



UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA – CCET
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE

DAVID CAVASSANA COSTA

TRANSMISSÃO PROGRESSIVA DE MAPAS BASEADA EM GIS
SERVICES

São Luís
2006

DAVID CAVASSANA COSTA

TRANSMISSÃO PROGRESSIVA DE MAPAS BASEADA EM GIS SERVICES

Dissertação apresentada ao curso de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão como parte dos requisitos para a obtenção do título de Mestre em Engenharia Elétrica na área de Ciência da Computação.

Orientador: Prof. Dr. Anselmo Cardoso de Paiva

Co-Orientador Prof. Dr. Mario Meireles Teixeira

São Luís
2006

DAVID CAVASSANA COSTA

TRANSMISSÃO PROGRESSIVA DE MAPAS BASEADA EM GIS SERVICES

Dissertação apresentada ao curso de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão como parte dos requisitos para a obtenção do título de Mestre em Engenharia Elétrica na área de Ciência da Computação.

Aprovada em: ___ / ___ / _____

BANCA EXAMINADORA

Prof. Dr. Anselmo Cardoso de Paiva
Universidade Federal do Maranhão – UFMA
(Orientador)

Prof. Dr. Mário Antônio Meireles Teixeira
Universidade Federal do Maranhão – UFMA
(Co-Orientador)

Prof. Dr. Marcelo Gattass
Pontifícia Universidade Católica do Rio de Janeiro – PUC – RJ

Prof. Dr. Aristófares Corrêa Silva
Universidade Federal do Maranhão – UFMA

A minha mãe.

AGRADECIMENTOS

No desenvolvimento e conclusão desse trabalho, nada seria possível sem a ajuda e a iluminação de Deus, principalmente com as idéias e soluções, mesmo nas horas mais inusitadas. Além Dele, outras pessoas foram importantes para a conclusão e principalmente ao dar forças para que não desistisse no meio do caminho, sabendo entender a falta de tempo e atenção tirados por esse trabalho.

Minha mãe Olímpia e meu pai José Cícero, que, mesmo sem entender a fundo o que envolve a área de informática e os pormenores no desenvolvimento desse trabalho, não me deixaram desistir quando fui tentado e já não agüentava mais, sempre me apoiando, dando forças e todas as condições necessárias, direta ou indiretamente, para que chegasse até o final.

Minha namorada Francislene, que me incentivou a começar, continuar e terminar esse trabalho e, principalmente, soube me entender nas vezes em que estava física e psicologicamente sem condições de lhe corresponder e acompanhar e que abriu mão de muita coisa em prol desse objetivo.

Meu orientador Anselmo, pela oportunidade que me foi dada e pela escolha do tema. Pelas idéias, nem sempre seguidas de início, e pelo suporte que me deu desde o início do trabalho e principalmente pela ajuda na escrita dos artigos e dissertação, nos quais teve papel fundamental.

Meu orientador Mário, que mesmo entrando no segundo tempo, deu um novo fôlego ao time e foi importante com idéias e sugestões ao trabalho, sendo também fundamental nas escritas dos artigos.

Meus amigos André, Bysmarck, Carlos Eduardo, Daniel, João Carlos, Mark Renato, Robertson, Valdeci, com os quais pude passar grandes momentos, mesmo que pelo MSN, durante os últimos anos.

Meus irmãos e toda minha família, que também souberam entender a falta de tempo e atenção e me incentivaram a terminar este trabalho.

O pessoal do NTI, que deu todo o apoio e condições para que pudesse desenvolver o trabalho da melhor maneira possível, e principalmente por terem me dado tempo para que pudesse concluir este trabalho, além de compreender os momentos em que não pude estar presente.

Todas as pessoas que, direta ou indiretamente, contribuíram para o desenvolvimento e conclusão deste trabalho.

RESUMO

A Internet criou um ambiente adequado para o compartilhamento de dados geoespaciais, permitindo aos usuários a transferência, visualização, manipulação e interação com esses dados. Esse ambiente não apenas fornece novas oportunidades na utilização dos dados espaciais, como também introduz novos problemas que devem ser resolvidos a fim de possibilitar a utilização eficiente e efetiva desses conjuntos de dados. Um desses problemas está relacionado com a utilização dos dados espaciais em ambientes com baixa velocidade de transmissão, o que, devido ao grande volume dos dados espaciais, torna-se um empecilho para a disponibilização desses dados na Internet. Este trabalho apresenta um algoritmo para transmissão progressiva de mapas vetoriais na Web, que utiliza técnicas baseadas na resolução do dispositivo cliente com o objetivo de reduzir a quantidade de dados transmitidos, enviando os dados aos poucos, aumentando gradativamente o nível de detalhes do mapa. São combinadas técnicas de transmissão progressiva, simplificação e transmissão dependente de escala, a fim de maximizar o desempenho do sistema no ambiente Web. Ademais, apresenta-se uma arquitetura baseada em serviços Web que implementa a transmissão progressiva de mapas e possibilita uma maior flexibilidade no uso desta funcionalidade e na construção de clientes em diferentes plataformas.

Palavras-chave: Sistemas de Informação Geográfica, transmissão progressiva, GIS Services, transmissão eficiente de dados, mapas vetoriais.

ABSTRACT

The Internet has created an good environment for the sharing of geospatial data. In this environment users can transfer, visualize, manipulate and interact with these data sets. This environment not only provides new opportunities for the utilization of geospatial data, but also introduces new problems that should be addressed in order to provide an efficient and effective use of these datasets. One of such problems is related to the use of these spatial datasets in low transfer rate environments, what often becomes an impediment for the publication of those data in the Internet, given the huge volume of data to be transmitted. This paper presents a progressive transmission algorithm for vector maps on the Web, which uses techniques based on the resolution of the client device with the aim of reducing the amount of transmitted data, sending the data incrementally, thus gradually increasing the level of map details. Scale-dependent transmission techniques are combined with simplification and progressive ones, in order to maximize the overall system performance in a Web GIS environment. Furthermore, an architecture based on Web services is presented, that implements the progressive transmission of maps and allows a greater flexibility in the use of this functionality and in the development of clients in different computing platforms.

Keywords: Geographic Information Systems, progressive transmission, GIS Services, efficient data transmission, vector maps.

SUMÁRIO

LISTA DE FIGURAS	11
LISTA DE TABELAS	13
LISTA DE SIGLAS.....	14
1 INTRODUÇÃO	12
2 CONCEITOS BÁSICOS	18
2.1 Transmissão de Dados Geográficos	22
2.2 Escala Cartográfica	23
2.3 Generalização	24
2.4 Multiresolução e Multirepresentação	26
2.5 Transmissão Progressiva	28
2.6 Representação de Mapas	30
3 TRABALHOS RELACIONADOS	33
3.1 Armazenamento	33
3.2 Transmissão de dados na Web	36
4 FERRAMENTAS UTILIZADAS	47
4.1 O <i>Framework</i> iGIS	47
4.1.1 Camada de Apresentação	48
4.1.2 Camada de Dados	49
4.1.3 Camada de Aplicação	50
4.1.4 Vantagens do <i>Framework</i>	51
4.2 Web Services	53
4.3 GIS Services	56
5 ARQUITETURA DE TRANSMISSÃO PROGRESSIVA	59
5.1 Algoritmo de Transmissão Progressiva	59

5.2	Implementação do algoritmo no iGIS.....	71
5.3	Aplicação de Referência.....	77
5.4	GIS Service de Transmissão Progressiva de Mapas.....	79
5.4.1	Arquitetura Geral	79
5.4.2	Interface do <i>GIS Service</i>.....	81
5.4.3	Aplicação de Referência.....	83
6	AVALIAÇÃO DE DESEMPENHO	85
6.1	Ambiente de testes	85
6.2	Avaliação do tamanho dos mapas gerados	87
6.3	Avaliação dos clientes desktop e web.....	95
7	CONCLUSÃO.....	103
	REFERÊNCIAS.....	105

LISTA DE FIGURAS

Figura 4.1. Arquitetura geral do iGIS (MIRANDA, 2002).	48
Figura 4.2. Arquitetura geral da tecnologia de Web Services.	55
Figura 5.1 Algoritmo de geração do mapa-base e incrementos.	64
Figura 5.2. Mapa exemplo, na resolução mais detalhada.	66
Figura 5.3. Decomposição em mapa base e incrementos.	67
Figura 5.4. Reconstituição da Versão 2 do mapa.....	68
Figura 5.5. Reconstituição da Versão 3 do mapa.....	68
Figura 5.6. Exemplo: Pontos de um elemento em alta resolução	69
Figura 5.7. Decomposição inicial do mapa (mapa-base)	69
Figura 5.8. Incrementos da Versão 2 (Pontos 2, 7 e 11)	70
Figura 5.9. Incrementos da Versão 3 (Pontos 3, 4 e 12).....	70
Figura 5.10. Incrementos da Versão 4 (Pontos 4, 6 e 10).....	70
Figura 5.11. Extensão de Classes Progressivas a partir das classes iGIS.	72
Figura 5.12. Classes formatadoras de geometrias.....	72
Figura 5.13. Classes que processam as geometrias progressivamente.	73
Figura 5.14. Cenário de geração de mapas progressivos.	74
Figura 5.15. Cenário de utilização dos mapas progressivos	76
Figura 5.16. Arquitetura do iGIS com módulo de Transmissão Progressiva.	77
Figura 5.17. Cliente Web do iGIS.....	78
Figura 5.18. Arquitetura de GIS Services Proposta.....	79
Figura 5.19. Cliente desktop implementado em Java.....	84
Figura 6.1. Mapas Brasil1, Brasil2 e Brasil3.....	86
Figura 6.2. Gráfico comparativo para o mapa Brasil1	90
Figura 6.3. Gráfico comparativo para o mapa Brasil2	91

Figura 6.4. Gráfico comparativo para o mapa Brasil3	92
Figura 6.5. Taxas de Compressão dos mapas.....	95
Figura 6.6. Tempo de transmissão para o mapa Brasil2, a 25KB/s.	101

LISTA DE TABELAS

Tabela 6.1. Valores comparativos para o mapa Brasil1	89
Tabela 6.2. Valores comparativos para o mapa Brasil2	91
Tabela 6.3. Valores comparativos para o mapa Brasil3	92
Tabela 6.4. Tamanho dos mapas e respectivos tempos de transmissão	96
Tabela 6.5. Tempos de processamento entre os clientes (mapa Brasil1)	97
Tabela 6.6. Tempos de processamento entre os clientes (mapa Brasil2)	98
Tabela 6.7. Tempos de processamento entre os clientes (mapa Brasil3)	98
Tabela 6.8. Tempos de resposta para o mapa Brasil1	99
Tabela 6.9. Tempos de resposta para o mapa Brasil2	100
Tabela 6.10. Tempos de resposta para o mapa Brasil3	100

LISTA DE SIGLAS

CGM	– Computer Graphics Metafile
DAG	– Directed Acyclic Graph
DOM	– Document Object Model
DPI	– Dots Per Inch – Pontos Por Polegada
GIF	– Graphics Interchange Format
GIS	– Geographic Information Systems
GML	– Geography Markup Language
HTML	– Hypertext Markup Language
HTTP	– Hypertext Transfer Protocol
J2EE	– Java 2 Enterprise Edition
JPEG	– Joint Photographic Experts Group
JSP	– Java Server Pages
MVC	– Model View Controller
OGC	– Open GeoSpatial Consortium
PNG	– Portable Network Graphics
SGBD	– Sistema Gerenciador de Banco de Dados
SOA	– Service – Oriented Architecture
SOAP	– Simple Object Access Protocol
SQL	– Structured Query Language
SVG	– Scalable Vector Graphics
TWF	– TeCGraf Web Format
UDDI	– Universal Description, Discovery and Integration
URI	– Uniform Resource Identifier
W3C	– World Wide Web Consortium
WSDL	– Web Services Description Language
XML	– eXtensible Markup Language

1 INTRODUÇÃO

A visualização desempenha um importante papel para um melhor entendimento de fenômenos nas mais diversas áreas, e com os dados geográficos não é diferente (Cecconi, 2003). Os mapas geográficos são utilizados há muito tempo para a visualização de dados espaciais, ajudando-nos a entender melhor os relacionamentos desses dados. Com a crescente utilização dos Sistemas de Informações Geográfica (GIS¹), tem crescido o número de mapas criados para serem analisados e manipulados.

Devido à crescente demanda para a visualização de dados geográficos, pesquisas vêm sendo realizadas a fim de formalizar o processo de visualização, definindo regras e princípios, pois, na cartografia, diferentes tipos de dados podem ser visualizados em diferentes formas, e para todas as possibilidades de visualização, métodos e técnicas devem ser encontrados de maneira a otimizar ao máximo o processo.

Os mapas podem ser definidos como representações gráficas do nosso ambiente. Os mapas Web também são mapas, porém são visualizados em navegadores Web, sujeitos, portanto, a restrições do próprio navegador e das redes de comunicação a que são submetidos para alcançarem seu destino. Essas restrições implicam em considerações especiais no projeto e implementação por parte dos desenvolvedores, responsáveis pela disponibilização do mapa nesse ambiente.

Muitos princípios foram propostos e testados de maneira a formalizar mapas geográficos impressos em papel, como (Bertin, 1983) e (Dent, 1990), porém,

¹ GIS: sigla em inglês referente a Geographic Information Systems (Sistemas de Informação Geográfica)

alguns requisitos podem mudar para a visualização de tais mapas em meios multimídia e na Web, por meio de telas com as mais diversas características.

Com o desenvolvimento do *software* (aplicações GIS) e *hardware* (melhores resoluções e representações de cores, além do crescente avanço na área de dispositivos móveis) a demanda por mapas digitais tem crescido enormemente. Assim, novos requisitos para tais tipos de mapas devem ser definidos. Os limites de representação de dados na tela, por exemplo, são intrínsecos da tecnologia, como resolução da tela e de cores ou o próprio tamanho da tela (van Elzakker, 2001).

Outra razão para o crescimento do uso de mapas digitais tem sido a possibilidade de interações do usuário com o mapa, e a facilidade de criação de mapas digitais temporários, manipuláveis pelo usuário, permitindo também a visualização de fenômenos dinâmicos e a mudança da informação ou do tema apresentado em um mapa de uma mesma área, de acordo com necessidades do usuário.

Com o crescimento da Internet e do uso de PCs nas duas últimas décadas, os mapas digitais e em especial os mapas na Web ganharam um novo significado.

Com o avanço das tecnologias de dispositivos móveis e a popularização de tais dispositivos no mercado, em especial os PDAs, foi adicionada a vantagem da flexibilidade aos mapas digitais, antes restritos ao uso com PCs, unido assim as vantagens dos mapas de papel aos mapas digitais. Tudo isso torna possível a cada usuário, individualmente, criar e manipular o mapa sob-demanda, de acordo com suas necessidades, a qualquer lugar e hora (Reichenbacher, 2001).

Independente do meio de publicação do mapa ser de papel ou digital, a adaptação do conteúdo do mapa será fortemente determinada pela sua escala.

Mapas de escala menor contêm informações menos detalhadas que mapas de escala maior, para uma mesma região. O processo de reduzir a quantidade de dados e ajustar a informação para uma determinada escala é chamado de generalização cartográfica (Weibel, 1999). Esse processo abstrai e reduz informações reais seguindo especificações cartográficas, e mantém as características significantes da área do mapa. Tal processo é bastante complexo e pode consumir bastante tempo, dependendo do volume do mapa.

Com o crescimento da multimídia e Internet, a velocidade de entrega da informação se tornou ainda mais importante, tornando fundamental o papel da generalização dos mapas na Internet (Feringa, 2001). A generalização, assim, deve ser incluída aos novos requisitos do processo de criação de mapas (Kraak, 2001).

Atualmente, a Internet tem se tornado um grande meio de publicação de conteúdo, sendo assim um ambiente favorável para os usuários de GIS trocarem dados, conduzirem análises e apresentarem resultados geográficos (ESRI, 2000). A informação geográfica na Internet tem evoluído rapidamente com o avanço das tecnologias Web (Arcnews, 1997), (ESRI, 1997), (Peng, 1997), (Cho, 1998).

Cria-se assim um novo ambiente de compartilhamento de dados geoespaciais, onde fornecedores de dados tornam suas bases geográficas disponíveis, para que os usuários possam utilizar a Web para visualização, análise e/ou manipulação (Bertolotto, 1999). Enquanto tal configuração provê novas oportunidades de utilização, tanto para domínios públicos como comerciais, de conjuntos de dados geográficos, novos problemas emergem de tal arquitetura, como a disponibilidade de grandes quantidades de dados armazenados em repositórios sem estrutura de transferência eficiente (Flewelling, 1999).

Mesmo com o crescente acesso a uma maior velocidade de transmissão, existem sempre situações onde o custo de transmissão é fator crítico, como no caso de redes sem fio, dispositivos móveis ou mesmo de acesso discado. Essas características adversas têm feito surgir vários métodos para a transmissão eficiente dos dados, necessária para a diminuição do tempo de resposta em sistemas com velocidade de transmissão considerável e tornando possível a própria publicação de informações geográficas em formato vetorial em sistemas com baixas taxas de transmissão.

Existem dois formatos básicos de dados para informações geográficas. O formato matricial provê uma maior eficiência, devido principalmente ao tamanho constante dos arquivos que independem da complexidade dos elementos do mapa. Há também o formato vetorial, que apesar de possuir um volume maior, possibilita a construção de aplicações muito mais ricas, utilizando técnicas de processamento geométrico e topológico (Buttenfield, 2002).

Uma vez que os dados vetoriais tendem a ser bastante volumosos na maioria dos bancos de dados geográficos, o processo de transferência pode gerar grande espera pelo usuário, ou até mesmo se tornar proibitivo. Para resolver esses problemas, técnicas como transmissão progressiva e generalização têm sido desenvolvidas e utilizadas. Tais técnicas vêm sendo propostas, avaliadas e combinadas nos últimos tempos com o objetivo de amenizar o problema da transferência de mapas geográficos vetoriais na Internet. A complexidade das técnicas e algoritmos que se destinam a solucionar tal problema é a maior motivação para a pesquisa nessa área, pois cada abordagem possui um enfoque e escopo diferentes umas das outras.

Este trabalho propõe um método diferenciado de transmissão progressiva de mapas em relação aos métodos tradicionais utilizados atualmente. Nele, o mapa é transmitido ao cliente em níveis crescentes de resolução, de acordo com as necessidades do cliente de obterem dados mais refinados. A principal característica do método proposto é a transmissão apenas dos dados que puderem ser visualizados pelo cliente em seu dispositivo, evitando o envio de dados supérfluos que não contribuiriam para o aprimoramento visual do mapa, economizando dessa forma a quantidade de dados transmitidos. Tal abordagem difere de outras técnicas de transmissão progressiva, que desconsideram a resolução de exibição do mapa. Difere também de técnicas de transmissão dependente de resolução, que não apresentam características progressivas de transmissão, enviando apenas o mapa adequado para uma resolução específica.

É realizada uma combinação das técnicas de generalização, adaptação de conteúdo e transmissão progressiva, a fim de tornar mais eficiente o processo de transmissão de mapas do servidor ao cliente, causando um mínimo impacto na consistência dos dados e aumentando a usabilidade da aplicação, com a diminuição do tempo de resposta experimentado pelo usuário.

O método de transmissão progressiva é, portanto, a principal contribuição desse trabalho. Entretanto, apresenta-se ainda uma investigação das arquiteturas propostas na literatura para transmissão progressiva de mapas, além de apresentar um serviço de mapas (*GIS Service*) com suporte a transmissão progressiva de mapas utilizando o método desenvolvido. Foram também implementados clientes de forma que se possa testar e avaliar sua funcionalidade e flexibilidade.

O presente trabalho encontra-se organizado em mais seis capítulos descritos a seguir. O segundo capítulo apresenta alguns conceitos importantes

relacionados aos GIS. O terceiro capítulo apresenta alguns trabalhos relacionados ao tema de armazenamento e transmissão eficiente de dados geográficos na Internet. Em seguida, no capítulo quatro, apresenta-se as tecnologias utilizadas para a implementação da arquitetura e algoritmo propostos. O capítulo cinco apresenta o trabalho desenvolvido, onde são detalhados a arquitetura de transmissão progressiva proposta, o algoritmo implementado e o serviço web de transmissão progressiva de mapas desenvolvido. O capítulo seis apresenta alguns experimentos realizados a fim de avaliar a flexibilidade da arquitetura proposta e seu desempenho. Finalmente, no capítulo sete, serão apresentadas as considerações finais a respeito do trabalho, destacando a contribuição e possíveis trabalhos futuros a serem desenvolvidos a partir deste.

2 CONCEITOS BÁSICOS

GIS (Sistemas de Informação Geográfica) são ferramentas largamente utilizadas na coleta, gerenciamento e apresentação de informações sobre o mundo em que vivemos. São utilizados desde áreas científicas, em aplicações de coleta e análise de dados ambientais, até áreas de administração, na manutenção de registros de propriedades e controle de veículos de emergência (Frank, 1994).

Os GIS gerenciam dados relacionados à localização espacial e os dados são apresentados graficamente como um mapa. Bancos de dados espaciais são utilizados para armazenar os dados desses sistemas, disponibilizando funcionalidade das mais diversas aplicações, nas mais diversas áreas.

De acordo com Burrough (Burrough, 1986), GIS é definido como um conjunto de ferramentas que contém módulos de aquisição, armazenamento, análise, transformação, manutenção e visualização de objetos e fenômenos espaciais do mundo real.

Por outro lado, Cowen (Cowen,1988) enfatiza o aspecto da integração, afirmando que um GIS é melhor caracterizado como um sistema de suporte a decisão que envolve a integração de dados espacialmente referenciados em um ambiente de resolução de problemas. As definições deixam claro que o modelo de dados desempenha papel fundamental e que os sistemas atuais são resultados de esforços combinados de diversas áreas.

A principal vantagem dos GIS e seus mapas eletrônicos sobre os mapas de papel é a capacidade de o usuário final interagir com o mapa, além da facilidade de manutenção e disponibilização de dados espaciais em diversos meios eletrônicos (Cecconi, 2003).

A Web é o mais recente meio para apresentar e disseminar dados geoespaciais, nas mais diversas áreas. Isso ocorre principalmente devido às características de acessibilidade e disponibilidade propiciadas por esse novo ambiente. Diante desse quadro, os mapas desempenham um papel fundamental, podendo exercer desde funções tradicionais, como possibilitar a interpretação sobre padrões e relacionamentos geoespaciais, sendo usados da mesma forma que seriam no caso de Atlas de papel, até função de interface ou índice para informações adicionais, podendo ligar locais específicos do mapa a fotos, texto, sons ou mesmo outros mapas, num ambiente multimídia com grande riqueza de interação (Kraak, 2001).

O GIS baseado na Internet, que denominaremos Internet GIS, é uma ferramenta orientada a rede que usa a Internet como principal meio para acesso e transmissão de dados, possibilitando a análise e visualização de dados geográficos.

Os mapas podem ser definidos como representações gráficas do nosso ambiente. Na Web, esses mapas são visualizados em navegadores Web, sujeitos, portanto, a restrições do próprio navegador e das redes de comunicação a que são submetidos para alcançarem seu destino. Essas restrições implicam em considerações especiais no projeto e implementação por parte dos desenvolvedores, responsáveis pela publicação do mapa nesse ambiente.

A Web cria um novo ambiente para os mapas, que não são mais usados apenas em sua funcionalidade natural, de exibição de locais, exploração de padrões espaciais e navegação sobre o espaço geográfico. Novos papéis emergem desse ambiente, como na busca e disseminação desses dados (Couclelis, 1998). A informação na Web é virtualmente independente de plataforma, incomparável em sua capacidade de atingir grande quantidade de usuários com custos mínimos e

facilidade de atualização dos dados. Além disso, possibilita uma disseminação interativa e dinâmica dos dados espaciais, oferecendo novas técnicas de mapeamento e uso de possibilidades não observadas em mapas impressos tradicionais, como a integração multimídia.

