

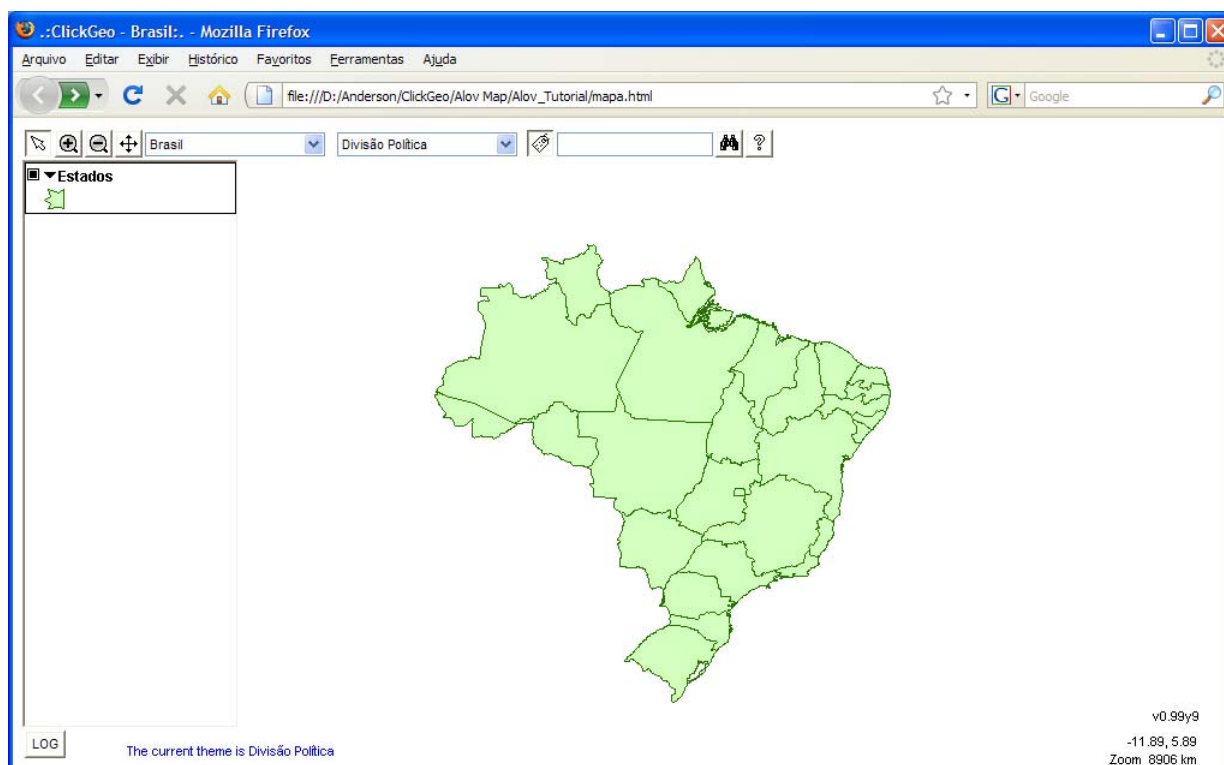
# CRIAÇÃO DE APLICAÇÃO WEBMAPPING COM O AUXILIO DO ALOV MAP

## INTRODUÇÃO

O **ALOV Map** é uma aplicação construída usando-se a linguagem Java que permite visualizar mapas tanto em formato vetorial do tipo *shapefile* (**\*.shp**) como também *raster* através da internet, com qualidade bastante satisfatória. Veremos nesse tutorial como montar uma aplicação básica com arquivos em formato vetorial.

## EXEMPLO PRÁTICO

Vamos criar uma aplicação que permitirá visualizar os atributos gráficos e alfanuméricos de um **shapefile** com a divisão política do Brasil. O resultado do que será mostrado nesse tutorial, quanto à visualização, será semelhante ao mostrado na figura 1.



