

# SERVIÇOS *WEB* GEOGRÁFICOS E SUA INTEROPERABILIDADE

## *Geographical Web Services and its Interoperability*

**Jonas Bezerra de Melo Junior<sup>1</sup>**  
**Ana Lucia Bezerra Candeias<sup>2</sup>**  
**João Rodrigues Tavares Junior<sup>1</sup>**

<sup>1</sup>**Universidade Federal de Pernambuco - UFPE**

**Pós-Graduação em Ciências Geodésicas e Tecnologias da Geoinformação- DECART**  
Av. Acadêmico Hélio Ramos, s/n - Cidade Universitária - CEP 50740-530 - Recife-PE - Brasil  
jonas@recife.pe.gov.br  
joaoufpe@gmail.com

<sup>2</sup>**Universidade Federal de Pernambuco - UFPE**

**Centro de Tecnologia e Geociências, Departamento de Engenharia Cartográfica**  
Av. Acadêmico Hélio Ramos, s/n - Cidade Universitária - CEP 50740-530 - Recife-PE - Brasil.  
analucia@ufpe.br

## RESUMO

Os Sistemas de Informações Geográficas (SIG) foram concebidos como ilhas de informação. Eles foram desenvolvidos de forma autônoma e utilizavam tecnologias proprietárias. Não havia a preocupação com o intercâmbio de dados nem com a comunicação entre as aplicações. A necessidade de compartilhamento de dados entre os SIGs motivou a comunidade científica a propor novas tecnologias que dessem suporte para tal fim. Recentemente uma tecnologia denominada de Serviços *Web* foi apontada como um paradigma que facilita a construção de SIG interoperáveis. Estes serviços são independentes de plataformas e linguagens de programação. Tendo-se em vista as vantagens e potencialidades desta tecnologia para as instituições que lidam com informações geográficas. Este artigo apresenta um aplicativo para verificar uso prático dos Serviços *Web* em interoperabilidade de SIG.

**Palavras chaves:** SIG, Serviços *Web*, Interoperabilidade, Cartografia Digital.

## ABSTRACT

Geographical Information Systems started based on information islands. They were autonomously developed and used proprietary technologies. There was no concern about data exchange or applications communication. The data sharing between GIS makes the scientific community to propose new technologies to support data exchange. Recently, a technology called *Web* services has been seen as a good paradigm for building interoperable GIS. These services are independent of platform and program languages. *Web* services have advantages and potentialities for the institutions who deal with geographic information. This work shows a practical experiment of *Web* Services use with GIS interoperability.

**Keywords:** GIS, *Web* Services, Interoperability, Digital Cartography.

## 1. INTRODUÇÃO

Os Sistemas de Informações Geográficas (SIG) estão presentes nas mais diversas áreas, dando suporte a aplicações tais como o cadastro imobiliário, saúde pública, prospecção de petróleo, entre outros.

Nas instituições públicas e privadas já se observa o novo comportamento frente aos SIGs e suas necessidades de aquisição e manipulação dos dados geográficos.

A replicação de informação pode comprometer o bom desempenho de um SIG. Além disto, uma boa

administração destes dados pode gerar um menor custo de tempo, de recursos computacionais e financeiros. Um exemplo disto é o dado: imagem de satélite de alta resolução que possui alto custo e necessita de grande volume de memória e espaço de armazenamento. Caso vários órgãos estaduais precisem de uma mesma imagem, isto não deveria implicar para o Estado comprar várias vezes esse mesmo dado, bem como ocupar espaço em disco de vários órgãos para armazená-lo.

A interoperabilidade de dados pode solucionar este e vários outros problemas que os desenvolvedores e gerenciadores de SIGs possuem hoje em dia.

No início da década de 90, os SIG desenvolvidos foram configurados como ilhas de informação. E, portanto, os sistemas gerados com esta filosofia eram necessariamente independentes na forma de tratar e modelar os dados bem como nas escolhas de bancos de dados, paradigmas de linguagens de programação e sistema operacional (EGENHOFER et al. 1997).

As empresas e instituições públicas que, atualmente, ainda utilizam SIG com a filosofia da década de 90 possuem grande dificuldade em compartilhar seus dados e geralmente só podem acessá-los no local onde estão instalados.

Este problema vem sendo enfrentado pela comunidade científica da área de SIG e várias pesquisas trabalham no sentido de propor tecnologias que possibilitem as instituições compartilharem seus acervos cartográficos e recursos computacionais.

Tendo-se em vista as vantagens e potencialidades dessa tecnologia para as instituições que lidam com informações geográficas, foi desenvolvido um aplicativo para verificar uso prático dos Serviços *Web*.

Este artigo investiga uma tecnologia denominada Serviços *Web* que possibilita aos SIG o intercâmbio de dados e funcionalidades, independentemente de plataforma tecnológica.

## 2. INTEROPERABILIDADE DE SIG

O IEEE define a interoperabilidade como a habilidade de dois ou mais sistemas ou componentes de trocar informação e usar a informação que foi trocada. (IEEE, 2004).

Desta definição conclui-se que existem dois problemas a serem resolvidos. O primeiro é prover a comunicação entre os sistemas (aspecto físico). O segundo parte da premissa que o dado está disponível e o desafio é entendê-lo para que se possa utilizá-lo (nuance semântica).

### 2.1 Interoperabilidade Física

Os SIG são mais complexos que os sistemas puramente descritivos. Eles se diferenciam dos outros por causa de algumas peculiaridades como o alto custo dos dados e a complexidade nas operações com estes dados. Por este motivo sempre foram estudados e desenvolvidos de forma diferenciada dos sistemas con-

vencionais onde os dados são apenas descritivos, ou seja, não têm uma representação gráfica espacial (TSOU, 2001).

Na busca da solução para o compartilhamento dos dados surgiu a linha de pesquisa da interoperabilidade. Este termo possui significados como sistemas abertos, capacidade de intercâmbio de dados, compartilhamento de aplicações e uniformidade na interface com os usuários (EGENHOFER et al, 1997).

O caminho inicial na integração de SIG foi a utilização de conversores de dados. Os arquivos de um determinado fabricante de SIG eram convertidos para o formato que o outro fabricante pudesse ler. Surgiram também formatos padrão de dados (STDS, DXF, GML, VPF, ShapeFile) que facilitaram o intercâmbio de dados entre os SIG. Não havia interação entre os sistemas o que se buscava era o acesso aos dados (FONSECA, 2001).

Atualmente, os SIG vêm acompanhando a evolução tecnológica e incorporando novos paradigmas que auxiliam o processo de integração. Um exemplo disso é a migração dos dados armazenados em arquivos para banco de dados espaciais (TSOU, 2001). Neste sentido, os Serviços *Web* é uma proposta da comunidade científica para integrar sistemas geográficos distribuídos (ALAMEH, 2002) (MELO JR e CANDEIAS, 2004) (MELO JR e CANDEIAS, 2005).

## 3. SERVIÇOS WEB

Hoje, o principal uso da *World Wide Web* é o acesso interativo a documentos e aplicações. Na maioria dos casos, tal acesso é realizado por usuários humanos, tipicamente trabalhando através de browsers, áudio *players*, ou outros sistemas *front-end* interativos. A *Web* pode crescer significativamente em poder e escopo se ela for estendida para suportar comunicação entre aplicações, de um programa para outro (W3C, 2001).

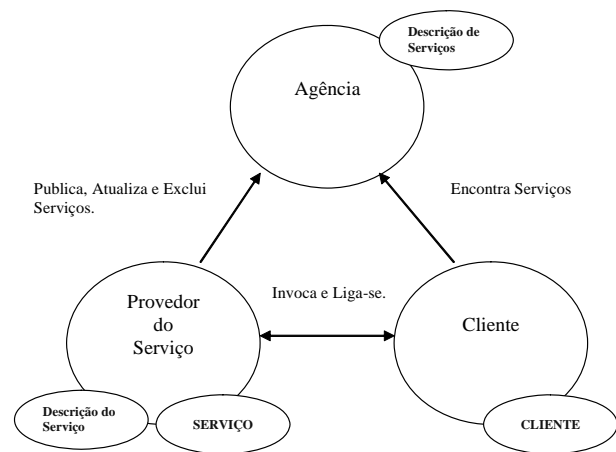
O uso da Internet vem crescendo e as empresas vêm direcionando seus sistemas para a *Web* em busca de visibilidade global, redução de custos e maior automação (TSALGATIDOU e PILIOURA, 2002). Espera-se que os atuais sistemas centralizados sejam substituídos por uma estrutura de serviços distribuídos e dinâmicos (TSOU, 2001).

