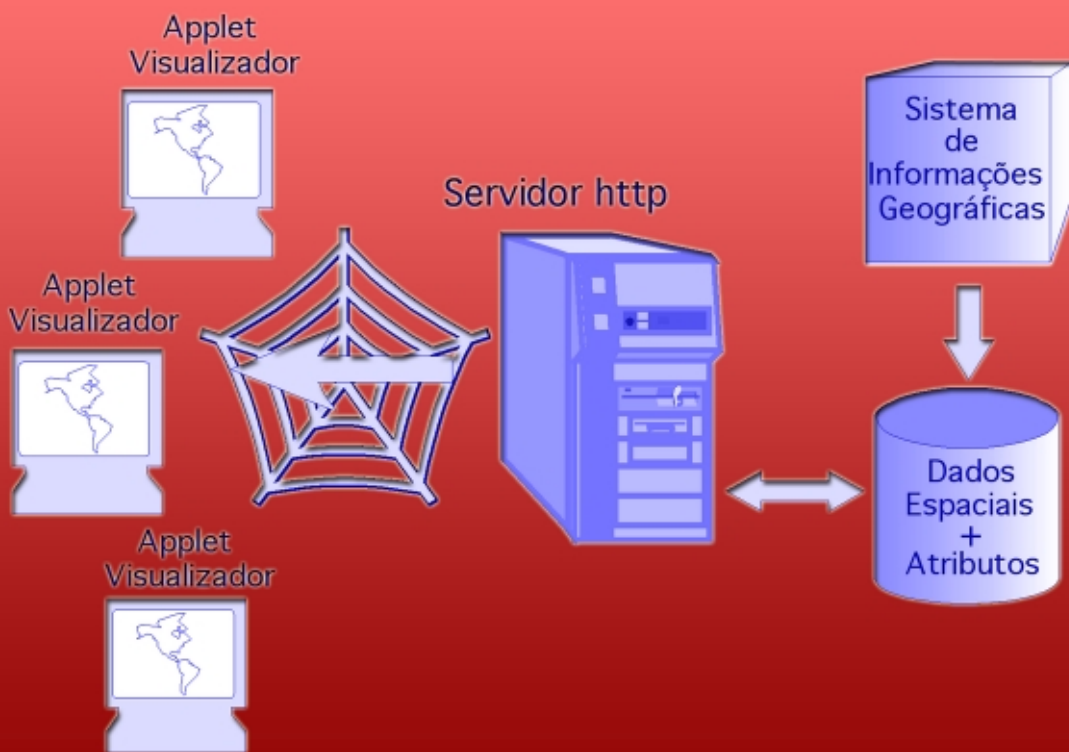


ISSN 1677-9274

Diretivas para Disponibilizar Mapas na Internet



República Federativa do Brasil

Fernando Henrique Cardoso
Presidente

Ministério da Agricultura, Pecuária e Abastecimento

Marcus Vinicius Pratini de Moraes
Ministro

Empresa Brasileira de Pesquisa Agropecuária - Embrapa

Conselho de Administração

Márcio Fortes de Almeida
Presidente

Alberto Duque Portugal
Vice-Presidente

Dietrich Gerhard Quast
José Honório Accarini
Sérgio Fausto
Urbano Campos Ribeiral
Membros

Diretoria Executiva da Embrapa

Alberto Duque Portugal
Diretor-Presidente

Bonifácio Hideyuki Nakasu
Dante Daniel Giacomelli Scolari
José Roberto Rodrigues Peres
Diretores-Executivos

Embrapa Informática Agropecuária

José Gilberto Jardine
Chefe-Geral

Tércia Zavaglia Torres
Chefe-Adjunto de Administração

Kleber Xavier Sampaio de Souza
Chefe-Adjunto de Pesquisa e Desenvolvimento

Álvaro Seixas Neto
Supervisor da Área de Comunicação e Negócios

Documentos 14

Diretivas para Disponibilizar Mapas na Internet

José Iguelmar Miranda

Embrapa Informática Agropecuária
Área de Comunicação e Negócios (ACN)

Av. André Tosello, 209
Cidade Universitária "Zeferino Vaz" – Barão Geraldo
Caixa Postal 6041
13083-970 – Campinas, SP
Telefone (19) 3789-5743 - Fax (19) 3289-9594
URL: <http://www.cnptia.embrapa.br>
e-mail: sac@cnptia.embrapa.br

Comitê de Publicações

Amarindo Fausto Soares
Ivanilde Dispato
José Ruy Porto de Carvalho (Presidente)
Luciana Alvim Santos Romani
Marcia Izabel Fugisawa Souza
Suzilei Almeida Carneiro

Suplentes

Adriana Delfino dos Santos
Fábio Cesar da Silva
João Francisco Gonçalves Antunes
Maria Angélica de Andrade Leite
Moacir Pedroso Júnior

Supervisor editorial: *Ivanilde Dispato*
Normalização bibliográfica: *Marcia Izabel Fugisawa Souza*
Capa: *Intermídia Produções Gráficas*
Edição eletrônica: *Intermídia Produções Gráficas*

1ª. edição

on-line - 2002

Todos os direitos reservados

Miranda, José Iguelmar.

Diretivas para disponibilizar mapas na Internet / José Iguelmar
Miranda. — Campinas : Embrapa Informática Agropecuária, 2002.
29 p. : il. — (Documentos / Embrapa Informática Agropecuária ; 14)

ISSN 1677-9274

1. Publicação de mapas. 2. Mapas na Internet. 3. Criação de mapas. I. Título. II. Série.

CDD – 21st ed.
005.720223

© Embrapa 2002

Autor

José Iguelmar Miranda

Ph.D. em Geoprocessamento, Pesquisador da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo 13083-970 – Campinas, SP.