**Fig. 1**

Antes de prosseguir com a explicação, podemos resumir o funcionamento de um *Webmapping* desse tipo, afirmando que ele faz uso de um arquivo **HTML**, um **XML** (Com as devidas configurações), o arquivo **shapefile** compactado e o **applet** do Alov – Todos estes devendo, preferivelmente, estar em um mesmo nível ou diretório.

## FAZENDO O DOWNLOAD DO ALOV MAP

O **Alov Map** é gratuito e você pode fazer o *download* na página oficial do Alov <<http://alov.org/>>. Procure a sessão de *downloads*, depois disso, faça seu cadastro e baixe a versão *applet*. (O Alov também possui uma versão *servlet*, que não será considerada neste tutorial).

## CRIAÇÃO DOS ARQUIVOS HTML E XML

Para criar os arquivos **.html** e **.xml** que serão usados para o funcionamento da aplicação, poderemos usar um dos diversos editores de código fontes, como por exemplo o SciTE (*software livre*), ou mesmo o bloco de notas.

Abaixo estão os códigos dos arquivos **XML** e **HTML**. Alguns trechos estão com formatação especial para facilitar a compreensão durante as explicações dos mesmos.

- **XML**

O código fonte usado para nosso exemplo foi:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<project zoomunits="km" mapunits="degrees">
<domain name="Brasil" startup="no" full="yes" xmin="-78.71" ymin="-35.94"
  xmax="-20.29" ymax="12.67"/>
<map name="Divisão Política" index="m1" />
  <layer name="Estados" visible="yes" showlegend="yes" zoom="50">
    <dataset url="shape.zip" />
    <renderer>
      <symbol fill="211:255:190" outline="38:115:0" />
    </renderer>
  </layer>
</project>
```

Entenda que a primeira linha é padrão para arquivos **XML** e diz a respeito de certas características peculiares do arquivo. Códigos da linguagem XML, como também da **HTML** (**XHTML**) são divididos no que chamamos de *tags*. A seguir, temos a explicação de algumas das *tags* usadas neste caso acima.

A *tag* <project ... </project> indica o início e o término do projeto do arquivo XML. Na maioria dos casos, para que seja indicado o fechamento da *tag* usa-se o sinal “</” ou “/>”.

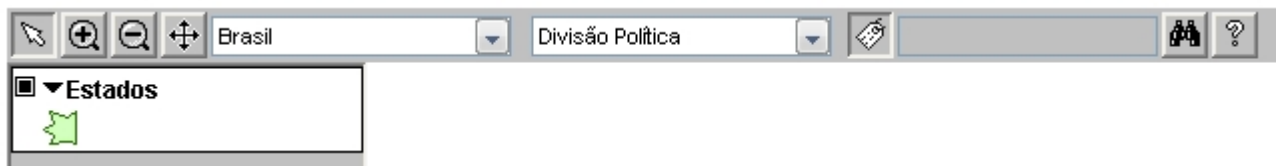
No trecho <domain name="Brasil" ... xmin="-78.71" ymin="-35.94" xmax="-20.29" ymax="12.67"/> é necessário entender que “domain name="Brasil” se refere ao domínio, ou seja a um campo que aparecerá na aplicação que será usado, por exemplo, para dar um zoom total no *layer* (Figura 2), já o restante: “xmin="-78.71" ymin="-35.94" xmax="-20.29" ymax="12.67” é a delimitação do retângulo envolvente, também chamado *bounds*,

que delimita a camada da informação. Perceba que nesse caso foram usadas coordenadas geográficas.

**OBS:** Vale lembrar que o retângulo envolvente é determinado pelas coordenadas do canto inferior esquerdo e superior direito que delimitam a área representada pelo *layer*.

A parte do código onde está escrito: `<map name="Divisão Política" index="m1" />` está relacionado com o nome do mapa a ser visualizado. Nesse nosso exemplo só temos um único mapa, ou nomeado como “Divisão Política” (Figura 2). Essa linha poderá exercer a função de controlar a exibição dos mapas temáticos, se existirem na aplicação, o que ainda não é o nosso caso.

O trecho `<layer ... </layer>` define as diversas camadas de informação. Em nosso exemplo, também só temos um único layer, chamado “Estados”. Note também que a *tag layer* traz a declaração: `name="Estados" visible="yes" showlegend="yes" zoom="50">`, que significa, respectivamente, que ao carregar a aplicação, o *layer* “Estados” estará visível, mostrando uma legenda na lateral da tela e ao vermos a tabela de atributos do *shapefile* teremos uma opção de zoom de índice 50 (Figura 2).



**Fig. 2**

Já a parte onde aparece: `<dataset url="shape.zip" />` indica o nome do arquivo compactado onde estão os *shapefiles*.

Por fim, o trecho `<renderer> <symbol fill="211:255:190" outline="38:115:0" /> </renderer>` indica características da visualização da camada de informação. Os números entre aspas e separados por dois pontos estão relacionados ao padrão RGB (Red, Green, Blue) de composição das cores. Por exemplo, neste código, a cor da linha de contorno das feições e o preenchimento dos polígonos são tonalidades do verde.

Para essa nossa primeira experiência com o Alov, o arquivo XML já está suficientemente bem configurado. Mas é de suma importância também prepararmos bem o arquivo HTML.

- **HTML**

O código fonte do arquivo **\*.html** usado para nosso exemplo foi:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>.:ClickGeo - Brasil:.</title>
</head>
  <body>
    <applet codebase = "." code="org.alov.viewer.SarApplet"
      archive="alov_applet.jar"
      width="980" height="530" align="center">
      <param name="pid" value="mapa.xml">
    </applet>
  </body>
</html>
```

