

14 Mapeamento Web e Mobile

Michael P. Peterson, EUA

Resumo

Mapear sempre dependeu de um conjunto de ferramentas, tanto para medir o mundo tanto para criar o mapa. Desde 2005, um determinado conjunto de ferramentas em Nuvem (Cloud) é desenvolvida para manipulação de mapas – chamadas de Interface de Programação de Aplicações ou Application Programming Interfaces (API) em inglês. Essas ferramentas possuem o benefício de ser automaticamente disponível a qualquer pessoa com conexão a Internet. Esse capítulo examina essa nova era no mapeamento e mostra como construir e distribuir mapas usando ferramentas de mapeamento gratuitas.

14.1 Introdução

É difícil sobrestimar a importância dos mapas como uma forma de comunicação acerca do mundo. Eles nos ajudam a entender tanto o nosso entorno como também o espaço que está além de nossa percepção direta. Os mapas influenciam o nosso pensar sobre o mundo e como nós agimos nele – eles nos conectam ao nosso ambiente. Cada um de nós é um cartógrafo, no sentido que todos nós criamos mapas mentais. Por exemplo, algumas vezes desenhamos esses mapas para alguém a fins de explicar como chegar a um local específico.

A criação de mapas e a análise da informação subjacente evoluíram para uma ciência, sendo habilidades valiosas para diferentes tipos de trabalho. Muitas dessas ferramentas de mapeamento são hoje hospedadas na Nuvem – um sofisticado sistema de hardware e software

acessível por meio da Internet. Essas ferramentas baseadas na Nuvem permitem formas avançadas de mapeamento, onde os mapas online construídos são facilmente disponibilizados a terceiros.



Order Free Web Hosting

I want to host my own domain (domain must be registered already)

www.

or, I will choose your free subdomain (recommended)

www. .hostei.com

Your name

Your email (account details will be sent there)

Password (at least 6 symbols, both letters and numbers)

Type password again

Figura 14.1a mostra à esquerda o painel da página de inscrição para <http://000webhost.com>. A informação da conta é exibida à direita. Um pagamento é necessário para a especificação de um nome de domínio como <http://www.peterson.com>, porém é gratuita à escolha de um subdomínio como <http://geographyprof.hostei.com>. (© 2014 First Class Web Hosting)

Criar mapas dentro da Nuvem requer um servidor. Apesar de ser possível transformar praticamente qualquer computador em um servidor, é mais simples e mais seguro utilizar um serviço de hospedagem online. Os serviços de hospedagem em Nuvem como o Amazon Web Services e o Microsoft Azure implementam escalabilidade no sentido que eles podem ser expandidos conforme a demanda a fins

servir até milhões de usuários simultaneamente. A cobrança desses serviços é baseada na utilização do espaço de armazenamento e utilização dos ciclos de processamento do 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

Figure 14.1b ilustra as informações da conta do serviço.

Outra opção de hospedagem para páginas web é os serviços que cobram somente pelo armazenamento de disco. Embora esses serviços não sejam escaláveis, eles são de fácil utilização e oferecem planos, por exemplo, com subdomínios gratuitos com 1.500 MB de espaço em disco. Duas dessas hospedagens gratuitas são o 000webhost.com e podserver.info. As seções seguintes descrevem como criar uma página e servir mapas utilizando um desses serviços de hospedagem gratuitos.

14.2 Servidores na Nuvem

14.2.1 Criando um espaço na Nuvem

A Figura 14.1 ilustra a página de inscrição para o provedor 000webhost.com e as informações resultantes da conta criada. Observe que um subdomínio está sendo requisitado sob o domínio hostei.com. Apesar de um endereço como <http://geographyprof.com> pode ser configurado na linha superior da tela, ele incorreria em uma cobrança uma vez que representa um novo domínio. Para obter uma conta gratuita, é necessário utilizar um subdomínio do hostei.com.

Ressalta-se que um endereço de e-mail é necessário para registrar o subdomínio gratuito. A página Account Information relata o endereço de Web <http://geographyprof.hostei.com> (ou <http://64.120.177.162>) fornecido pelo provedor de serviço junto com 1.500 MB de armazenamento gratuito. Essa página também informa a disponibilidade do Apache Web Server e de outras ferramentas online incluindo o PHP e o MySQL.