Em relação à sua disponibilização na Web, os mapas podem ser classificados em duas categorias: mapas estáticos e mapas dinâmicos. Essas categorias são ainda subdivididas de acordo com sua funcionalidade, em mapas usados apenas para visualização e mapas com suporte a interação e manipulação do usuário. Os termos estáticos e dinâmicos se referem à tecnologia de geração de conteúdo do mapa.

O mapa estático usado apenas para visualização é o tipo mais comum encontrado na Web, geralmente originados de produtos cartográficos e disponibilizados na Web no formato de imagens matriciais.

Mapas estáticos interativos podem funcionar como interface a outros recursos na Web, inclusive outros mapas. Podem fornecer a possibilidade de *zoom*² e *pan*³ sobre o mapa pelo usuário, seleção de subconjuntos de dados do mapa (camadas) a serem exibidos ou ocultados, e variação nas características visuais dos mapas, como cores, simbologias e transparências.

Os mapas dinâmicos incorporam as capacidades de visualização ou interação e permitem interações e consultas mais complexas sobre os dados, utilizando funcionalidades disponíveis apenas no servidor, como por exemplo, consultas espaciais sobre os dados geográficos armazenados no bancos de dados.

² *Zoom* é uma operação de aproximação e detalhamento de dados exibidos na tela do computador.

³ *Pan* é a operação de translação do mapa, movendo-o sobre a tela e alterando a área sendo visualizada.

Alguns problemas e limitações dos mapas Web não são exclusivos de tal ambiente, sendo conseqüências do meio computacional a que estão sujeitos e à própria natureza cartográfica, como a portabilidade limitada e certa dificuldade em sua manipulação.

Uma aplicação Internet GIS deve responder às requisições dos clientes de maneira eficiente, com baixo tempo de resposta e alta usabilidade. Assim, o principal objetivo dessas aplicações é ser simples o suficiente para usuários sem grandes conhecimentos específicos, sem perder sua funcionalidade, permitindo aos usuários não-especialistas tomarem decisões e conclusões diante dos dados (Torun, 2000).

Peterson (Peterson, 2001) observa que a Internet modificou significativamente o uso dos mapas, a forma que eles são feitos e como são distribuídos. Os mapas digitais podem ser personalizados, facilmente atualizados e são interativos. Porém não possuem a principal vantagem dos mapas de papel tradicionais: a mobilidade.

Para suprir essa necessidade, a cartografia móvel combina as principais vantagens dos GIS tradicionais com a flexibilidade e acessibilidade dos dispositivos móveis, valendo-se de seu crescimento constante no mercado, assim como aconteceu com o PC há alguns anos. A informação apresentada é individualizada para cada usuário e o conteúdo é gerado dinamicamente, sendo dessa forma flexível para qualquer informação e localização. Tal configuração de GIS em dispositivos móveis dá a possibilidade ao usuário de criar seu próprio mapa sob demanda, em qualquer lugar e a qualquer momento (Cecconi, 2003).

Este novo ambiente apresenta diversas vantagens, principalmente com relação à distribuição e disponibilização de dados geográficos, nas mais diversas

áreas de aplicação, contribuindo assim para a sua popularização. Junto a isso, porém, emergem novos problemas e são lançados novos desafios com relação à disponibilização eficiente dos dados geográficos, geralmente bastante volumosos, o que dificulta seu uso em ambientes de rede com baixa taxa de transmissão, principalmente no emergente mercado dos dispositivos móveis.

2.1 Transmissão de Dados Geográficos

Mesmo que técnicas de compressão auxiliem muito a entrega de conteúdo de maneira mais otimizada, diminuindo o tamanho dos dados em dezenas de vezes, o tamanho dos dados geográficos ainda é grande após a compressão (Tu, 2001). Considerando ainda a evolução na capacidade de memória, processamento e velocidade de transmissão dos sistemas computacionais, esses requisitos serão sempre um limitante, de certo modo, pois o conjunto dos dados a serem analisados e processados está em constante crescimento. Conforme Sayood (Sayood, 1996), pode-se afirmar que a necessidade de armazenamento e transmissão de dados se expande pelo menos duas vezes mais rápido que o crescimento dos recursos disponíveis.

Assim, pode-se constatar que sempre vai haver limitação na transmissão de dados, uma vez que sua demanda cresce conforme as limitações a que estão sujeitos. Como uma analogia à Primeira Lei de Parkinson⁴, pode-se verificar que o tamanho dos mapas expande-se até ocupar toda a largura de banda disponível. Quando houver mais largura de banda, surgirá um mapa ainda maior a ser compartilhado. A pesquisa na eficiência de transmissão de dados geográficos se

⁴ Primeira Lei de Parkinson: “O trabalho se expande de forma a preencher todo o tempo disponível para sua realização”.

justifica, portanto, mesmo em face às crescentes evoluções e inovações tecnológicas.

O problema da produção de arquivos pequenos que representem os mapas digitalmente está, de certa forma, relacionado com o conceito de generalização da Cartografia, em que versões simplificadas de representações gráficas dos objetos são produzidas. Puppo e Dettori (Puppo, 1995) apresentam um resumo das técnicas de generalização cartográfica, formalizando matematicamente as operações que transformam as representações do mapa original em representações mais simples e eficientes, a serem utilizadas em comunicações visuais. A generalização de linhas poligonais é definida como simplificação, sendo uma das técnicas mais utilizadas em sistemas GIS.

As linhas poligonais são geralmente simplificadas de acordo com alguns critérios geométricos. Tais critérios utilizam uma tolerância normalmente baseada na escala do mapa que se deseja obter após o processo de simplificação. O objetivo desses critérios é produzir a mesma representação visual com um número reduzido de pontos poligonais, mantendo as características perceptíveis dos objetos simplificados e baseado na escala do mapa.

2.2 Escala Cartográfica

Escala Cartográfica é um termo que diz respeito ao tamanho relativo ou absoluto de um mapa, e envolve um conjunto fundamental de conceitos na geografia. A escala se preocupa principalmente com o espaço na geografia, sendo também denominada escala espacial. Ela se refere ao tamanho descrito de um elemento no mapa relativo ao seu tamanho no mundo real. Assim, escalas grandes

significam um nível mais preciso de detalhes, enquanto escalas pequenas mostram os mesmos dados geográficos com um nível menor de detalhamento.

Um mapa é exibido de acordo com uma determinada escala. Em diferentes escalas, a mesma informação é geralmente exibida de maneira diferente, do ponto de vista de seu conteúdo, e não apenas tendo seu tamanho ampliado ou reduzido. Segundo Spaccapietra (Spaccapietra, 2000), isso ocorre em virtude de fatores como:

- A exibição de convenções que podem mudar de uma escala para outra
- Elementos que podem aparecer ou desaparecer ou serem agregados/desagregados devido o seu tamanho poder se tornar visível ou invisível de acordo com a escala adotada.
- Formas de elementos visíveis podem ser modificadas (tornando-se mais simples ou mais precisas)
- A informação simplesmente não é disponível na escala requisitada.

2.3 Generalização

A generalização, no contexto dos GIS, pode ser definida como o processo de geração de um conjunto de dados derivados mais importantes e significativos, geralmente de propriedades geométricas e temáticas menos complexas, comparadas aos dados originais. Ela ocorre com a mudança da visualização de mapas de uma escala maior para escalas menores (Torun, 2000).

Os métodos de generalização automatizada são semelhantes aos manuais, e uma lista detalhada de operadores de generalização pode ser encontrada em (Ruas, 1993), incluindo: simplificação, seleção/eliminação,

agregação, estruturação, compressão, alisamento, exageração, caricaturização, dilatação e deslocamento.

A simplificação de linhas é um processo de generalização que pode ser definido como a eliminação de detalhes não desejados. Este processo pode ser concebido selecionando um subconjunto de pontos que melhor represente as propriedades geométricas de uma linha, eliminando os pontos restantes (Bernon, 2005).

Os algoritmos de simplificação de linhas podem filtrar e suavizar os dados, eliminando os vértices desnecessários e mantendo a qualidade de sua aparência gráfica. Outro objetivo da simplificação é a geração de uma nova versão da linha, mais adequada para a representação do mesmo fenômeno geográfico, em uma escala menor que a original, ocorrendo, nesse caso, uma generalização da linha (McMaster, 1992). A simplificação organiza os vértices de maneira a tornar possível a produção de versões generalizadas adequadas para uma escala específica, sem introduzir dados redundantes (van Oosterom, 1995) (Davis Jr, 2005).

A eliminação dos pontos deve seguir restrições geográficas de aproximação entre as duas representações, de maneira a manter uma boa representação geométrica. Essas restrições podem ser definidas de várias maneiras (Li, 1992), mas é em geral alguma medida da proximidade geométrica entre as poligonais, tais como o máximo deslocamento perpendicular permitido (mais utilizada) ou o mínimo deslocamento angular permitido. Um problema eventualmente abordado na literatura é a escolha do parâmetro de tolerância, e sua correlação com a escala da representação simplificada. Um critério interessante para a determinação da tolerância é o que usa o tamanho do menor objeto visível em uma determinada escala.

Um dos algoritmos mais populares de Simplificação de Linhas foi proposto por Douglas e Peucker (Douglas, 1973), o qual é muito eficiente e amplamente aceito pela comunidade, pois suas simplificações produzidas são muito parecidas com as soluções subjetivas produzidas manualmente por especialistas, no passado.

Esses critérios, projetados originalmente para o processo de impressão de mapas em papel, não levam em conta a natureza escalável das representações vetoriais em navegadores Web (pelo menos na maneira como foram concebidos). Nesse novo ambiente, o tamanho da superfície de visualização, da janela na tela do monitor, não é tão controlado como o tamanho do mapa em uma impressora. Diferentes monitores podem ter diferentes tamanhos e capacidades. Além disso, se capacidades de ampliação (*zoom*) são oferecidas ao usuário, este pode perceber diferenças entre as versões do mapa de acordo com o nível de ampliação, por exemplo, certo objeto passa a ser exibido ao se alcançar um maior nível de detalhes.

Existem várias abordagens para os critérios de simplificação em ambiente Web. É senso comum que os dispositivos de visualização possuem uma resolução definida, que limita a escala máxima de visualização dos dados. Uma entidade no mapa pode ser generalizada se sua geometria é representada por coordenadas inteiras, isto é, o critério de simplificação é definido baseado na resolução do dispositivo de visualização.

2.4 Multiresolução e Multirepresentação

Novos conceitos são introduzidos nesse novo ambiente de mapas digitais, como multiresolução, multiescala, multirepresentação.

Os termos *multiresolução e multiescala*, se referem ao mesmo mapa, com os mesmos elementos e mesma topologia, porém com suas linhas e polígonos

representado em níveis de detalhamento diferentes, por meio de algoritmos de simplificação, e os objetos mantêm a mesma representação geométrica entre as várias versões. É um mesmo mapa, contendo os mesmo elementos, sendo visualizado em níveis de resoluções diferentes.

A resolução é geralmente definida como a dimensão geométrica mínima que um objeto deve ter a fim de ser visualizado. A resolução de uma informação no banco de dados é ou a resolução que foi utilizada no processo de aquisição de dados ou o resultado de um processo de generalização cartográfica. Se diferentes resoluções são associadas a um mesmo mapa, tem-se o conceito de multiresolução (Spaccapietra, 2000).

O termo *multirepresentação*, entretanto, refere-se ao fato de que, em cada resolução, o mapa é formado por um conjunto de representações de cada elemento, e um mesmo elemento pode, em níveis diferentes do mapa, ser representado de várias maneiras, em diferentes dimensões, ou até mesmo não ser representado. Por exemplo, pode ser exibido como um ponto em uma representação e como um polígono em outra. Nesse caso, dependendo do nível de detalhes, os mesmos objetos espaciais podem ter, em diferentes níveis de detalhes do mapa, diferentes dimensões, podendo ser representados por pontos, linhas, polígonos ou combinações desses elementos.

Uma *resolução* menos precisa significa que os dados contêm simplificações do mapa original, porém, a topologia não deve ser alterada. Por outro lado, a redução dos detalhes do mapa por meio de *alteração nas representações* dos objetos do mapa pode mudar as estruturas topológicas dos objetos espaciais, assim como sua forma.

2.5 Transmissão Progressiva

Associada a essas técnicas de simplificação e representação dos mapas, que visam tornar o processo de mapeamento na Web mais eficiente, encontra-se a técnica de transmissão progressiva.

Codificar um mapa para transmissão progressiva significa organizá-lo de tal forma que porções do mapa parcialmente transmitidas possam ser visualizadas e progressivamente refinadas assim que a informação é recebida. Dessa forma, o cliente não precisa esperar que todo o mapa seja transmitido antes de ser exibido, fornecendo uma resposta mais imediata ao usuário. Esse método organiza o mapa em um mapa-base, de baixa resolução, responsável pela visualização inicial, e uma seqüência de incrementos, responsáveis pela incorporação de detalhes ao mapa-base, incrementando sua resolução por meio da adição de novas informações a ele. Dessa forma, o mapa tem sua resolução aumentada gradativamente, até que seja alcançada a maior resolução, quando todos os incrementos forem incorporados ao mapa-base.

O mapa-base é construído por meio de técnicas de simplificação de poligonais, com a remoção de vértices, que são agrupados e separados, formando os detalhes do mapa. Esses vértices são, posteriormente, enviados ao cliente, e incorporados ao mapa-base, aumentando sua resolução (To, 1999).

Na transmissão progressiva de dados matriciais, detalhes da imagem são gradativamente preenchidos, permitindo ao usuário obter uma primeira impressão visual da imagem, antes que todos os dados sejam recebidos. No caso de dados vetoriais, os usuários são capazes de realizar operações sobre os dados (além de visualizá-los), mesmo em um nível de detalhe e precisão mais baixo. Entretanto, o

usuário tem a vantagem de não precisar receber os dados em sua totalidade, a fim de iniciar o processo de análise e manipulação. A transmissão progressiva vetorial objetiva não só dar uma “primeira impressão” sobre os dados, mas também entregar dados com fidelidade suficiente para que sua manipulação, mesmo em níveis baixos de resolução, produza resultados consistentes e próximos da realidade. A idéia da transmissão progressiva é reduzir a quantidade de dados enviados ao cliente inicialmente, permitindo-o, porém, realizar tarefas úteis, além da visualização dos dados (Yang, 2004).

O caso matricial é relativamente simples, pois apenas é necessário adicionar *pixels* a uma imagem incompleta a fim de torná-la mais refinada. O caso vetorial, entretanto, é um pouco mais complexo, devido à criação de representações e estruturas de dados hierárquicas adequadas em vários níveis de detalhes não ser um processo trivial.

Alguns dos desafios da transmissão progressiva de dados vetoriais são a manutenção da consistência topológica e a redução da complexidade computacional, de maneira a permitir a simplificação dos mapas em tempo real, bem como a integração dos incrementos ao mapa-base em um tempo razoável de processamento, para que o uso da técnica de transmissão progressiva seja justificado e não se torne o responsável pelo baixo desempenho (Bertolotto, 2001).

A transmissão progressiva pode ser vista em duas diferentes perspectivas. De um lado, o servidor, que necessita de métodos para construir, manipular e transmitir uma seqüência de representações do mapa em diferentes níveis de detalhes, progressivamente, e de outro lado, o cliente, que deve ser capaz de atualizar o mapa integrando os detalhes recebidos, reconstruindo o mapa

correspondente à resolução do detalhe, para em seguida visualizá-lo (Bertolotto, 1999).

2.6 Representação de Mapas

Os dados espaciais em sistemas GIS possuem dois componentes principais: dados geométricos e dados topológicos. Dados geométricos têm natureza quantitativa e são usados para representarem coordenadas e equações de linha, ou seja, representar o mapa em algum formato. Dados topológicos descrevem os relacionamentos entre dados geométricos, tais como conectividade, adjacência, inclusão, entre outros, e permitem a construção de consultas sobre dados espaciais, tais como quais áreas são vizinhas de outras, quais polígonos são conectados e formam uma rede rodoviária, ou quais lagos existem em uma cidade. Os dados topológicos geralmente não são armazenados explicitamente, pois podem ser derivados de dados geográficos.

Os mapas utilizados em GIS são modelos abstratos de fenômenos geográficos reais, auxiliando na análise de informação topológica e métrica. Existem dois meios de representar esses mapas: em formato matricial ou vetorial (Han, 2004).

Dados matriciais organizam sua informação em uma matriz de *pixels*. Cada elemento dessa matriz especifica um valor numérico descrevendo a cor do respectivo *pixel* que será apresentado no dispositivo de saída.

Os dados vetoriais são representados por meio de objetos individuais, definidos por pontos georeferenciados por meio de linhas, ao invés de *pixels*. Os pontos são representados como pares de coordenadas. Pontos ligados uns aos outros formam linhas e são armazenados como uma lista, contendo as coordenadas

do início ao fim. Formas geométricas são linhas onde o ponto inicial está na mesma coordenada do ponto final. Eles representam informação por meio de primitivas geométricas, como linhas, pontos, polígonos, curvas, são chamados tipos geométricos primitivos e podem ainda serem combinados entre si, formando tipos compostos. Para cada geometria ou conjunto de geometrias, há comumente anexado um conjunto de atributos, tais como texto, cor, largura da linha (Sallem, 2003).

Embora as duas abordagens a princípio sejam válidas e possam co-existir, na prática, análise de alta precisão posicional e estruturas topológicas precisas não são alcançadas no formato matricial, desestruturado. As representações estruturadas vetoriais possuem vantagens com relação à riqueza de funcionalidade, processamento e interação do usuário com o mapa, por exemplo, permitindo consulta sobre adjacências e outros relacionamentos topológicos, além de um armazenamento e gerenciamento mais eficiente, em bancos de dados espaciais. Com o rápido crescimento das aplicações Internet GIS, cresceu também a necessidade de acessar esses dados vetoriais em diferentes níveis de detalhes para navegação e análise de informações espaciais, uma vez que esses dados, embora apresentem as vantagens apresentadas, possuem um tamanho muito maior, comparado aos dados matriciais (Han, 2004).

É interessante notar o processo de digitalização⁵ do mapa, em ambos os formatos. No formato vetorial, os dados são digitalizados no cliente para exibição na tela, formada por *pixels*, num processo conhecido como rasterização, calculando os valores das coordenadas em uma posição correspondente, no formato matricial. Já no caso de imagens matriciais, estas são rasterizadas no servidor, e enviadas já

⁵ Conversão da informação do mapa para a representação matricial conforme uma determinada resolução.

rasterizadas ao cliente, exibindo-a na tela do usuário. As imagens matriciais são diretamente dependentes da resolução, onde a resolução se refere ao número de *pixels* em uma imagem, sendo geralmente expressa em pontos por polegada (DPI). O conceito de resolução também é aplicado à tela do dispositivo de exibição do mapa (Waldén, 2003).

Inicialmente, em virtude das tecnologias de visualização e dos algoritmos matriciais serem mais simples, a maioria dos mapas digitais, em especial dos Internet GIS, eram disponibilizados em formato matricial. Porém, com o avanço tecnológico, popularização de formatos vetoriais com suporte dos navegadores (SVG), novos algoritmos e técnicas para transmissão eficiente de mapas vetoriais, além das limitações dos dados matriciais, houve uma crescente demanda por dados em formato vetorial na Internet, e os sistemas GIS atualmente vêm sendo amplamente desenvolvidos tendo esse formato como base de seus mapas.

3 TRABALHOS RELACIONADOS

Visando tratar a adaptação dos mapas aos ambientes de transmissão com limitações de recursos, de maneira a torná-la o mais eficiente possível, diversas estratégias são utilizadas para o armazenamento dos dados espaciais e sua transmissão, de modo a tratar a natureza multiescala dos dados geográficos e considerando as características peculiares desses dados. Nesse capítulo, são apresentados algumas dessas estratégias, apresentando inicialmente os trabalhos relacionados ao armazenamento de dados com suporte a multiescala, e em seguida, trabalhos com o enfoque na transmissão desses dados espaciais, com o uso de técnicas e conceitos que visam otimizar a publicação desses dados na Internet.

3.1 Armazenamento

Com a necessidade de armazenamento de dados espaciais em diferentes níveis de detalhes, a fim de satisfazer diferentes demandas de diversos usuários, conforme relatado no capítulo anterior, foi desenvolvido o conceito de bancos de dados multiescala, a fim de garantir a consistência entre as diversas representações de mapas e reduzir a carga global de atualização.

Uma abordagem para o armazenamento de mapas em vários níveis de detalhes pode ser o armazenamento de cada representação do mapa separadamente, para cada nível de detalhes. Porém essa abordagem torna difícil a manutenção e a manipulação de estruturas de maneira consistente entre as várias versões do mapa. Uma solução alternativa é a aplicação de métodos hierárquicos, capazes de armazenar objetos espaciais em diferentes níveis de detalhes,

mantendo uma conexão entre as várias representações, o que torna o processo de manipulação das versões do mapa mais simplificado.

Algumas dessas estruturas são as Árvores de Particionamento Binário (van Oosterom, 1993), a Gap-Tree (Árvore de Particionamento de Área Generalizado) (van Oosterom, 1995), as Árvores Reativas (van Oosterom, 1993), o Grafo Acíclico Direto (DAG – Directed Acyclic Graph) (Timpf, 1997), e as Árvores-R Multi-Escala (Chan, 2002).

Zhou e Jones (Zhou, 2001) apresentam um estudo sobre vários conceitos de projeto e implementação de bancos de dados espaciais multiescalares, como codificação de vértices em intervalos de escala, suporte a recorte para geometrias multiescalares, estratégias para reconstrução de geometrias em *cache* no cliente e relacionamentos entre a indexação sobre a escala e sobre o espaço geométrico, concentrando-se no acesso às geometrias de objetos espaciais multiescala individuais, principalmente linhas e polígonos. Adicionalmente, o trabalho apresenta uma solução para facilitar a obtenção de geometrias em determinada escala, com base no acréscimo de um índice de prioridade e números seqüenciais explícitos aos vértices da geometria.

Poorten ((Poorten, 1999) e (Poorten, 2002)) mostra um procedimento de generalização topologicamente consistente, que pode ser utilizado para aumentar a qualidade de dados multiescala na construção de bancos de dados multiescala. O método identifica e seleciona seções de uma linha e remove-as segundo certos critérios.

O método é baseado na triangulação do espaço entre as linhas, e fornece maior flexibilidade que outros procedimentos topologicamente consistentes com relação ao controle do estilo da generalização, além de possibilitar a generalização

simultânea de múltiplos elementos do mapa, e garantir a consistência topológica de todos os níveis de detalhes de objetos compostos por grupos de linhas e polígonos.

Em (Han, 2004) é descrita uma estrutura de dados hierárquica para múltiplas representações de conjuntos de dados vetoriais implicitamente consistentes. A seqüência de representações é codificada de tal forma que apenas os dados correspondentes ao nível mais baixo de resolução são explicitamente armazenados, enquanto os outros níveis, mais detalhados, contêm apenas novos elementos e refinamentos dos elementos do nível anterior. Essa estrutura de dados adiciona ligações verticais a essas estruturas, ligando-as entre os diferentes níveis de detalhes, possibilitando uma navegação eficiente entre eles.

Cheng (Cheng, 2004) introduz o conceito de granularidade e apresenta um método para a construção de um banco de dados multigranular. Esse banco pode ser configurado de acordo com os requisitos da aplicação e ambientes de visualização, de maneira a alcançar melhor desempenho e/ou menor volume de armazenamento, fazendo um balanceamento entre desempenho e armazenamento. Utiliza um método de conversão vetorial-matricial-vetorial no processo de generalização, na construção do banco de dados multigranular, a fim de evitar sobreposições e lacunas entre os polígonos por meio de um *framework* de modelo conceitual de objetos, que inclui restrições topológicas e semânticas entre as várias representações, sem prejudicar o processo de visualização.

Vangenot (Vangenot, 2004) propõe um modelo de dados conceitual com suporte a multirepresentações de entidades do mundo real. A proposta é baseada em dois processos complementares: um responsável pela definição de elementos customizados do banco de dados e outro responsável pelas ligações entre as representações.