Telefone (19) 3789-8767 - e-mail: miranda@cnptia.embrapa.br

Apresentação

Quando a disseminação de informações começou na Internet, todo o esforço foi dedicado para divulgar textos e gráficos. Não houve iniciativa para manipular dados espaciais contendo topologia. Isto porque os algoritmos para tratar topologia de mapas eram complexos e linguagens de programação para desenvolver aplicações para a rede eram incipientes. Com o amadurecimento destas linguagens, como Java, *scripts*, ou C, C++, Delphi para desenvolver CGI (Common Gateway Interface), favoreceu-se projetos com o objetivo de disponibilizar mapas topológicos na Web.

Empresas tradicionais da área de sistemas de informação geográfica (SIG) começaram a desenvolver aplicativos dedicados. Iniciativas públicas também surgiram, como uma opção aos sistemas comerciais. Hoje existem ofertas de sistemas comerciais e públicos para a divulgação de mapas pela Internet. Este trabalho apresenta tópicos a serem considerados para se divulgar mapas pela rede e uma lista, não exaustiva, de sistemas comerciais e públicos.

José Gilberto Jardine
Chefe-Geral

Sumário

Introdução	9
Questões sobre Arquitetura da Aplicação ...	11
Qual o propósito da página?	11
Qual a capacidade operacional?	12
Como deve acontecer a interação usuário/mapas?	12
De que lado estará o aplicativo?	13
Administração dos dados	15
Ambiente Web e a Tecnologia de Mapas	16
Está crescendo?	16
A Web espacial está mais interativa	17
O que já existe no Mercado	19
Servidores de mapas proprietários	19
Servidores de mapas não-proprietários	20
Visualizadores de mapas	23
Instalação do servidor de mapas	24
Considerações Finais.....	25
Conclusões	25
Referências Bibliográficas	28

Diretivas para Disponibilizar Mapas na Internet

José Iguelmar Miranda

Introdução

A Internet revolucionou a comunicação nos últimos dez anos. Com a Internet móvel, mais recente, coloca-se mais atores no processo da comunicação. Espera-se uma demanda crescente do público em receber informações, genericamente classificadas como textos ou imagens. No início da Internet o tipo de documento que fluía era texto. A linguagem de comunicação da rede — HTML — tem a palavra texto na sua definição — *HyperText Markup Language*. *Hyper* significa associar palavras no texto como ponteiros ou apontadores para outros documentos ou partes do mesmo texto (Phillips, 1998). Anos atrás, a velocidade de transmissão de dados era baixa, os computadores pessoais eram limitados em sua capacidade de armazenamento e de processamento. Estes pontos restringiam o fluxo de imagens, por serem arquivos de grande volume.

O fluxo de imagens cresceu à medida que a velocidade de transmissão aumentou, computadores mais rápidos surgiram e o custo da área de armazenamento declinou. Desenvolveram-se algoritmos para compactar imagens, produzindo padrões como GIF (Graphics Interchange Format) e JPG (Joint Photographic Group). A restrição na transferência de imagens estava, e ainda está, no seu formato matricial, que requer grande área de armazenamento. Como vantagem, o armazenamento requer algoritmos simples para manipulá-lo. Armazenamento no formato vetorial é uma solução, mas sua manipulação não é trivial.

A Internet é uma fonte de pesquisa e de troca de informações. Entre estas encontram-se os mapas, ou informações espaciais, com potencial de análise em diferentes áreas do conhecimento. Existem grupos de pesquisa, empresas, universidades, órgãos do poder público — municipal, estadual e federal — e indivíduos com o propósito de colocar dados georreferenciados na Internet. A “Web espacial” oferece troca ou aquisição de dados entre órgãos, mas com limitação, devido ao direito de propriedade.

Com o avanço nas pesquisas, facilidades de uso acontecem para o usuário, embora não implique o mesmo para o desenvolvedor. Nas aplicações para a Internet, o ônus do processamento tem se transferido do cliente, residente no equipamento do usuário, para o servidor, residente em um computador central.

Utilidades surgem, e ainda surgirão, com o uso de informações espaciais na Web. Como exemplo, os serviços de localização (ruas, estradas, lojas, etc.), disponíveis nos computadores convencionais, nos celulares e nos computadores de bordo dos carros mais modernos, usando a tecnologia WAP (Wireless Application Protocol) (Open Mobile Alliance, 2002). Tem-se também o potencial de uso do GPS (Global Positioning System), que pode estar embutido no aparelho móvel.

O objetivo deste trabalho é esclarecer como mapas podem usar o potencial oferecido pela rede, listando uma série de desafios e também de oportunidades disponíveis para a análise de dados espaciais.

A definição de dois conceitos vai ajudar no entendimento deste trabalho: servidor de mapas na Web e visualizador de mapas. Servidor de mapas é qualquer aplicativo que pode disponibilizar informações espaciais — dados espaciais e seus atributos — na Web, via servidor ou cliente, utilizando um conjunto de funções de análise espacial. Visualizador de mapas é um aplicativo local, executado no computador pessoal, acessando dados locais utilizando também um conjunto de funções de análise espacial.

Outra diferença entre o servidor e o visualizador de mapas refere-se ao conceito de interoperabilidade. O servidor de mapas é executado em qualquer sistema operacional, precisando apenas de uma versão do aplicativo. O visualizador só é executado no sistema operacional em que seu código foi desenvolvido. Porém, servidor de mapas na Web não é

necessariamente um sistema de informações geográficas (SIG). A Fig. 1 (esquema a e esquema b) mostra esquematicamente as duas versões possíveis do servidor de mapas e sua ligação com um SIG.

O documento está dividido em cinco seções. A primeira trata da arquitetura do projeto para disponibilizar os mapas na Internet. A segunda seção trata das condições atuais de mapas na Web. A terceira traz uma lista de opções disponível no mercado. A quarta seção apresenta considerações finais e a quinta seção, as conclusões. A escolha dos tópicos a serem discutidos neste documento se baseia em parte nos trabalhos de Limp (2002a, 2002b) e Mckee (2002).