Uma interface gráfica é disponibilizada pela a maioria dos serviços de hospedagem de páginas, sendo normalmente chamadas de painel de controle, ou cPanel (veja Figura 14.2). Tais ferramentas manuseiam e-mails, edição de arquivos, agendamento de tarefas e gerenciamento da conta. Todas essas ferramentas representam projetos de código aberto que são escritos e mantidos por uma pequena legião de programadores. O File Manager (gerenciador de arquivos) é a ferramenta mais utilizada para gerenciar arquivos e construir a página de Internet. Já o MySQL e o phpMyAdmin são utilizados para construir bancos de dados. A maioria dos serviços de hospedagem de sites utilizam ferramentas simulares ao cPainel para acessar os recursos do servidor.

A Figura 14.3 apresenta a janela do File Manager com acesso a ferramentas para carregar e criar novos arquivos e diretórios (subpastas). Essas ferramentas também permitem mover, apagar e renomear arquivos. A listagem dos arquivos exibe o nome, tipo e tamanho dos mesmos, enquanto os campos Owner (proprietário), Group (grupo) e Perms (permissões) dizem respeito às configurações de segurança. Já o Mod Time indica quando o arquivo foi modificado pela última vez. Os arquivos podem ser editados diretamente nesta janela com o clique em “Edit” ao final de cada nome de arquivo.

A pasta public_html é o diretório onde todos os arquivos são servidos na Web, ou seja, se um arquivo HTML deve ser apresentado por meio da página Web, ele deve residir nesse diretório. Usualmente, esse diretório contém um arquivo chamado index.htm (ou index.php) que é a primeira página a ser acessada quando o site é endereçado. Por exemplo, se um endereço como <http://geographyprof.hostei.com/CloudMapping/> é digitado no navegador, o servidor, em geral, irá fornecer o arquivo chamado index.htm dentro do diretório (pasta) chamado CloudMapping que está, por sua vez, localizada no diretório public_html. Isso significa que os dois endereços a seguir vão exibir o mesmo arquivo:

http://geographyprof.hostei.com/Online_Mapping/ e http://geographyprof.hostei.com/Online_Mapping/index.htm

Normalmente esse arquivo index.htm serve o ponto de entrada à página da Internet com ligações a outros arquivos deste diretório.



Figura 14.2 ilustra um painel de controle padrão de hospedagem Web, chamado cPanel, que fornece acesso a diferentes ferramentas. O File Manager (gerenciador de arquivos) é o principal programa para carregar e editar arquivos. (© 2014 First Class Web Hosting)

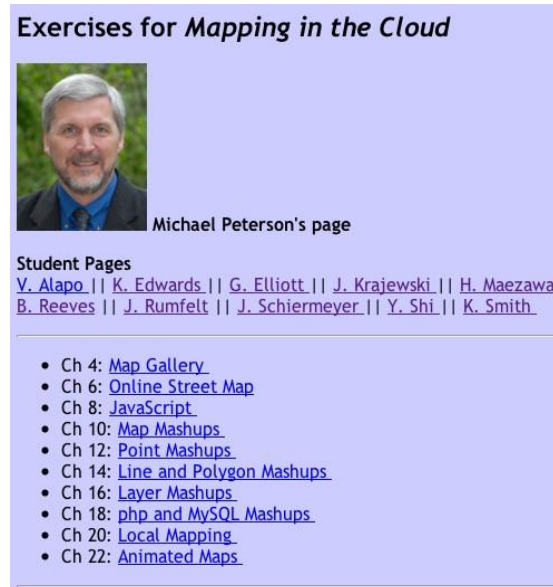


Figura 14.3 mostra a janela do File Manager de um serviço de hospedagem na Internet. Esse serviço permite que arquivos sejam criados e editados. Geralmente os arquivos a serem servidos por meio da Internet estão na pasta **public_html**. (© 2014 First Class Web Hosting)

O arquivo index.htm possui uma estrutura relativamente simples – uma linha de título seguido pelas ligações (links) para seus conteúdos internos. Esse arquivo poderia ter uma imagem do responsável da página e link para página de outros estudantes, como mostra a Figura 14.4. Para inserir uma figura devemos inserir a seguinte tag:

```
<img src=filename>
```