3.2 Transmissão de dados na Web

Em ((Bertolotto, 1999) e (Bertolotto, 2001)), são discutidos problemas e soluções relacionados à transmissão progressiva de mapas, descrevendo um modelo multirepresentação para mapas, que possibilita a sua transmissão progressivamente. Além disso, é definido um *framework* para o problema da transmissão progressiva vetorial, que pode ser visto como parte de um problema mais geral: a criação de sumários que ajudam o usuário na consulta de grandes bancos de dados.

O trabalho propõe a criação de sumários vetoriais, definindo um método para a geração de uma seqüência de representações vetoriais dos mapas em diferentes níveis de detalhes a serem transmitidos progressivamente. No método proposto, um conjunto de operadores de generalização define transformações consistentes dos mapas em diferentes níveis de detalhes.

Após a transmissão da primeira representação, o servidor continua a transmissão das representações subseqüentes a fim de aumentar os detalhes até que o usuário esteja satisfeito. O mapa menos detalhado é completamente armazenado no servidor, enquanto que, nos outros níveis, apenas porções das representações subseqüentes são armazenadas. Assim, o modelo proposto economiza espaço de armazenamento no servidor e acelera a transmissão dos dados ao cliente. No cliente, um *buffer* de integração é utilizado para incorporar os incrementos às versões do mapa previamente transmitidas. Ligações lógicas conectam as entidades entre as várias representações das mesmas entidades do mapa, em diferentes níveis de detalhes, chamadas ligações verticais, ou ligações intra-níveis. Essas ligações facilitam a reconstrução de representações em certo

nível do mapa, permitindo saber quais entidades são modificadas entre um nível e outro, permitindo uma navegação eficiente entre os vários níveis, sem a necessidade de consultar todo o modelo, sucessivamente, para a formação de cada nível.

Em ((Gattass, 1999) e (Ferreira, 1998)) são apresentados um conjunto de componentes de *software* que visam suportar a visualização eficiente de mapas na Web, por meio de um formato vetorial específico, apresentando exemplos com o objetivo de avaliar a arquitetura proposta. O *framework* proposto estende o formato TWF de maneira a suportar a visualização de mapas na Web, e apresenta componentes de autoria e de visualização.

Os trabalhos apresentam, entre as técnicas para diminuição do tamanho dos arquivos, as técnicas de quantização de coordenadas e a codificação de coordenadas relativas. A quantização é o processo de reduzir o tamanho necessário para o armazenamento das coordenadas do mapa, mapeando as coordenadas do mapa no espaço dos inteiros positivos Z^+ . As coordenadas relativas expressam cada posição do ponto na linha ou polígono baseada na posição do último ponto, sendo que o ponto inicial do segmento é sempre expresso em coordenadas absolutas. São representadas em um sistema de coordenadas fixo, armazenando apenas o deslocamento do ponto, e não sua posição absoluta e diminuindo, dessa forma, a quantidade de armazenamento necessário para se representar os pontos.

Se for utilizada uma resolução específica para visualização, por exemplo, 1024x1024, cada coordenada não deve possuir mais de 10 bits, representado em coordenadas inteiras, e não em coordenadas reais, geralmente de precisão muito grande, o que aumenta o tamanho dos arquivos ainda mais. Além disso, o vértice de uma linha poligonal que representa um objeto geralmente está bem próximo ao seu antecessor, o que favorece o armazenamento dessas coordenadas de maneira

relativa umas às outras, diminuindo ainda mais o número de bits necessários para o armazenamento (Gattass, 1999).

Mustafa (Mustafa, 2001) apresenta um algoritmo e um sistema de visualização, capaz de realizar a simplificação dependente de visualização de grandes mapas geográficos, através do uso de *hardware* gráfico. O trabalho foca no problema de simplificação, subdividindo interativamente os polígonos em vários níveis de detalhes, preservando as características geométricas e topológicas dos elementos. O uso de *hardware* gráfico possibilita que a simplificação se dê em tempo real, além de permitir a manipulação da escalabilidade dos dados e robustez de maneira mais eficiente que as técnicas geométricas tradicionais.

Liang (Liang, 2001) propõe um método de Transmissão Dependente de Escala, capaz de manipular grandes volumes de dados vetoriais em ambientes com baixa taxa de comunicação como a Internet, baseado na idéia de que nem todos os elementos são visíveis em um mapa em determinada escala, e propõe um método de transmissão apenas do que pode ser visto no dispositivo cliente.

O método proposto utiliza técnicas de generalização que permitem ao servidor eliminar detalhes invisíveis dos elementos, de acordo com a escala de exibição do cliente, para que possa, então, transmitir apenas os elementos visíveis ao usuário, reduzindo o tempo de resposta das operações de manipulação do mapa e aumentando a eficiência da transmissão.

A proposta explora as limitações de visualização dos mapas pelos dispositivos, onde alguns detalhes não podem ser visualizados pelo usuário, aumentando o custo de transmissão sem que isso aumente a qualidade visual do mapa. A eliminação desses detalhes é o ponto chave, que permite a redução do

tempo de resposta do Internet GIS. De acordo com os resultados obtidos, o tempo de resposta foi reduzido substancialmente.

Oh (Oh, 2002) propõe um método de transmissão progressiva no qual, primeiramente, os vértices significantes do mapa são selecionados baseados em um algoritmo de Estimação de Ordem de Prioridade, que utiliza o algoritmo de simplificação de Douglas-Peucker (Douglas, 1973), em seguida, os vértices selecionados são inseridos em uma estrutura de dados chamada Fila de Ordem de Prioridade, que permite ao cliente visualizar, requisitar e eventualmente rejeitar dados espaciais, sem ter que esperar pela transferência de todo o mapa. As etapas de transmissão são processadas progressivamente, e os vértices significantes são enviados ao cliente de acordo com a ordem de prioridade, previamente definida. Finalmente, caso o cliente necessite de mais detalhes, é enviada uma estrutura de transmissão para que o mapa original possa ser exibido corretamente. Este trabalho obteve redução de 22% sobre o tempo de resposta, além de uma redução de 21% no armazenamento dos dados.

Buttenfield (Buttenfield, 2002) apresenta o projeto e implementação de um algoritmo de transmissão de dados vetoriais em níveis progressivos de resolução, preservando propriedades geométricas e topológicas. A solução é diretamente aplicável aos modelos espaciais de dados vetoriais atualmente utilizados, e o algoritmo utilizado opera sobre os dados em seu nível mais fundamental, as coordenadas, e não sobre entidades abstratas, possibilitando a incorporação do método nos bancos de dados espaciais existentes.

O procedimento implementa a generalização *online* de atributos, adaptando a resolução dos dados transmitidos. Em uma fase de pré-processamento, os dados são divididos hierarquicamente, armazenando os resultados no servidor. O

processo de transmissão adiciona detalhes apropriados ao mapa, de maneira a manter a sua topologia.

A criação dos pacotes é realizada utilizando uma decomposição hierárquica baseada no algoritmo de simplificação de Douglas-Peucker, gerando uma estrutura hierárquica de árvore para a transmissão progressiva. Cada nível da árvore representa o mapa em um determinado nível de detalhes. A transmissão progressiva é realizada com a transmissão dos filhos dos elementos do nível do mapa correspondente.

Em (Cecconi, 2003), é apresentada uma técnica de integração da generalização cartográfica com bancos de dados multiescala de maneira a tornar mais eficiente e flexível a publicação de mapas na Web. Para gerar os mapas em diferentes resoluções, os dados correspondentes devem estar disponíveis na escala requisitada, ou então algum processo de derivação e simplificação deve ser realizado.

O trabalho divide o processo de generalização em duas fases, dando mais flexibilidade ao processo. A primeira fase é *offline*, e corresponde à construção do banco de dados multiescala, gerando uma série de mapas em diferentes escalas, ou níveis de detalhes. A segunda fase é iniciada pela requisição do usuário, gerando o mapa apropriado em tempo real, de acordo com a escala desejada, dependendo dos requisitos do usuário. A segunda fase é, em grande parte, simplificada pelo processo de generalização da primeira. O trabalho se concentra na definição de estratégias e aplicação de técnicas de generalização para o mapeamento sob demanda na Web.

Em (Lee, 2003), é proposto um método de transmissão dependente de escala, baseado na idéia de transmitir apenas o que pode ser visto pelo dispositivo

de visualização. Além disso, o trabalho propõe um algoritmo de generalização de mapas baseado em *Wavelets*, que adapta os dados às necessidades de clientes heterogêneos. Os resultados obtidos demonstram uma diminuição no tempo de resposta das operações GIS.

O método proposto filtra e elimina os dados invisíveis do mapa para certa escala de exibição, com ajuda do algoritmo de generalização baseado em *Wavelets*, transmitindo apenas dados que possam contribuir visualmente com a qualidade do mapa. Ao receber as requisições do cliente, o servidor extrai do banco o mapa original e aplica, então, o algoritmo de generalização, de acordo com a escala desejada, requisitada pelo cliente, transmitindo apenas os dados visíveis. Quando recebe o mapa resultante, o cliente exibe os novos elementos que não eram visíveis na resolução anterior e atualiza os objetos que passaram de um nível de detalhamento menor para um maior, atualizando a escala dos objetos em *cache*.

Testes foram realizados buscando mensurar o tempo de resposta de acordo com a mudança na escala e no tamanho dos objetos requisitados, e foi constatada uma diminuição substancial no tempo de resposta, principalmente em redes de baixa velocidade.

O trabalho de Komzák (Komzák, 2003) introduz técnicas de transmissão e visualização escaláveis, baseado em uma hierarquia dos dados através de níveis de detalhes. Descreve técnicas de visualização foco-e-contexto juntamente com transmissão progressiva, combinando essas duas técnicas com diferentes tipos de dados espaciais, durante a exploração interativa de grandes volumes de dados espaciais.

O método proposto implementa as visualizações *fisheye* (olho-de-peixe) e transmissão progressiva baseada no contexto de diversos tipos de dados em um

ambiente GIS distribuído. O método é especialmente adequado para dispositivos móveis com pequena capacidade de visualização e baixa velocidade de transmissão.

A visão *fisheye* integra detalhes de uma região de interesse (foco) juntamente com a área envolvente, bem menos detalhada (contexto) em uma mesma visualização, reduzindo os requisitos de visualização e a quantidade de dados necessários ao usuário. As técnicas de *fisheye* e transmissão progressiva utilizam níveis de detalhes baseados na hierarquia de cada tipo de dados manipulado.

Walden (Waldén, 2003) visa analisar um conjunto de propostas de fornecimento de dados espaciais para visualização em ambientes móveis a fim de superar certas limitações. Realiza a análise de uma representação vetorial eficiente de dados espaciais adequada para apresentação e interpretação em ambientes móveis. A representação deve ser adequada para uma transferência eficiente de dados sobre redes com taxa de transferência de dados limitada.

As propostas apresentadas visam aumentar a eficiência dos Internet GIS em dispositivos móveis e envolvem uma combinação de operações funcionais, cartográficas, de projeto e de desempenho do sistema. O trabalho concentra-se principalmente no projeto teórico dessas operações, não chegando a implementar os algoritmos e arquiteturas propostas.

Paiva (Paiva, 2004) descreve uma abordagem multiresolução com o objetivo de reduzir o tempo de resposta de aplicações GIS na Internet e implementa a arquitetura proposta no *framework* iGIS. Essa abordagem, porém, não envia o mapa por completo, em determinada resolução, já que o usuário talvez necessite trabalhar apenas com uma parte do mapa enviado. Assim, primeiramente, o mapa é

decomposto em várias células, permitindo a transmissão de versões mais e mais detalhadas do mapa, de acordo com a região de interesse.

A região do mapa é, inicialmente, subdivida em várias sub-regiões (nove), algumas das quais se interceptam, permitindo que o usuário visualize as áreas ao redor do mapa sem requisitar uma nova sub-região ao servidor, uma espécie de zona de *buffer*.

Ao aplicar o esquema de subdivisão hierárquica, é gerada uma árvore multiresolução, onde cada nível representa um conjunto de células que cobrem todo o mapa em determinado nível de detalhe. O número de níveis é definido de acordo com o nível de detalhes dos dados, onde mais detalhes significam uma árvore mais profunda.

A abordagem proposta dá ao usuário um ambiente GIS mais eficiente e adequado para visualizar e interagir com grandes volumes de dados, onde nem sempre ele precisa manipular todo o mapa de uma vez.

Yang (Yang, 2004) projeta e desenvolve um modelo hierárquico para compressão de dados vetoriais com o objetivo de transmiti-los progressivamente na Internet. Esse modelo visa extrair um subconjunto representativo dos dados originais em uma versão de baixa resolução, utilizando operações de remoção de vértices. Propõe ainda um conjunto de regras de maneira a controlar a validade topológica dos modelos obtidos, e um algoritmo de reconstrução, para integrar detalhes às versões de baixa resolução, reconstruindo o conjunto de dados originalmente detalhado, após o envio dos vértices removidos.

Os resultados obtidos demonstram uma redução no tempo de resposta, uma manutenção das características geométricas dos objetos e uma eficiente simplificação dos dados vetoriais. Alguns testes realizados demonstram a redução

do tempo de resposta da requisição de um mapa de 29,4 segundos para 19,5 segundos, em virtude da diminuição do volume de dados, a serem transmitidos, devido à simplificação. Vale destacar que a arquitetura proposta de simplificação é sob-demanda, *online*, e um processo *offline* poderia diminuir o tempo de resposta ainda mais.

Baptista (Baptista, 2005) apresenta várias técnicas de otimização, avaliando-as em uma aplicação Internet GIS real. Utiliza técnicas como simplificação dos dados, coordenadas relativas, geração *offline* dos mapas, multiresolução, compressão e carga sob-demanda.

Os mapas estáticos (gerados em processo *offline*) se justificam pelo fato de que os Internet GIS armazenam mapas em Bancos de Dados Espaciais, e cada sessão de um usuário necessita estabelecer uma conexão com o banco e carregar todos os dados, e, posteriormente, transformá-los em algum formato compatível com a Web, como SVG, Flash, WebCGM, entre outros, o que demandaria muito tempo e contraria os requisitos de certas (ou muitas) aplicações Internet GIS.

A compressão, técnica amplamente utilizada em diversas áreas da computação, também pode ser concebida nos Internet GIS, ajudando na compressão dos dados transmitidos. Como o trabalho renderiza dados em formato SVG, cujos visualizadores suportam mapas comprimidos por filtros Gzip, esse foi o método de compressão implementado.

O carregamento sob-demanda é definido como a criação do mapa de acordo com a requisição do usuário, e, apesar de contrastar com a geração estática de mapas, foi incluída de maneira a contemplar adaptações relacionadas à escala e área de interesse requisitada pelo usuário. Assim, as geometrias do mapa são

transmitidas ao cliente sob demanda, conforme realiza certas operações de *zoom* e *pan*.

Os resultados obtidos demonstram que a combinação de várias técnicas, independentes e complementares, pode ser utilizada para aumentar a eficiência de WebGIS, chegando a reduzir o tempo de resposta em até 58%, dependendo das técnicas utilizadas.

Em face aos trabalhos apresentados, constata-se que, em relação ao armazenamento multiescala, muito trabalho já foi realizado, e atualmente, muitas das propostas vêm sendo utilizadas em sistemas GIS, em combinação com as técnicas de multirepresentação e multiresolução, tornando possível a geração de mapas em determinados níveis de detalhes, permitindo certa flexibilidade aos sistemas GIS em relação à visualização dos mapas em diferentes níveis de detalhamento. Essas técnicas de geração de mapas multiescala/multiresolução são baseadas, grosso modo, basicamente em uma consulta espacial com parâmetros que determinam o nível de detalhamento desejado ao mapa e seus elementos, por vezes baseados em uma determinada resolução de visualização do mapa.

Outros trabalhos visam o desenvolvimento de técnicas eficientes de transmissão de grandes volumes de dados, valendo-se da característica de interatividade dos sistemas GIS com o usuário final. A utilização da Transmissão Progressiva possibilita um ganho no tempo de resposta e eficiência da usabilidade do sistema, porém, os trabalhos apresentados não combinam tal técnica com as características do dispositivo de visualização do usuário, onde o mapa será visualizado. Isso ocorre em virtude dos algoritmos clássicos de simplificação de poligonais utilizados nessas técnicas serem, antes de tudo, baseados nas teorias

cartográficas, desconsiderando, assim, as características e limitações de visualização do dispositivo onde o mapa será exibido.

Essas limitações podem ser mais bem exploradas, em conjunto com a técnica de transmissão progressiva, a fim de alcançar um ganho ainda maior em relação às técnicas tradicionais, garantindo que apenas os dados relevantes sejam transmitidos, sem prejuízo de visualização nem alterações topológicas no mapa. A combinação entre as técnicas de transmissão dependente de resolução e transmissão progressiva é, portanto, um problema em aberto aos desenvolvedores de Internet GIS.

4 FERRAMENTAS UTILIZADAS

Neste capítulo apresentam-se duas tecnologias que representam a base para o desenvolvimento do trabalho aqui descrito. A primeira delas é o *framework* iGIS ((Miranda, 2002), (Baptista, 2004) e (Paiva, 2004)), um *framework* GIS para publicação de mapas na Web, que foi estendido para a incorporação da geração de mapas com suporte a transmissão progressiva. Em seguida apresenta-se a arquitetura de *Web Services* e *GIS Services*, destacando suas características e benefícios, que possibilitaram a utilização para a construção de um servidor de mapas com suporte à transmissão progressiva baseada na resolução do dispositivo de visualização.

4.1 O *Framework* iGIS

iGIS é um *framework* para disponibilização de mapas na Web, em conformidade com o padrão OpenGeospatial (OGC, 2005), que permite o desenvolvimento rápido de aplicações de geoprocessamento a um baixo custo. O iGIS possibilita a construção de uma solução completa de mapeamento na Web sem a necessidade de aquisição de *softwares* proprietários de visualização e manipulação de dados espaciais.

Sua arquitetura cliente-servidor é baseada no padrão de arquitetura MVC (Model-View-Controller) (Buschmann, 1996), o que possibilita uma maior separação e independência dos módulos e camadas do sistema, permitindo a implementação e adaptação de novos módulos ao *framework*, incrementando suas funcionalidades.

O *framework* é composto de três camadas, cada uma executando operações específicas e encapsuladas das outras: apresentação, com páginas JSP

gerando HTML e SVG; aplicação, composta de classes Java que implementam a lógica de negócio; dados, na qual um SGBD objeto-relacional com extensão espacial controla o acesso e a manipulação dos dados geográficos. Na Figura 4.1 são detalhadas as principais camadas e módulos do iGIS.

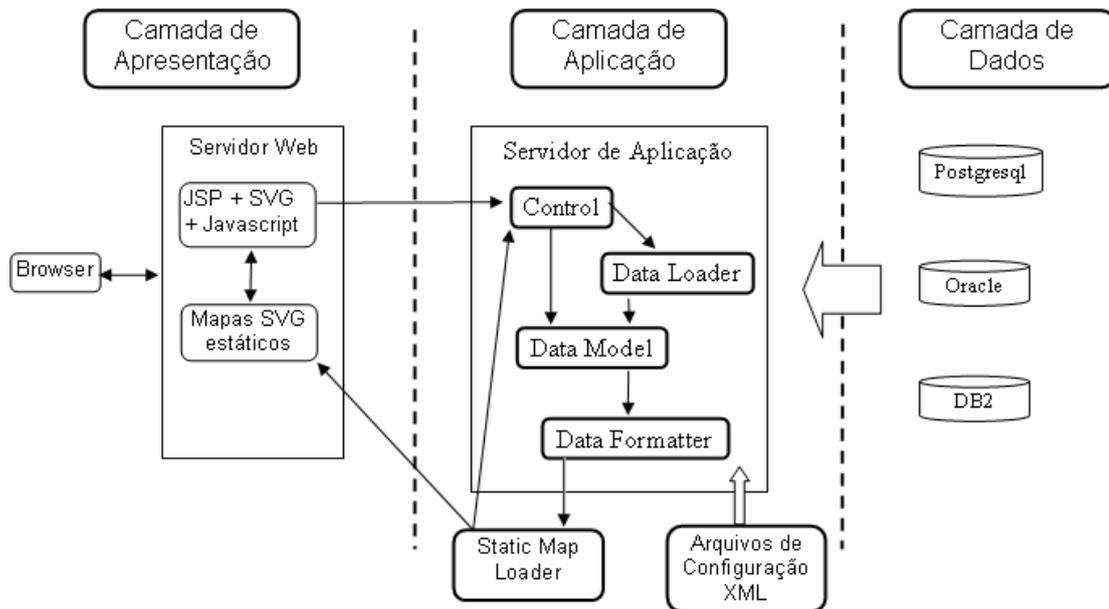


Figura 4.1. Arquitetura geral do iGIS (MIRANDA, 2002).

4.1.1 Camada de Apresentação

A Camada de Apresentação é responsável pela interface e interação com o usuário. Ela constrói a interface gráfica a partir de páginas JSP, que geram código HTML com funções *javascript* responsáveis pela interação usuário-mapa, além do desenho do mapa propriamente dito, renderizado em SVG. Essas páginas podem ser divididas em duas categorias: as responsáveis pela *funcionalidade do sistema*, que exibem as camadas do mapa e ferramentas de manipulação e consulta sobre o mapa; e o mapa propriamente dito, renderizado em SVG.

As páginas funcionais possuem código *javascript* que permite ao usuário manipular o mapa e interagir com ele, através de chamadas DOM, possibilitando

tratar o mapa dinamicamente no cliente, sem que seja necessário, para certas operações, novas requisições ao servidor.

A camada de aplicação (páginas funcionais) envia requisições por meio de um objeto *IGISRequest*, com informações sobre a origem dos dados do mapa, repassando-o a um objeto *Map*, que retorna os dados do mapa renderizados em SVG. Observa-se que a formatação visando a renderização do mapa é realizada a nível de aplicação (e não de apresentação), assim, é possível estender o iGIS de maneira a suportar outros formatos de imagens para o mapa, como GML, PNG, GIF, JPEG, ArcInfo, MapView. Atualmente, o iGIS suporta os formatos de mapas SVG (vetorial) e JPEG (matricial).

Algumas funcionalidades de manipulação dessa camada são: *zoom*, *pan*, *buffer*, exibição de localização, consultas espaciais, consultas textuais.

4.1.2 Camada de Dados

A Camada de Dados é constituída por uma instância de um SGBD objeto-relacional com suporte a dados geográficos, como por exemplo, o Oracle (Oracle, 2006) e o PostgreSQL (PostgreSQL, 2006). O papel dessa camada é permitir a manipulação dos dados geográficos de forma independente do SGBD, seguindo o padrão OpenGeoSpatial. Aspectos como independência dos dados, segurança, concorrência e tolerância a falhas são tratados pelo próprio SGBD.

Nessa camada estão as classes de mapeamento com o banco de dados, seguindo o padrão OpenGeoSpatial, permitindo a manipulação de dados espaciais armazenados no banco em um alto nível de abstração, além de garantir a independência do SGBD Espacial utilizado, uma vez que, para novos bancos de dados, é necessário apenas estender algumas classes dessa camada, de acordo

com as particularidades do banco sendo utilizado. Atualmente, o iGIS suporta os bancos de dados espaciais Oracle, MySQL, PostgreSQL e DB2, além do formato de dados proprietário ShapeFile, da ESRI (ESRI, 1998).

4.1.3 Camada de Aplicação

A Camada de Aplicação é composta de classes Java que implementam a lógica de negócio e é responsável por receber as requisições do cliente, obter as informações desejadas na camada de dados, convertê-las para o modelo OpenGeoSpatial, formatar as informações para um tipo de saída específico e as retornar para a camada de apresentação. Essa camada é dividida nos quatro módulos a seguir.

O módulo *Control* (Módulo Controlador) define a seqüência de ações que serão executadas para atender à requisição do cliente e retornar a resposta à interface de apresentação, solicitando o carregamento e a formatação dos dados para os módulos responsáveis.