As duas primeiras linhas dizem a respeito das características do tipo de documento **HTML**. A *tag* `<html ... </html>` se referem ao início e o encerramento do uso da linguagem **XHTML**. (Isso mesmo, a linguagem se chama **XHTML** e não **HTML**, embora a extensão do arquivo continue sendo **\*.html**)

No trecho copiado abaixo, colocamos o título da página, nesse caso escolhemos “.:ClickGeo - Brasil:.”. Perceba que colocamos isso dentro da *tag* **head**, que se refere ao cabeçalho.

```
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>.:ClickGeo - Brasil:.</title>
</head>
```

Dentro da *tag* **body** colocamos outra *tag* muito importante chamada **applet**. Segue a baixo o detalhamento do que está dentro da *tag* **applet**.

- `archive="alov_applet.jar"` – Faz referência ao applet do Alov, note a extensão **.jar** mostrando que se trata de uma aplicação montada em Java.
- `width="980" height="530" align="center">` - Nessa linha definimos as dimensões da janela na qual será exibida o *applet* do Alov.
- `<param name="pid" value="mapa.xml">` - Nessa parte fazemos referência ao arquivo XML criado anteriormente, neste caso “**mapa.xml**” (Que poderia ter sido nomeado aleatoriamente)

Assim terminamos de configurar o arquivo HTML. Abra o arquivo `mapa.html` com seu *browser* e veja como ficou aplicação Alov Map.

Você pode consultar a tabela de dados alfanuméricos clicando com o cursor sobre o mapa representado. Poderá encontrar exemplos de aplicações semelhante a essa na página **WebGis** do **ClickGeo**.

Em breve no site do **ClickGeo** ([www.clickgeo.com.br](http://www.clickgeo.com.br)) haverá outros tutoriais relacionados com o **Alov Map**, mostrando como aumentar as funcionalidades dessa poderosa ferramenta de publicação de mapas na Web.

Como sugestão extra: Confira um excelente exemplo de utilização do Alov Map, presente no site oficial do desenvolvedor, mostrando um Mapa Mundi e diversas informações: <<http://www.alov.org/sample/world.html>>

Espero que este tutorial como os demais já publicados possam servir para disseminação de conhecimento, o qual não deve ser escondido, mas compartilhado.

© *Anderson Medeiros* 2009