As ligações às páginas dos estudantes são separadas por duas linhas verticais (“|”). O código de todo o arquivo index pode ser obtido procurando por “Peterson Mapping in the Cloud”.



```
<html>
<head></head>
<body bgcolor="#CCCCFF">
<h2> Exercises for Mapping in the
Cloud</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 exibe um exemplo de arquivo index.htm que inclui uma figura, ligações (links) para todos os estudantes e aos capítulos.

14.2.2 HTML

HTML é o bloco de construção da Web. Ele é a linguagem que torna possível apresentar informação por meio de páginas Web, sendo também um recipiente para linguagens de script como Javascript e PHP. Algum conhecimento de HTML é necessário para apresentar mapas por meio da Internet.

A linguagem HTML consiste de tags que definem o layout da página (Willard 2009). São incluídos nele códigos de texto simples, cercados de delimitadores “<” e “>”, que especificam como o documento irá ser exibido no navegador. Outras tags HTML também criam ligações a documentos ou exibem um arquivo gráfico. Um editor de texto comum pode ser utilizado para criar e editar um arquivo HTML (Notepad no Windows ou TextEdit no Macintosh, utilizando as configurações apropriadas). Ao contrário dos processadores de texto, esses citados programas são destinados para entrada de texto sem formatação. Uma vez que esses arquivos são criados, eles podem ser abertos com navegadores como Explorer, Firefox ou Chrome.

Todos os arquivos HTML começam com a tag `<html>` e terminam com esse mesmo marcador antecedido com uma barra `</html>`, indicando que o código HTML está terminado. Tecnicamente todos as tags HTML possuem um início e um fim, com a terminação indicada com `“/.”` Por exemplo, o comando `<h1>` é utilizado para iniciar um texto de cabeçalho – texto grande utilizado em títulos – e o código `</h1>` para interromper a formatação do texto de cabeçalho.

Nós podemos exibir um mapa com a tag `img`. Todos os nomes de arquivos referenciados com `img` devem ser de extensão GIF, JPG/JPEG ou PNG uma vez que estes são os tipos de arquivos usuais aos diferentes navegadores. A opção `src` da tag `img` é utilizada para especificar o endereço URL do arquivo. No exemplo abaixo, observe que a designação da imagem é `.gif` ao final da URL. O comando `img` também possui um conjunto de opções que pode ser utilizada para alterar o tamanho da imagem ou mudar o seu posicionamento na página. Como a tag `hr`, a `img` não possui a usual tag de terminação. Na estrita formatação XHTML, o código é escrito como:

```
<img src=map.png />
```

A tag `<embed>` é utilizada para exibir arquivos gráficos que não são GIF, JPEG, ou formatos usuais de imagem. Os exemplos incluem Adobe PDF, Flash, SVG e QuickTime. Seu formato é idêntico às opções do `img`:

```
<embed src="http://maps.unomaha
.edu/Cloud Mapping/Chapter4/MapExample4.pdf"
width="500" height="389">
```

Uma das tags mais utilizada em HTML é a âncora ou ligação ou link, que usualmente é utilizada com um texto para criar um hipertexto. Podemos também utilizar essa tag junto com a tag `img` para criar uma hiperimagem. O formato dessa tag âncora segue o exemplo:

```
<a href="http://maps.unomaha
.edu/Cloud Mapping/Chapter4/MapExample4.pdf">
Click for PDF file</a>
```

14.2.3 JavaScript

Por si só, HTML é uma simples linguagem para formatação de páginas. Porém em combinação com JavaScript, uma página HTML pode executar códigos computacionais (W3Schools.com 2011). Uma vez relegada aos geeks, a programação hoje é percebida como uma forma de expressão, a “amplificação do pensamento”, e uma habilidade necessária. O conceito de “alfabetização à codificação” promove novas ideias sobre a importância de ser capaz de programar. Páginas online de programação como o CodeAcademy estão se tornando cada vez mais populares. Já a Khan Academy introduziu um conjunto de exercícios online gratuitos para programação. Os seguintes exemplos compõem uma breve introdução ao JavaScript e como ele pode ser utilizado para chamar outras funções.

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

Figura 14.5 ilustra um cálculo executado com JavaScript dentro do corpo de um arquivo HTML.

As funções são fundamentais para a estrutura dos códigos em JavaScript. Uma função é um procedimento, ou seja, um conjunto de comandos para executar uma tarefa específica. Essas funções geralmente são definidas na seção de cabeçalho (head) do documento HTML, pois assim fica garantido que elas estejam definidas antes que seja exibido qualquer conteúdo da página. O exemplo na Figura 14.6 define uma simples função na seção de cabeçalho de um documento HTML. Observe que a função é chamada no corpo do documento.

A função **square** (quadrado, em português) recebe um argumento chamado *number*. A função executa apenas um comando: **return** number * number, indicando que a função retorna o argumento multiplicado por ele mesmo. Ou seja, o comando **return** determina o valor retornado pela função JavaScript.

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 apresenta o código de uma função que retorna o quadrado do número passado como argumento na chamada da função.

Um arquivo externo pode conter funções na linguagem JavaScript, sendo que este arquivo pode estar no mesmo computador que o arquivo HTML, como implementado na Figura 14.7, ou pode se encontrar em outro computador ou servidor. Essa última maneira é como um código de Application Programming Interface (API) – ou Interface de Programação de Aplicações, em português – é geralmente distribuído. Uma referência a uma biblioteca de código de uma API torna possível que um criador de página Web tenha acesso a milhares de funções.

Em outras palavras, ao invés de inserir o código JavaScript diretamente no arquivo HTML, seja no corpo ou no cabeçalho, é possível também colocar as funções JavaScript em um arquivo separado. O atributo SRC da tag <script> especifica o arquivo externo onde o código JavaScript é encontrado. A Figura 14.8 apresenta o arquivo externo chamado *common.js* e como ele é referenciado dentro da seção cabeçalho de um documento HTML. O arquivo

externo JavaScript pode conter múltiplas funções, porém não pode ter código HTML.

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

Figura 14.7 apresenta a função que está localizada no documento externo *common.js*, sendo referenciado no código HTML com a tag <script src="common.js">.

A Figura 14.8 exemplifica como o código do Google Maps API é referenciado. Para auxiliar a depuração do código, o código do Google Maps API pode ser executado no seu computador, ou seja, sem a necessidade de transferi-lo a um servidor. Entretanto, para que terceiros acessem o mapa, o código precisa estar hospedado em um 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 mostra a implementação de uma chamada ao Google Maps API, que dá acesso um vasto número de funções relacionadas a mapas digitais. Aqui, o parâmetro sensor é associado com o valor "false". O valor desse parâmetro deve ser associado como "true" se utilizamos um dispositivo móvel que pode fornecer a sua posição atual.

14.3 Google Maps API

Lançado um pouco depois do Google Maps em 2005, o Google Maps Application Programming Interface consiste em uma série de funções que controla a aparência do mapa, incluindo escala e localização, e qualquer informação adicional na forma de pontos, linhas ou áreas e suas descrições associadas. O uso do Google Maps API é essencialmente gratuito, porém colocando uma limitação no número de mapas que podem ser servidos. Um site não pode gerar mais do que 25.000 carregamentos de mapa por dia durante 90 dias consecutivos, sendo um carregamento uma exibição de mapa com a API do Google Maps. Uma vez carregado, o grau de interação do usuário com o mapa não impacta neste cálculo. Seria muito difícil um usuário mediano do Google Maps API exceder o limite dos 25.000 carregamentos, pois mesmo que o site tivesse muitos acessos, seria necessário exceder tal limite por 90 dias consecutivos. Todavia, se o site exceder, o Google demandará um registro e o pagamento de US\$ 0,50 para cada 1.000 visualizações de mapas além do

limite. Por exemplo, se sua página utiliza o Google Maps API para servir 100.000 mapas por dia em 90 dias consecutivos, você será cobrado em US\$ 37.50 ($75.000 \div 1.000 \times 0,5$) um mês depois do período de 90 dias.

O serviço web Specialized Google Maps API possui limites adicionais, incluindo:

- Direções – prove direções no formato texto – limitado a 2.500 por dia;
- Matriz de Distância – retorna a distância e tempo de viagem – limitado a 100 elementos por consulta e 2.500 por dia;
- Elevação – elevação em pontos – limitado a 2.500 requerimentos por dia, onde cada requerimento retorna até 512 elevações;
- Geocodificação – converte um endereço em latitude e longitude – limitado a 2.500 por dia;
- Lugares – retorna estabelecimentos comerciais e outros pontos de interesse próximos a um ponto – requer uma chave de API e é limitado a 1.000 requisições por dia.