O *Data Loader* (Carregador de Dados) provê uma interface comum para a obtenção de uma coleção de dados espaciais a partir de consultas SQL, tornando o iGIS independente do SGBD utilizado, desde que seja implementado um carregador específico para cada plataforma alvo obedecendo à interface comum.

Os dados oriundos do bando de dados são tratados no módulo *Data Model* de acordo com o modelo de dados da aplicação, segundo a definição do OpenGeospatial. Tais dados são manipulados em alto nível pelo *framework*, independente do Carregador ou Formatador utilizados.

A independência entre manipulação e apresentação dos dados é garantida pelo módulo *Data Formatter* (Formatador de Dados), analogamente ao que

foi feito para os carregadores. O iGIS utiliza o formato gráfico vetorial SVG para visualização, mas outros formatos podem ser contemplados criando-se novos formatadores para os mesmos.

Para a publicação de um mapa no iGIS, inicialmente são gerados arquivos de configurações do mapa e do banco de dados daquele mapa. Em seguida, o *framework* lê esses arquivos e toma as medidas necessárias, acessando o banco de dados especificado e gerando os arquivos a serem visualizados no navegador do usuário.

4.1.4 Vantagens do *Framework*

As principais vantagens que o iGIS oferece são a reusabilidade do código e produtividade alcançada na construção de aplicações. O *framework* visa disponibilizar aos desenvolvedores ferramentas que tornem o processo de desenvolvimento o mais eficiente possível, além de ser projetado de tal forma a se adaptar rapidamente a mudanças tecnológicas repentinas, o que é comum no caso de aplicações Web.

A reusabilidade garante a independência do banco de dados, que pode ser alcançada apenas com a extensão e adaptação das classes do *framework* ao novo banco a ser utilizado, mapeando os formatos e rotinas específicas do banco de dados com os métodos do *framework* e as classes definidas no modelo OpenGeospatial.

Isso garante flexibilidade na escolha do banco de dados a ser utilizado, podendo-se levar em conta o custo-benefício, a qualidade e robustez do SGBD, medidas políticas de utilização de *software* livre, adaptação do sistema GIS a bases

de dados já existentes e mesmo integração de informações de várias fontes de dados.

Quanto à rapidez da construção de novas aplicações, o *framework* garante flexibilidade por meio de arquivos XML que definem características das fontes de dados a serem utilizadas, como por exemplo, o local de armazenamento dos dados, informações de autenticação com o banco de dados, como o mapa será exibido aos usuários, quais informações exibir, entre outras. O processo de geração desses arquivos XML, assim como a publicação em si da aplicação GIS se dá por meio de uma ferramenta visual chamada *Deployer*, que permite fazer a publicação da aplicação Web em um servidor J2EE, o que facilita e torna ágil a manutenção e publicação dos GIS.

Quanto à camada de aplicação, o iGIS visa incorporar, em um ambiente Web, as principais características visuais e funcionais dos principais fabricantes de aplicações GIS, de maneira a não se apresentar aos usuários como uma nova ferramenta a ser aprendida, valendo-se de princípios de *design* e *layout* consagrados no mercado dos GIS, como a estruturação do mapa em camadas e legendas e ferramentas de consulta e manipulação espacial dos dados, como *zoom*, *pan*, *buffer*, distância, área, informações sobre uma localidade específica, entre outros, destacando-se principalmente as consultas espaciais.

Como um *framework*, o iGIS dispõe de alguns pontos de extensão, que permitem adaptá-lo à resolução de problemas que não foram projetados e/ou implementados em seu desenvolvimento inicial. Além dos *Data Loaders*, outro ponto de extensão disponível no iGIS é o *Data Formatter*, que permite gerar o mapa que será exibido no dispositivo cliente, assim como também a sua manipulação instantes antes de sua versão final ser processada. O *framework*, portanto, dispõe de abertura

suficiente para implantação e adaptação da solução proposta de transmissão progressiva, de maneira clara e transparente para o restante do sistema, possibilitando o desenvolvimento de novas funcionalidades sem tanta preocupação com o comportamento e funcionamento de todo o sistema, apenas de suas partes principais, permitindo uma solução de baixo acoplamento, bastante atrativa do ponto de vista da engenharia e qualidade de *software*.

4.2 Web Services

Os *Web Services* são programas identificados por uma URI que oferecem serviços a aplicações remotas, denominadas consumidoras, utilizando a Internet como canal de comunicação (W3C, 2002), (W3C, 2004). Tanto os *Web Services* quanto as aplicações consumidoras podem ser desenvolvidos em diferentes plataformas, utilizando diferentes linguagens de programação, o que é permitido pelo uso de um conjunto de padrões, tais como XML, SOAP, HTTP, WSDL (W3C, 2004). As mensagens são trocadas por meio do protocolo SOAP (Simple Object Access Protocol), o que permite a interoperabilidade entre diferentes plataformas. As características do serviço e suas interfaces são publicadas através de documentos WSDL (Web Services Description Language), que descrevem quais dados serão trocados, e como estes dados estarão organizados nas mensagens SOAP. Adicionalmente, os *Web Services* podem ser publicados através de UDDI (Universal Description, Discovery and Integration), um formato utilizado para seu armazenamento em repositórios disponíveis na Internet. Assim, se um desenvolvedor precisar resolver uma determinada tarefa, pode encontrar o *Web Service* que mais se adequar à sua necessidade (Schuenck, 2004).

Entre os benefícios de utilização da tecnologia pode-se destacar: distribuição, integração, infra-estrutura, flexibilidade, integração e automação. O benefício da distribuição advém do fato de tornar mais simples distribuir dados diferentes através de diferentes plataformas, sistemas operacionais e linguagens de programação. A integração, resultada da facilidade para a composição de diversas funcionalidades e dados de diferentes provedores. A infra-estrutura, devido à crescente pesquisa sendo desenvolvida em torno dos *Web Services*, possibilitando o impulso de ferramentas de desenvolvimento, servidores de aplicação, protocolos de mensagens, infra-estrutura de segurança, entre outros (Schuenck, 2004).

A flexibilidade, devido ao desacoplamento das interfaces do serviço de suas implementações, podendo estas serem mudadas, desde que mantenham a mesma interface. A automação, que resulta da possibilidade de os serviços poderem interagir entre si de maneira automática ou semi-automática, o que é possível devido ao uso da linguagem XML e o protocolo SOAP de comunicação (Panatkool, 2002). Tais benefícios derivam das descrições de interface e acesso serem descritas no padrão WSDL, que torna possível a existência de uma Arquitetura Orientada a Serviços de maior poder para a integração de aplicações na Web (Ferris, 2003).

Os *Web Services* apresentam ainda a característica de um modelo de Arquitetura Orientada a Serviços (Service Oriented Architecture – SOA). Nesse modelo, o provedor de serviço possui um serviço que pode ser utilizado pelos clientes, chamados consumidores do Serviço. O provedor cria uma descrição WSDL do serviço (as operações do serviço e entradas e saídas de cada operação). Em seguida é feita uma ligação (*binding*) à implementação do serviço, que descreve como enviar cada mensagem na rede, quando o serviço for solicitado. Dessa forma, o WSDL contém toda informação necessária para invocar o serviço. O provedor de

serviço publica então a descrição WSDL em uma ou mais agências de descoberta, tipicamente com o papel de disponibilizar registros UDDI, com informações adicionais sobre o serviço, permitindo que este possa ser encontrado por meio de critérios de busca e por categoria. Finalmente, o requisitante do serviço encontra, então, a descrição (UDDI) do serviço em uma agência de descoberta, e utiliza a descrição WSDL do serviço a fim de desenvolver ou configurar um cliente computacional que irá interagir com o serviço por meio do provedor de serviços (Ferris, 2003) (Panatkoool, 2002) (Papazoglou, 2003).

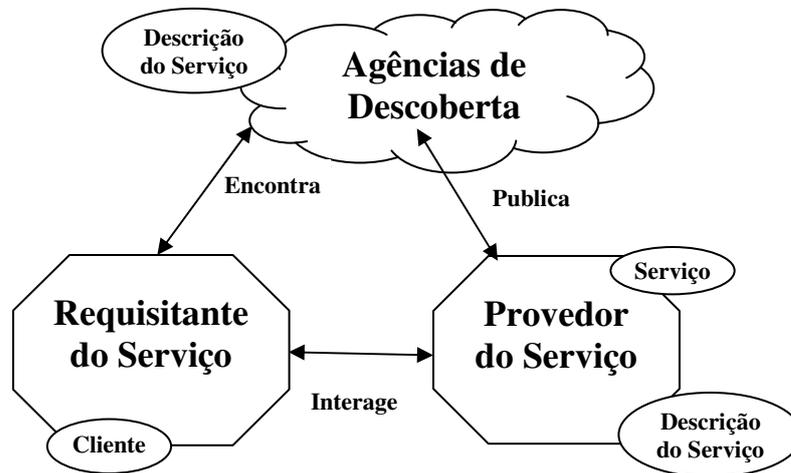


Figura 4.2. Arquitetura geral da tecnologia de Web Services.

A Arquitetura Orientada a Serviços relaciona os componentes de um sistema em um ambiente distribuído onde são disponibilizados serviços que podem ser acessados dinamicamente através de uma rede de comunicação (Papazoglou, 2003), como pode ser observado na Figura 4.2. A tecnologia de *Web Services* implementa a maioria das características desta arquitetura, propondo a exposição das transações e das regras de negócios por meio de protocolos que podem ser processados por qualquer linguagem de programação, em qualquer sistema operacional, rodando em qualquer dispositivo. Desta forma, os *Web Services* são um

caminho para a redução de custos através da redução da redundância dos dados e serviços.

4.3 GIS Services

Gis Services é um termo emergente após a junção dos GIS com a Internet, e englobam serviços de dados e funções geográficas em geral. Porém, quando se fala de *GIS Services*, frequentemente associa-se o termo aos GIS baseados em *Web Services*. Entretanto, não existe uma definição formal. Pode-se definir *GIS Services* como componentes e aplicações GIS modulares e auto-contidas que podem ser publicadas, localizadas e acessadas dinamicamente na Web. Uma outra definição, encontrada em (Yumin, 2004) trata os *GIS Services* como uma série de ferramentas ou funções interoperáveis com a finalidade de expressar, processar, administrar, analisar e visualizar dados geoespaciais, e que geralmente realizam uma função GIS específica que pode ser integrada como parte de uma aplicação maior. Os *GIS Services* fornecem acesso a um conjunto de operações por meio de uma ou mais interfaces padronizadas que permitem que os componentes geográficos possam ser registrados e disponibilizados (Alameh, 2003). Com tal arquitetura, dados e funções de geoprocessamento distribuídos pela rede podem ser acessados, montados e integrados por uma rede de usuários GIS (Tao, 2001).

A tecnologia GIS tem evoluído do tradicional modelo de sistemas monolíticos em plataformas do tipo *mainframe* e posteriormente *desktop*, com os dados espaciais fortemente acoplados ao sistema utilizado para manipulá-los, para um modelo cada vez mais distribuído, baseado inicialmente em aplicações cliente-servidor, finalmente chegando a um modelo de componentes distribuídos em ambientes heterogêneos, o modelo de GIS *Web Services* especializados,

interoperáveis e fornecidos de forma independente. Essa evolução é impulsionada por fatores como o crescimento da importância dos GIS nas organizações, dados espaciais cada vez mais disponíveis e reutilizáveis, maturidade das tecnologias Web e *Web Services* e computação distribuída, e o papel chave dos sistemas GIS em mercados promissores, como por exemplo, Serviços Baseados em Localização (Alameh, 2003), (Tao, 2001).

Assim, o modelo baseado em serviços torna possível a construção dinâmica de aplicações a partir de GIS *Web Services* existentes. O mecanismo para tal montagem de serviços é conhecido como Encadeamento de Serviços, o processo de combinar ou enfileirar resultados a partir de vários serviços complementares, criando aplicações customizadas, possibilitando estender os limites dos GIS tradicionais a fim de englobarem uma ampla comunidade de usuários e um vasto escopo de serviços (Alameh, 2003). Assim, mesmo que um *GIS Service* possa processar uma grande quantidade de funcionalidades, geralmente eles constituem parte de um todo, o que geralmente acontece quando tarefas mais sofisticadas devem ser implementadas pelo serviço (Sliwinski, 2004).

A maioria dos usuários dos GIS tradicionais utiliza apenas uma pequena parte das funcionalidades do sistema. Dessa forma, o modelo baseado em serviços distribuídos permite aos usuários utilizar apenas os serviços de seu interesse, acessando servidores heterogêneos, independente de plataforma ou sistema operacional, característica essencial numa arquitetura de GIS distribuídos (Alameh, 2003). Assim, os componentes clientes e servidores podem interagir e interoperar livremente na Internet, dando aos usuários liberdade de escolha de quais módulos utilizar de um fabricante específico, podendo, inclusive, compor módulos de fabricantes diferentes, adequando-os às reais necessidades de sua aplicação, de

acordo com parâmetros de eficiência, flexibilidade ou custo dos módulos. Essa arquitetura permite que os *softwares* sejam individualizados e otimizados de acordo com sua funcionalidade ou serviço necessários (Tsou, 1998). Tal abordagem facilita também o trabalho dos fabricantes, que podem subdividir seu grande e robusto pacote GIS em vários componentes funcionais interoperáveis. O conceito chave dos *GIS Services* é que o *software* GIS multifuncional, fornecido por fabricantes de sistemas GIS tradicionais, pode ser substituído por vários componentes de geoprocessamento interoperáveis distribuídos pela rede (Tao, 2001).

5 ARQUITETURA DE TRANSMISSÃO PROGRESSIVA

Neste capítulo será apresentado um algoritmo de transmissão progressiva de mapas, que se configura uma das importantes contribuições deste trabalho. Inicialmente, descreve-se o funcionamento do algoritmo computacional, sua construção e características. Em seguida, é mostrado o projeto do algoritmo e sua inserção no *framework* iGIS, estendendo esse *framework* para que pudesse suportar a transmissão progressiva. Por fim, descreve-se a arquitetura e implementação de um *GIS Service* com suporte a transmissão progressiva de mapas, utilizando toda a arquitetura de transmissão progressiva descrita anteriormente.

5.1 Algoritmo de Transmissão Progressiva

Esta seção descreve o algoritmo para transmissão progressiva de dados vetoriais, em níveis progressivos de detalhes, de maneira a diminuir o tempo de resposta do usuário e aumentar a usabilidade e acessibilidade do sistema de maneira adaptativa, de acordo com a resolução do dispositivo alvo onde o mapa será visualizado.

Uma solução baseada em uma prévia classificação dos elementos dos mapas em várias resoluções, como na abordagem multirepresentação, está fora dos objetivos que se deseja alcançar, que é o de operar sobre grandes volumes de dados já existentes. Deseja-se, portanto, alcançar uma solução que atue sobre os elementos vetoriais mais fundamentais do mapa: os seus pontos.

Soluções baseadas na simplificação dos pontos utilizando principalmente o algoritmo de Douglas-Peucker (Douglas, 1973) com algumas customizações, como as de Buttenfield (Buttenfield, 2002), desconsideram a resolução onde o mapa é

renderizado no cliente, concentrando-se apenas em cálculos geométricos absolutos sobre os pontos em si, utilizando parâmetros empíricos a fim de alcançar o nível de detalhes desejado. Isso pode resultar em duas situações problemáticas.

No primeiro caso, mesmo com a simplificação do mapa, podem ser enviados dados supérfluos que não serão visualizados pelo usuário em uma determinada resolução. O outro problema que pode ocorrer é a eliminação de detalhes importantes do mapa, que seriam visualizados no dispositivo, mas, em virtude do parâmetro de simplificação escolhido, foram eliminados de determinada versão do mapa.

No processo de visualização no dispositivo, os pontos do mapa são transformados de coordenadas do mapa em coordenadas do dispositivo. Nesse processo, alguns pontos são mapeados para o mesmo *pixel* da tela. O método desenvolvido antecipa esse processo de rasterização e transfere essa tarefa do cliente para o servidor, no momento da construção do mapa e, portanto, antes do seu envio, eliminando pontos consecutivos de linhas e polígonos que seriam mapeados em um mesmo *pixel* quando o mapa fosse visualizado no cliente, reduzindo a quantidade de dados transmitidos.

Nossa idéia é fornecer um algoritmo que considere a resolução do dispositivo cliente, enviando apenas os dados necessários para a visualização em uma determinada resolução, eliminando redundâncias. A simplificação do mapa é baseada na eliminação de pontos e linhas invisíveis para uma determinada resolução. São enviados ao cliente apenas as informações do mapa estritamente necessárias para formar a versão correta do mapa em determinada resolução, eliminando dados que não seriam visíveis e diminuindo a quantidade de dados

transmitidos. Assim, combinam-se as técnicas de transmissão dependente de escala e transmissão progressiva, aumentando a eficiência alcançada pelo sistema.

No algoritmo proposto, inicialmente, é enviado um mapa gerado em uma resolução baixa (mapa-base). Posteriormente, à medida que o usuário necessita de um maior detalhamento do mapa, são enviados apenas os incrementos daquele mapa que o cliente já possui (economizando tráfego na rede) e os dados são integrados ao mapa existente, dando origem a um novo mapa, mais detalhado. Vale destacar que apenas os detalhes que serão visíveis no dispositivo do cliente serão enviados. Tal abordagem, centrada na resolução do cliente, é uma técnica diferenciada, enquanto que os algoritmos clássicos de simplificação para transmissão progressiva desconsideram a resolução em que o mapa será exibido, interessando apenas alguns cálculos realizados entre os pontos do mapa.

À medida que o cliente recebe os dados do servidor, é realizado um processo de integração dos detalhes ao mapa do cliente, formando então um mapa mais detalhado. Ao mapa residente no cliente, em formato SVG, são realizadas operações DOM resultantes do processamento do algoritmo de integração.

O processo de simplificação e eliminação dos pontos de um determinado segmento de dados (*path* de pontos consecutivos) é realizado no servidor, com cálculos de verificação do *pixel* da tela (resolução alvo) em que um determinado ponto será exibido. Se o *pixel* do ponto corrente for o mesmo *pixel* do último ponto desenhado na tela, esse ponto deve ser eliminado, pois não acrescenta nenhuma informação, uma vez que a exibição do mapa se dará em dispositivos matriciais.

A geração dinâmica sob demanda (*online*) de todas as versões do mapa, além de sobrecarregar o servidor, aumentaria o tempo de espera pelo usuário, anulando os benefícios da transmissão progressiva. Dessa forma, a geração do

mapa em várias resoluções é realizada por um processo *offline*. Quando o cliente requisita os níveis do mapa, estes já estão previamente gerados e armazenados no servidor. Neste ponto, o servidor tem apenas o trabalho de escolher o subconjunto de dados a enviar e transferi-los ao cliente. Assim, o método pode ser dividido em duas fases: a fase *offline*, onde os diversos níveis do mapa são divididos em vários níveis, e o processo interativo, onde cliente requisita detalhes ao servidor, e, ao recebê-los, integra-os ao mapa corrente.

Os pontos enviados são exatamente os pontos do mapa quantizado para a resolução mais alta, não sendo, portanto, realizadas translações dos pontos: a única simplificação é a eliminação dos pontos e linhas redundantes. Essa eliminação não altera o mapa resultante, pois, como o mapa será visualizado em um dispositivo matricial – uma tela gráfica – pontos do mapa real que seriam mapeados em um mesmo *pixel* da tela tornam-se dispensáveis. Outras formas de simplificação alteram o formato do mapa para uma mesma resolução (variando-se o parâmetro de simplificação), pois são baseadas em cálculos entre os pontos do mapa, desconsiderando a resolução em que o mapa seria visualizado.

Quando a resolução do mapa aumenta, devem ser enviados apenas os pontos anteriormente eliminados e que passam a ser visíveis na resolução atualmente existente no cliente. Os pontos são enviados ao cliente, e após uma operação de integração desses pontos com o mapa já existente, é gerada uma nova versão do mapa em uma escala maior. Tais detalhes, na versão anterior do mapa, eram dispensáveis, e não alteravam a forma como o mapa seria visualizado para sua respectiva resolução. Com o aumento da resolução, novos pontos, que antes eram renderizados em um mesmo *pixel*, se “espalham”, passando a ocupar as

vizinhanças daquele *pixel*, aumentando assim os detalhes do mapa. Isso explica porque novos detalhes “surgem” quando a resolução aumenta.

Explicando o processo em outras palavras, é como se a rasterização que seria realizada pelo visualizador do mapa no cliente fosse antecipada como uma tarefa do servidor, economizando tráfego de dados desnecessários na rede. As técnicas de transmissão dependente de escala, transmissão progressiva e simplificação são combinadas, a fim de alcançar os benefícios de cada uma e incrementar a usabilidade e eficiência no uso dos Internet GIS.

A seguir é apresentado o algoritmo de simplificação do mapa e geração dos detalhes pelo servidor (*fase offline*), detalhes estes que serão enviados ao cliente para integração com o mapa em baixa resolução e geração do mapa de maior resolução.

Para gerar o mapa adequado para a visualização em determinada resolução, foi adaptado o processo proposto por Ferreira (Ferreira, 1998) utilizando duas fases: a quantização de coordenadas e a codificação de coordenadas relativas.

Ao fim da quantização, alguns pontos consecutivos são mapeados no mesmo ponto inteiro, gerando segmentos de tamanho zero. Nesse caso, é mantido apenas um desses pontos coincidentes, descartando os outros pontos, o que reduz o número dos pontos necessários para representar um mapa vetorial. O formato em coordenadas relativas é suportado pela especificação SVG e seus *plugins* de visualização e renderização.

O método inicialmente discretiza o mapa para a resolução mais detalhada, definida empiricamente na fase anterior, realizando a quantização de coordenadas e codificação em coordenadas relativas. Nesse processo, são eliminados os pontos consecutivos que seriam mapeados para o mesmo *pixel* da

tela. Essa versão, quantizada, do mapa, é a versão mais detalhada que será disponível ao cliente. Posteriormente, são computadas as versões menos detalhadas de cada elemento do mapa, gerando duas listas de pontos: E^1 , que representam os pontos necessários para renderizar o mapa naquela resolução, e S , que representa os pontos supérfluos para a resolução 1 (resolução-base), que não devem ser transmitidos. O algoritmo continua em um laço que calcula, para cada resolução r : a respectiva versão do elemento E^1 ; os pontos que são necessários adicionar a E^{r-1} para gerar E^r , armazenados em Inc^r . São mantidos em S os pontos supérfluos de E^{r-1} para a geração de E^r . Esse processo é realizado de acordo com o algoritmo (Figura 5.1):

```

1. anterior <- primeiro ponto de  $E^n$  quantizado para a resolução 0
2. para cada ponto  $p_j$  em  $E^n$ 
    $p^q$  <- ponto  $p_j$  quantizado para a resolução 0
   se  $p^q = anterior$ 
     adiciona  $p_j$  em  $S$  //ponto deve ser descartado: mesmo pixel
   else
     adiciona  $p^q$  em  $E^0$  //ponto cai em pixel diferente
     anterior =  $p^q$ 
3. para cada resolução  $r$  de 1 até  $n$ 
    $E^r = E^{r-1}$ 
4. para cada ponto  $p_j$  em  $S$ ,  $j$  de 1 até  $np$ 
    $p^q =$  ponto  $p_j$  quantizado para a resolução  $r$ 
   anterior = ponto anterior de  $p_j$  retirado de  $E^r$ 
   se  $p^q <> anterior$  //se for um ponto significante
     adiciona  $p^q$  em  $E^r$ 
     adiciona  $p^q$  em  $Inc^r$ 
     retira  $p_j$  de  $S$ 

```

Figura 5.1 Algoritmo de geração do mapa-base e incrementos.

Na fase *offline*, são gerados o mapa-base, em sua menor resolução, em formato SVG, e os incrementos do mapa, na forma de arquivos texto, em um formato de dados específico para integração, conforme será explicado adiante. O número de incrementos do mapa, sua resolução inicial e suas resoluções posteriores são definidos empiricamente para cada mapa.

