposición de una imagen es que la superposición se puede hacer rápidamente. No hace falta realizar transformaciones importantes o dibujos para colocar la información, porque el mapa subyacente está en el mismo formato. Lo ficheros de mapa de bits se pueden superponer como entidades únicas o divididas, para que coincidan con precisión con el mapa subyacente.

El ejemplo de la figura 14.15 muestra un mapa que ha sido escaneado y guardado en formato JPEG. Se han estimado la latitud y la longitud de las esquinas suroeste y noreste, y luego se han definido utilizando **imageBounds**. Esas coordenadas se combinan con la dirección de la imagen en el objeto **oldmap**.

14.5 Cartografía a partir de móviles

Los dispositivos que informan de la localización (o ubicación) son cada vez más comunes. Ahora, prácticamente todos los teléfonos móviles pueden ser localizados dentro de un radio de unos pocos metros. Los teléfonos inteligentes tienen la capacidad adicional de mostrar su ubicación real en un mapa. Los dispositivos de tipo «tablet» basados en el **iOS** de Apple o en el **Android** de Google, pueden, por lo general, hacer lo mismo pero con el beneficio añadido de mostrar una imagen mucho más grande.

Hay muchos tipos diferentes de dispositivos móviles y muchas formas diferentes de determinar su ubicación. Para proporcionar un enfoque estandarizado, el World Wide Web Consortium (W3C) ha creado una API de geolocalización de acceso libre. Con el apoyo de casi

```
function initialize() {
var newark = new google.maps.LatLng (40.740,-
74.18);
var imageBounds = new
google.maps.LatLngBounds(
new google.maps.LatLng (40.712216,-74.22655),
new google.maps.LatLng (40.773941,-74.12544));
var myOptions = {
zoom: 12,
center: newark,
mapTypeId: google.maps.MapTypeId.ROADMAP
}
var map=new
google.maps.Map(document.getElementById("map_c
anvas"),myOptions);
var oldmap = new google.maps.GroundOverlay(
"http://www.lib.utexas.edu/maps/historical/newark_nj_192
2.jpg",
imageBounds);
oldmap.setMap(map);
}
```

Figura 14.15. Superposición de un mapa escaneado en formato JPEG. Mapa de Newark, NJ, cortesía de las bibliotecas de la Universidad de Texas, Universidad de Texas en Austin. (© 2014 Google).

todos los navegadores, la API utiliza varios métodos para encontrar la ubicación del ordenador o del dispositivo móvil (Svennerberg 2010, P. 235).

El Sistema de Posicionamiento Global (GPS) es un método para determinar la ubicación, pero solo funciona con una vista despejada del cielo. En las zonas urbanas, el método más común para determinar la ubicación se basa en la triangulación wifi y en las señales de los repetidores de telefonía. El *software* de localización desarrollado por la empresa Skyhook, con sede en Boston, utiliza una red masiva de referencia,

```
if(navigator.geolocation) {
browserSupportFlag = true;
navigator.geolocation.getCurrentPosition(funct
ion(position) {
initialLocation = new google.maps.LatLng
(position.coords.latitude,position.coords.longitude);
contentString = "Pt:
"+position.coords.latitude+",
"+position.coords.longitude ;
map.setCenter(initialLocation);
infowindow.setContent(contentString);
infowindow.setPosition(initialLocation);
var marker = new google.maps.Marker({
position: initialLocation,
map: map,
title:"Hello World!"
});
google.maps.event.addListener(marker, 'click',
function() {
infowindow.open(map,marker);
});
}, function() {
handleNoGeolocation(browserSupportFlag);
});
}
}
```

Figura 14.16. Ejemplo de texto. En la burbuja emergente, en la que se puede hacer clic, aparecen la latitud y longitud del dispositivo móvil. El «+» de la cuenta contentString se utiliza para concatenar los números en una cadena de texto. (© 2014 Google).

compuesta por los lugares conocidos de más de 250 millones de puntos de acceso wifi y repetidores de telefonía. Para desarrollar la base de datos, *Skyhook* desplegó una serie de operadores para que inspeccionasen cada calle, carretera o callejón en decenas de miles de ciudades y pueblos en todo el mundo, explorando puntos de acceso wifi y repetidores de telefonía, anotando sus ubicaciones geográficas precisas.

La cartografía de la ubicación de un dispositivo a través de un navegador utilizando la API W3C se muestra en la figura 14.16. La declaración

«**navigator.geolocation.getCurrentPosition (function(position)**»

resuelve la posición del dispositivo. Si la posición no puede ser determinada por GPS, la API utiliza el método de triangulación basado en una red inalámbrica. En este ejemplo se presenta una ventana de información (*infowindow*) en la ubicación actual.

El ejemplo de la figura 14.16 sustituye a la ventana de información con un marcador seleccionable mediante un clic.

El contentString del texto de información que aparece en una burbuja emergente, muestra la latitud y longitud.

14.6 Conclusiones

Vivimos en tiempos increíbles para la cartografía. En cuestión de 20 años, desde 1970 hasta la década de 1990, los mapas han pasado de ser objetos estáticos en papel a convertirse en presentaciones interactivas que se pueden descargar a través de una red informática. En los años transcurridos desde entonces, los mapas se han vuelto aún más interactivos, en el sentido de ser capaces de añadir información –tanto información temática a través de «*mash-ups*» (mezclas de capas), como por la edición del mapa de base subyacente–. Los ejercicios de este capítulo suponen una introducción al nuevo mundo de la cartografía a través de Internet. Las herramientas que se han presentado aquí pueden utilizarse para hacer mapas muy sofisticados.

Referencias

Google Maps JavaScript API V3 Basics (2011). (buscar: Google Maps JavaScript API V3 Basics).

Neumann, A., Winter A. M. (2003). «Web-mapping with Scalable Vector Graphics (SVG): Delivering the promise of high quality and interactive web maps.» En: Peterson, M. P. (ed.) *Maps y el Internet*. Elsevier, Amsterdam, pp.197–220.

Peterson MP (2008) *International Perspectives on Maps and the Internet*. Springer, Berlin.

Svennerberg, Gabriel (2010). *Beginning Google Maps API 3*. New York, NY: Apress.

W3Schools.com (2011). JavaScript Tutorial. (http://www.w3schools.com/js/default.asp). (buscar: Learning JavaScript).

Willard, Wendy (2009). HTML: *A Beginner's Guide*. Berkeley, CA: Osborne/McGraw-Hill.

Nota: El material de este capítulo se basa en el libro del autor titulado «Mapping in the Cloud», publicado por Guilford Press.