A chave da API do Google Maps é um identificador numérico que registra seu site junto ao Google, sendo que ele não é necessário para aplicações normais; é necessário somente se o limite do pacote gratuito é excedido ou o serviço Lugares (Places) é utilizado.

O exemplo na Figura 14.9 apresenta um código em JavaScript. A API do Google Maps é chamada para exibir um mapa simples que está centralizado em uma localização determinada. O nível de zoom, que compreende uma faixa

numérica entre 0 e 21, está configurada para 15 em **myOptions**. A centralização é definida com um específico valor de latitude e longitude, e a opção **ROADMAP** é selecionada para definir o estilo do mapa. Todas as chamadas de API estão realizadas na função **initialize**. Por sua vez, essa função é chamada dentro de **onload**, que está dentro da tag **body** do arquivo HTML.

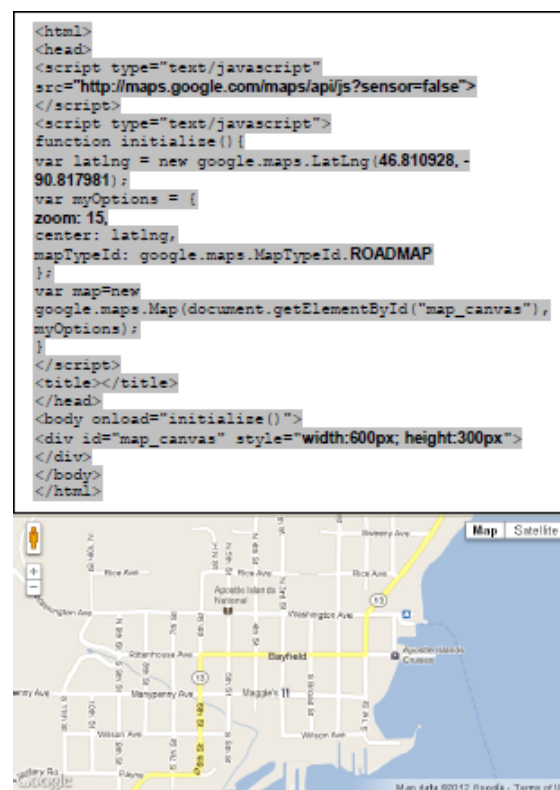


Figura 14.9 apresenta uma requisição básica utilizando o API do Google Maps. O centro do mapa pode ser alterado, juntamente com o nível de zoom e o tipo de mapa. (© 2014 Google)

Uma simples mudança pode ser realizada no código para substituir os valores de latitude e longitude. É possível determinar a latitude e longitude de uma coordenada específica por diferentes formas, a citar algumas:

- No Google Maps, clique com o botão direito (tecla control + clique no Mac) e selecione a opção “What’s here?”. As coordenadas do ponto serão exibidas em uma linha superior da janela do Google Maps;
- Clique com o botão direito do mouse na janela do MapQuest a fim que os valores das coordenadas apareçam em uma janela;
- No Bing Maps, a latitude e longitude são exibidos com um clique com o botão direito do mouse;
- Para exibir as coordenadas no formato de graus decimais com o Google Earth, selecione a opção Tools/Options no menu suspenso e selecione a opção decimal degrees;
- Por último, a Internet possui um conjunto de sites com ferramentas para encontrar latitudes e longitudes. A maioria desses sites utiliza o Google Maps, incluindo o exemplo: <http://findlatitudeandlongitude.com>

Outra mudança que podemos aplicar é o tipo de mapa ou estilo que é exibido. O Google oferece quatro tipos de visualizações:

- **MapTypeId.ROADMAP** apresenta a visualização padrão de ruas;
- **MapTypeId.SATELLITE** apresenta as imagens de satélite do Google Earth;
- **MapTypeId.HYBRID** exibe uma mistura entre as visualizações padrão e satélite;
- **MapTypeId.TERRAIN** apresenta um mapa físico baseado em informação de terreno.

O nível de zoom inicial também pode ser alterado. O valor “0” designa um zoom afastado, um mapa em pequena escala. Conforme esse valor aumenta, então também aumenta a escala do mapa. Geralmente, 20 níveis de detalhes estão disponíveis. Algumas partes do mundo tem mais do que 20 níveis de zoom.

14.4 Combinando ponto, linha, área e camada

Todos os mapas são compostos por pontos, linhas e áreas. Em adição, múltiplos mapas podem ser combinados como camadas individuais, uma função que está na base dos sistemas de informação geográfica. Os exemplos nessa seção demonstram como esses elementos podem ser adicionados ao Google Maps.

14.4.1 Pontos

O marcador padrão do Google é uma gota de cabeça para baixo, porém um grande número de ícones alternativos está disponível, ainda sendo possível criar símbolos personalizados porque eles são simplesmente imagens de 32 por 32 pixels no formato PNG. Marcadores podem ser estáticos ou interativos, sendo a principal interatividade por meio de cliques.

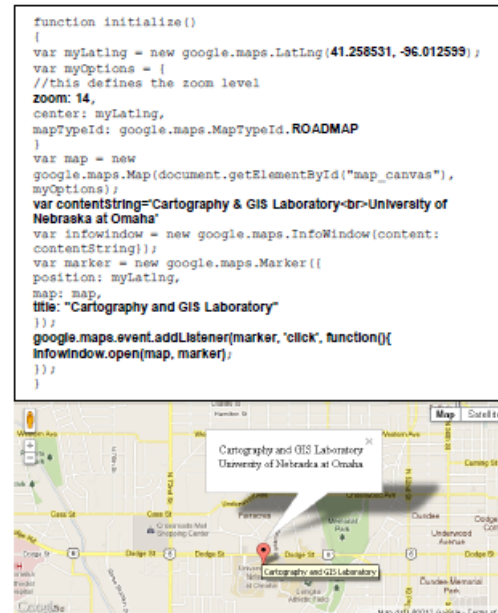


Figura 14.10 apresenta um exemplo de marcador que pode ser clicado. A variável texto *contentString* é definido no HTML. (©2014 Google)

No exemplo da Figura 14.10, a variável *contentString* é definida como um texto formatado em HTML. Este é associado a variável *infoWindow*, que é subsequentemente associado com o *google.maps.event.addListener*. Quando o usuário clica sobre o marcador, o texto é exibido em uma janela pop-up. O código HTML para essa janela pode incorporar uma figura tão como um vídeo por meio das tags *img* ou *embed*.

No caso onde um grande número de pontos precisa ser mapeado, principalmente quando esses pontos são frequentemente atualizados, um formato web chamado Really Simple Syndication (RSS) é em muitas vezes utilizado. Há diversas vantagens em utilizar o RSS, onde seus publicadores servem seu conteúdo automaticamente, enquanto os consumidores possuem o benefício de receber periodicamente atualizações das informações mapeadas. A padronização do formato de arquivo publicado permite que a informação seja acessada por diferentes programas.

O formato KML (Keyhole Markup Language) descreve espaços de duas e três dimensões, sendo originalmente desenvolvido para o Google Earth. Todavia, ele hoje é oficialmente um padrão chamado OpenGIS® KML Encoding Standard (OGC KML) mantido pela Open Geospatial Consortium (OGC). Esse formato especifica feições como marcadores (placemarks), imagens, polígonos e modelos 3D. As localidades são sempre especificadas com latitudes e longitudes. Um grande número de arquivos KML está disponível por toda a Internet.

A função *google.maps.KmlLayer* lê um formato KML endereçado em um endereço HTTP. Os mapas criados desta forma são exibidos rapidamente, pois os símbolos do mapa são definidos dentro do arquivo KML.

O exemplo da Figura 14.11 exemplifica uma aplicação de um feed de RSS com os registros de terremotos. Esse KML em particular é atualizado diariamente e apresenta os registros de terremotos nos últimos sete dias. Cada marcador apresenta informações sobre esses eventos quando clicados. O código da Figura 14.11 apresenta como criar o mapa para uma determinada parte do mundo.

Earthquakes in the past week