O cliente possui um algoritmo de integração que faz o inverso do trabalho realizado no servidor (fase interativa): inicialmente, o cliente faz uma requisição do

mapa na resolução mais baixa possível. Quando for necessária a visualização de mais detalhes pelo usuário, o cliente envia uma requisição dos detalhes da resolução n , a fim de formar um mapa com uma resolução $n+1$. Após receber os detalhes, os dados são incorporados à resolução " n " existente, dando origem à resolução " $n+1$ ". Esse processo é repetido enquanto houver necessidade de mais detalhes.

A fim de adicionar e integrar os pontos ao cliente progressivamente, foram adicionados metadados que especificam onde cada ponto deve ser adicionado. Essa informação é codificada e armazenada no vetor Inc^r , adicionando como índice um ponto de referência, indicando que o novo ponto será incluído após esse ponto de referência. É gerada uma *string* no formato "P0 0 1 1 2 1 1 0 1", onde P0 indica o ponto de referência (ponto 0), e os pares de números seguintes indicam as coordenadas relativas dos novos pontos a serem inseridos. O cliente, ao receber esses incrementos – o vetor de pontos Inc^R – que contém os pontos que devem ser adicionados à versão atual do mapa que ele possui, gerando a próxima versão do mapa, com um maior nível de resolução. Com base nos metadados, o cliente adiciona os incrementos apropriadamente, atualizando a resolução (nível de detalhes) de cada elemento do mapa.

A seguir, o funcionamento do algoritmo é explicado por meio de um exemplo real. Inicialmente o mapa é quantizado e discretizado para a resolução mais alta possível (Figura 5.2), passando seus pontos de coordenadas reais e absolutas para coordenadas inteiras e relativas. Supondo-se que essa máxima resolução seja de 8×8 , que a resolução mínima seja de 2×2 e que a resolução intermediária seja de 4×4 .

Supondo que os pontos de um segmento do mapa, discretizado para a resolução 8x8, sejam os seguintes: P0(0,0), P1(0,1), P2(1,1), P3(2,2), P4(2,3), P5(1,4), P6(0,5), P7(0,6), P8(1,7), P9(2,6), P10(3,7), P11(4,6), P12(5,6), P13(6,5), P14(7,5), P15(7,4). O algoritmo de simplificação para a geração dos mapas nas diferentes resoluções é explicado a seguir.

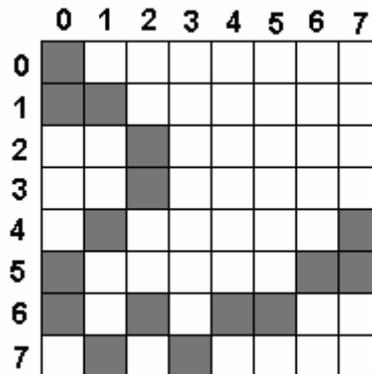


Figura 5.2. Mapa exemplo, na resolução mais detalhada.

Para a geração do mapa na resolução mais baixa (mapa-base: versão 1), o ponto P0 seria adicionado à versão 1. Porém, os pontos P1, P2, P3 e P4 não seriam adicionados a essa versão, pois seriam mapeados para o mesmo *pixel* do ponto P0: o *pixel* (0,0). Já o ponto P5 seria adicionado a essa versão do mapa, pois o último ponto renderizado (P0) está mapeado no *pixel* (0,0) enquanto que o ponto P5 será renderizado no *pixel* (0,1). Os pontos P6, P7, P8, P9 e P10 seriam renderizados no mesmo *pixel* que o último ponto renderizado (P5), sendo, portanto, dispensáveis para essa versão inicial do mapa. O ponto P11 será renderizado no *pixel* (1,1), que é diferente do *pixel* do último ponto renderizado da seqüência de pontos (P5), logo, este ponto é adicionado ao mapa na resolução 2x2, o que não acontece com os pontos P12, P13, P14 e P15, que seriam renderizados no mesmo *pixel* que P11, sendo dispensáveis para essa versão. Assim, a versão inicial do mapa teria apenas os pontos P0, P5 e P11, como ilustrado na Figura 5.3.

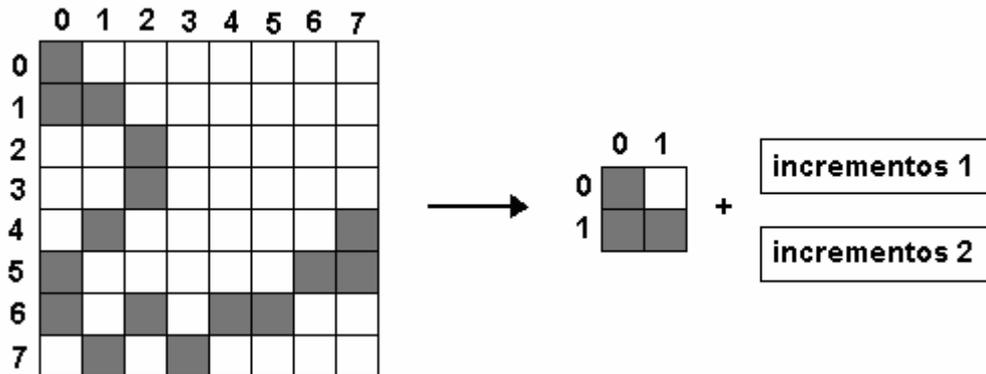


Figura 5.3. Decomposição em mapa base e incrementos.

Além da eliminação dos pontos supérfluos e adição dos pontos significativos para a formação do mapa em um determinado nível de resolução, informações adicionais também são geradas nesse processo, a fim de integrar, em resoluções mais altas, os pontos eliminados ao mapa em uma versão “n”, formando assim o mapa na versão “n+1”. Essas informações são codificadas em um formato mínimo, informando apenas um ponto de referência e o deslocamento, em *pixels*, do ponto eliminado a esse ponto de referência. Para os pontos P1, P2, P3 e P4, essas informações seriam: P0(0,1), P0(1,1), P0(2,2), P0(2,3), onde cada par de coordenadas indica a distância, nas duas coordenadas, do ponto a ser adicionado, em relação ao ponto de referência.

Tantas informações são dispensáveis para um processamento computacional, sendo o formato simplificado a fim de reduzir o número de bytes necessários para codificar tais informações, ficando o formato exemplificado anteriormente como “P0 0 1 1 0 1 1 0 1”, em que P0 indica o ponto de referência e os próximos números, agrupados 2 a 2, as distâncias em *pixels* ao ponto exatamente anterior a ele. Observa-se que como o ponto de referência não muda, essa informação (sobre o ponto corrente) é dispensável e sua eliminação diminui

ainda mais o tamanho das informações que serão enviadas ao cliente para processamento e formação do novo mapa.

Para a formação da versão 2 do mapa, utiliza-se as informações de integração formadas na etapa anterior, juntamente com o mapa na versão 1 (mapa-base), já existente no cliente (Figura 5.4). Assim, são enviadas para o cliente apenas as informações de integração. De posse dessas informações, o cliente faz cálculos simples (somam e subtrações), já que o formato das informações é otimizado para a simplificação e redução da complexidade das operações no cliente, “criando” novos pontos e adicionando-os ao mapa já existente.

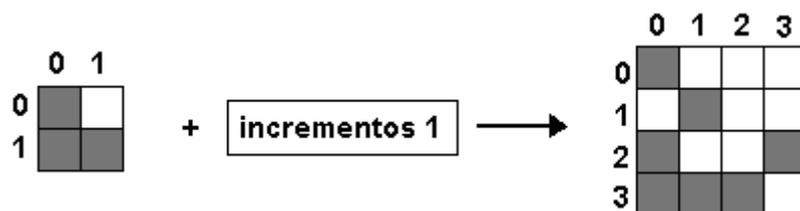


Figura 5.4. Reconstituição da Versão 2 do mapa.

Vale destacar que o processo de simplificação ocorre a cada formação de certa versão do mapa, ou seja, para a formação do mapa na resolução 4x4, alguns pontos que foram eliminados para a versão de resolução 2x2 passam a ser visíveis. Outros, porém, ainda são supérfluos e serão exibidos apenas na versão de resolução 8x8.

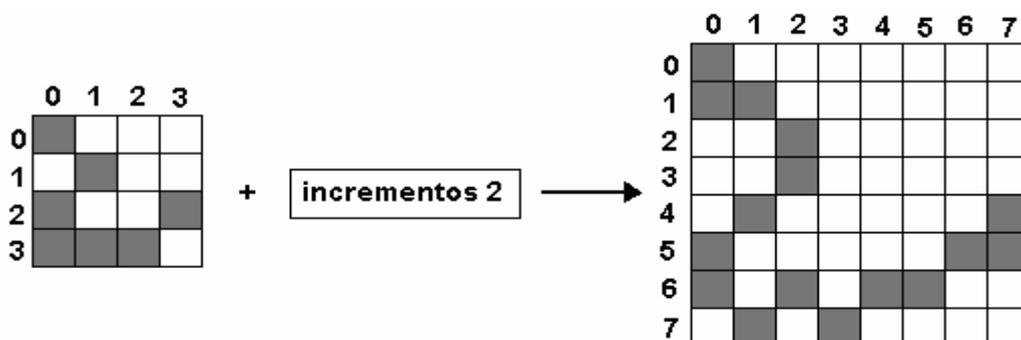


Figura 5.5. Reconstituição da Versão 3 do mapa.

No exemplo, os pontos P1, P3, P7, P9 e P13 são adicionados ao mapa, enquanto os pontos P2, P4, P6, P8, P10, P12, P14 e P15 são dispensáveis à versão de resolução 4x4 (Figura 5.4), sendo visíveis apenas na versão de resolução 8x8 (Figura 5.5).

Quanto às estruturas de dados dos elementos do mapa em seus diversos níveis de resolução, são utilizadas duas listas encadeadas (para cada elemento do mapa a ser visualizado), onde uma armazena os pontos do mapa de determinada resolução e a outra, os pontos eliminados para tal resolução, que serão apenas visíveis em escalas maiores.

Como um exemplo dessas estruturas de pontos, suponha-se um elemento de um mapa (um rio, por exemplo) constituído, em sua escala quantizada máxima, pelos pontos P1, P2..., P12, representado na Figura 5.6.



Figura 5.6. Exemplo: Pontos de um elemento em alta resolução

Na menor escala, após aplicação do algoritmo de subdivisão, são escolhidos os pontos P1, P5 e P9, enquanto os pontos restantes são eliminados, por serem mapeados ao mesmo *pixel* destes: P2, P3 e P4 no *pixel* de P1, P6, P7 e P8 no *pixel* de P5 e P10, P11 e P12 no *pixel* de P9 (Figura 5.7).

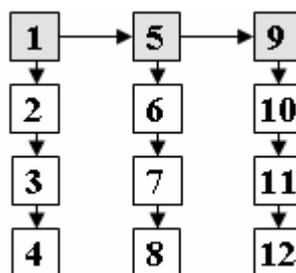


Figura 5.7. Decomposição inicial do mapa (mapa-base)

O processo continua, e na segunda versão do mapa, os pontos P2, P7 e P11 passam a ser visíveis, sendo, portanto, adicionados a lista de pontos daquela resolução (Figura 5.8). Os outros pontos são eliminados. Vale destacar que os “pais” dos pontos eliminados continuam sendo os mesmos. O “pai” de um ponto é o responsável pela eliminação deste de certa resolução, já que seu pai ocupou o lugar, no mesmo *pixel* onde ele estaria.

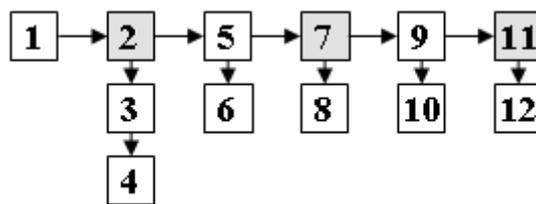


Figura 5.8. Incrementos da Versão 2 (Pontos 2, 7 e 11)

Na próxima versão (versão 3, Figura 5.9) os pontos P3, P8 e P12 passam a ser visíveis, e os pontos P4, P6 e P10 continuam sendo eliminados.

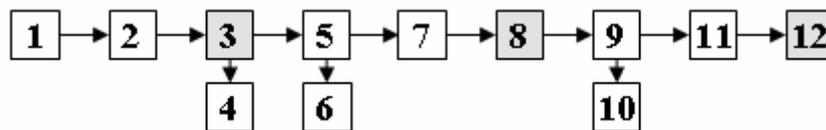


Figura 5.9. Incrementos da Versão 3 (Pontos 3, 4 e 12)

Na última versão, mais detalhada (Figura 5.10), todos os pontos já podem ser visualizados, e são adicionados à versão anterior, aumentando a resolução e escala do mapa.



Figura 5.10. Incrementos da Versão 4 (Pontos 4, 6 e 10)

5.2 Implementação do algoritmo no iGIS

No trabalho proposto, optou-se pela incorporação do núcleo principal da arquitetura de transmissão progressiva adaptado ao iGIS dentro do pacote responsável pela formatação dos mapas chamado *formatter*. As novas classes, relativas à solução de transmissão progressiva a ser incorporada ao *framework* estão situadas em um subpacote de *formatter*, denominado *progressive*. Toda a implementação situa-se dentro do módulo *Formatter*, presente na Figura 4.1.

O módulo de transmissão progressiva estará, portanto, situado em uma camada intermediária, entre a leitura dos dados do banco de dados e a geração dos arquivos do mapa pelo *Formatter*, a serem visualizados no navegador cliente. Mais especificamente, existe um subpacote *progressive* dentro do pacote *formatter*, responsável pela formatação dos mapas com características progressivas.

A seguir é apresentada a modelagem das novas classes da arquitetura proposta e como as mesmas se adaptaram às classes já existentes no *framework* iGIS. Apresenta-se, portanto, a hierarquia de classes do *framework* a fim de facilitar a leitura e o entendimento da arquitetura como um todo. Além disso, as classes cujos nomes não estão totalmente qualificados pertencem ao pacote *igis.formatter.svg.progressive*, e o nome de todas elas possuem a seqüência “*Progressive*” em sua formação.

No diagrama da Figura 5.11 encontra-se a classe *SVGProgressiveFormatterFactory*, que implementa a interface *FormatterFactoryIF* e tem a responsabilidade de criar formadores progressivos. Já a classe *SVGProgressiveFormatter* herda de *AbstractSVGFormatter* e tem por objetivo

formatar as geometrias dos mapas utilizando o algoritmo de transmissão progressiva proposto.

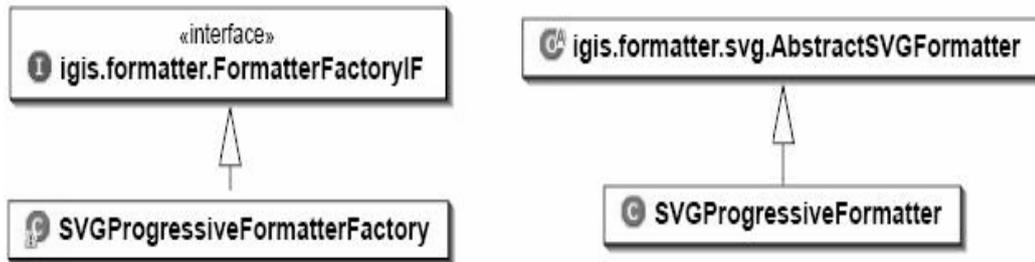


Figura 5.11. Extensão de Classes Progressivas a partir das classes iGIS.

No diagrama da Figura 5.12, as quatro classes que herdam de *GeometryFormatter* são responsáveis por formatar as saídas das respectivas geometrias utilizadas na transmissão progressiva (*Polygon*, *MultiPolygon*, *Line* e *MultiLine*), utilizando coordenadas relativas (W3C, 2003).

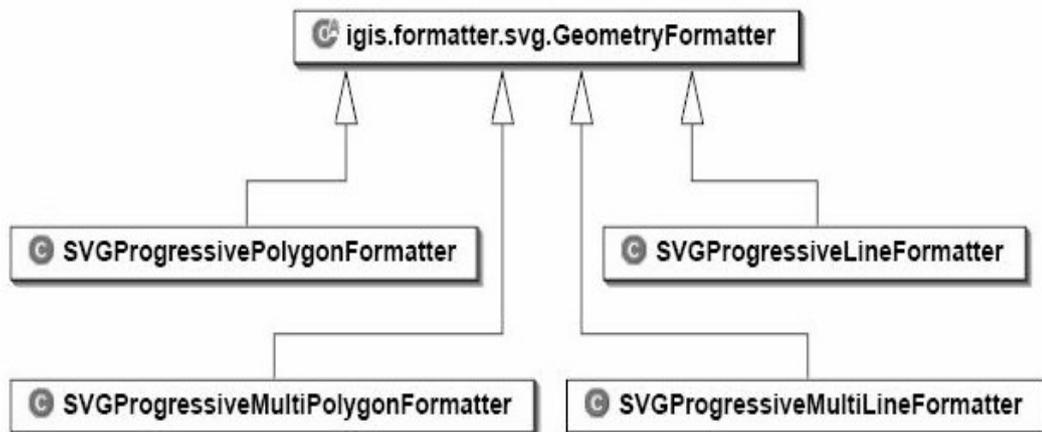


Figura 5.12. Classes formatadoras de geometrias.

Na Figura 5.13 estão as classes que fazem uso direto do algoritmo de transmissão progressiva, que herdam de *AbstractSVGProgressiveFormatter*. Elas têm o papel de gerar as versões em várias resoluções da respectiva geometria: a versão em baixa resolução e os incrementos, utilizados para alcançar resoluções mais altas.

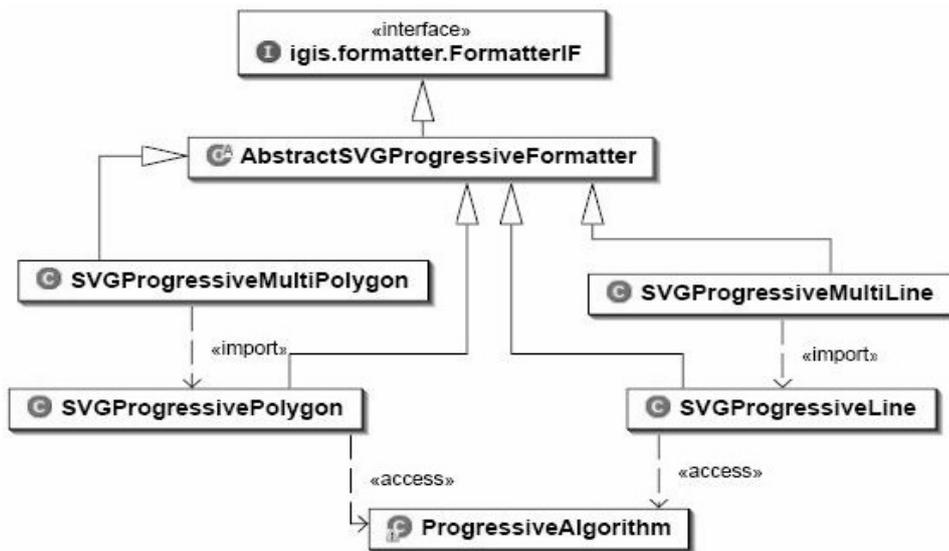


Figura 5.13. Classes que processam as geometrias progressivamente.

No processo de disponibilização do mapa na Web, inicialmente deve ser realizada a geração dos mapas a partir do banco de dados espacial. Isso é feito por meio do *framework* iGIS, cujo fluxo de tarefas é ilustrado resumidamente na Figura 5.14.

Inicialmente, deve ser escolhido um mapa a ser disponibilizado a partir dos mapas existentes no banco de dados. Uma vez escolhido, são recuperados os metadados do banco, a fim de auxiliar parametrização do mapa a ser gerando, como por exemplo, quais camadas serão geradas, quais os *labels* de cada tabela, as legendas do mapa, os tipos de geometria de cada tabela, entre outros, todos relacionados à visualização do mapa. Após a escolha das camadas do mapa a serem disponibilizadas, as definições são salvas em arquivos de configuração XML, para então poder ser iniciado o processo de geração e publicação propriamente dito.

Esse processo inicia por meio do *Map Loader* (pertencente ao módulo *Data Loader*), responsável por carregar o mapa do banco de dados, que carrega as definições do mapa, por meio das quais consegue descobrir como acessar o banco de dados a fim de carregar suas estruturas de dados em alto nível na memória, ou

seja, para cada camada do mapa, são carregados as linhas, pontos e polígonos que representam cidades, rios, lagos e vegetação do mapa, por exemplo. Essas estruturas, pertencentes ao *Data Model* (Figura 4.1), ficam armazenadas em memória principal.

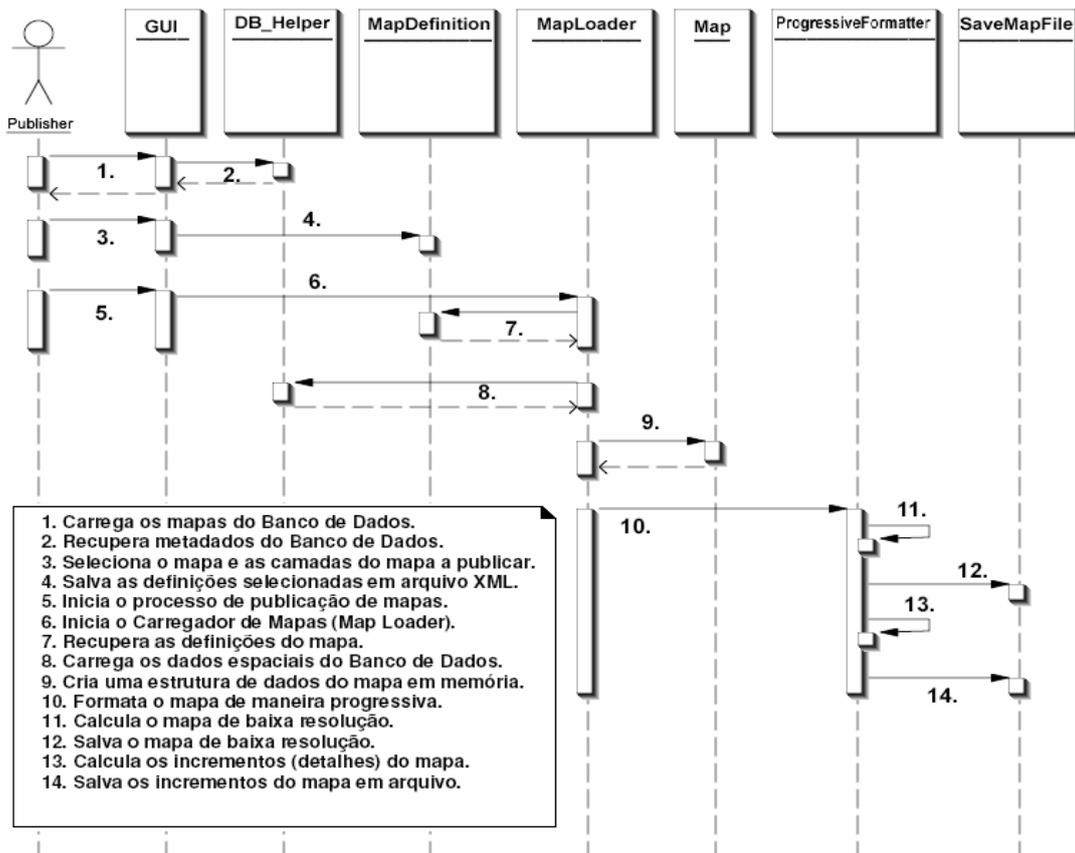


Figura 5.14. Cenário de geração de mapas progressivos.

A partir desse ponto entra em cena o Formatador Progressivo de Mapas, que cria inicialmente o mapa-base, salvando-o em arquivo e em seguida processa os diversos incrementos do mapa, que também são armazenados em arquivo, no servidor Web. A geração do mapa base e incrementos é realizada manipulando as estruturas em memória e mapeando seus pontos em resoluções específicas, de acordo com as definições de geração do mapa.