Questões sobre Arquitetura da Aplicação

Manipular mapas na Web não é a mesma coisa que em um SIG convencional. Primeiro, porque as soluções de mapas para a Web são direcionadas para um público diferente. Segundo, porque a arquitetura de desenvolvimento é diferente. Terceiro, porque a segurança dos dados, a interação analista-aplicação, a gerência dos dados, a base de suporte para o sistema funcionar, os equipamentos e outros fatores são abordados de maneira diferente entre os dois modelos. Os tópicos a seguir não são exaustivos, mas cobrem os pontos fundamentais no estudo da questão.

Qual o propósito da página?

O ponto inicial do projeto é definir objetivo e audiência. Perguntas que servem de orientação: o propósito é mostrar um mapa estático ou dinâmico? De que maneira o usuário interage com o mapa, de forma restrita ou ampla?

O usuário apenas visualiza o mapa, não podendo salvá-lo no seu disco local? O objetivo é mostrar espacialmente dados de um banco de dados (BD) ou realizar análises geográficas? Qual o tipo de linha de comunicação, de alta velocidade Intranet ou de baixa velocidade — linha discada — da Internet? Os usuários serão externos ou internos à empresa? Haverá níveis de acesso diferenciado para os dados serem visualizados? Outras questões poderiam ser colocadas.

Qual a capacidade operacional?

Para servir mapas na Web deve-se levar em consideração os diferentes computadores, navegadores Web, resolução dos monitores, etc. existentes no mercado. Não se pode pensar em desenvolver um sistema para Web delimitando o usuário a uma série de requisitos de sistema operacional (SO), capacidade de processamento da máquina, resolução do monitor, quantidade de cores, etc.

A velocidade de conexão na Internet (modem, linha de alta velocidade ou mesmo sem linha física) afeta o fluxo de dados. O tipo de dado afeta o volume de transferência, que por sua vez influencia a velocidade, fechando o ciclo.

A capacidade de processamento do computador que hospeda o servidor deve ser considerada. A quantidade de acessos que se espera deve fazer parte do planejamento. Se o sistema for operar em Intranet, talvez esta questão não seja tão relevante. Mas se for para Internet, pode-se esperar dezenas de milhares de acessos. Esta informação orienta na definição do sistema servidor.

Como deve acontecer a interação usuário/mapas?

Posto de outra forma: o usuário necessita consultar mapas ou analisá-los? A resposta a esta questão influenciará na abordagem de programação e implementação do aplicativo. Em alguns casos, uma visualização da área geográfica de interesse é suficiente. Permite-se nesta situação uma interação leve com o mapa, como mudança de escala e navegação.

Mesmo não necessitando das funcionalidades de análise espacial, os dados apresentados no mapa devem estar atualizados. É de responsabilidade do servidor de mapas consultar a base de dados e recuperar os atributos mais atualizados para mostrá-los. Mas esta decisão fica a cargo do fornecedor da informação.

Na hipótese do usuário necessitar de análises no mapa, a abordagem deve ser diferente. Funções de manipulação devem estar disponibilizadas. Na maioria dos casos, este tipo de aplicação está restrito para uso interno das empresas. Dificilmente alguma empresa abrirá seus dados e mapas

para manipulação por usuários da Internet. O mais comum é disponibilizar este tipo de sistema na Internet restrita, ou Intranet. A empresa pode manipular seus dados e mapas e tornar um ou outro disponível na Web apenas para consulta.

De que lado estará o aplicativo?

Na Web, existem duas maneiras de se interagir com um aplicativo: no próprio navegador local (Netscape ou Explorer), também chamado de *cliente* ou no computador central que hospeda o *servidor*.

Quando a interação é pelo servidor, uma imagem do mapa é gerada, normalmente GIF ou JPG, e enviada para o navegador (cliente). Na interface do navegador, podem existir opções para o usuário interagir com o mapa. Uma parte do mapa pode ser selecionada para mudar de escala. Neste momento, o navegador local (*cliente*) faz uma requisição para o *servidor*, que responde à solicitação redesenhando um quadrante da imagem no navegador. Outra opção seria mover o mouse para uma área do mapa, selecionar e depois apontar para o botão “aproximar.” O processo requisição/resposta é realizado novamente.

O trabalho no servidor pode ser feito por um *servlet* ou um *Common Gateway Interface* (CGI). Um *servlet* é um programa escrito em linguagem Java executado no servidor. Um CGI pode ser escrito em qualquer linguagem de programação, como C, C++, Delphi, Visual Basic, ou por um *script*, como PERL (Practical Extraction and Report Language), também executado no servidor (Felton, 1997; Hunter & Crawford, 1998).

Um mapa-imagem geralmente é pequeno e rapidamente baixado no cliente. Interações sucessivas, como obter uma forma final desejada do mapa, podem aumentar o tráfego requisição/resposta. Se a demanda no servidor for grande, o tempo de resposta cai (Fig. 1, esquema (b)).

Quando a interação é pelo cliente, a responsabilidade de manipular o mapa fica por conta de um aplicativo executado no computador onde está o navegador. O aplicativo pode ser um *applet* — programa escrito na linguagem de programação Java (Flanagan, 1996, 1999). Embora executado no cliente, ele precisa ser carregado a partir do servidor.

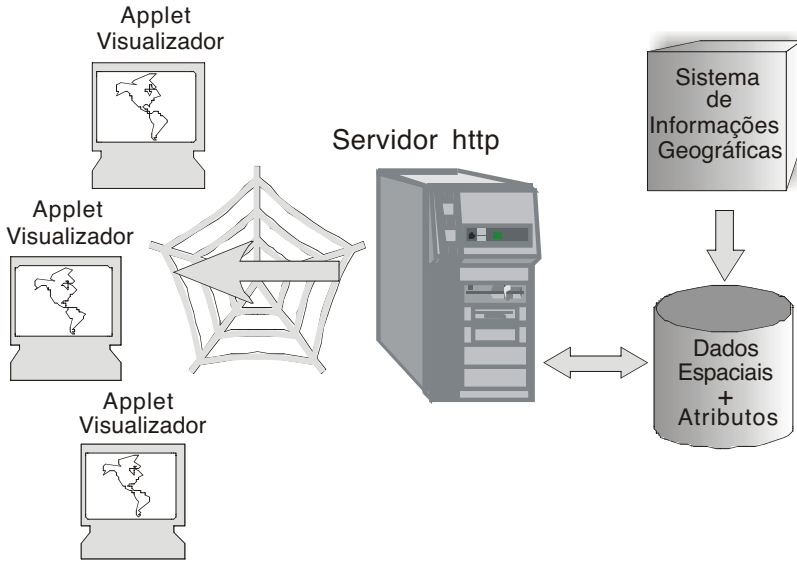


Fig. 1.a. Servidor de Mapas na Internet com uso de *applet*.