```
var georssLayer = new
google.maps.KmlLayer('http://earthqu
ake.usgs.gov/earthquakes/
catalogs/eqs7day-M2.5.xml');
Earthquakes in the past week
```



```
var ctaLayer = new var ctaLayer = new
google.maps.KmlLayer('http://earthqu
ake.usgs.gov/earthquakes/
catalogs/eqs7day-
M2.5.xml', {preserveViewport:true});
ctaLayer.setMap(map);
```

Figura 14.11 apresenta o conteúdo de um feed de RSS no formato KML provido pelo Serviço Geológico Americano (USGS, em inglês). Cada ponto descreve um terremoto que, quando clicado, detalha o evento. O mapa inferior é apresentado utilizando uma chamada básica de uma camada KML que ignora o centro e o nível de zoom e, portanto, duplica a maior parte do mundo. O mapa superior é apresentado com a opção `{preserveViewport:true}`, aplicando então o zoom e centro de visualização definido pelo usuário. (© 2014 Google)

14.4.2 Linhas

A função de Polyline (polinha) é utilizada para desenhar linhas com a API do Google Maps. Na Figura 14.12, a função Polyline conecta pontos que foram previamente definidos. As opções para controle da aparência da linha incluem as propriedades `strokeColor`, `strokeOpacity` e `strokeWeight`. Como sempre, é necessário definir o nível de zoom e o centro de visualização, este último pode ser o ponto médio da linha.



Figura 14.12 apresenta uma linha criada a partir de quatro pontos. (© 2014 Google)

A melhor distância entre dois pontos em um mapa é possível, pois ele foi projetado a partir da Terra esférica. A maioria das projeções, incluindo a Mercator que é utilizada nos principais serviços de mapas online, o menor caminho sobre a superfície é representado como uma curva sobre o mapa – aparentemente uma distância longa entre os dois pontos quando vista sobre o mapa projetado. A dita linha ortodrômica contém a menor distância dentro dois pontos sobre a Terra, dividindo-a em dois hemisférios iguais. Esse recurso é suportado na API do Google Maps por meio da opção da polilinha `geodesic: true` (veja a 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 apresenta linha com a opção `geodesic: true` conectando dois pontos por meio da linha ortodrômica, definindo a menor distância entre duas localizações sobre a esfera. A linha aparenta ser uma curva longa devido à projeção cartográfica. (© 2014 Google)

14.4.3 Áreas

Um polígono pode ser entendido como uma linha com traçado fechado, sendo formada a partir de uma série de pontos onde o último é igual ao primeiro. Dois atributos adicionais que precisam ser definidos para `google.maps.Polygon` são o sombreamento e opacidade da área interior.