Uma vez que o mapa base e os incrementos estão armazenados no servidor web como arquivos texto simples, com código SVG, estão facilmente acessíveis à aplicação Web que requisita e processa esses incrementos, como será descrito a seguir.

Após a geração do mapa e decomposição de seus detalhes (incrementos), o cliente pode acessá-lo progressivamente. A interação entre o módulo cliente e o módulo servidor pode ser melhor entendida observando-se a Figura 5.15, de acordo com as etapas descritas a seguir:

Inicialmente, o usuário inicia a aplicação cliente. O cliente solicita ao servidor os mapas disponíveis e escolhe um mapa a ser carregado. Requisita também o número de níveis do mapa e a resolução de cada versão do mapa, armazenando-as para posterior utilização, a fim de adaptar as resoluções mais detalhadas à tela de exibição do mapa. Em seguida, o cliente requisita o mapa em baixa resolução (mapa-base) e redimensiona a área de visualização do mapa (janela) para aquela resolução, exibindo-o na tela. Em seguida, são requisitadas as camadas do mapa, exibindo-as também na interface gráfica. Nesse ponto, o usuário é capaz de visualizar o mapa completo, numa resolução baixa, e todas as suas camadas. É possível, então, requisitar os detalhes das camadas do mapa, que são processados e integrados ao mapa atual, redimensionando, em seguida, a janela de visualização do mapa para a próxima resolução, adequada para aquele nível de resolução do mapa.

A arquitetura do iGIS com o módulo de Transmissão Progressiva incorporado pode ser ilustrada como na Figura 5.16. Pode-se perceber que foram apenas adicionadas novas funcionalidades ao *framework*, nas partes onde foi necessária uma adaptação às características de transmissão progressiva.

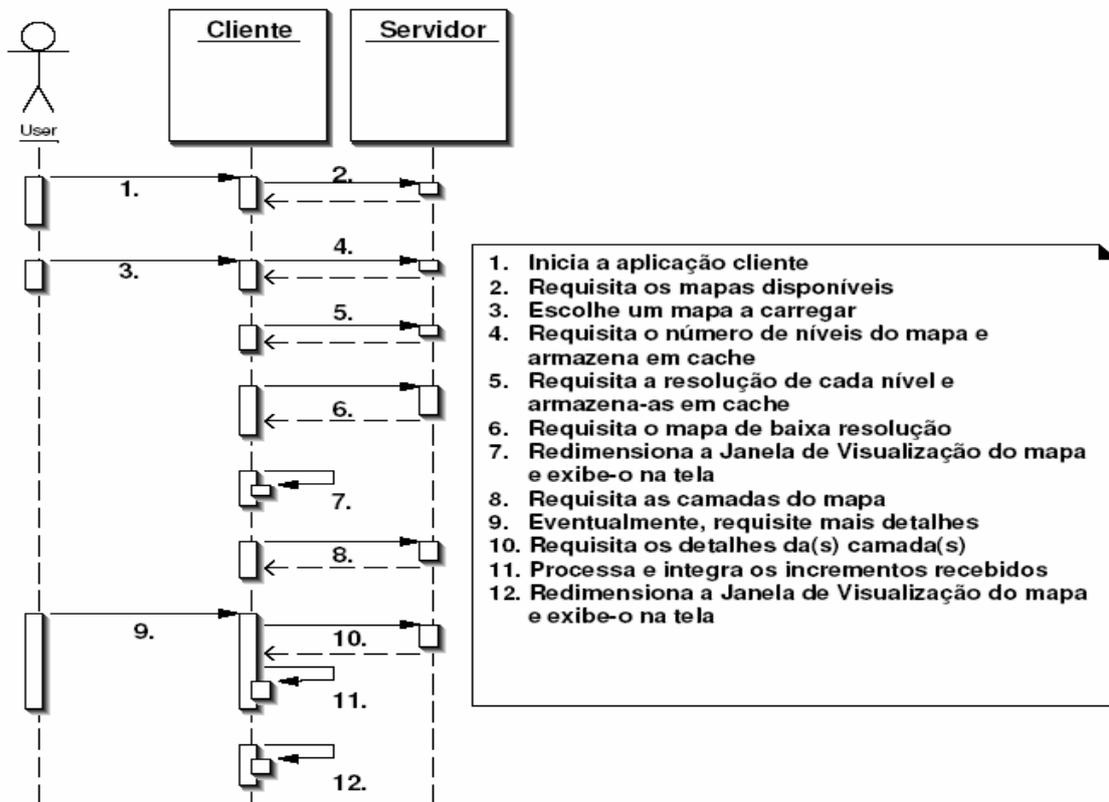


Figura 5.15. Cenário de utilização dos mapas progressivos

No lado do servidor, a principal modificação deu-se no módulo *Data Formatter*, onde foi implementado um módulo específico para a formatação de mapas com suporte a transmissão progressiva. Os arquivos XML de configuração foram aprimorados, incorporando definições e parâmetros para a geração de mapas progressivos dependentes de resolução: as resoluções do mapa-base, intermediárias e máxima. O Carregador de Mapas estáticos foi estendido, passando a carregar também os incrementos de um mapa, de acordo com as requisições do cliente.

O cliente também foi modificado, a fim de tornar-se um cliente com capacidade de requisição de incrementos e incorporação destes ao mapa corrente. A geração de mapas únicos pelos formadores foi substituída pela geração de mapas-base e incrementos, continuando com a natureza estática e armazenamento

em arquivo existente na arquitetura anterior. Além disso, foram implementados um *applet* Java e incorporadas novas funções *javascript* para a manipulação do mapa residente no cliente e dos incrementos recebidos. O *applet* Java é responsável por receber os incrementos via *javascript*, processá-los e devolvê-los ao *javascript*, que, por meio de operações DOM, integra esses novos elementos, detalhados, ao mapa atual.

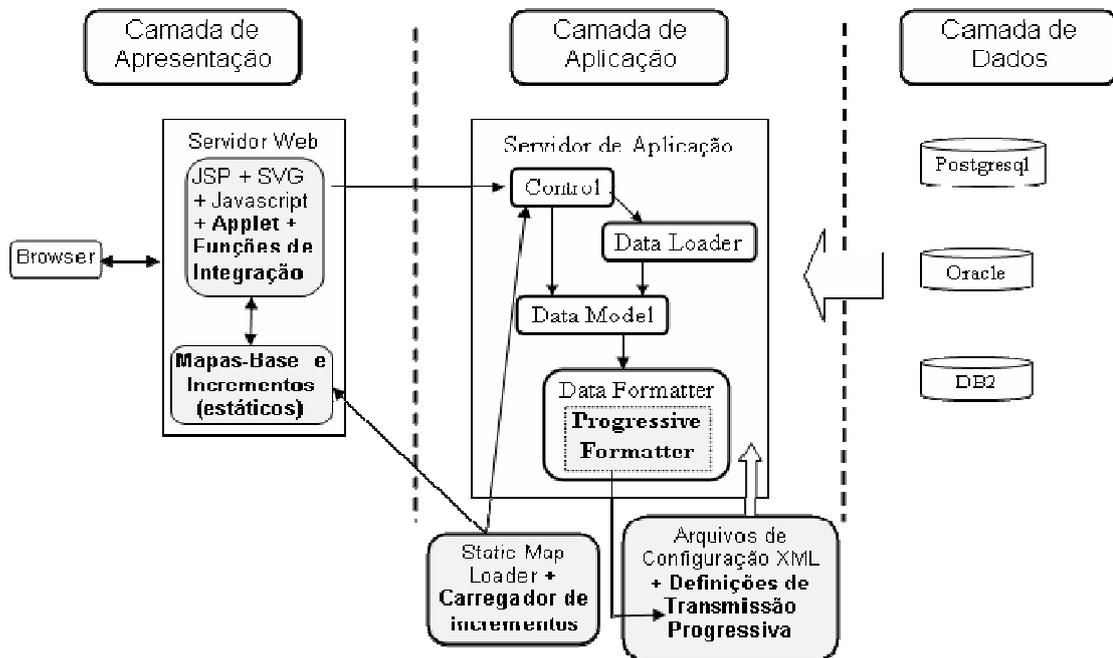


Figura 5.16. Arquitetura do iGIS com módulo de Transmissão Progressiva.

5.3 Aplicação de Referência

Como foi destacado, foram implementados módulos tanto no cliente quanto no servidor do *framework* iGIS, de maneira a incorporar os serviços de transmissão progressiva. O funcionamento do servidor foi explicado anteriormente, e sua função básica é a geração dos mapas e seus incrementos, adaptados a diversas resoluções, que se dá de maneira *offline* e resulta em arquivos-texto armazenados no servidor.

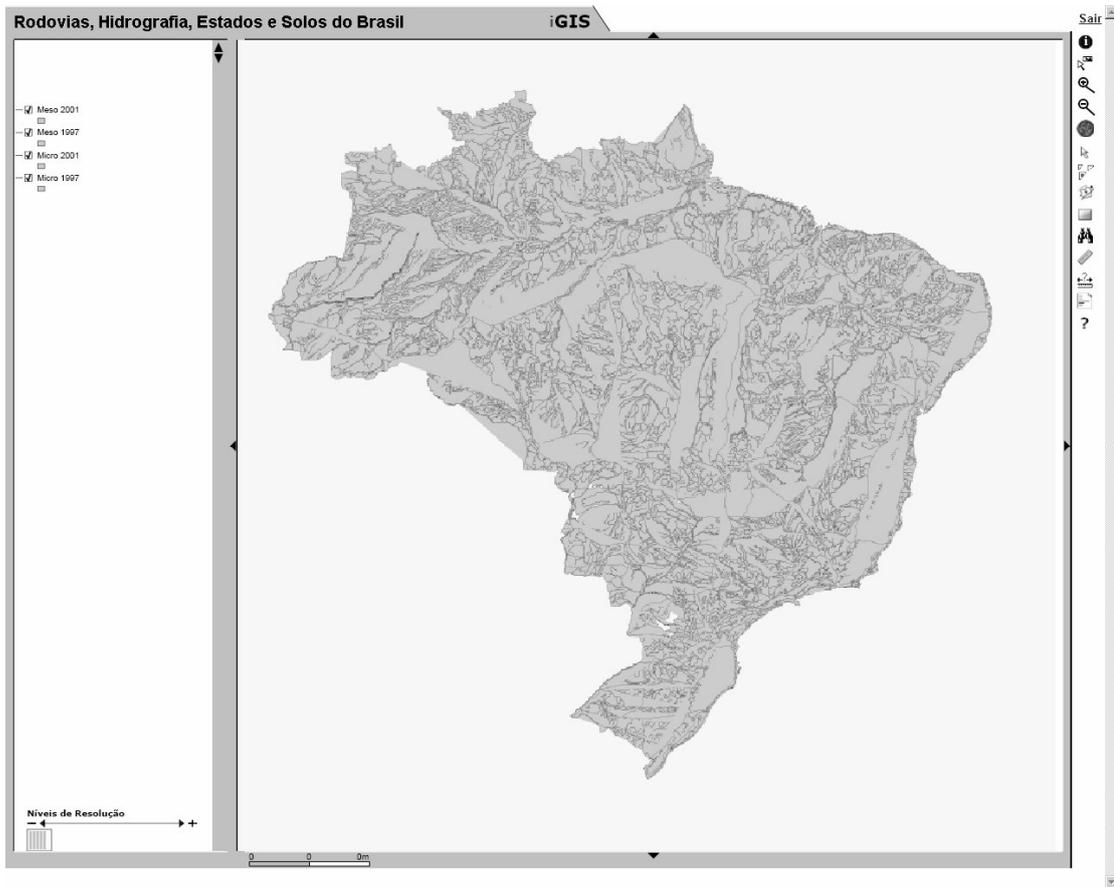


Figura 5.17. Cliente Web do iGIS.

O cliente (Figura 5.17), por sua vez, tem a função de requisitar os detalhes, recebe-los e integrá-los ao mapa já existente e armazenado atualmente. Quando o cliente dá um *zoom* no mapa, é enviada uma requisição aos detalhes relativos ao nível de resolução da próxima versão do mapa. Esses detalhes são recuperados por meio de uma página JSP, que é requisitada ao servidor e retorna ao cliente com todos os incrementos daquela versão, e que contém todos os detalhes da versão requisitada, em formato texto. Em seguida, uma função *javascript* recupera cada elemento do mapa SVG via DOM (os pontos do mapa, presentes no atributo “*d*” da *tag* “*path*” do SVG) e seus respectivos incrementos (formato específico), passando-os a um *applet* Java responsável pela integração dos detalhes ao elemento do mapa atual. O *applet* retorna os pontos do mapa mais detalhados,

resultados da integração, à função *javascript* que realizou a chamada ao *applet*, que os incorpora ao mapa, dando origem a um mapa mais detalhado.

5.4 GIS Service de Transmissão Progressiva de Mapas

5.4.1 Arquitetura Geral

O *GIS Service* de transmissão progressiva faz uso do *framework* iGIS e seu processo de geração de mapas progressivos a partir de bancos de dados espaciais. Primeiramente, é realizado o carregamento dos dados a partir do banco de dados do mapa que se deseja publicar, onde todas essas informações são definidas por meio de arquivos XML, que especificam o banco, as tabelas e camadas do mapa a ser publicado. Essa carga dá origem a vários arquivos para um mesmo mapa. Um deles é relativo ao mapa de mais baixa resolução, os outros contêm os incrementos, detalhes capazes de transformar um mapa de um nível de resolução n a um nível $n+1$ por meio de operações de integração dos dados.



Figura 5.18. Arquitetura de GIS Services Proposta.

Uma vez que os arquivos estejam gerados, o mapa pode ser acessado pelo *GIS Service* por meio de seus métodos, como demonstrado na Figura 5.15. O Serviço Web possibilita que seja adquirido inicialmente um mapa em uma versão de baixa resolução e posteriormente possibilita a aquisição de mais detalhes do mapa

para um determinado nível de resolução, aumentando sua escala. Existe a possibilidade ainda de o cliente solicitar os detalhes de apenas determinadas camadas do mapa, por exemplo, solicitar mais detalhes das rodovias, mas nem tantos detalhes dos limites dos municípios. As interações entre o cliente e o servidor de mapas é realizada por meio do protocolo SOAP, entre o cliente, que pode ser móvel, *desktop*, ou mesmo uma página Web dinâmica, e o *GIS Web Service* implementado, como pode-se observar na Figura 5.18. Nela, a arquitetura do iGIS é condensada em um único bloco, e o *GIS Service* implementado utiliza sua infraestrutura a fim de prover seus serviços aos clientes de mapas progressivos.

O cliente deve ser capaz de receber esses dados e integrá-los ao mapa que já detém em seu *cache*. Ao receber os dados de incrementos, realiza chamadas DOM ao mapa SVG, gerando versões mais detalhadas dos *paths* SVG relacionados, aqueles que devem receber atualização ao passarem do nível de resolução n ao nível $n+1$.

A proposta dos *GIS Services* é disponibilizar esse serviço de entrega de mapas com suporte a transmissão progressiva de maneira independente de plataforma ou linguagem de programação. A única restrição que o cliente deve ter é ser capaz de renderizar o mapa em SVG e poder atualizá-lo via operações DOM, o que é possível em clientes Web, clientes *desktop* implementados em Java, .NET, Delphi, ou mesmo clientes móveis, por exemplo. O serviço é o mesmo, e sua utilização pode ser adaptada de acordo com a aplicação. Uma vantagem da utilização em clientes *desktop* é o ganho de desempenho que se obtém em relação ao cliente Web. Isso porque, no caso do cliente Web, as operações DOM/*Javascript* tornam o processo de integração do mapa um pouco lento comparados ao cliente

desktop, porém não o bastante para que não se justifique sua utilização também para clientes Web.

O *GIS Service* garante a interoperabilidade, independência de plataforma e de linguagem de implementação, garantindo uma maior flexibilidade aos clientes. Uma aplicação cliente Java pode estar necessitando de ainda mais desempenho, a fim de garantir uma maior qualidade de serviço, para o caso de aplicações críticas, por exemplo. A implementação de um novo cliente, nativo para a plataforma-alvo, já pode ser o suficiente para o aumento de desempenho. Ou mesmo o algoritmo de integração dos dados pode ser modificado a fim de aumentar ou otimizar a utilização de recursos, de maneira transparente para o servidor, que continua disponibilizando seus dados por meio de seus métodos de maneira padronizada, com um formato específico.

Além disso, o serviço de transmissão progressiva pode ser integrado em outros sistemas SIG ou mesmo serem adicionados novos módulos ao sistema que utiliza o serviço, uma abordagem conhecida como encadeamento de serviços, onde o resultado (saída) de um serviço é utilizado como entrada de outro serviço. Enfim, uma vez que o componente esteja disponibilizado e seja acessível por uma interface bem definida, este pode ser utilizado como uma “caixa preta”, de acordo com requisitos específicos das aplicações.

5.4.2 Interface do *GIS Service*

A seguir são apresentados os métodos do servidor de mapas progressivo, juntamente com uma breve explicação sobre o comportamento da operação realizada. O método ***getMapNames()*** obtém os nomes dos mapas com suporte a transmissão progressiva armazenadas no servidor. O nome dos mapas será utilizado pelos outros métodos, identificando unicamente um mapa. O método

getLayerNames(mapName) obtém os nomes das camadas de um determinado mapa, que serão utilizados por outros métodos do serviço. Esses dois métodos são primordiais para a utilização dos outros, que receberão o nome do mapa ou camada como parâmetro.

O método ***getLowResMap(mapName)*** obtém o mapa na resolução mais baixa, em sua versão menos detalhada, no formato SVG. Essa versão será exibida no cliente inicialmente e, conforme este requisitar mais detalhes, o mapa SVG será atualizado via operações DOM, aplicando-se o algoritmo de integração no lado do cliente. O método ***getNextLayerIncr(mapName, layerName, level)*** obtém os detalhes de uma determinada camada de um mapa para um determinado nível de resolução. Esses detalhes serão processados e integrados ao mapa existente no *cache* do cliente por meio de operações DOM do mapa em SVG.

Os métodos seguintes são de suma importância ao cliente, pois, por meio deles, é possível adequar a área de visualização do mapa ao mapa recebido e detalhado. O método ***getLevelRes(mapName, level)*** obtém a resolução de uma determinada versão do mapa. Assim como o método anterior, é útil para o cliente ajustar sua janela ao serem recebidos e integrados os detalhes de um mapa, aumentando seu nível de detalhes. O método ***getMinResolution(mapName)*** obtém a menor resolução do mapa, possibilitando ao cliente ajustar sua janela de visualização ao mapa menos detalhado, enviado no início do processo da transmissão progressiva. E o método ***getLevels(mapName)*** obtém o número de níveis (versões) de um determinado mapa.

Alguns métodos são apenas utilitários, e não estão diretamente relacionados aos serviços de transmissão progressiva. O método ***getMapDescription(mapName)*** obtém as informações sobre o mapa com o nome

dado. Semelhantemente, o método ***getLayerDescription(String mapName, layerName)*** obtém informações sobre uma determinada camada, de um determinado mapa. Essas informações são apenas de uso do usuário final. O método ***mapToScreen(x, y)*** converte um ponto em coordenadas reais para um ponto no mapa na tela. Já o método ***screenToMap(x, y)*** faz o contrário, convertendo um ponto na tela para um ponto em coordenadas reais. Por fim, o método ***getDetails(x, y)*** obtém informações sobre um determinado ponto do mapa, ou seja, informações sobre os objetos a que aquele ponto pertence, por exemplo, rios, estradas e cidades existentes naquele ponto.

5.4.3 Aplicação de Referência

Foram desenvolvidos um cliente desktop (Figura 5.19) e um cliente Web (Figura 5.17) para testar as funcionalidades dos serviços implementados, além de realizar uma avaliação de desempenho entre a abordagem baseada na Web e a arquitetura baseada em *Web Services*. Utilizou-se o Axis (Axis, 2005) para a geração dos *Web Services*, juntamente com o servidor Tomcat 5.5.

Grande parte do funcionamento da aplicação cliente, demonstrada na Figura 5.15, foi comentado anteriormente, já que a aplicação nada mais é do que um protótipo da utilização dos serviços implementados. Entretanto, a maneira como são realizadas as requisições aos detalhes merece algum destaque. Ela pode ser realizada de duas formas: síncrona e assíncrona.

Na abordagem assíncrona, o cliente fica livre para realizar outras tarefas enquanto aguarda a chegada dos incrementos de determinada camada. Isso é útil no caso de o cliente estar interessado, por exemplo, em detalhes de todas as camadas do mapa. Nesse cenário, são feitas várias requisições assíncronas, via *Web Services*, de detalhes das camadas do mapa e, à medida que vão sendo

recebidos, esses detalhes são processados e integrados ao mapa, enquanto detalhes de outras camadas continuam sendo obtidos, possibilitando ao cliente realizar outras tarefas, além de processamento e integração de detalhes. Além disso, o restante da aplicação consiste basicamente em chamadas aos serviços, detalhados anteriormente. O cliente Web foi implementado na etapa inicial do trabalho, quando ainda não se pretendia a construção de uma arquitetura de *GIS Services*. Com a possibilidade dos *Web Services*, a fim de testar sua flexibilidade, o cliente Web foi adaptado às chamadas dos *GIS Services* e foi também implementado o cliente *desktop*, a fim de mensurar o desempenho de ambos os clientes, além de comprovar a flexibilidade oferecida pela nova tecnologia.

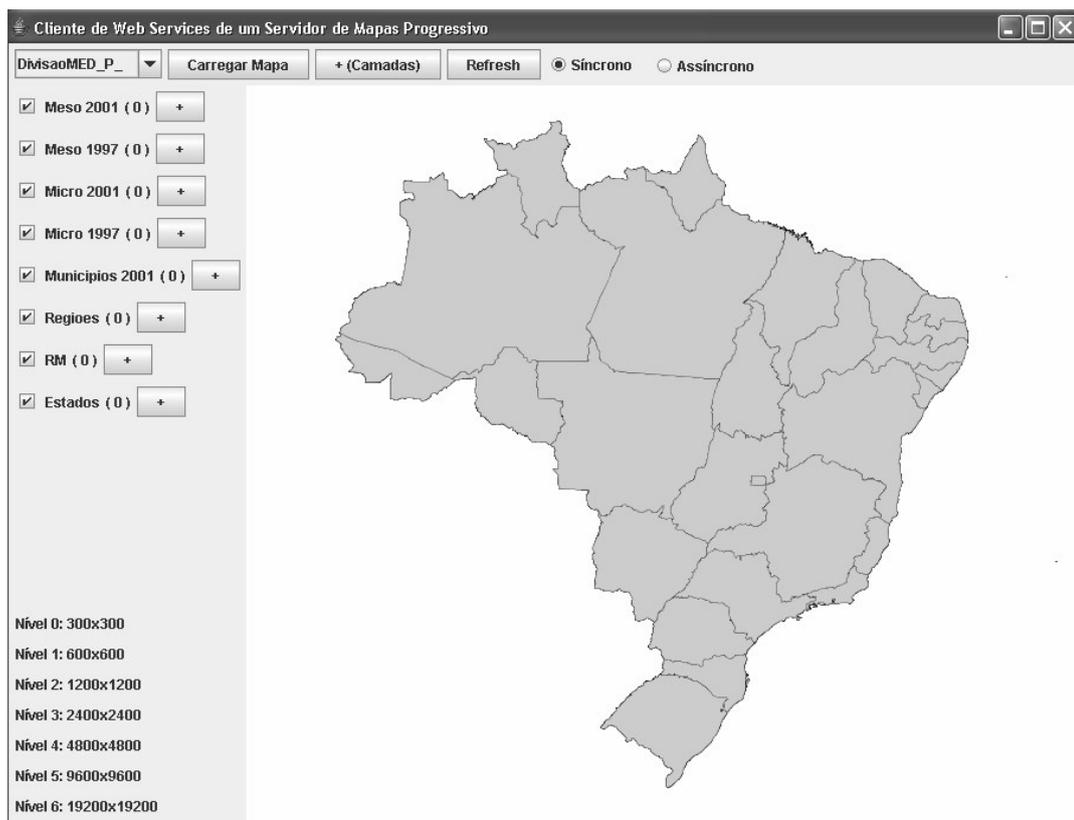


Figura 5.19. Cliente desktop implementado em Java.