Nesse contexto surge a tecnologia dos Serviços *Web* que provê uma arquitetura de interação sistemática entre aplicativos na *Web* (CURBERA et al, 2001). O modelo dos Serviços *Web* é uma evolução dos sistemas orientados a objetos e do paradigma dos componentes. Ele incorpora conceitos fundamentais de orientação a objeto como encapsulamento, troca de mensagens (*message passing*) e ligação dinâmica. Os Serviços *Web* são componentes leves (*lighweight*), fracamente acoplados e independente de plataformas e linguagens de programação (TSALGATIDOU e PILIOURA, 2002).

### 3.1 Definições

Serviço *Web* é um sistema de software identificado por um URI (Identificador Uniforme de Recursos), cujas interfaces públicas e ligações são definidas e descritas utilizando-se XML (Linguagem de Marcação Extensível) (FALLSIDE e WALMSLEY, 2004). Sua definição pode ser descoberta por outros sistemas de software. Estes sistemas podem então interagir com o Serviço *Web* numa maneira prescrita na sua definição, usando mensagens baseadas em XML transportadas por protocolos da internet (W3C, 2003).

Serviços *Web* são aplicações modulares auto-contidas e auto-descritas que podem ser publicadas, descobertas e invocadas através da *Web* (CHAPPELL e



JEWELL, 2002) (Figura 1).

Fig. 1 - Serviços *Web*: Arquitetura Orientada a Serviços

Os tópicos subsequentes descrevem as definições de papéis, componentes e operações dos Serviços *Web* de acordo com TSALGATIDOU e PILIOURA (2002).

#### 2.1 Papéis

**Provedor de Serviço:** É aquele que provê o serviço através de sistemas aplicativos. A partir de uma perspectiva comercial, ele é o proprietário do serviço. A partir de uma perspectiva arquitetural, ele é a plataforma que hospeda acessos ao serviço. Ele também tem sido referenciado como um ambiente de execução do serviço ou o container do serviço. Ele disponibiliza seus serviços na Internet através da atividade de publicação. A ele também compete a atualização bem como a retirada de seus serviços da internet.

**Cliente:** Na perspectiva comercial, ele é o negócio que requer que certas funções sejam satisfeitas. Na perspectiva arquitetural, ele é a aplicação que está procurando por um serviço e invocando ou iniciando uma interação com ele. O papel do solicitante pode ser realizado por uma pessoa através de um browser, um programa aplicativo ou outro Serviço *Web*. Seu papel é

o de cliente no padrão de troca de mensagens cliente-servidor. O cliente encontra o serviço desejado através da agência.

**Agência:** É um conjunto pesquisável de descrição de serviços. Ela funciona como uma intermediária entre os provedores e os clientes. Os provedores a utilizam para publicam as descrições de seus serviços. Já os clientes pesquisam na agência as informações necessárias para se ligar aos provedores. A agência pode ser centralizada ou distribuída.

Estes três atores interagem de uma forma sistemática segundo tecnologias padrão independentes de linguagens e plataformas. Mais adiante será visto que esta independência se baseia no fato da utilização do padrão XML como base do protocolo de comunicação e das mensagens trocadas entre os atores.

#### 2.2 Operações

Para que uma aplicação possa utilizar os Serviços *Web*, três ações precisam ocorrer: publicação de descrições de serviços, descobrimento e recuperação de descrição de serviços, e ligação ou invocação de serviços baseados em suas descrições. Estas ações podem ocorrer isoladamente ou iterativamente, com qualquer cardinalidade entre os papéis. Estas operações são:

**Publicar:** A fim de ser acessível, um serviço precisa publicar suas descrições de tal forma que um solicitante possa subsequentemente encontrá-lo. Onde ele é publicado pode variar dependendo dos requisitos da aplicação.

**Descobrir:** Na operação de descoberta, o solicitante de serviço recupera uma descrição do serviço diretamente ou consulta o registro pelo tipo de serviço requerido. A operação de descoberta pode envolver duas fases de ciclo de vida para o solicitante do serviço: a primeira, em tempo de design, a fim de recuperar a descrição da interface de serviço para o desenvolvimento do programa, e a segunda, em tempo de execução, a fim de recuperar a ligação e descrição da localização do serviço para invocação.

**Interagir:** Eventualmente, um serviço precisa ser invocado. Na operação de interação o solicitante de serviços invoca ou inicializa uma interação com o serviço em tempo de execução usando os detalhes de ligação, contidos nas descrições dos serviços, para localizar, contatar, e invocar o serviço

#### 2.3 Componentes

O Serviço *Web* é composto basicamente de dois elementos: o serviço em si e o descritor dele.

**Descritor do Serviço** – O descritor do serviço contém os detalhes da interface e de sua implementação. Isso inclui seus tipos de dados, operações, informações de ligação e sua localização na rede. Ele também inclui uma categorização de outros metadados para facilitar a descoberta e utilização pelos clientes.

**Serviço** – É a implementação do serviço conforme estabelecida no descritor do serviço. O serviço é um módulo de programa fornecido pelo provedor do serviço e disponibilizado em plataformas acessíveis de rede. Ele existe para ser invocado pelo solicitante do serviço ou para interagir com ele. O serviço pode ser composto por outros serviços, ou seja, ele pode também desempenhar o papel de cliente usando outros Serviços *Web* na sua implementação.

## 2.4 Vantagens dos Serviços

### Implantação Rápida e Fácil

Projetos que utilizem o modelo de Serviços *Web* podem criar novos produtos através da reutilização e/ou combinação dos serviços existentes. Isto leva a uma diminuição no investimento e tempo de desenvolvimento em relação ao modelo tradicional de confecção de sistemas.

### Interoperabilidade

Qualquer Serviço *Web* pode interagir com outros Serviços *Web*. A interação realizada através de trocas de mensagens entre os participantes é baseada em XML. A tecnologia é independente de plataforma e linguagem de programação. Isto significa que os desenvolvedores não precisam modificar seus ambientes de desenvolvimento a fim de produzir ou consumir Serviços *Web*. Além do mais se permite que aplicações legadas sejam expostas como serviços. Essa arquitetura permite facilmente a interoperabilidade entre aplicações legadas bem como entre Serviços *Web* e aplicações legadas.

### Integração Imediata

Arquiteturas de sistemas tradicionais, não baseadas em conceitos de camadas, incorporam acoplagens relativamente sensíveis à mudança. Uma nova implementação ou uma mudança na saída de um subsistema irá geralmente causar quebra nas colaborações entre os subsistemas.

A tecnologia dos Serviços *Web* traz no seu bojo uma desacoplagem significativa que facilita o processo de construção “*just-in-time*” de sistemas, ou seja, seus componentes são integrados de forma imediata no momento exato de sua necessidade.

### Redução de complexidade pelo encapsulamento

Os serviços podem ser enxergados como componentes distribuídos na internet. O importante é saber o que o serviço faz e não como ele foi implementado. Isto reduz a complexidade dos sistemas a serem desenvolvidos, uma vez que os projetistas não têm que se preocupar com os detalhes de implementação dos serviços que eles estão invocando..

## 3. EVOLUÇÃO TECNOLÓGICA LEVANDO O SIG PARA A INTERNET

Ao longo dos anos os SIG vêm incorporando a evolução tecnológica da área de informática melhorando seu desempenho e facilitando a integração entre sistemas.

Este tópico irá contextualizar a evolução da tecnologia até o estágio dos Serviços *Web* e seu reflexo na interoperabilidade física entre SIG.

### 3.1 Internet, Web e Serviços Web

Quando dois ou mais computadores se interligam diz-se que estão ligados em rede. As regras que gerenciam esta interligação são denominadas de protocolos. Ao longo dos anos foram propostas várias tecnologias para suportar redes de computadores. A grande vencedora foi aquela que propôs uma arquitetura simples baseada na família de protocolos TCP/IP (*Transmission Control Protocol/Internet Protocol*) (ALBUQUERQUE, 2001).