```

<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>

```

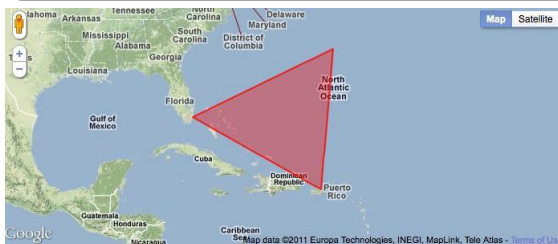


Figura 14.14 A função Polygon desenha uma geometria fechada. As opções incluem strokeColor, strokeOpacity, strokeWeight, fillColor e fillOpacity. (© 2014 Google)

A Figura 14.14 apresenta o Triângulo das Bermudas no Oceano Atlântico. Quatro pontos são definidos para descrever o triângulo. Esses pontos são carregados em um

conjunto chamado triangleCoords. Esse conjunto então é passado como parâmetro ao google.maps.Polygon, assim como strokeColor, strokeOpacity, stroke-Weight, fillColor e fillOpacity.

14.4.4 Camadas (Layers)

Até este ponto do texto, temos sobrepostos pontos, linhas e polígonos que foram definidos a partir de latitudes e longitudes. Agora, nós vamos sobrepor uma imagem como uma foto aérea, uma cena de satélite ou um mapa escaneado. Uma vantagem de sobrepor uma imagem ao mapa é que ela pode ser realizada rapidamente sem a necessidade de maior conversão ou desenho, uma vez que o mapa base também é uma imagem. Arquivos de imagem podem ser apresentados no mapa de forma inteira ou dividindo a imagem em blocos coincidentes com os blocos que dividem o mapa base.

O exemplo da Figura 14.15 apresenta um mapa que foi escaneado e salvo no formato JPEG. A latitude e longitude dos extremos sudoeste e nordeste foram definidos utilizando o imageBounds. Essas coordenadas foram combinadas com o endereço online para a imagem dentro do objeto oldmap.