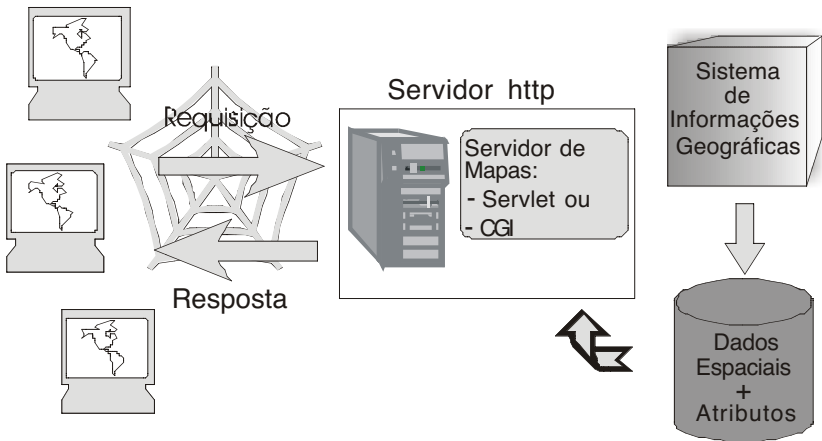


Fig. 1.b. Servidor de Mapas na Internet com uso de *servlet*.

Um *applet* não é como um programa residente no computador local. Quanto maior interatividade for oferecida, maior complexidade terá o aplicativo. O que implica em um aplicativo de maior tamanho, que por sua vez leva a um tempo maior de carga.

Administração dos dados

Interação com um banco de dados

Na Internet é comum a interação entre dados espaciais e seus atributos armazenados em um banco de dados (BD). Dificilmente uma aplicação para a Web não tem uma interação com um BD. Por exemplo, desenvolver um aplicativo que armazene mapas sobre propriedades rurais e suas parcelas ligados a um BD contendo os atributos das propriedades e parcelas. O usuário acessa o mapa da propriedade, escolhe uma parcela, e pode ficar sabendo quem é o proprietário, o tamanho da parcela, qual a cultura, quantas armadilhas para monitoramento de pragas ou coletas de controle fitossanitário foram realizadas, etc.

É comum a aplicação permitir a entrada de dados em um BD pela Internet. A escolha do sistema de gerência de banco de dados (SGBD) passa a ser importante. Sistemas de gerência de BD podem ser de código proprietário ou código aberto. Sistema proprietário deve ser comprado de um fornecedor, como o Oracle, um dos mais usados. Geralmente são sistemas muito caros.

Os sistemas abertos são de uso livre e estão disponíveis de forma gratuita. Os dois mais conhecidos são MySQL (www.mysql.com) e PostgreSQL (www.postgresql.org). Enfatiza-se a interatividade entre o aplicativo de mapas e o banco de dados. Com que SGBD ele é — se já estiver pronto — ou deverá ser — se ainda for ser desenvolvido — compatível? Esta não é uma pergunta irrelevante.

Formato de dados

O servidor de mapas geralmente é direcionado para um formato de dado. Para casos de incompatibilidade, existem facilidades de conversão de formatos. Alguns servidores podem acessar diferentes formatos de dados.

A maioria dos servidores, se não todos, se comunicam com os SGBDs existentes no mercado, tanto os de código proprietário como aberto.

Propriedade dos dados

Nem todo dado que circula na Internet é de domínio público. Alguns fornecedores restringem o uso de seus dados usando sistemas criptográficos. Outros permitem visualização dos dados, mas não permitem operação de salvamento. Outros requerem um cadastramento do interessado, compra ou disponibilizam seus dados somente em CD. Existem os livres, mas com reconhecimento de direitos de cópia (*copyright*) ou citação da fonte.

Compatibilidade com soluções existentes

O servidor de mapas deve se adequar a soluções já existentes, como o SGBD, o sistema operacional e, principalmente, o SIG. Outro caso de compatibilidade diz respeito ao navegador. O servidor de mapas, teoricamente, não deveria sofrer restrição do navegador. O mais comum é o servidor ser usado com os dois navegadores mais comuns, Netscape e Internet Explorer, porém, usando versões mais atualizadas. Por exemplo, se o *Javascript* for usado, uma versão 5+ do navegador é requerida.

Ambiente Web e a Tecnologia de Mapas

Está crescendo

O número de acessos à Internet para obter informações espaciais cresce continuamente. Para atender esta demanda, empresas tradicionais de tecnologia espacial (ESRI, INTERGRAPH, AutoDesk, MapInfo, etc.) estão desenvolvendo aplicações orientadas à Web. Buscando conformidade na troca de informações, a maioria adota o padrão OpenGIS do Open GIS Consortium (2002).

O Open GIS Consortium (OGC) é um consórcio internacional de indústrias abrangendo mais de 220 empresas, agências governamentais e universidades com o propósito de desenvolver especificações de geoprocessamento publicamente disponíveis. Ainda segundo o OGC, constam dos seus objetivos definir padrões para interfaces e protocolos do tipo aberto ou livre, que suportem soluções de interoperabilidade na Web, WAP e serviços de localização (Open GIS Consortium, 2002).