A Internet é uma grande rede mundial de computadores que se comunica através dos protocolos TCP/IP. Cada computador é unicamente identificado pelo seu endereço IP (*Internet Protocol*). A Internet também realiza a interconexão entre redes, tornando-se uma rede de redes. Neste trabalho a Internet é vista como o conjunto de computadores interligados, porém sem considerar o que está armazenado neles.

A Internet vem revolucionando a maneira com a qual lidamos com a informação. Ela encurtou distâncias e abriu o acesso instantâneo a milhões de fontes de conhecimento. Como consequência disto, notamos uma corrente migratória dos sistemas de informação para a Internet. O sucesso dessa plataforma está na simplicidade dos seus protocolos e na capacidade de distribuição da informação através de redes heterogêneas.

Nota-se também que, devido à popularidade da Internet, o paradigma dos Sistemas de Informações Geográficas está se deslocando para uma nova direção que é a de Serviços de Informações Geográficas. A idéia é que a computação passe a ser distribuída. Cada nó da rede pode consumir quanto prover serviços aos outros nós (TSOU,2001).

A *Web (World Wide Web)* é o conjunto dos recursos disponíveis na Internet. Ela é considerada uma rede de recursos de informação (W3C, 2001). Ela tornou-se um veículo popular de distribuição e compartilhamento de informação (CURBERA et al, 2001; SHEKHARL, 2001). Atualmente seu principal uso é o acesso interativo a documentos e aplicações. Na maioria dos casos, tal acesso é realizado por usuários, tipicamente trabalhando através de *browsers*, áudio *players*, ou outros sistemas *front-end* interativos (W3C, 2001). O acesso a estes recursos é efetuado através de protocolos.

los que utilizam um esquema de nomeação uniforme URL.

Um exemplo de recurso é a página do Departamento de Cartografia <http://www.ufpe.br/decart/index.htm> onde HTTP é o protocolo de acesso ao recurso **index.htm** localizado no diretório **decart** da máquina **www** que faz parte da rede **ufpe.br**.

Uma evolução natural da *Web* é que seus recursos sejam consumidos também por sistemas computacionais. A interação sistemática entre aplicativos na *Web* é o grande salto esperado nesta evolução. A tecnologia dos Serviços *Web* surge como uma proposta para implementar a **interação sistemática entre aplicativos**. Ela é uma tecnologia que permite a integração de sistemas heterogêneos e distribuídos (TSALGATIDOU e PILIOURA, 2002).

A tecnologia dos Serviços *Web* surge no contexto de proporcionar a interação entre sistemas baseados na *Web*. Ela é capaz de encapsular os sistemas para que sejam acessados externamente através dos métodos da sua interface. Outra característica fundamental da tecnologia dos serviços *Web* é a sua independência de sistema operacional e linguagem de desenvolvimento. Sua tecnologia se baseia na Internet e toda a interação entre os sistemas é feita através de troca de mensagens baseadas em XML (Linguagem de Marcação Extensível) (TSALGATIDOU e PILIOURA, 2002).

Uma característica muito positiva desta tecnologia é que se pode desenvolver uma interface também para os sistemas legados (herdados). Neste artigo, o termo sistema legado significa um sistema antigo que tenha sido desenvolvido com tecnologia anterior ao serviço *Web*.

### 3.2 Sistemas de Informações Geográficas Distribuídos

“SIG Distribuído é uma coleção de sítios (*sites*) geralmente distanciados entre si e conectados por uma rede de comunicação de dados. Cada sítio é um SIG autônomo que contém poder de processamento e dados geo-espaciais” (WANG, 2000).

A distribuição se faz necessária, pois SIG tradicionais têm arquitetura fechada implicando em pouca interoperabilidade, reusabilidade e flexibilidade (TSOU e BUTTENFIELD, 2002).

Devido ao alto custo envolvido na aquisição, produção, exploração e disseminação de dados geoespaciais, tais como imagens de satélite, é praticamente impossível para uma simples organização possuir todos os dados que ela necessita. Um novo paradigma para o geoprocessamento é necessário para que a troca de informações saia da simples transferência de arquivos para um nível mais avançado. Ele deverá superar as barreiras causadas por plataformas de SIG incompatíveis. A partir deste paradigma deseja-se que seja possível aos usuários compartilharem dados e integrar fun-

cionalidades de SIG diferentes em um ambiente distribuído (WONG *et al*, 2002).

As tecnologias independentes de plataforma propostas para dar suporte à interoperabilidade e distribuição de SIG estão entre CORBA (Common Object Request Broker Architecture) e Serviços *Web* (WONG *et al*, 2002) (ALAMEH, 2002). Na opção CORBA as arquiteturas propostas para a distribuição dos serviços de SIG são distintas. O mesmo acontece na alternativa de Serviços *Web*.

DOYLE e REED (2001) e CURBERA *et al* (2001) afirmam que o serviço *Web* é uma opção melhor que CORBA com suporte a interoperabilidade.

Uma característica muito importante dos serviços é que sua tecnologia é baseada em um padrão do mercado que é o XML (CURBERA *et al*, 2001). Essa tecnologia pode conviver com todas as anteriores. Isto é uma grande vantagem, pois anteriormente não havia um grande vencedor na disputa pelo padrão de sistemas distribuídos e a mudança de paradigma era onerosa (DOYLE e REED, 2001). O consórcio OGC (Open Geospatial Consortium) também vem trabalhando na padronização da utilização dos Serviços *Web* na área de informações geográficas (OWS-2, 2004).

## 4 SERVIÇOS WEB GEOGRÁFICOS

Os sistemas antigos para conviver com novas tecnologias necessitam ser adaptados para esta realidade. O fato é que nem sempre é possível compatibilizá-los com tecnologias modernas. No caso dos SIG tem-se uma incorporação de novas tecnologias ao longo do tempo. Um exemplo disso é a tecnologia dos bancos de dados geográficos. Alguns SIG atuais já armazenam seus dados nesta nova plataforma, uma vez que eles já foram construídos com esta arquitetura. Obviamente que um SIG desenvolvido nos anos 70 terá grande dificuldade ou impossibilidade de armazenar ou consultar dados que estejam em tais bancos. Em resumo, o SIG deve ser construído de forma a incorporar avanços tecnológicos.

Como existem diversos sistemas espalhados pelo mundo onde se investiu na compra de programas, equipamentos e aquisição dos dados, freqüentemente surge a pergunta: “Como fazer para que os sistemas antigos (legados) usufruam das tecnologias novas?”

A resposta, para indagação anterior, é construir sistemas intermediários (*middleware*) que sirvam de ponte entre os sistemas antigos e as tecnologias novas (WONG *et al*, 2002) (TANIN *et al*, 2002).

Sistemas antigos não acessam diretamente tecnologias novas, o que pode ser feito é a construção de adaptadores que propiciem esse acesso. Como o acesso não é nativo, é de se esperar que existam diferenças de performances entre sistemas que utilizam diretamente as novas tecnologias daqueles que acessam por via indireta.

Outro ponto relevante é que SIG antigos ou novos enfrentam o problema da qualidade dos dados e também da qualidade do operador do sistema. A qualidade das informações geradas por um SIG está diretamente relacionada com qualidade dos dados que o alimentam. Se o dado for ruim, não tem como os sistemas gerarem informação boa, independentemente se eles utilizem tecnologias mais modernas ou não. A qualidade do operador de um SIG também é fundamental, pois mesmo tendo dados acurados e precisos é possível que ele incorra em erros graves tais como análise e/ou geração de cartas a partir dos dados espaciais e não espaciais existentes.

Serviços *Web* Geográficos são serviços que na sua produção se utilizam direta ou indiretamente de dados georeferenciados.

O paradigma de Serviços *Web* é simples e muito versátil permitindo novas formas de interação e construção de Sistemas Geográficos.

As possibilidades do uso desta tecnologia são inúmeras. Estas configurações podem ser extrapoladas e combinadas para gerar novas configurações, pois serviços são como peças de um LEGO.