6 AVALIAÇÃO DE DESEMPENHO

Nesse capítulo serão apresentados os resultados obtidos com a arquitetura de transmissão progressiva, avaliando os tamanhos dos arquivos gerados e os tempos de transmissão em redes de com taxas ideais de transmissão, avaliando o comportamento da arquitetura proposta e comparando-a com a transmissão do mapa sem nenhuma técnica de otimização e com a técnica de transmissão dependente de resolução. Devido ao fato de outros métodos, tradicionais, de transmissão progressiva desconsiderarem a resolução do dispositivo de visualização, esses métodos não foram utilizados na comparação, e a comparação com esses métodos seria subjetiva, uma vez que o objetivo e os parâmetros das técnicas têm naturezas diferentes.

6.1 Ambiente de testes

Nos testes realizados, cujos resultados serão apresentados a seguir, foram utilizados três diferentes mapas: Brasil1, Brasil2 e Brasil3, com tamanhos de 9527, 19139 e 35847 KB, respectivamente. O mapa Brasil1 apresenta informações sobre vegetação, solo, rodovias e hidrografia do Brasil. Os mapas Brasil2 e Brasil3 contêm informações sobre a divisão política do Brasil, diferindo entre si no número de suas camadas, entre as quais estão as macro, meso e microregiões, estados e municípios de várias épocas. A diferença entre eles é que o Brasil2 contém apenas os dados mais recentes existentes no Brasil3, sendo um subconjunto deste, enquanto o Brasil3 apresenta todas as camadas do banco de dados. Para cada mapa será apresentada uma tabela com os valores das variáveis descritas para cada resolução, tornando possível uma comparação da técnica proposta, tanto com

o mapa original quanto com a técnica de quantização. Em seguida, são apresentados os gráficos de algumas variáveis e, por fim, telas do sistema em funcionamento. Os tamanhos dos mapas/incrementos são medidos em *kilobytes* (KB) e os tempos de processamento dos incrementos foram medidos em segundos (s). Os mapas podem ser visualizados na Figura 6.1



Figura 6.1. Mapas Brasil1, Brasil2 e Brasil3

Ao analisar o tempo de resposta do sistema, o tempo de transmissão dos incrementos somado ao tempo de processamento destes, para integração ao mapa, os resultados foram simulados em redes hipotéticas de baixas velocidades de transmissão, cujas taxas foram escolhidas empiricamente, equivalentes a 5, 12, 25 e 50 KB/s. As avaliações em torno do tempo de resposta não foram realizadas em um ambiente de rede real, sendo realizados apenas cálculos com base no tamanho dos mapas e nas taxas citadas, consideradas idealmente constantes. Tal fato decorre de que a análise feita em torno da arquitetura de transmissão progressiva tem por objetivo apenas dar uma idéia comparativa da abordagem implementada em relação a outras abordagens, e não um conhecimento exato de um sistema de transmissão progressiva em produção, o que foge do escopo desse trabalho. Além disso, os tempos de resposta são constantes e ideais em todas as arquiteturas analisadas.

Os resultados foram obtidos utilizando um computador Athlon64 de 2 GHz, com 1Mb de RAM, tanto para o módulo cliente quanto para o servidor de mapas.

6.2 Avaliação do tamanho dos mapas gerados

O principal objetivo do trabalho, conforme discutido em (Costa, 2006) é diminuir o tempo de resposta, aumentando a usabilidade do sistema GIS, e isso pôde ser alcançado, em vários níveis quantitativos de eficiência, de acordo com os parâmetros do algoritmo e com as resoluções finais desejadas pelo usuário final do sistema, ou seja, a resolução que ele julga ser suficiente para suas necessidades, que será denominada resolução suficiente.

Os parâmetros de entrada do algoritmo são as resoluções inicial e máxima do mapa (menos e mais detalhadas, respectivamente), os fatores multiplicadores e o número de incrementos.

Os fatores multiplicadores dizem respeito a quantas vezes os mapas das resoluções intermediárias (entre a inicial e a máxima) são maiores que a resolução inicial. Isso dá a liberdade de utilizar diversos tipos de funções na formação das escalas dos mapas, como por exemplo, funções exponenciais ou lineares, enquanto que o número de incrementos é o número de passos necessários para se chegar da resolução mínima até a máxima. Este número é derivado do parâmetro anterior (fatores multiplicadores) e foi citado apenas para fins explicativos.

Para a avaliação da eficiência do algoritmo, analisa-se nesta seção os ganhos do algoritmo sobre o tamanho dos dados transmitidos, em *kilobytes* (KB). Várias medidas foram escolhidas a fim de obter os resultados da eficiência do algoritmo, e são descritas a seguir.

A versão do mapa (**V**), onde a versão 1 significa o mapa na versão de mais baixa resolução, menos detalhada, a resolução do mapa (**R**), em *pixels*, o tamanho dos incrementos do mapa (**TI**), para cada resolução, que são enviados ao cliente para aumentarem o nível de resolução do mapa, o tamanho do mapa quantizado (**TMQ**), adaptado para uma determinada resolução de exibição (técnica dependente de visualização), o tamanho real do mapa (**TRM**), sem o uso de nenhuma técnica de otimização.

Além dessas medidas, outras podem ser derivadas dessas, de acordo com os níveis de resolução crescente dos mapas: a soma dos incrementos transmitidos (**SIT**), que mede o tamanho total dos detalhes enviados ao cliente desde a versão 1 (mapa-base) até certa versão do mapa, a soma dos mapas quantizados transmitidos (**SMQT**), que mede o tamanho total dos mapas transmitidos, se fosse utilizada a técnica de quantização, a razão entre o tamanho dos incrementos e o tamanho dos mapas quantizados (**TI / TMQ**), a fim de comparar o ganho da técnica de transmissão progressiva proposta sobre a técnica de quantização, a razão entre a soma dos incrementos transmitidos e a soma dos mapas quantizados transmitidos (**SIT / SMQT**) e a razão entre a soma dos incrementos transmitidos e o tamanho real do mapa (**SIT / TRM**). A seguir serão apresentados os resultados obtidos com a implementação do método proposto no *framework* iGIS e os ganhos que se obteve com a sua utilização.

Pode-se perceber na Tabela 6.1 que se não for utilizada nenhuma técnica de otimização, utilizando o mapa original, tem-se uma sobrecarga inicial de transmissão muito grande, com o custo de transmissão de 9527 KB. Observando a coluna **TRM**, todo o mapa é transmitido em um único passo, e o tamanho dos outros níveis do mapa é zero, pois na primeira versão o mapa alcança seu detalhamento

máximo. A subdivisão do mapa em conjuntos de incrementos segmenta o mapa original em fragmentos menores do mapa para cada nível de resolução, como se pode observar na coluna **TI**. Na coluna **TMQ**, observa-se que o tamanho do mapa quantizado para um nível específico de resolução é menor que o mapa sem utilização de nenhuma técnica, porém, maior que o tamanho dos incrementos de cada versão.

V	R	TRM	TI	TMQ	SIT	SMQT	TI/TMQ	SIT/SMQT	SIT/TRM
1	300x300	9527	819	819	819	819	100%	100%	9%
2	600x600	0	1126	1325	1945	2144	85%	91%	20%
3	1200x1200	0	1137	1976	3082	4120	58%	75%	32%
4	2400x2400	0	916	2525	3998	6645	36%	60%	42%
5	4800x4800	0	676	3013	4674	9658	22%	48%	49%
6	9600x9600	0	430	3345	5104	13003	13%	39%	54%
7	19200x19200	0	226	3460	5330	16463	7%	32%	56%

Tabela 6.1. Valores comparativos para o mapa Brasil1

Para a versão 4, por exemplo, o tamanho dos incrementos é de 916 KB, enquanto que o tamanho do mapa quantizado é 2525 KB. Observa-se também que, com o aumento do nível de detalhes, enquanto o **TMQ** aumenta, o **TI** diminui. As colunas **SIT** e **SMQT** permitem analisar a quantidade de dados transmitidos até determinado nível de resolução com as técnicas de transmissão progressiva e quantização, respectivamente. Por exemplo, até alcançar a versão 4 do mapa, a técnica de transmissão progressiva utiliza 3998 KB enquanto que a de quantização soma um total de 6645 KB transmitidos, contra os 9527 KB do mapa real. Observa-se que apenas a transmissão da versão 4 quantizado consome apenas 2525 KB, e caso conheça-se previamente o nível de detalhes que se deseja visualizar ou analisar os dados, a técnica de quantização pode ser mais adequada em alguns casos, embora o seu tempo de resposta inicial seja maior que com a técnica de transmissão progressiva.

As três últimas colunas apresentam razões entre as medidas das colunas anteriores. A coluna **TI/TMQ** mostra a diminuição crescente do tamanho dos incrementos em relação ao tamanho do mapa quantizado. A coluna **SIT/SMQT** mostra também uma diminuição alcançada pela técnica de transmissão progressiva em relação à de quantização, agora analisando o somatório dos mapas transmitidos até certo nível de detalhes. Finalmente, a coluna **SIT/TRM** mostra que inicialmente, a soma dos incrementos do mapa transmitidos para uma determinada resolução é apenas uma pequena parte do mapa real, 9% e 20% do tamanho original, para os níveis 1 e 2, respectivamente, e mesmo com o aumento até níveis maiores de detalhes, o total de dados transmitidos incrementalmente ainda é menor que o uso do mapa sem nenhuma técnica de otimização. Esses valores podem ser visualizados na Figura 6.2.

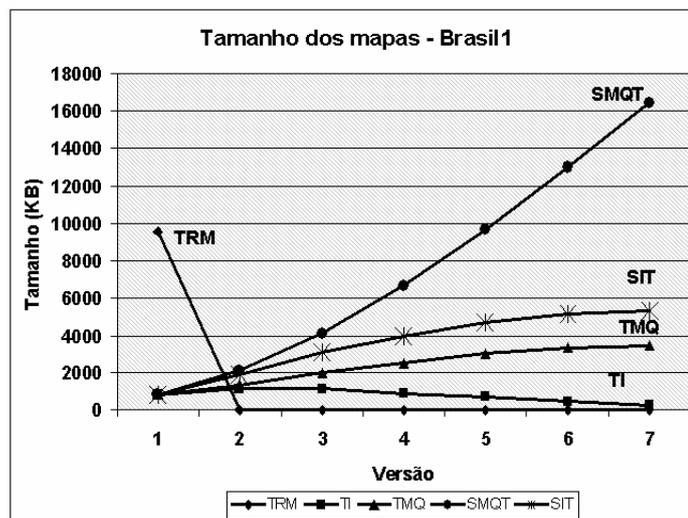


Figura 6.2. Gráfico comparativo para o mapa Brasil1

No mapa Brasil2, o comportamento geral dos tamanhos dos mapas e incrementos gerados é basicamente o mesmo (Tabela 6.2 e Figura 6.3). Nesse caso, trabalhou-se com um mapa maior que o primeiro caso, com tamanho real de 19139 KB, mais que o dobro do anterior, demonstrando que o aumento da

quantidade de dados não interfere na geração dos mapas para transmissão progressiva.

V	R	TRM	TI	TMQ	SIT	SMQT	TI/TMQ	SIT/SMQT	SIT/TRM
1	300x300	19139	1427	1427	1427	1427	100%	100%	7%
2	600x600	0	2147	2309	3574	3736	93%	96%	19%
3	1200x1200	0	2203	3596	5777	7332	61%	79%	30%
4	2400x2400	0	2012	5011	7789	12343	40%	63%	41%
5	4800x4800	0	1462	6196	9251	18539	24%	50%	48%
6	9600x9600	0	690	6670	9941	25209	10%	39%	52%
7	19200x19200	0	309	6728	10250	31937	5%	32%	54%

Tabela 6.2. Valores comparativos para o mapa Brasil2

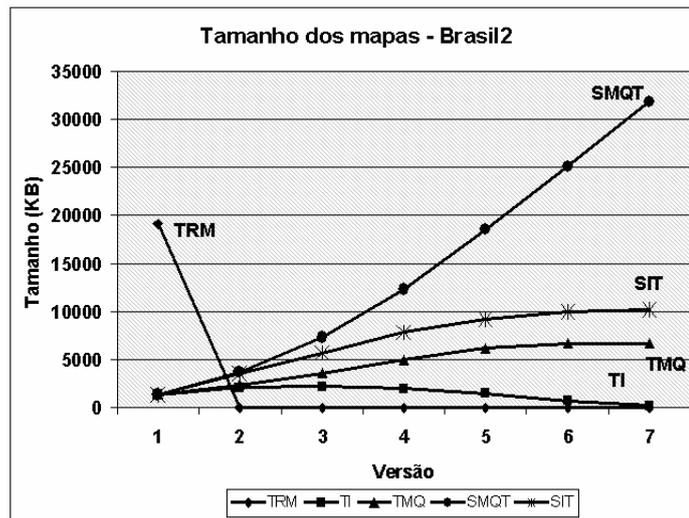


Figura 6.3. Gráfico comparativo para o mapa Brasil2

No mapa Brasil3 foram utilizadas resoluções menores para os níveis do mapa, a fim de analisar o comportamento nesse tipo de situação (Tabela 6.3 e Figura 6.4). Observa-se nesse mapa que as razões TI/TQM e SIT/SMQT não apresentam uma queda constante, como observado nos dois primeiros casos. Analisando o tamanho dos incrementos em razão do tamanho do mapa quantizado, observa-se que o mapa quantizado é ainda menor que o tamanho dos incrementos, para os níveis 2 e 3, assim como a soma dos incrementos totais enviados é maior que a soma dos mapas quantizados, pelo menos até ser alcançada a versão 3. Isso

se deve à quantidade de metadados para processamento e integração dos detalhes existentes nos incrementos ser relativamente maior, quando a resolução não aumenta muito, o que faz com que os metadados tornem-se uma sobrecarga, pois nesses casos existem mais metadados e menos dados de detalhes que comparado com o aumento mais significativo entre as resoluções.

V	R	TRM	TI	TMQ	SIT	SMQT	TI/TMQ	SIT/SMQT	SIT/TRM
1	75x75	35847	1476	1476	1476	1476	100%	100%	4%
2	150x150	0	2789	2016	4265	3492	138%	122%	11%
3	300x300	0	2988	2933	7253	6425	101%	112%	20%
4	600x600	0	2925	4322	10178	10747	67%	94%	28%
5	1200x1200	0	3016	6665	13194	17412	45%	75%	36%
6	2400x2400	0	2830	9405	16024	26817	30%	59%	44%
7	4800x4800	0	2007	11222	18031	38039	17%	47%	50%

Tabela 6.3. Valores comparativos para o mapa Brasil3

Entretanto, com o aumento da versão do mapa (4, 5, 6 e 7), o ganho da técnica de transmissão progressiva torna-se mais evidente. Pôde-se concluir que o ganho com a técnica de transmissão progressiva não é tão evidente quando não se aumenta significativamente as resoluções do mapa entre uma versão e outra, passando a ser observado apenas nos últimos níveis de resolução, com a progressão sucessiva de interações pelos usuários.

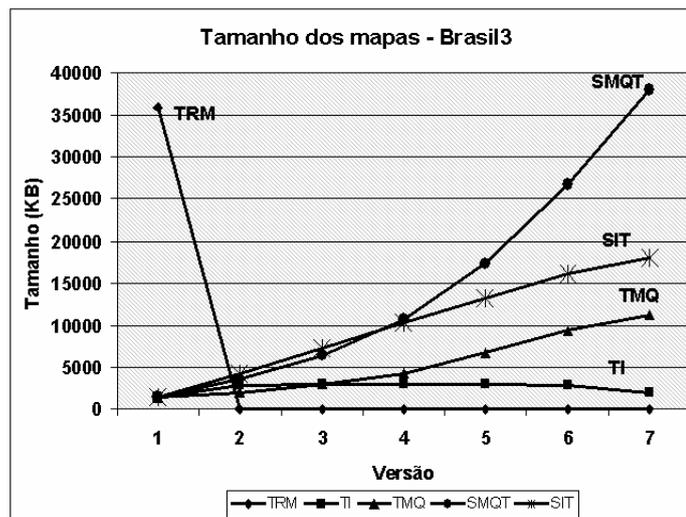


Figura 6.4. Gráfico comparativo para o mapa Brasil3

Diante desses dados, evidenciam-se alguns fatos que justificam a eficiência do algoritmo em relação à economia do tráfego na rede. Os fatores escolhidos seguem uma função exponencial, justificada pelo fato de que há um grande número de detalhes a serem adicionados nas baixas resoluções, mas à medida que a resolução aumenta, os detalhes diminuem, e isso se dá de maneira exponencial. Testes empíricos foram realizados com um aumento linear da resolução dos mapas, e constatou-se que não existe um ganho significativo, como acontece quando se usa uma função exponencial para as resoluções. Isso acontece devido ao fato de que o aumento no número de detalhes não é grande o suficiente para suprir a sobrecarga dos incrementos, as informações de integração dos incrementos ao mapa. É como dar vários passos sem sair praticamente do lugar, dar quatro passos quando na verdade fosse possível dar apenas um. Nesse exemplo, os fatores aumentam sempre a uma taxa onde o mapa da resolução é o dobro da anterior.

Após avaliar os resultados obtidos, pode-se perceber que a quantidade de informação a ser transmitida para cada requisição do usuário consiste apenas de uma pequena porcentagem do tamanho original do mapa. Isso torna possível a utilização de grandes volumes de mapas mesmo em ambientes com baixa velocidade de transmissão. Além disso, percebe-se que o tamanho dos incrementos diminui à medida que a resolução do mapa aumenta. Isso é uma característica bastante interessante, já que o usuário do sistema GIS consegue cada vez mais detalhes com cada vez menos dados transmitidos, em outras palavras, menos informações são necessárias para cada alcançar níveis cada vez maiores de detalhes.

Comparando o esquema de transmissão progressiva com o envio do mapa quantizado para cada resolução, pode-se inferir que a taxa de compressão entre os incrementos do mapa e o mapa quantizado é sempre menor que 1, o que mostra que, no pior caso, ambas as abordagens possuem um desempenho equivalente, porém, o método desenvolvido nesse trabalho possui uma grande vantagem em relação à quantização, ao analisar o processamento de níveis progressivos de resolução do mapa.

Outra importante característica do método é a eficiência que o sistema alcança à medida que são necessários mais e mais detalhes. Quanto mais aumenta a resolução do mapa, maior é o ganho com a utilização do método, comparada à simples utilização da técnica de quantização. Por exemplo, na Tabela 2, se forem requisitados os mapas nos níveis 1, e em seguida os detalhes dos níveis 2, 3 e 4, seriam transmitidos um total de 7789 KB de dados com o uso da técnica proposta. Caso fossem requisitados em seqüência os mapas quantizados (técnica de quantização), nos mesmos níveis, seriam trafegados cerca de 12343 KB de dados, e o ganho é ainda maior caso o usuário necessite de dados mais e mais detalhados. Nesse cenário hipotético, o usuário não sabe, a priori, qual das versões do mapa contém os dados com os detalhes suficientes que ele deseja observar. Daí a justificativa principal do uso da técnica de transmissão progressiva: à medida que são realmente necessários mais detalhes, as requisições são feitas; é um processo sob demanda, aumentando a eficiência e usabilidade do sistema. Isso se deve principalmente ao tempo de resposta, que é menor quando se entrega inicialmente versões menos detalhadas do mapa e posteriormente os incrementos, num processo que pode ser descrito como “entregue os dados aos poucos, à medida que sejam necessários”.

A Figura 6.5 apresenta as curvas obtidas para a taxa de compressão dos mapas em relação ao número de *pixels* (a resolução do mapa). Pode ser observado que as curvas apresentam uma geometria semelhante, e a largura de banda máxima necessária progressivamente diminui quando o usuário aumenta a resolução do mapa, requisitando mais detalhes. Além disso, o esquema proposto fornece em cada passo uma compressão mínima de 16%, evidenciando que possibilita um uso da largura de banda eficiente.

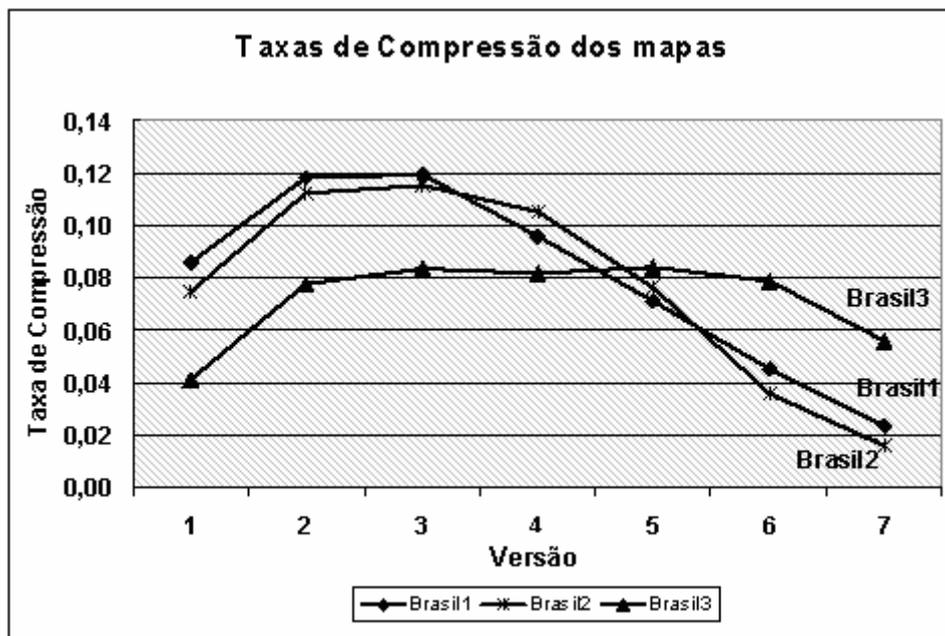


Figura 6.5. Taxas de Compressão dos mapas

Com relação à geração dos mapas, alcançou-se uma média de 260 KB/s em relação ao tamanho do mapa real, ou seja, 1 segundo para cada 260 KB do mapa real, para gerar o mapa em baixa resolução e todos os seus incrementos.

6.3 Avaliação dos clientes desktop e web

A fim de evidenciar a flexibilidade oferecida pela tecnologia de *Web Service*, foram implementados um cliente Web e um cliente *desktop*. Os testes

apresentados a seguir têm o objetivo de comparar as duas abordagens de transmissão progressiva, utilizando os mesmos mapas e a mesma técnica. A primeira abordagem é a utilizada pelo iGIS atualmente (cliente Web), adaptada às chamadas de *GIS Services*. O cliente *desktop* é totalmente independente do cliente Web previamente existente no iGIS, e utiliza apenas as chamadas ao *GIS Service* implementado, não tendo, portanto, qualquer outra ligação direta com o *framework* iGIS, como pode-se observar na Figura 5.18. O objetivo desta avaliação é comparar o desempenho das duas abordagens e observar o ganho que se obteve com a técnica de *GIS Services*, que garante flexibilidade e independência de plataforma e de linguagem de programação ao cliente.

Medida		Brasil1	Brasil2	Brasil3
Tamanho original (KB)		9527	19139	35847
Tamanho quantizado (KB)		3466	6728	11223
Tempo de Transm. do Mapa Original (s)	a 5KB/s (s)	1905,4	3827,8	7169,4
	a 12KB/s (s)	793,9	1594,9	2987,3
	a 25KB/s (s)	381,1	765,6	1433,9
	a 50KB/s (s)	190,5	382,8	716,9
Tempo de Transm. do Mapa Quant.(s)	a 5KB/s (s)	693,2	1345,6	2244,6
	a 12KB/s (s)	288,8	560,7	935,3
	a 25KB/s (s)	138,6	269,1	448,9
	a 50KB/s (s)	69,3	134,6	224,5

Tabela 6.4. Tamanho dos mapas e respectivos tempos de transmissão

Na análise, foram utilizados os mesmos mapas descritos na seção anterior: Brasil1, Brasil2 e Brasil3. Os tamanhos reais e quantizados dos mapas estão dispostos na Tabela 6.4. O tamanho real não utiliza nenhuma técnica para transmissão eficiente dos mapas e o tamanho quantizado diz respeito ao mapa que é gerado por meio da técnica de quantização, para a resolução máxima escolhida. Além disso, estão dispostos na tabela os tempos de transmissão desses mapas

(mapa original e mapa quantizado), a fim de comparações com outras tabelas, que mostram dados relativos a esses mapas com a utilização da técnica de transmissão progressiva nas duas diferentes abordagens (cliente *javascript* e Cliente *desktop*).