A adoção das especificações do consórcio pelas empresas de SIG para a Web é benéfico. Para enfatizar o valor desta iniciativa, ressalta-se a grande dificuldade atual na área de geoprocessamento com a quantidade de formatos para armazenar arquivos vetoriais e matriciais.

A Web espacial está mais interativa

A liberação de dados vetoriais, com atributos de topologia, está facilitando a interatividade. Imagens matriciais usadas na Internet, como GIF e JPG, não têm topologia. A ausência de topologia dificulta a interação com o mapa. Atributos geográficos, como rios, lagos ou parcelas, não podem ser determinados. Com o uso de topologia isto é possível.

Outra tendência é o uso de dados e métodos não proprietários, através da GML (Geography Markup Language), que compartilha dado espacial (vetorial ou matricial) através da codificação do dado espacial em XML (Extended Markup Language). A GML é uma iniciativa do consórcio OpenGIS. As iniciativas OpenGIS e GML oferecem um padrão, livre do vendedor, para codificar qualquer tipo de dado geográfico bem como qualquer método para mostrá-lo. O objetivo da GML é o transporte e armazenamento da informação geográfica, incluídos a geometria e propriedades do atributo geográfico (Open GIS Consortium, 2002).

Exemplos de especificações com a GML:

- Codificando uma linha:

```
<coordinates decimal = "." cs = "," ts = "whitespace">  
  1.03,2.167 4.16,2.5 4.87,3.0 10.00,4.5  
</coordinates>
```

coordinates indica ao interpretador GML um conjunto de coordenadas. As coordenadas devem ser lidas conforme os parâmetros: o separador de casas decimais (*decimal*) é o ponto; o separador do par *x,y* (*cs*) é a vírgula; e o separador dos pares de coordenadas (*ts*) é um espaço branco. Os valores *decimal*, *cs* e *ts* podem mudar, como vírgula, ponto-e-vírgula e espaço, respectivamente.

- Codificando um ponto:

```
<point srsName = "SC:33" >
  <coordinates>
    44.6,50.8
  </coordinates>
</point>
```

srs (spatial reference system) é o sistema de referência espacial no qual as coordenadas devem ser calculadas. No caso, elas devem estar dentro de um retângulo com coordenadas previamente definidas, tendo por nome (*srsName*) SC:33.

- Codificando uma estrada:

```
<feature typeName="Estrada" >
  <description>
    PI-11
  </description>
  <property typeName="classifica" >
    estrada_estadual
  </property>
  <property typeName="number" type="integer" >
    11
  </property>
  <geometricProperty typeName="linearGeometry" >
    <lineString srsName="SC:33">
      <coordinates>
        0.0,100.0 100.0,0.0
      </coordinates>
    </lineString>
  </geometricProperty>
</feature>
```

O padrão de interface do OpenGIS, o Web Map Server (WMS), acessa mapas de múltiplos servidores. A interface facilita a interatividade, permitindo capturar e fazer mapas pela Web. Enquanto mais fácil para o usuário, torna-se mais difícil para o desenvolvedor. A aplicação a ser desenvolvida tem que dar suporte às solicitações dentre uma multiplicidade de informações espaciais.

O que já existe no Mercado

Servidores de mapas proprietários

Servidores de mapas proprietários disponíveis: MapGuide (Autodesk), ModelServer Discovery (Bentley Systems), ArcIMS (ESRI), GeoMedia Web Map (Intergraph), MapXtreme (MapInfo). Existem outras empresas, listou-se as mais conhecidas no mercado de SIG. Os preços geralmente são cobrados pelo servidor. Como o cliente é o navegador, não há o que ser cobrado.

Alguns fornecedores oferecem seu produto com um *plug-in* que deve ser instalado na máquina cliente, podendo ou não ser cobrado. Sobre preços, a GeoMicro Inc. oferece o servidor AltaMap Server pelo custo de US\$ 4.995,00. Não há despesas adicionais de cliente. Já a Smallworld pede pelo Smallworld Web US\$ 240.000,00, cobrados pelo servidor e mil assentos (Limp, 2002b).

Sobre o navegador que pode ser usado com estes sistemas, há grande variação. Por exemplo, o MapGuide release 4 suporta as versões 3 e 4, mas não a versão 5 do Explorer. Outros só aceitam versões superiores à versão 3 do Explorer, enquanto alguns aceitam qualquer versão. As mesmas observações são válidas para o Netscape. Mapas podem não aparecer em alguns navegadores que não sejam compatíveis com as exigências do servidor adquirido. Os clientes podem ser executados em qualquer sistema operacional (SO), como Unix e suas variantes, Windows NT/9x e MacOS.

Todos os servidores proprietários citados são compatíveis com o ambiente Windows NT, poucos aceitam Unix e MacOS e menos ainda o Windows 9x (Limp, 2002b). Isto é significativo para uma empresa que usa apenas um destes três SO. Ao adquirir um servidor de mapas, pode ser necessário um investimento significativo na compra de novas máquinas com o Windows NT, além de ter que se adaptar a uma nova cultura de SO. Se as máquinas forem Windows 9.x apenas uma atualização será necessária. Mesmo assim, vai depender da configuração, pois as exigências para se executar o NT são bem maiores, em termos de CPU e área de armazenamento em disco, do que para Windows 9.x.

Servidores de mapas não-proprietários

Existem servidores de mapas disponíveis gratuitamente na Internet. O MapServer, uma aplicação do tipo CGI, usa o padrão OpenSource (www.opensource.org) para construir aplicações executáveis na Internet. O sistema foi construído usando ferramentas de domínio público. Ele foi desenvolvido para a plataforma Linux/Apache, mas tem executáveis para o Windows NT/9x. O MapServer suporta o MapScript, permitindo linguagens populares do tipo *script*, como Perl, Python, Tk/Tcl, Guile e mesmo Java serem executadas no cliente acessando diretamente as API (Application Program Interface) em C do MapServer (2002).