Doravante o termo Serviços *Web* Geográficos será denominado apenas pela sigla SWG.

#### 4.1 Funcionalidades de Serviços WEB Geográficos

Um SWG pode conter uma ou mais funcionalidades que são as operações disponibilizadas pelo serviço.



Fig. 2 - Encadeamento de Serviços.

Na Figura 3 é mostrado outro exemplo de encadeamento de serviços. O serviço 8 tem como entrada a saída dos serviços 7 e d, os quais por sua vez têm seus dados de entrada originados de encadeamentos de outros serviços. O serviço 4 tem seus dados de entrada provenientes dos serviços 1, 2 e 3. O que se deseja mostrar nessa figura é que não há restrição na forma de encadear serviços.

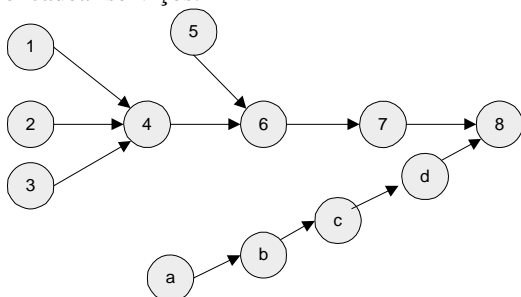


Fig. 3 - Encadeamento de Serviços

**Exemplo1:** Serviço que fornece a imagem do Recife na projeção UTM/SAD 69 no formato GEOTIFF gerada pelo *QuickBird*.

**Exemplo2:** Serviço do Departamento de Cartografia que disponibiliza as seguintes funcionalidades:

i) Dada uma posição geográfica ele informa o histórico pluviométrico bem como informações sobre composição do solo, vegetação e relevo.

ii) Fornecimento de Base cartográfica vetorial no formato DXF na escala 1:100000 do estado de Pernambuco.

iii) Geração de mapa temático de vegetação através da Classificação de imagens SPOT enviada pelo cliente.

#### 4.3 Encadeamento de Serviços

O conceito de encadeamento está relacionado ao fato do dado de entrada de um serviço ser oriundo de outro serviço.

Na Figura 2 tem-se uma representação gráfica de encadeamento de serviços. Os dados de entrada do serviço  $n$  são provenientes da saída do serviço  $n-1$ . Os dados de entrada do serviço  $n$  podem ser vistos como um fruto das transformações resultantes do encadeamento dos serviços 1 a  $n-1$ .

Para exemplificar os conceitos, supõe-se que um cliente deseja um mapa do Brasil no formato DXF, na escala de 1:50.000, projeção cônica de Lambert e Datum Córrego Alegre. Supõe-se também que só existam disponíveis três serviços: o primeiro só gera o mapa do Brasil no formato SHP, projeção UTM e Datum SAD69; o segundo efetua conversões de formatos (SHP, DXF) e o terceiro faz uma transformação de projeção de mapas de qualquer sistema para a projeção cônica de Lambert no Datum Córrego Alegre. Efetuando-se o encadeamento dos três serviços pode-se obter o resultado desejado pelo cliente.

Na Figura 4, o serviço 1 fornece o mapa desejado, porém ele está em formato e projeção diferentes do almejado pelo cliente (SIG). Utiliza-se então o serviço 2 para efetuar a mudança de formato e finalmente o

serviço 3 realiza a reprojeção.

#### 4.4 SWG e Interoperabilidade de SIG

A interoperabilidade entre SIG pode ser alcançada através do uso de SWG (Serviços Web Geográficos). Esses serviços são sistemas intermediários que

podem efetuar operações nos dados para propiciar esta integração.

Na Figura 5 observa-se que existe um SIG em cada lado. Para que haja interoperabilidade é necessário que sejam utilizados um ou mais serviços de modo que seja possível o intercâmbio dos dados entre eles.

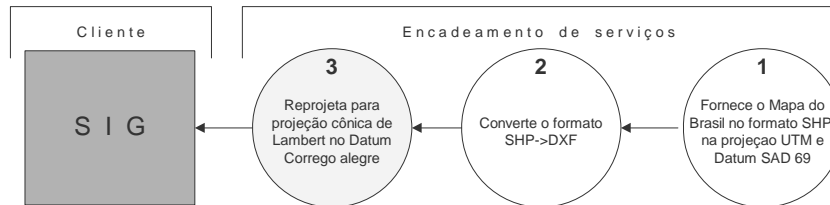


Fig. 4 - Exemplo de Encadeamento de Serviços

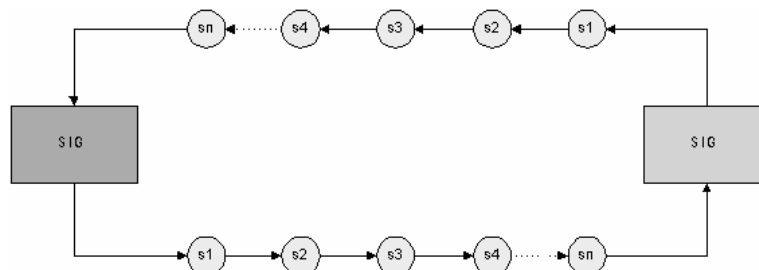


Fig. 5 – SIG e encadeamento de serviços.

O encadeamento dos serviços da Figura 5 serve como uma ponte entre os SIG. Nestes serviços podem estar presentes funcionalidades como conversão entre formatos de dados, reprojeção, *overlay* e quaisquer outras operações sobre dados georeferenciados.

**Exemplo de aplicação:** Supõe-se dois SIG comerciais: um baseado no ArcView 3.2 e outro em Auto-

desk® Map e que ambos trabalhem com cartas de igual escala, precisão e acurácia de uma mesma região. O ArcView trabalha com formato de arquivo SHP enquanto o Autodesk® Map o formato é DWG. Pode-se conceber um serviço que converta o arquivo SHP em DWG e vice-versa. A Figura 6 ilustra este exemplo, nela temos um serviço que converte os dados entre os sistemas.

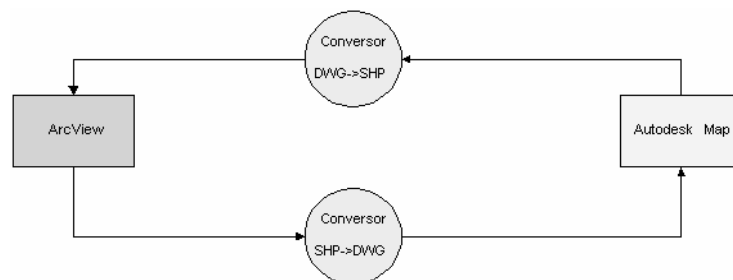


Fig. 6 - Serviço Conversor de Dados

#### 4.5 Sistemas não Geográficos e SWG

Existe uma grande demanda por informações georeferenciadas e os SWG são uma boa opção para os sistemas descritivos que não possuem funcionalidades de geoprocessamento. Entre eles estão prefeituras, concessionária de água, luz, correios e diversos serviços de atendimento ao público.

**Exemplo:** Suponha um sistema de *delivery* (entrega) onde o banco de dados contenha apenas informações descritivas dos clientes tais como nome, telefone, CEP e número do imóvel para a entrega das encomendas. Suponha também que exista um serviço geográfico capaz de gerar rotas bastando-se apenas fornecer os CEP de origem e de destino. Através da tecnologia dos serviços *Web* o sistema de *delivery* pode

se comunicar com o serviço de rotas e solicitar mapas para cada entrega que for efetuar.

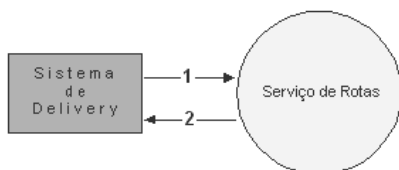


Fig. 7 - Sistema não geográfico acessando um serviço de rotas

Na Figura 8 um sistema não geográfico acessa um serviço de rotas. Quando o sistema de *delivery* necessita efetuar uma entrega ele entra em contato com o serviço (1) que gera a melhor rota para aquela situação (2).

A tecnologia dos SWG além de servir para a interoperabilidade de SIG mostra uma nova perspectiva para os sistemas geográficos. Pelo motivo de ser uma tecnologia versátil e independente de plataforma, estes serviços podem ser consumidos por qualquer sistema, independentemente se ele é um SIG ou não.