```

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 mostra a camada de um mapa escaneado no formato JPEG. O Mapa de Newark, New Jersey (EUA), é uma cortesia da Universidade do Texas em Austin. (© 2014 Google)

14.5 Mapeamento Mobile

Os dispositivos que aferem a posição geográfica estão cada vez mais acessíveis. Praticamente todos os celulares hoje podem medir coordenadas com erro de poucos metros. Smartphones possuem a capacidade adicional de exibir sua posição atual sobre o mapa. Computadores do tipo tablet baseados no iOS da Apple ou o Android da Google também podem apresentar mapas, porém em telas maiores.

Há diferentes tipos de dispositivos móveis e diferentes formas para determinar a localização. Para alcançar uma abordagem padronizada, o World Wide Web Consortium (W3C) criou uma API de geolocalização que está disponível gratuitamente. Suportado por quase todos os navegadores, essa API utiliza diversos métodos para determinar a localização de um computador ou dispositivo móvel (Svennerberg 2010, P. 235).

O Sistema de Posicionamento Global (sigla GPS, em inglês) é um dos métodos para se determinar a localização, porém ele apenas funciona com uma visão desobstruída do céu. Em áreas urbanas, o método mais comum de determinar as localizações é utilizando técnicas de triangulação a partir de sinais de Wi-Fi e de torres de celular. O aplicativo de localização desenvolvido por uma empresa baseada em Boston, Skyhook, utiliza uma extensa rede de referência para determinar a localização de 250 milhões de pontos de acesso Wi-Fi e de torres de celular. Para desenvolver essa base, a Skyhook desenvolveu o monitoramento em cada rua, avenida e estrada em dezenas de milhares de cidade em todo o mundo, verificando a presença de pontos de acesso Wi-Fi e de torres de celular a fim de registrar suas coordenadas geográficas com precisão.

A determinação da posição atual de um dispositivo por meio de um navegador que utiliza a API da W3C é exemplificada na Figura 14.16. O comando `navigator.geolocation.getCurrentPosition(function(position))` computa a posição atual do dispositivo. Se a posição não puder ser determinada pelo GPS, então a API utiliza o método de triangulação baseado na rede sem fio. Nesse exemplo da Figura 14.16 é apresentado uma janela informativa com a posição corrente.



```
if(navigator.geolocation) {
  browserSupportFlag = true;
  navigator.geolocation.getCurrentPosition(function(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 apresenta o marcador que exibe a latitude e longitude do ponto corrente do dispositivo móvel. O “+” presente no código junto ao comando `contentString` é utilizado para concatenar números em um texto. (© 2014 Google)

14.6 Conclusões

Nós vivemos em tempos incríveis para o mapeamento. Em questão de 20 anos – de 1970 até a década de 1990 – os mapas saíram de objetos estáticos em papel e se tornaram interativos e acessados por meio de uma rede eletrônica. Nos anos seguintes, os mapas se tornaram ainda mais interativos, onde hoje somos capazes de adicionar informações temáticas e realizar edições sobre o mapa base. Os exercícios desse capítulo abrangem uma introdução ao novo mundo do mapeamento por meio da Internet. As ferramentas aqui introduzidas podem ser utilizadas para criar um sofisticado conjunto de mapas.

Referências

- Google Maps JavaScript API V3 Basics (2011). (busca: 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.” In: Peterson, M. P. (ed.) *Maps and the 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>]. (busca: Learning JavaScript).
- Willard, Wendy (2009). *HTML: A Beginner's Guide*. Berkeley, CA: Osborne/McGraw-Hill.

Nota: O material desse capítulo é baseado no livro *Mapping in the Cloud* deste autor e publicado pela Guilford Press.