O MapServer foi originalmente desenvolvido pela Universidade de Minnesota em cooperação com a NASA (National Aeronautics & Space Administration) e o Departamento de Recursos Naturais de Minnesota, recebendo também contribuições do Centro de Informações e Conservação de Solos de Minnesota (MapServer, 2002). Exemplo de uma interface do MapServer pode ser visto na Fig. 2.

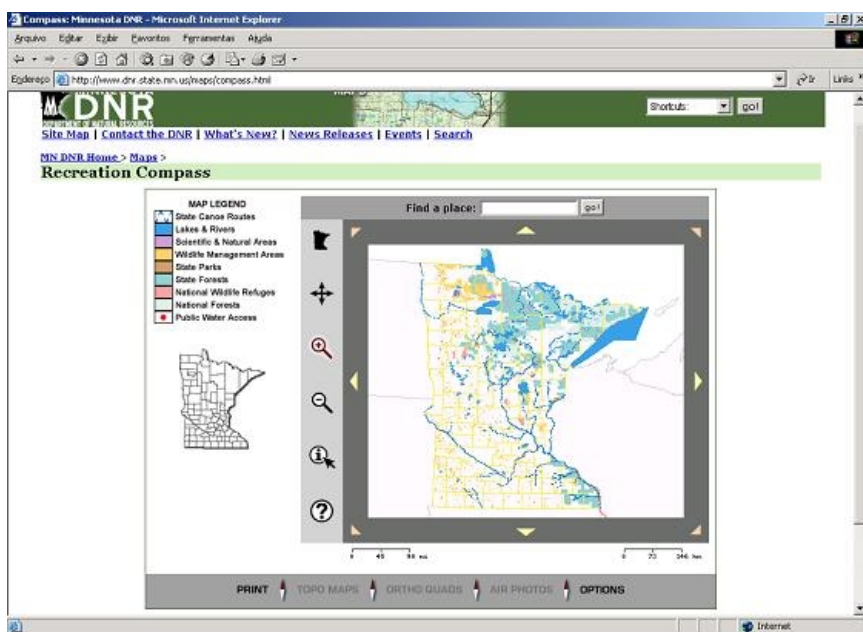


Fig. 2. Tela do MapServer desenvolvida pelo Minnesota Department of Natural Resources.

O Mapit! é um CGI do tipo *script*, escrito em Python, para uso com dados espaciais matriciais. Pode ser usado nas plataformas Linux e Windows NT4/ME/98. O Mapit!, mostrado na Fig. 3, apresenta opções de aproximação (a lupa), navegação (as setas) e também a possibilidade de escolher um tema em "Markers", como estações de trem, metrô, correios, etc. Estas opções serão inseridas no mapa após clicar o botão "Show." A arquitetura do sistema é cliente/servidor. Como o Mapit! foi escrito em Python, será necessário ter no servidor o Python e o Python Imaging (Mapit, 2002).

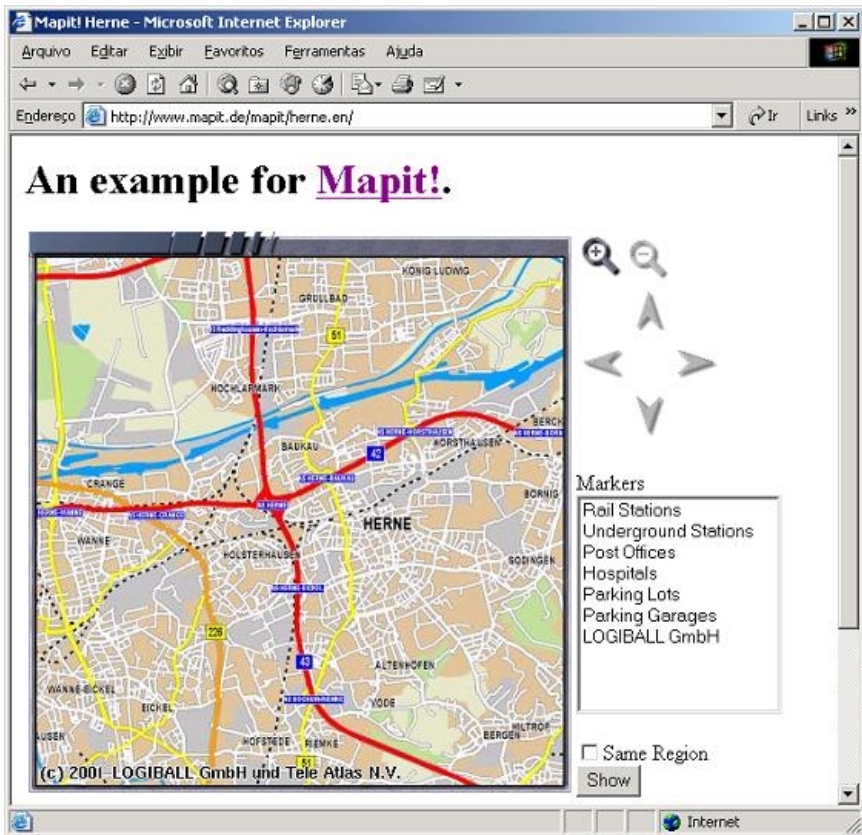


Fig. 3. Janela principal do Mapit!

O GeoTools é um *applet*. Ele é usado para mostrar mapas de maneira interativa em um navegador, sendo executado na máquina cliente. A equipe do GeoTools disponibiliza uma biblioteca Java que pode ser usada para desenvolver solução do tipo cliente-servidor (GeoTools, 2002).

O ALOV Map é uma aplicação Java que permite publicar mapas vetoriais e matriciais na Internet. Ele pode ser usado de duas maneiras: cliente ou cliente/servidor. A versão cliente é um *applet* para visualização de dados no formato SHAPE da ESRI. Como é característica de todo *applet*, cada vez que ele for inicializado, o servidor http baixará todos os arquivos de mapas na máquina cliente. Isto requer cuidado com o tamanho dos arquivos, senão o tempo de transmissão pode se tornar longo.