#### 4.6 SIG LEGADOS E SWG

O termo legado significa herdado ou antigo. Portanto, um SIG legado é um sistema que na sua construção não utilizou a tecnologia dos serviços *Web*.

Estes SIG antigos podem tirar proveito desta nova tecnologia através de sistemas intermediários (*middleware*). O grau de aderência vai depender fundamentalmente do seu grau de interatividade com sistemas externos.

Estes SIG legados, dependendo de sua API, podem desempenhar ou não os papéis de provedor e cliente de serviços.

#### 4.7. SIG Legado Provedor de Serviços

Embora estes sistemas tenham sido desenvolvidos com tecnologias antigas é possível que eles sejam enxergados como provedores de serviço. A técnica é desenvolver um sistema intermediário (*middleware*) para estabelecer a ponte entre o SIG e o SWG. Basicamente este sistema intermediário mapeia a API do SIG em funcionalidades do serviço a ser oferecido.

Por exemplo, supõe-se que se deseja desenvolver um SWG capaz de gerar um mapa temático das regiões de queimadas do Brasil. Supõe-se também que esta funcionalidade já tenha sido desenvolvida em *avenue* (API do ArcView). A partir destes dados é possível disponibilizar a funcionalidade presente neste *script* do ArcView como um SWG. O primeiro passo seria desenvolver um sistema intermediário capaz de chamar o ArcView e fazer com que ele rodasse o *script* da geração do mapa e armazenasse o resultado em um

diretório pré-determinado. O segundo passo seria disponibilizar este sistema intermediário como um SWG. Toda vez que um cliente necessite do mapa das queimadas do Brasil ele entra em contato com o SWG. E ativa o *script* do ArcView para gerar o mapa que seria então repassado ao cliente.

Na Figura 8 O cliente solicita o mapa ao serviço (1). Este por sua vez ativa o *script avenue* do ArcView que gera o mapa das queimadas (2). O serviço busca o mapa em um diretório previamente estabelecido pelo *script* do *avenue* (3). No passo seguinte (4) o cliente recebe o mapa do serviço.



Fig. 8 - SIG Legado fornecendo serviço

#### 4.7.1 SIG Legado Cliente de Serviços

No item anterior foi exemplificado o papel do SIG como provedor de serviço, porém ele também pode desempenhar o papel de cliente. Nas duas situações esta interação é feita de forma indireta, ou seja, através de sistemas intermediários.

Um exemplo de um SIG desempenhando o papel de cliente seria novamente o ArcView 3.2. Supõe-se que a Prefeitura trabalhe como o ArcView e necessite de uma carta que está disponível em um serviço *Web* geográfico da concessionária de água. a Prefeitura teria que desenvolver um *script avenue* para ativar um programa externo que teria como função entrar em contato com o serviço que disponibiliza o mapa, para consumir o serviço da concessionária de água. Esse programa receberia a carta desejada e a colocaria em um diretório, previamente acertado. Feito isto o ArcView 3.2 poderia acessar a carta normalmente.

Na Figura 9 o ArcView através de um *script avenue* acessa um sistema intermediário (1). Este por sua vez solicita a carta desejada a concessionária de água (2). Ele entrega a carta ao sistema intermediário (3). Finalmente o sistema intermediário armazena a carta em diretório, previamente acordado, para que o ArcView possa acessá-la (4).

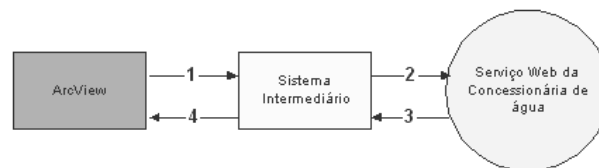


Fig. 9 - SIG Legado acessando um serviço de concessionária de água.

## 5. IMPLEMENTAÇÃO E RESULTADOS



A primeira fase foi a elaboração do serviço. O objetivo foi criar um serviço simples, pois se desejava acompanhar todas as etapas do processo. Optou-se por desenvolver um serviço que disponibilizasse um mapa de uma região previamente escolhida. Ele conteria informações vetoriais e matriciais do local escolhido.

A teoria diz que se pode construir um serviço utilizando-se sistemas legados. Surgiu então a idéia de aproveitar um produto já disponível no mercado. O publicador de mapas *MapServer* (MAPSERVER, 2004) foi escolhido como ponto de partida na construção do serviço. O motivo da escolha foi o fato dele ser gratuito e seguir algumas das especificações do consórcio OGC, conhecidas como *OpenGis*® (OGC, 2004).

WMS (*Web MAP Server*) *OpenGis*® para a publicação de mapas na Internet. Ele é o mais básico dos modelos e encontra-se disponível na maioria dos publicadores de mapas, inclusive no *MapServer*. O serviço desenvolvido utilizou o padrão WMS. Nele, através do comando GETMAP, pode-se gerar o mapa da região desejada.

Com a configuração e instalação do programa *MapServer* que tem-se o mapa desejado. Porém o publicador de mapas foi desenvolvido para gerar o mapa e enviar para um cliente http (*browser*, por exemplo). O objetivo é criar um serviço para que seja consumido por outros programas. A fim de viabilizar esse intento foi desenvolvido um programa para trabalhar como intermediário. Ele é enxergado pelo mundo externo como

um serviço. Na realidade ele repassa os pedidos dos clientes para o *MapServer* e retorna os mapas para eles. O cliente só enxerga a interface do serviço que diz o que o sistema faz e não como ele foi implementado. Este processo é análogo ao termo “*information hiding*” que é empregado em engenharia de software orientada a objetos (BOGGS e BOGGS 1999).

O primeiro passo foi colocar o serviço no ar e depois foram concebidos dois clientes em arquiteturas distintas (Delphi e Java) para verificar na prática se os mapas chegavam a eles de forma transparente, ou seja, se os pressupostos de encapsulamento e independência de plataformas estavam sendo obedecidos.

Na Figura 10 tem-se um pequeno resumo da aplicação. Um serviço será criado e disponibilizará a geração de mapas. Dois clientes concebidos em plataformas tecnológicas distintas irão consumir o serviço (mapa de uma região) utilizando o protocolo http. Com a implementação do aplicativo será possível a observação de um ciclo completo da tecnologia que vai desde a produção do serviço até o seu consumo.

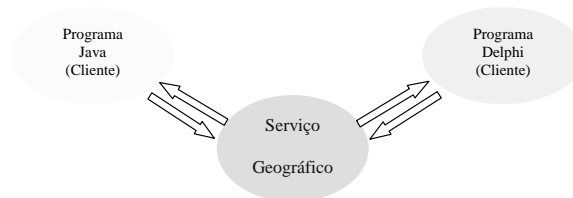


Fig. 10 - Aplicação de um Serviço Geográfico.

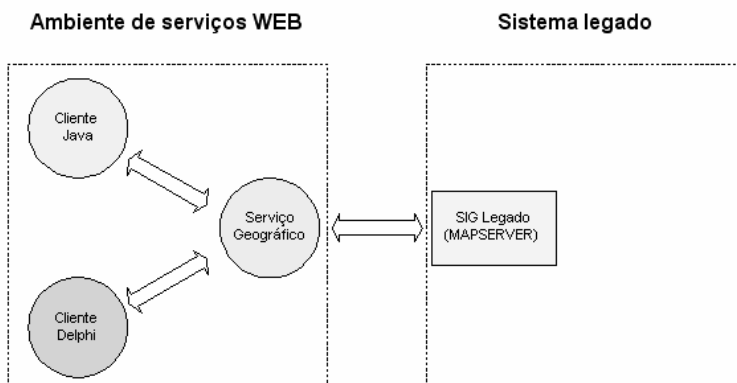


Fig. 11 - Visão do sistema implementado

Na Figura 11 se tem outra visão do aplicativo desenvolvido. De um lado está um sistema antigo (legado) e do outro existem clientes que necessitam do mapa que ele gera. Funcionando como uma ponte entre estes dois mundos existe um serviço geográfico. Ele recebe os pedidos dos clientes e se comunica com o servidor de mapas para prover o mapa solicitado.

Este aplicativo representa um caso particular da arquitetura vista anteriormente no SIG Legado Provedor de Serviços onde um SIG concebido em uma tecnologia antiga disponibiliza suas funcionalidades através de um serviço geográfico.

### 5.1 Aspectos Cartográficos do projeto

A elaboração de um serviço geográfico tem como premissa uma sólida formação em cartografia devido às peculiaridades dos dados geográficos (Datum, projeção, escala, etc.).

Os conceitos de Datum, projeção e sistema de coordenadas foram fundamentais para o projeto. Eles foram aplicados na configuração dos servidores de mapas. No aplicativo utilizaram-se dados vetoriais e matriciais que estavam no mesmo sistema de projeção e Datum. É importante frisar que mesmo que os dados não estivessem no mesmo sistema, o servidor de mapas

poderia fazer uma transformação de projeção de forma a compatibilizá-los. Claro que isto naturalmente exigiria um pequeno passo a mais na configuração do servidor.

Observe as seguintes linhas presentes em uma das configurações dos servidores de mapas utilizadas no projeto.

```
# Definição da Projeção#
PROJECTION
"init=EPSG:29185"      # Significa Datum SAD69
Projeção UTM zona 25S
END

# Definição da extensão do mapa em coordenadas
UTM
EXTENT 290662 9102017 291768 9102783
```

Este pequeno extrato da configuração de um servidor de mapas mostra o papel fundamental dos conceitos de cartografia para o desenvolvimento do projeto. O serviço geográfico foi construído a partir dele. Portanto, o sucesso deste serviço depende da correta aplicação e domínio dos conceitos cartográficos ao lidar com servidores de mapas.

### 5.2 Histórico do desenvolvimento

Inicialmente, com o objetivo de acelerar o desenvolvimento do aplicativo, tudo foi simulado em um só computador. Depois de testado, procedeu-se à instalação do aplicativo nos computadores do departamento de Cartografia para simulação da distribuição dos sistemas.

Durante a construção do aplicativo e após estudar melhor o padrão WMS optou-se por um aprimoramento do serviço. O mapa a ser oferecido aos clientes é construído através de um processo de *overlay* de mapas. No primeiro servidor teríamos dados vetoriais e no segundo dados matriciais. A tecnologia permite que quaisquer servidores “construam” seus mapas com níveis de informação oriundos de outros servidores. Este processo é denominado de encadeamento de servidores de mapas e faz parte do padrão WMS. Na Figura 12 temos o funcionamento de um servidor que gera mapas com dados próprios.

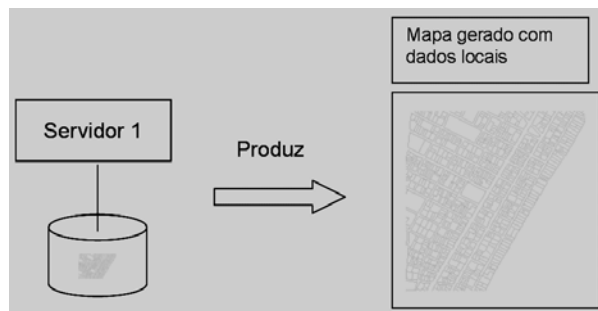


Fig. 12 - Servidor Vetorial

Na Figura 13 o dado imagem é armazenado no Servidor 2 (servidor matricial).

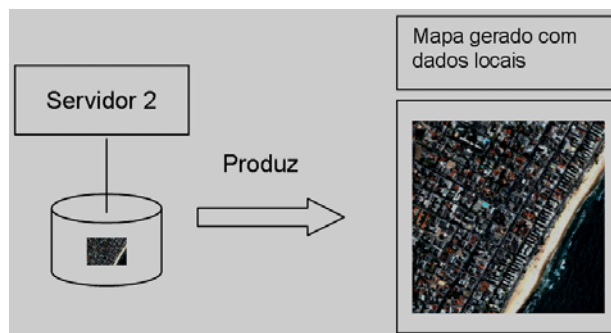


Fig. 13 - Servidor Matricial

O processo do encadeamento pode ser observado na Figura 14. O Servidor 2 para produzir um mapa necessita de um *layer* que está no Servidor 1. Ele requisita ao servidor 1 este *layer* e efetua o *overlay* com o dado armazenado localmente. Já a Figura 15 mostra o resumo do *overlay* dos mapas

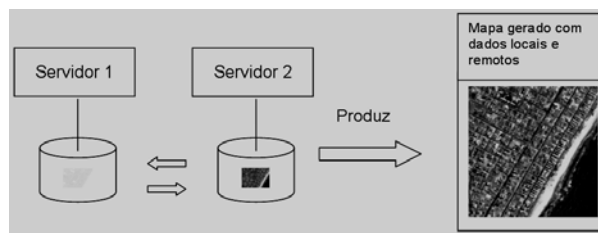


Fig. 14 - Encadeamento de Servidores.

O processo de implementação do projeto passou por várias etapas, a saber: Instalação e configuração dos servidores http e de mapas; Criação do serviço através de um aplicativo que comandava o servidor de mapas e elaboração de clientes para o serviço.

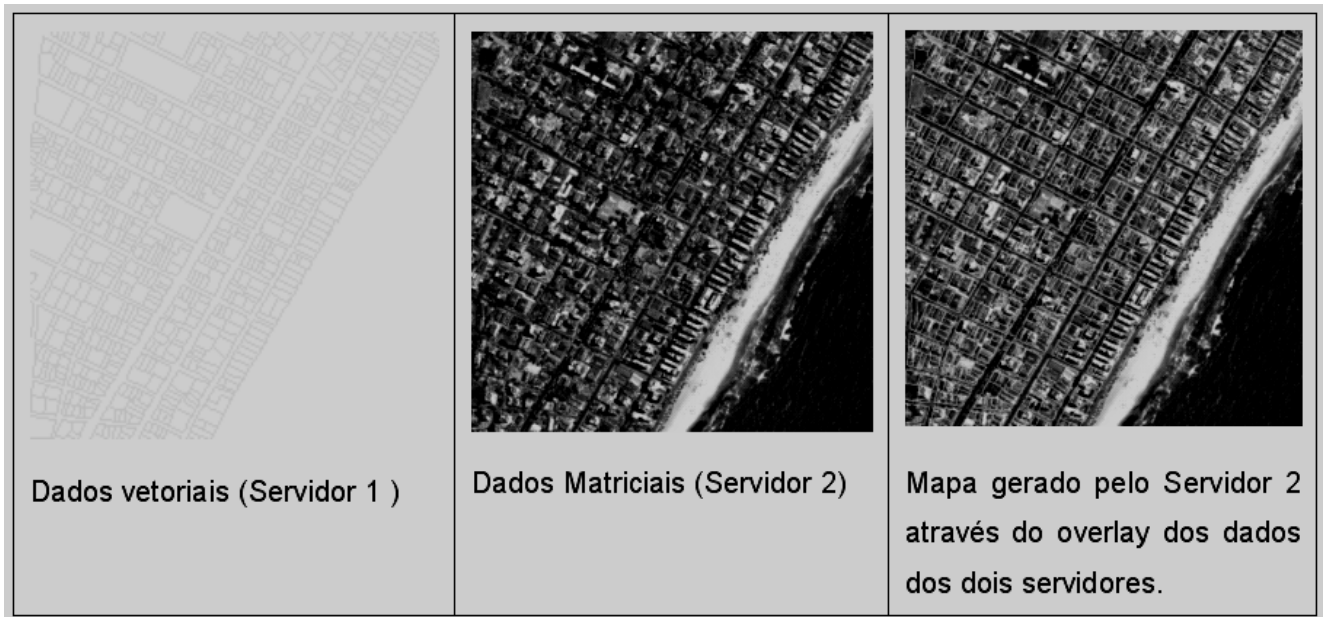


Fig. 15 - Resumo do *Overlay* dos Mapas

### 5.3 Servidor de Mapas

O servidor de Mapas escolhido foi o *MapServer*. O primeiro passo foi a instalação e configuração de um servidor http. Utilizou-se o Apache, pois é gratuito e roda em vários sistemas operacionais (APACHE, 2004).