A versão cliente/servidor do ALOV Map, um *servlet*, é mais flexível e permite transferir dados de forma incremental. O servidor de mapas envia dados vetoriais para o cliente de forma mais eficiente, através da técnica de pacotes, reduzindo o tráfego de rede. Os atributos espaciais são armazenados em um banco de dados SQL (Interbase ou MySQL) de acordo com padrões SFS da OpenGIS (ALOV, 2002).

O GIS-4.0, um *applet*, foi desenvolvido na Universidade da Califórnia em Berkley e funciona em ambiente Unix ou Windows. A Universidade libera apenas a versão *bin* (executável) do programa. O GIS-4.0 só trabalha com mapas no formato matricial (<http://elib.cs.berkley.edu/gis>).

O JShape é grátis apenas nas versões 1.x e 2.x. A partir da versão 3.x, ele passou a ser proprietário. As versões 1.x e 2.x são *applets* para leitura de arquivos no formato SHAPE (ESRI). A extensão servidor do JShape começou a partir da versão 3.x. Esta versão é melhor que as anteriores, liberando imagens com formatos diversos, tamanhos, cores e níveis de cinza de mapas na Web e em aparelhos WAP (Wireless Application Protocol) (Fig. 4) (www.jshape.com).

Como iniciativa nacional, tem-se a TerraLib (www.terralib.org), uma biblioteca de classes que permite a construção de um ambiente colaborativo e seu uso para o desenvolvimento de variados aplicativos e ferramentas para a análise geográfica. Seu principal objetivo é possibilitar o desenvolvimento de uma nova geração de aplicativos SIG, baseados nos avanços tecnológicos dos bancos de dados espaciais.

Como ferramenta de pesquisa, a TerraLib objetiva fornecer um ambiente para o desenvolvimento de protótipos SIG que incluam novos conceitos, como modelos de dados espaço-temporal, ontologias (semânticas) geográficas e técnicas avançadas de análise espacial. Segundo Rocha (2002), "em computação, uma ontologia é uma representação de categorias de coisas que existem numa determinada área de interesse, mais uma linguagem para manipular categorias dessa representação." Os SGBD que podem interagir com a biblioteca são MySQL, PostgreSQL, ORACLE e ACCESS. TerraLib está implementada como uma biblioteca de classes e funções C++, escrita no padrão ANSI-C++.

A TerraLib é um produto da Divisão de Processamento de Imagens (DPI) do Instituto Nacional de Pesquisas Espaciais (INPE), Tecgraf, o Grupo de Computação Gráfica da Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) e Fundação de Ciência, Aplicações e Tecnologia Espaciais (FUNCATE).



Fig. 4. Aplicação de mapas em WAP.

Visualizadores de mapas

Visualizadores de mapas podem ser adquiridos gratuitamente pela Internet. São aplicativos desenvolvidos com a finalidade exclusiva de se visualizar dados espaciais, geralmente de formatos proprietários. Os arquivos no formato SHAPE (ESRI) apresentam a maior compatibilidade com os visualizadores.

A ERDAS coloca disponível grátis o ViewFinder 2.1, que pode ser adquirido na página www.erdas.com, escolhendo a opção “Free Download”. A PCI Geomatics (www.pcigeomatics.com) oferece o Geomatica FreeView. Ele permite visualizar, realçar e analisar imagens de vários satélites e foto aérea, além de mapas no formato vetorial. O FreeView disponibiliza muitas ferramentas úteis, como aproximações no mapa, realce de imagens, mostra valores numéricos e lista tabela de atributos, mas não disponibiliza funções de análise. Os dados só podem ser vistos. Nenhuma operação de edição ou salvamento é permitida.

O TerraView (www.terralib.org) é um aplicativo construído com a biblioteca TerraLib — acima referenciada — com a finalidade de apresentar um visualizador de dados geográficos com recursos de consulta e análise no formato SHAPE. Comparado com os visualizadores anteriores, este apresenta mais flexibilidade, com a opção de análise de dados.

O TerraView manipula mapas vetoriais e imagens matriciais, ambos armazenados em banco de dados relacionais ou geo-relacionais de mercado, incluindo ACCESS, PostGress, MySQL e Oracle. O TerraView coloca à disposição do usuário uma extensa lista de ferramentas de análise.

Instalação do servidor de mapas

A instalação de um SIG convencional, por maior que seja a empresa e por mais complexo que seja o SIG, estará limitada às dependências físicas da empresa. O atendimento dos quesitos de segurança de acesso aos dados é mais fácil. Estes sistemas são comprados como um pacote fechado, treinamentos são oferecidos e a partir daí, começa seu uso. Sistemas de informação geográfica pela Web requerem procedimentos diferentes, desconsiderando que o treinamento é parte essencial. A questão da segurança dos dados é mais rígida, uma vez que não há mais a limitação das dependências físicas.

Antes do servidor de mapas estar operacional, tem que ser decidido que mapas estarão acessíveis aos usuários, como eles devem aparecer, que tipos de operações serão possíveis, entre outras considerações. A questão de invasões dos servidores por piratas eletrônicos, embora seja uma preocupação da empresa como um todo, deve também ser considerada pela equipe do SIG.

Ligação com um SGBD deverá ocorrer em praticamente todas as aplicações de servidor de mapa. Este é outro diferencial com o SIG convencional, pois a operação do servidor de mapa precisa de técnicos com habilidades para escrever programas em uma linguagem de programação, como Visual Basic, C++, Delphi ou Java, conhecer HTML ou mesmo XML e ainda ter conhecimento de linguagens do tipo *script*, como Javascript. Em um SIG convencional ligado a um BD também haverá especialistas, mas a filosofia de trabalho é bem diferente nos dois ambientes.

Considerações Finais

A Web espacial apresenta-se como uma ilha de informações sobre mapas. Existem muitas instituições — comerciais, acadêmicas, individuais, governamentais — oferecendo dados espaciais. Alguns dados não podem ser vistos, apenas adquiridos em forma de CD. Em outros casos, os metadados não estão padronizados, dificultando a visualização. Nos Estados Unidos o movimento de padronização é forte entre empresas.