Foram realizados os procedimentos padrões com a finalidade de instalar o *MapServer* e fazer com que ele trabalhe em conjunto com o Apache. Os detalhes do procedimento de instalação estão no site da Universidade de Minnesota (<http://mapserver.gis.umn.edu/>).

### 5.4 Esquema de funcionamento dos servidores

Foram concebidos dois servidores que conteriam informações vetoriais e matriciais do bairro de Boa Viagem da cidade do Recife. O de número 2 é responsável pela informação vetorial (lotes) e o de número 1 armazena a informação matricial (imagem de satélite). O Servidor 1 é o responsável por gerar um mapa de overlay dos lotes e da imagem de satélite.

O Servidor 1 foi instalado em outra máquina que utilizava o Windows XP® como sistema operacional e o Servidor 2 foi montado em uma máquina que rodava o sistema operacional linux. As especificações das máquinas estão no anexo II. Os servidores foram instalados em sistemas operacionais diferentes para enriquecer o aplicativo e começar a demonstrar as vantagens de se aderir a padrões abertos.

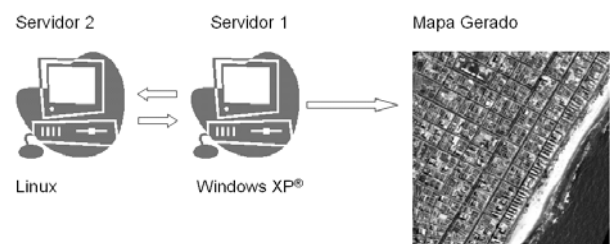


Fig. 14 - Esquema do funcionamento dos servidores

### 5.5 Interface para o Serviço

O mapa é agora oferecido aos clientes. O próximo passo é gerar um serviço *Web*. Foi desenvolvido um programa em Java que seria enxergado pelo mundo externo como um serviço que gera o mapa de Boa Viagem-PE. Na realidade ela serve apenas de interface, pois o mapa é gerado no servidor de mapas.

Quando um cliente necessita do mapa de Boa Viagem-PE ele se comunica com o programa JAVA e este por sua vez comanda o Servidor 1 para que gere o mapa. Posteriormente o cliente recebe o mapa gerado (Figura 15).



Fig. 15 - Visão geral do processo

Na implementação do programa este processo foi implementado através da seguinte configuração (Figura 16).

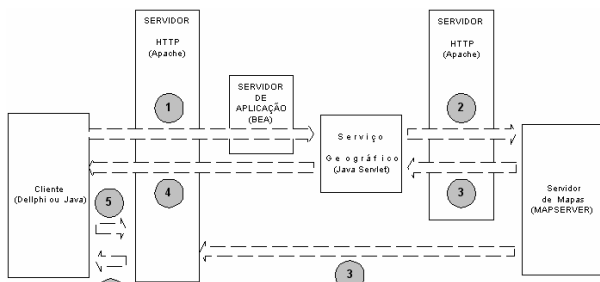


Fig. 16 - Funcionamento do processo

**Passo1:** O cliente solicita ao serviço geográfico, a localização (URL) do mapa.

**Passo2:** O serviço envia o pedido de geração de mapa ao servidor de mapas (MAPSERVER).

**Passo3:** O servidor gera o mapa e o remete ao servidor http, simultaneamente envia a localização (URL) deste mapa ao serviço.

**Passo4:** O Serviço envia a localização do mapa ao cliente.

**Passo5:** De posse da localização do mapa, o cliente solicita o mapa ao servidor HTTP.

**Passo6:** O cliente recebe o mapa.

Algumas considerações sobre a Figura 16: O cliente, o serviço e o servidor de mapas ao se comunicarem, necessariamente utilizam o servidor HTTP. O Servidor HTTP foi duplicado na ilustração com o objetivo de explicar melhor o funcionamento do processo. Como o serviço é um *servlet* então a comunicação entre cliente e o serviço envolve necessariamente o servidor servidor de aplicação (BEA)

No paradigma de serviços *Web* os aplicativos comunicam-se através do protocolo SOAP (Simple Object Access Protocol) que é baseado em XML, ou seja, a transferência de dados é feita através de texto. Se o serviço fosse transmitir o mapa (formato binário) para o cliente ele necessitaria de uma implementação onde a imagem fosse um “anexo” da mensagem texto que ele enviaria ao cliente. Na implementação do projeto decidiu-se que mensagens trocadas entre o serviço e os clientes seriam só textuais e a transferência do mapa (formato binário) ficaria a cargo do servidor HTTP.

### 5.5.1 Implementação

No paradigma de Java para se criar um serviço é necessário se desenvolver um componente (*bean*). Antes de criar o componente foi construída uma classe básica para servir de interface com o servidor de mapas. Esta classe foi denominada Mapa.

A classe Mapa tem um único método `getMapURL()`. Ele é responsável por gerar a URL do mapa confeccionado pelo MapServer.

```
public class Mapa {
    private String mapURL=null;
    public Mapa() {
    };
    public String getMapURL(){
```

```
        mapURL="http://127.0.0.1/cgi-
bin/mapserv?map=/web/dti/demo.map&VERSION=1.1.
0&REQUEST=GetMap";
        return this.mapURL;
    }
}
```

A parte importante a ser observada no código é <http://127.0.0.1/cgi-bin/mapserv?map=/web/dti/demo.map&VERSION=1.1.0&REQUEST=GetMap>. Isto é o comando para o MapServer gerar o mapa. “<http://127.0.0.1/cgi-bin/mapserv>” é o endereço do programa MapServer. “[map=/web/dti/demo.map](http://127.0.0.1/cgi-bin/mapserv)” é o parâmetro que indica qual o arquivo de configuração do mapa a ser gerado. “[VERSION=1.1.0](http://127.0.0.1/cgi-bin/mapserv)” é a versão do padrão WMS. “[REQUEST=GetMap](http://127.0.0.1/cgi-bin/mapserv)” é o comando para gerar o mapa. O segundo passo foi a criação de um componente denominado de MapaBean.

```
public class MapaBean implements java.io.Serializable{
    public MapaBean() {
    }
    public String getMapURL(){
        Mapa map=null;
        try{
            map = new Mapa();
        }
        catch(Exception e) {
            e.printStackTrace();
        };
        return map.getMapURL();
    }
}
```

No desenvolvimento do aplicativo utilizamos o programa Jbuilder 9 da Borland. Ele possui uma opção de gerar automaticamente o documento WSDL (Web Services Description Language) e hospedar o serviço em um servidor de aplicações. O servidor escolhido foi da empresa BEA *Weblogic* que gentilmente forneceu uma licença para o desenvolvimento do projeto.

Do documento WSDL gerado extraiu-se o seguinte:

```
<portType name="MapaBeanPort">
<operation name="getMapURL">
<soap:address location="http://localhost:7001/myweb/MapaBean">
```

Através de uma rápida olhada acima notamos que o serviço tem uma operação “`getMapURL`” e está disponível no endereço <http://localhost:7001/myweb/MapaBean>.

### 5.6 Clientes

Foram desenvolvidos dois clientes. Um utilizando Java (ferramenta Jbuilder) e o outro Pascal (ferramenta delphi). A criação deles é bastante similar onde

o primeiro passo é examinar o documento WSDL do serviço a ser consumido.

As descrições dos serviços devem ser armazenadas nos agentes (*brokers*). Atualmente existe um catálogo mundial de serviços disponíveis que se encontra espelhado nos servidores da IBM e Microsoft. No caso particular deste aplicativo não há a necessidade de registrar o serviço, pois os clientes e o provedor do serviço estão sendo desenvolvidos conjuntamente.

### 5.6.1 Cliente Java

Através da ferramenta Jbuilder® é possível se gerar classes em Java para consumir serviços descritos em um documento WSDL.

O aplicativo é simples. Ele se apresenta com uma tela contendo dois componentes: Um botão e um rótulo (*label*).

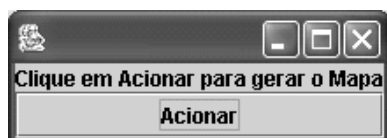


Fig. 17 - Cliente Java

Após o usuário clicar no botão “Acionar”, o serviço remoto é acionado e o mapa é apresentado.