No Brasil, ainda não há esta preocupação, ou, se existe, poder-se-ia dizer que o movimento é imperceptível. Seria o momento apropriado para começar a discutir a situação, e não deixar a proliferação de formatos. No caso das agências federais americanas, há uma tendência de conformidade com o OpenGIS e padrão metadado americano chamado de Federal Geographic Data Committee (FGDC) (McKee, 2002).

Conclusões

Enfatiza-se que um esforço deve ser feito para que os dados espaciais disponibilizados na Web sejam livre de um fornecedor de SIG e tenham um formato não proprietário. Este esforço pode ser dividido em duas frentes (McKee, 2002):

1. Interfaces livres

Interfaces são aplicações que permitem sistemas independentes se comunicarem. Fornecedores não mostram como funcionam suas interfaces de comunicação com sistemas independentes da sua linha de produtos por motivos óbvios. Mas existe uma tendência de se fugir de sistemas proprietários. Em 1994, o OGC começou a relacionar vendedores de SIG em um comitê técnico para achar consenso sobre interfaces abertas que permitam que os sistemas se comuniquem em rede. Em 1999, produziram o Web Mapping Testbed (WMT) para fazer de “testbed” seu principal método para desenvolver especificação.

2. Esquema de metadados padrão

Um memorial sobre os dados espaciais deveria ser anexado aos próprios dados. Informações sobre quando os dados foram coletados, que área geográfica foi coberta, que tipo de dado (matricial, vetorial, TIN), qual a escala, qual o sistema de referência, que atributos geográficos estão inclusos, como foram os atributos definidos. Todas estas informações, também conhecidas por linhagem, e muitas outras constituem o que se chama *metadados*, isto é, dados descrevendo dados.

Um esquema de metadado estabelece uma seqüência específica, vocabulário e formato de dados. Se múltiplos repositórios usam o mesmo esquema, é possível automatizar buscas no repositório. Como problemas de incompatibilidade vêm de muito tempo, existem muitos esquemas de metadados incompatíveis.

Conforme citado no tópico “a Web espacial está mais interativa” existe a iniciativa GML (Geography Markup Language), do OpenGIS, baseada na meta linguagem XML. Esta é uma ação positiva no sentido de amenizar o problema. Se a GML vier a ser adotada no ambiente Web espacial, abre-se oportunidade para a homogeneização da informação espacial. Segundo McLaughlin (2000), “[a meta linguagem XML] traz com ela uma imensa promessa: o que o Java fez para a portabilidade de código, XML anuncia fazer para a portabilidade de dados. A Sun tem anunciado um slogan ambicioso «Java + XML = Código Portável + Dados Portáveis».”

Web como uma ilha não se aplica apenas para o caso dos mapas, mas também para uma infinidade de outras fontes de informações. A falta de padrão para a divulgação de dados tem sido um impedimento para que os dados na Internet possam ser usados de forma coerente, embora existam bons algoritmos de busca. Uma padronização de dados na Web, espaciais ou não, facilitaria serviços como a de mineração de dados.

Referências Bibliográficas

ALOV. **ALOV map**: free Java GIS. Disponível em: <<http://alov.org/index.html>>. Acesso em: 27 ago. 2002.

FELTON, M. **CGI Internet programming with C++ and C**. Upper Saddle River: Prentice Hall, 1997. 514 p.

FLANAGAN, D. **Java in a nutshell**: a desktop quick reference for Java programmers. Sebastopol: O'Reilly & Associates, 1996. 438 p. (Nutshell handbook).

FLANAGAN, D. **Java in a nutshell**: a desktop quick reference. 3rd ed. Beijing: O'Reilly & Associates, 1999. 648 p. (The Java series).

GEOTOOLS. **GeoTools**: open source mapping toolkit. Disponível em: <<http://www.geotools.org>>. Acesso em: 27 ago. 2002.

HUNTER, J.; CRAWFORD, W. **Java servlet programming**. Beijing: O'Reilly & Associates, 1998. 510 p. (The Java series).

LIMP, W. F. **GeoWorld - Sept 99 - mapping hits warp speed on the World Wide Web!** Disponível em: <<http://www.geoplance.com/gw/1999/0999/999tec.asp>>. Acesso em: 6 set. 2002a.

LIMP, W. F. **GeoWorld - Nov 99 - don't hit warp speed with the wrong equipment!** Disponível em: <<http://www.geoplance.com/gw/1999/1199/1199Imp.asp>>. Acesso em: 6 set. de 2002b.

MAPIT! **MapIt!** [home page]. Disponível em: <<http://www.mapit.de/mapit.en.html>>. Acesso em: 27 ago. 2002.

MAPSERVER. **MapServer CGI reference**. Disponível em: <<http://mapserver.gis.umn.edu/doc36/cgi-reference.html>>. Acesso em: 27 ago. 2002.

MCKEE, L. **GEO resources**: web mapping guide: technology trends. Disponível em: <<http://www.geoplace.com/gr/webmapping/technology.asp>>. Acesso em: 26 ago. 2002.

MCLAUGHLIN, B. **Java and XML**. Beijing: O'Reilly & Associates, 2000. 479 p.

OPEN GIS CONSORTIUM. **OGC - Open GIS Consortium**. Disponível em: <<http://www.opengis.org>>. Acesso em: 6 set. 2002.

OPEN MOBILE ALLIANCE. **Wap forum**. Disponível em: <<http://www.wapforum.org>>. Acesso em: 6 set. 2002.

PHILLIPS, L. A. **Using HTML 4**. 4th ed. [Indianapolis]: Que, 1998. 688 p.

ROCHA, L. M. Insetos, neurônios e sistema imunológico: esses são os modelos das redes do futuro. **O Estado de São Paulo**, São Paulo, 16 set. 2002. Informática, p.6, CI. Entrevista.



Informática Agropecuária