A parte do código responsável pela geração do mapa pelo evento do clique no botão é o seguinte:

```
void jButton1_actionPerformed(ActionEvent e) {  
  
    try {  
        java.net.URL mapaURL=new java.net.URL(this.getMapURL());  
  
        this.jLabel1.setIcon(new ImageIcon(mapaURL));  
  
        this.jLabel1.setText("");  
        this.pack();  
        this.repaint();  
  
    }  
    catch (Exception ex) {  
        ex.printStackTrace();  
    }  
}
```

Quando se clica no botão, o rótulo `jLabel1` recebe o mapa gerado pelo método `getMapURL()`. Nota-se que poucas linhas de programação são necessárias para invocar serviços remotos.



Fig. 18 - Mapa no Cliente Java

### 5.6.2 Cliente Pascal

Foi desenvolvido um aplicativo similar ao anterior só que em outra linguagem. A linguagem escolhida foi Pascal orientado a objeto. O programa foi concebido através da ferramenta Borland Delphi®.

O aplicativo tem uma tela composta de um botão e um componente de imagem. Quando o usuário clica o botão o componente imagem recebe o mapa do serviço *Web* (Figura 19).



Fig. 19 - Cliente Delphi

Após o usuário clicar no botão “Ativar o Mapa”, o serviço remoto é acionado e o mapa é apresentado.

O código responsável pela ativação do serviço quando se clica o botão da Figura 19 é:

```
procedure TForm1.Button1Click(Sender: TObject);  
var servico :MapaBeanPort;  
begin  
    servico:=GetMapaBeanPort;  
    UrlToImage(servico.getMapURL,mapa);  
    Form1.Refresh;
```

end;



Fig. 20 - Mapa no cliente Delphi

## 6 CONCLUSÃO

Os SIG inicialmente eram ilhas de informação que se baseavam em tecnologias proprietárias e desenvolvidos de forma autônoma. Não havia a preocupação com importação/exportação de dados nem com a comunicação entre as aplicações. Mais tarde observou-se a necessidade de compartilhamento de dados entre os SIGs e isto motivou a criação de novas tecnologias que dessem este tipo de suporte. Os Serviços *Web* propõe um paradigma que facilita a construção de SIG interoperáveis. Estes serviços são independentes de plataformas e linguagens de programação.

A tecnologia dos serviços *Web* mostrou-se um paradigma simples e ao mesmo tempo poderoso na solução do problema da interoperabilidade.

A literatura afirma que é possível a interoperabilidade de SIG heterogêneos e distribuídos através de serviços *Web*. Comprovou-se através de um experimento que é factível e viável a utilização desta tecnologia

Um ambiente heterogêneo foi implantado com sucesso. Nele conviveram e interagiram diferentes sistemas operacionais e linguagens de programação. Isto foi possível devido à tecnologia dos serviços *Web* estar baseada na troca de informações através de textos (arquivos XML) o que facilita a interação entre sistemas com tecnologias distintas.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ALAMEH, Nadine *Service Chaining of Interoperable Geographic Information Web Services*, <http://citeseer.nj.nec.com/536832.html> 2002 acessado em março de 2004.
- APACHE <http://httpd.apache.org/> acessado em janeiro de 2004.
- ALBUQUERQUE, Fernando *TCP/IP- Internet Programação de Sistemas Distribuídos HTML, JavaScript e Java*, Axl Books do Brasil Editora, 2001
- BOGGS, Wendy; BOGGS, Michael Boggs *Mastering Uml With Rational Rose*, Sybex International, August 1999.
- CHAPPELL, David; JEWELL, Tyler, *Java Web Services*, O'Reilly First Edition March 2002 ISBN: 0-596-00269-6, 276 pages
- CURBERA, Francisco ; NAGY, William A. ; WEERAWARANA, Sanjiva, *Web Services: Why and How*, IBM T.J. Watson Research Center, August 9, 2001
- DOYLE, Allan; REED, Carl *Introduction to OGC Web Services* eds. 2001. <http://ip.opengis.org/ows/index.html> acessado em agosto de 2004.
- EGENHOFER M. J., FEGEAS R., GOODCHILD M.F., 1997 *Interoperating GISs Report* of a Specialist Meeting Held under the Auspices of the Zarenius Project, Panel on Computational Implementations of Geographic Concepts December 5-6, 1997, Santa Barbara, California <http://www.ncgia.ucsb.edu/conf/interop97/report.html> acessado em agosto de 2003.
- EPSG (European Petroleum Survey Group) Geodesy Parameters <http://www.ihsenergy.com/epsg/geodetic2.html> acessado em maio de 2004.
- FALLSIDE, David C.; WALMSLEY, Priscilla, *XML Schema Part 0: Primer Second Edition* W3C Recommendation 28 October 2004 <http://www.w3.org/TR/2004/REC-xmlschema-0-20041028/#Intro> (acessado em novembro de 2004)
- FONSECA, F. T., *Ontology-Driven Geographic Information Systems* PhD Thesis 2001
- IEEE – [WWW.IEEE.ORG](http://WWW.IEEE.ORG) acessado em março de 2004
- MAPSERVER <http://mapserver.gis.umn.edu/> acessado em janeiro de 2004
- MELO JR, Jonas Bezerra de; CANDEIAS, A. L. B., *Interoperabilidade de SIG através de WEB SERVICES*, . In: I Simpósio de Ciências Geodésicas e Tecnologias da Geoinformação, 2004, Recife. I Simpósio de Ciências Geodésicas e Tecnologias da Geoinformação. Recife: DECART-UFPE, 2004. p. 1-7.

- MELO JR, Jonas Bezerra de; CANDEIAS, A. L. B., *SIG e sua Interoperabilidade Utilizando Servidores de WEB*, XII Simpósio Brasileiro de Sensoriamento Remoto - XII SBSR 2005 (no prelo).
- NAGY, W. A.; WEERAWARANA S., *Web Services: Why and How*, Francisco Curbera, IBM T.J. Watson Research Center, August 9, 2001
- OGC, Open Geospatial Consortium <http://www.opengeospatial.org> acessado em março de 2004.
- OWS-2, *OGC Web Services Initiative Phase 2* <http://ip.opengis.org/ows2/> acessado em maio de 2004.
- TANIN, Egemen; BRABEC, František; SAMET, Hanan *Remote access to large spatial databases* Proceedings of the tenth ACM international symposium on Advances in geographic information systems 2002, McLean, Virginia, USA
- TSALGATIDOU A., PILIOURA, T., *An Overview of Standards and Related Technology in Web Services*, International Journal of Distributed and Parallel Databases, Special Issue on E-Services, 12(2): 135-162; Sep 2002.
- TSOU, Ming-Hsiang, *A dynamic Architecture for distributing Geographic Information Services on the Internet*, Phd Thesis 2001.
- TSOU, Ming-Hsiang; BUTTENFIELD, Barbara P., *A Dynamic Architecture for Distributing Geographic Information Services*, Transactions in GIS. 2002, 6(4): 355-381.
- WANG, Fangju *A Distributed Geographic Information System on the Common Object Request Broker Architecture (CORBA)*, GeoInformatica Volume 4, Issue 1, Mar 2000 Pages: 89-115.
- WONG, S.H.; SWARTZ, S.L.; SARKAR, D.; *A Middleware Architecture for Open and Interoperable GISs*, Multimedia, IEEE, Volume: 9, Issue: 2, April-June 2002 Pages: 62 – 76.
- WMS (Web Map Server) [http://portal.opengis.org/files/?artifact\\_id=5316](http://portal.opengis.org/files/?artifact_id=5316) acessado em setembro de 2004.
- W3C (World Wide Web Consortium) *Web Services Framework for W3C Workshop on Web Services* 11-12 April 2001, San Jose, CA USA <http://www.w3.org/2001/03/WSWS-pop/paper51> (acessado em maio de 2003)
- W3C (World Wide Web Consortium) <http://www.w3.org/> (acessado em maio de 2003)
- W3C (World Wide Web Consortium) *WSDL (Web Services Description Language)* [http://www.w3.org/TR/wsdl#\\_introduction](http://www.w3.org/TR/wsdl#_introduction) (2001) (acessado em maio de 2003)