Pretende-se avaliar o desempenho da arquitetura com cliente *desktop* e com o cliente Web com processamento *javascript*. Isso pode ser constatado nas tabelas Tabela 6.5, Tabela 6.6 Tabela 6.7. A coluna **Incr.** representa o tamanho dos incrementos para uma determinada versão do mapa. A coluna **TJS** é o tempo de processamento do cliente *javascript*, enquanto que a coluna **TD** representa o tempo de processamento do cliente *desktop*. A última coluna, **TJS / TD**, é uma razão entre os tempos das duas abordagens, e representa o ganho da abordagem do cliente *desktop* em relação ao cliente *javascript*, significando quantas vezes uma é mais rápida que a outra.

Brasil1	Resolução	Incr. (KB)	TJS (s)	TD (s)	TJS / TD (ganho)
Versão 1	300x300	819	0	0	0
Versão 2	600x600	1126	16,1	3,0	5,37
Versão 3	1200x1200	1137	16,7	3,1	5,39
Versão 4	2400x2400	916	17,2	3,3	5,21
Versão 5	4800x4800	676	17,7	3,3	5,36
Versão 6	9600x9600	430	18,1	3,3	5,48
Versão 7	19200x19200	226	18,1	3,3	5,48

Tabela 6.5. Tempos de processamento entre os clientes (mapa Brasil1)

O tempo de processamento para o mapa na versão 1 é igual a zero pois, na primeira versão é enviado um mapa em baixa resolução ao cliente, e nenhum processamento de integração é necessário para renderizar o mapa. Observa-se também que o tempo de renderização dos mapas foi desprezado, pois não apresentava muitas variações entre uma abordagem e outra, além de ser relativamente pequeno, comparado com os tempos de processamento.

Na Tabela 6.5 observa-se que o tempo de processamento do cliente *desktop* é, em média, 5,3 vezes mais rápido que o cliente *javascript*.

Brasil2	Resolução	Incr. (KB)	TJS (s)	TD (s)	TJS / TD (ganho)
Versão 1	300x300	1427	0	0	0
Versão 2	600x600	2147	70,4	7,3	9,64
Versão 3	1200x1200	2203	77,7	7,7	10,09
Versão 4	2400x2400	2012	80,0	8,2	9,76
Versão 5	4800x4800	1462	81,7	8,4	9,73
Versão 6	9600x9600	690	83,5	8,3	10,06
Versão 7	19200x19200	309	84,8	8,1	10,47

Tabela 6.6. Tempos de processamento entre os clientes (mapa Brasil2)

Na Tabela 6.6, que analisa o mapa Brasil2 é pouco mais que o dobro do mapa Brasil1, e conseqüentemente, seus incrementos a serem processados também são maiores, percebe-se que o tempo de processamento do cliente *desktop* é, em média, 6,8 vezes mais rápido que o cliente *javascript*.

Brasil3	Resolução	Incr. (KB)	TJS (s)	TD (s)	TJS / TD (ganho)
Versão 1	75x75	1476	0	0	0
Versão 2	150x150	2789	509,8	16,5	30,90
Versão 3	300x300	2988	696,5	16,5	42,21
Versão 4	600x600	2925	716,7	17,3	41,43
Versão 5	1200x1200	3016	718,4	18,1	39,69
Versão 6	2400x2400	2830	688,1	18,7	36,80
Versão 7	4800x4800	2007	706,6	18,6	37,99

Tabela 6.7. Tempos de processamento entre os clientes (mapa Brasil3)

Já na Tabela 6.7, que apresenta um aumento ainda maior dos incrementos e mapas a serem processados, observa-se um ganho do cliente *desktop* de médio de 38 vezes, comparado ao cliente Web. Conclui-se com isso que, quanto maior a quantidade de dados a serem processados no processo de integração dos incrementos ao mapa no cliente, a capacidade de processamento pode tornar-se um fator determinante para a eficiência do sistema.

Além das análises com relação ao tempo de processamento nos cliente *desktop* e *web*, obteve-se algumas informações adicionais a respeito do ganho real do cliente *desktop* (**D**) comparado com a utilização de clientes *Web javascript* (**JS**), mensurando o tempo de resposta (tempo de transmissão + tempo de processamento). Para tanto, foram simulados os resultados em redes com larguras de banda iguais a 5, 12, 25 e 50 KB/s. Os valores para cada rede e cada mapa estão ilustrados nas tabelas a seguir.

Brasil1	Tempo de Transmissão + Processamento (em s)											
	5 KB/s			12 KB/s			25 KB/s			50 KB/s		
	JS	D	JS/D	JS	D	JS/D	JS	D	JS/D	JS	D	JS/D
Versão 1	166,2	166,2	1,00	69,3	69,3	1,00	33,2	33,2	1,00	16,6	16,6	1,00
Versão 2	241,3	228,2	1,06	109,9	96,8	1,14	61,1	48,0	1,27	38,6	25,5	1,51
Versão 3	244,1	230,5	1,06	111,5	97,9	1,14	62,2	48,6	1,28	39,4	25,8	1,53
Versão 4	200,4	186,5	1,07	93,5	79,6	1,17	53,8	39,9	1,35	35,5	21,6	1,64
Versão 5	152,9	138,5	1,10	74,0	59,6	1,24	44,7	30,3	1,47	31,2	16,8	1,86
Versão 6	104,1	89,3	1,17	53,9	39,1	1,38	35,3	20,5	1,72	26,7	11,9	2,24
Versão 7	63,3	48,5	1,31	36,9	22,1	1,67	27,1	12,3	2,20	22,6	7,8	2,89

Tabela 6.8. Tempos de resposta para o mapa Brasil1

Na Tabela 6.8 observa-se que, em uma rede ideal de 5 KB/s, o ganho do cliente *desktop* em relação ao cliente *web* cresce com o aumento do nível de detalhes, chegando a ter um tempo de processamento 1,33 vezes menor que o cliente *javascript*. O ganho do cliente *desktop* aumenta com o aumento da velocidade de transmissão, uma vez que esta vai deixando de ser, gradativamente, um fator de impacto negativo para o sistema, e em altas velocidades de transmissão, o que interfere mais no tempo de resposta é a capacidade de processamento do cliente, no caso, o cliente *desktop* leva vantagem sobre o cliente *web*.

Na Tabela 6.9 observam-se as mesmas características da tabela anterior. Por exemplo, o aumento do nível de detalhes resulta em um aumento no ganho do cliente *desktop*, uma vez que, com mais dados a serem processados (mapas e

incrementos), maior é o ganho de quem tem mais capacidade de processamento. Devido a esse fato, nota-se ainda que os ganhos obtidos no mapa Brasil2 são maiores do que com o mapa Brasil1, já que este tem apenas a metade do tamanho do mapa Brasil2, e conseqüentemente, tamanho de incrementos e mapa-base maiores, demandando maior capacidade de processamento.

Brasil2	Tempo de Transmissão + Processamento (em s)											
	5 KB/s			12 KB/s			25 KB/s			50 KB/s		
	JS	D	JS/D	JS	D	JS/D	JS	D	JS/D	JS	D	JS/D
Versão 1	285,4	285,4	1,00	118,9	118,9	1,00	57,1	57,1	1,00	28,5	28,5	1,00
Versão 2	499,8	436,7	1,14	249,3	186,2	1,34	156,3	93,2	1,68	113,3	50,2	2,26
Versão 3	518,3	448,3	1,16	261,3	191,3	1,37	165,8	95,8	1,73	121,8	51,8	2,35
Versão 4	482,4	410,6	1,17	247,7	175,9	1,41	160,5	88,7	1,81	120,2	48,4	2,48
Versão 5	374,1	300,8	1,24	203,5	130,2	1,56	140,2	66,9	2,10	110,9	37,6	2,95
Versão 6	221,5	146,3	1,51	141,0	65,8	2,14	111,1	35,9	3,09	97,3	22,1	4,40
Versão 7	146,6	69,9	2,10	110,6	33,9	3,27	97,2	20,5	4,75	91,0	14,3	6,37

Tabela 6.9. Tempos de resposta para o mapa Brasil2

Assim como na tabela anterior, com o aumento da velocidade de transmissão, ocorre um aumento do ganho do cliente *desktop* em relação ao cliente web, novamente pelo fato de que o impacto positivo da capacidade de processamento aumenta com o aumento da capacidade de largura de banda, uma vez que, nesses casos, a transferência de dados não pode mais ser considerada um gargalo tão grande para o sistema. Tais fatos podem ser constatados, de maneira ainda mais evidente na Tabela 6.10.

Brasil3	Tempo de Transmissão + Processamento (em s)											
	5 KB/s			12 KB/s			25 KB/s			50 KB/s		
	JS	D	JS/D	JS	D	JS/D	JS	D	JS/D	JS	D	JS/D
Versão 1	295,2	295,2	1,00	123,0	123,0	1,00	59,0	59,0	1,00	29,5	29,5	1,00
Versão 2	1067,6	574,3	1,86	742,2	248,9	2,98	621,4	128,1	4,85	565,6	72,3	7,82
Versão 3	1294,1	614,1	2,11	945,5	265,5	3,56	816,0	136,0	6,00	756,3	76,3	9,92
Versão 4	1301,7	602,3	2,16	960,5	261,1	3,68	833,7	134,3	6,21	775,2	75,8	10,23
Versão 5	1321,6	621,3	2,13	969,7	269,4	3,60	839,0	138,7	6,05	778,7	78,4	9,93
Versão 6	1254,1	584,7	2,14	923,9	254,5	3,63	801,3	131,9	6,08	744,7	75,3	9,89
Versão 7	1108,0	420,0	2,64	873,9	185,9	4,70	786,9	98,9	7,96	746,7	58,7	12,71

Tabela 6.10. Tempos de resposta para o mapa Brasil3

Na comparação dos tempos presentes nas Tabelas Tabela 6.8, Tabela 6.9 e Tabela 6.10, e com os tempos da Tabela 6.4, tempos de transmissão do mapa original e do mapa quantizado, observa-se o ganho de usabilidade e eficiência da arquitetura proposta. Nela, um grande passo (transmissão do mapa completo, num tamanho enorme) é dividido em vários outros passos (transmissão dos incrementos), adequando-se às necessidades do usuário. Isso pode ser visualizado na Figura 6.6.

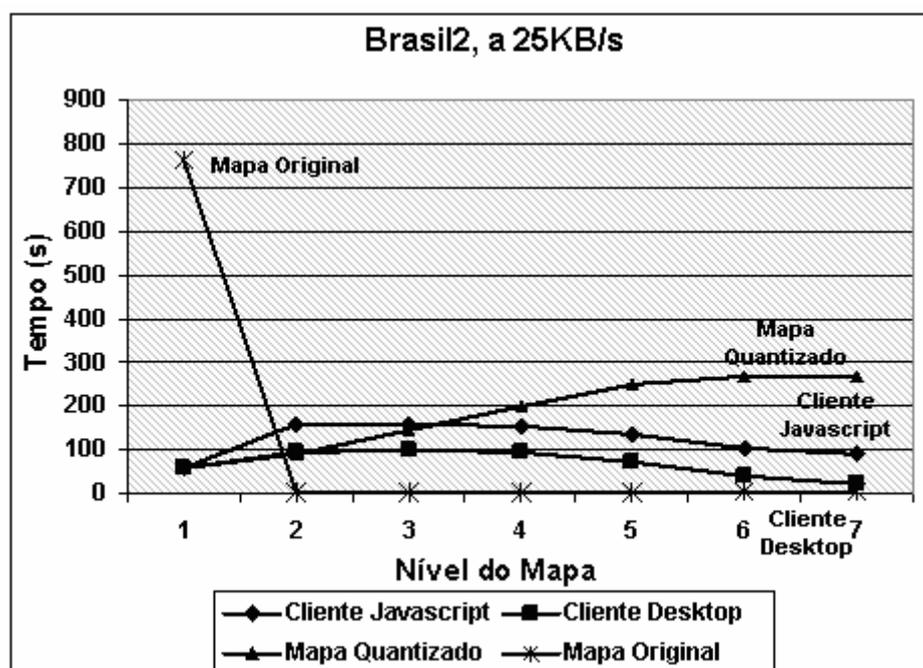


Figura 6.6. Tempo de transmissão para o mapa Brasil2, a 25KB/s.

O ganho do cliente *desktop* em relação ao cliente Web é observado comparando os tempos dos clientes em uma determinada taxa. Por exemplo, no mapa Brasil2, o cliente *desktop* tem um ganho total de até 6,37 vezes sobre o cliente Web.

No caso da transferência do mapa real, embora não requeira a obtenção de detalhes e processamento posteriores, a transferência do mapa real inicial (e único) demora demasiadamente, sendo este o principal problema de transmissão de grandes mapas na Internet. Comparando as técnicas de transmissão progressiva

com a de quantização, verifica-se que conforme são necessários mais detalhes, é mais vantajoso utilizar a técnica progressiva.

Observa-se que o ganho do cliente *desktop* torna-se relativamente mais evidente com o aumento da largura de banda, pois o tempo de transmissão dos incrementos, que é constante para as duas abordagens de transmissão progressiva, causa um impacto maior na aplicação que o tempo de processamento dos incrementos, para pequenas taxas de transmissão. Assim, para baixas velocidades de transmissão, um cliente com grande capacidade de processamento não é capaz de compensar o problema da transmissão eficiente de dados, o que, de certa forma, parece óbvio. Por outro lado, o uso da técnica de transmissão progressiva (independente do cliente) para baixas velocidades de transmissão gera uma grande eficiência no sistema, ao comparar-se a técnica de transmissão progressiva com a de quantização, por exemplo.

Diante dos testes e da arquitetura desenvolvida, foi possível observar algumas características e vantagens da utilização dos *GIS Services* na implementação de um Servidor de Mapas com transmissão progressiva. Alcançou-se maior flexibilidade de implementação de clientes, em diferentes plataformas, independente de linguagem de programação. Além disso, a característica de assincronismo adaptou-se muito bem ao problema de transmissão progressiva, visto que, enquanto espera pela transmissão dos detalhes das imagens, o cliente pode realizar outras tarefas em paralelo, como recepção e processamento dos detalhes já recebidos anteriormente. Outro benefício alcançado com os *Web Services* é prover ao cliente uma maior capacidade de processamento, pelo uso de aplicações *desktop*, em relação a scripts que executam em navegadores.

7 CONCLUSÃO

Neste trabalho foi apresentada uma arquitetura de *GIS Services* para um servidor de mapas com suporte a transmissão progressiva, utilizando o *framework* iGIS como arquitetura base para sua implementação.

O método proposto permite ao desenvolvedor GIS planejar uma discretização *offline* do mapa para vários níveis de detalhes, fornecendo ao usuário final da aplicação um ambiente mais eficiente para visualizar e interagir com os dados geográficos, especialmente considerando ambientes de baixas velocidades de transmissão.

O trabalho é baseado na resolução do dispositivo de visualização. Os elementos do mapa são simplificados em representações compactas do mapa, de maneira a serem transmitidos progressivamente, aumentando o nível de detalhes e a resolução do mapa incrementalmente, tendo como princípio transmitir apenas os dados que possam ser visíveis para aquela versão, mais detalhada, do mapa.

O algoritmo de subdivisão do mapa para transmissão progressiva e de integração dos detalhes com o mapa em baixa resolução é uma contribuição original, e a principal do presente trabalho.

Foi demonstrada a utilização dos *Web Services* em Sistemas de Informação Geográfica a fim de prover a tais sistemas as vantagens dos *Web Services*, de acordo com as características intrínsecas dos SIGs. Foi ilustrada a arquitetura proposta do *GIS Service* de transmissão progressiva, seu funcionamento e detalhados os métodos do serviço web implementado.

A arquitetura proposta permite a utilização de mapas vetoriais por clientes diversos, independente de plataforma ou linguagem de programação. Além disso, a

característica de transmissão progressiva provê uma maior usabilidade e melhor tempo de resposta para a aplicação cliente visualizar e interagir com os mapas, principalmente em casos de baixa velocidade de transmissão.

O principal objetivo do trabalho é a diminuição do tempo de resposta experimentado pelo usuário final do sistema GIS. A fim de avaliar o impacto da arquitetura implementada na visualização de mapas em ambiente Web, foram realizados vários testes baseados no tamanho dos mapas, tempos de processamento e tempos de resposta do sistema em diferentes configurações de ambientes de rede, a fim de comparar a técnica desenvolvida com a técnica de quantização e com o sistema sem o uso de nenhuma técnica de otimização.

Os resultados mostraram que a quantidade de informação transmitida diminuiu substancialmente no esquema proposto, sendo muito eficiente para sistemas Internet GIS e GIS móvel.

Como trabalhos futuros, pode-se aprimorar ainda mais a arquitetura de transmissão de mapas, agregando a ela outras técnicas existentes como, por exemplo, a segmentação do mapa em várias regiões de interesse, uma vez que, para o cliente GIS interessado em apenas uma pequena parte do mapa, apenas aquela parte possa ser transmitida progressivamente a ele, e não o mapa completo, aumentando ainda mais a eficiência do sistema. Técnicas de *cache* e provisionamento podem também ser incorporadas à arquitetura de transmissão progressiva, uma vez que são métodos que podem funcionar independentemente uns dos outros.

Pode-se ainda estudar estratégias de armazenamento em Banco de Dados da informação para transmissão progressiva evitando o processo de geração *offline* dos mapas.

REFERÊNCIAS

ALAMEH, N. **Chaining Geographic Information Web Services**. IEEE Internet Computing, Vol. 07, pp. 22-29, September, 2003.

ARCNEWS, **Scalable Internet Mapping Solutions from ESRI: Maps on the New Frontier**, ArcNews, Fall, 1997.

Axis. Apache Software Foundation. **The Axis2 project**. Disponível em <http://ws.apache.org/axis2>. Acessado em 26/10/2006. 2005.

BAPTISTA, C. S., LEITE JR., F. L., SILVA, E. R., PAIVA, A. C.: **Using Open Source GIS in e-Government Applications**, Lecture Notes in Computer Science, DEXA e-GOV 2004.

BAPTISTA, C. S., NUNES, C. P., SOUSA, A. G., SILVA, E. R., LEITE JR., F. L., PAIVA, A. C. **On Performance Evaluation of Web GIS Applications**. Proceedings of the 16th International Workshop on Database and Expert Systems Applications (DEXA'05). 2005.

BERNON, N. **Web-Based Demonstrations of Line Simplification Algorithms**. Project Report. 2005.

BERTIN, J. **Semiology of Graphics**. University of Wisconsin Press, Madison, Wisconsin. 1983.

BERTOLOTTO, M. EGENHOFER, M.J. **Progressive Vector Transmission**. In: Proceedings ACMGIS'99, Kansas City, MO, pp.152-157. 1999.

BERTOLOTTO, M. EGENHOFER, M.J., **Progressive Transmission of Vector Map Data over the World Wide Web**. Geoinformatica, 5(4), pp. 345-373. 2001.

BUSCHMANN, F., MEUNIER, R., ROHNERT, H., SOMMERLAD, P., STAL, M. **Pattern-Oriented Software Architecture: A System of Patterns**. John Wiley & Sons Ltd., 1996.

BUTTENFIELD, B. **Transmitting Vector Geospatial Data across the Internet**. In: M. J. Egenhofer and D. M. Mark (editors), Second International Conference, GIScience 2002, Lecture Notes in Computer Science, pp. 51-64, Berlin. 2002.

CECCONI, A. **Integration of cartographic generalization and multi-scale**

databases for enhanced web mapping. Dissertação. Zurich, 2003.

CHAN, E. P. F., CHOW, K. **On Multi-Scale Display of Geometric Objects**, in Data and Knowledge Engineering (DKE) 40, 2002, pp.91-119. 2002.

CHENG, C., LU F. **Building A Multi-Granularity Based Spatial Database.** Geoinformatics 2004 Proc. 12th Int. Conf. on Geoinformatics – Geospatial Information Research: Bridging the Pacific and Atlantic University of Gävle, Sweden, 7-9 June 2004.

CHO, Y.S., KIM, H.Y., KIM, J.H., BAE, H.Y., **The Design and Implementation of Componentized Web-enabled GIS**, ICIMU 98, Setembro 1998.

COUCLELIS, H. **Worlds of information: The geographic metaphor in the visualization of complex information.** Cartography and Geographic Information Systems v25 p209-220, 1998.

COSTA, D. C., PAIVA, A. C., TEIXEIRA, M. M., BAPTISTA, C. S., SILVA, E. R.: **A Progressive Transmission Scheme for Vector maps in low-bandwidth environments based on device rendering.** 3rd International Workshop on Conceptual Modeling for Geographic Information Systems (CoMoGIS'06), ER2006, 2006.

COWEN, D. J. **GIS versus CAD versus DBMS: What are the differences?** Photogrammetric Engineering and Remote Sensing, 54(11):1551-1555, Novembro 1988.

DAVIS JR., C. A., QUEIROZ, G. R. **Algoritmos geométricos e relacionamentos topológicos**, Bancos de Dados Geográficos, p. 43-83. MundoGEO, Curitiba, 2005.

DENT, B. D. **Principles of Thematic Map Design.** Addison-Wesley, Reading, Mass. 1990.

DOUGLAS, D. H.; PEUCKER, T. K. **Algorithms for the reduction of the number of points required to represent a line or its caricature.** The Canadian Cartographer, v. 10, n. 2, p. 112-122, 1973.

ESRI White Paper, **The Future of GIS on the Internet**, ESRI Inc. June 1997.

ESRI, **ESRI Shapefile Technical Description**, disponível em <http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>, acessado em

26/10/2006. **1998.**

ESRI White Paper, **Map Generalization in GIS: Practical Solutions with Workstation ArcInfo Software**, ESRI Inc. July **2000.**

FERINGA, W. **File formats and plugins.** In: Webcartography : developments and prospects / M.J. Kraak and A. Brown (eds.) pp. 177-193. **2001.**

FERREIRA, C. C. F. **A Study about Vector Files for Map Visualization in the Web.** Tese de Mestrado (em Português) PUC, Rio de Janeiro, Brasil, **1998.**

FERRIS, C., FARRELL, J. **What Are Web Services**, Communications of the ACM 46(6), p. 31. **2003.**

FLEWELLING, D., EGENHOFER, M., **Using digital spatial archives effectively.** International Journal of Geographical Information Science, Vol 13(8). P.1-8, **1999.**

FRANK, A.U., TIMPF, S. **Multiple Representations for cartographic objects in a multi-scale tree - an intelligent graphical zoom.** In Computers and Graphics Special Issue on Modelling and Visualization of Spatial Data in GIS, 18 (6), pp: 823-829. **1994.**

GATTASS, M., FERREIRA, C. C. F., VILAR, A. S., GLASBERG, M. S. **Efficient Map Visualization on the Web.** I Workshop Brasileiro de Geoinformática - Campinas, SP, Brasil. **1999.**

HAN, Q., Bertolotto, M. **A Multi-level Data Structure for Vector Maps.** GIS'04, Washington, DC, USA. November 12-13, **2004.**

KOMZÁK, J., SLAVÍK, P. **Scaleable GIS Data Transmission and Visualisation.** Proceedings of the Seventh International Conference on Information Visualization (IV'03). **2003.**

KRAAK, JAN, M., Brown, A. **Web Cartography: Development and Prospects.** London: Taylor and Francis. **2001.**

LEE, C. H., CHEN, L., LEE, J. D., BAE, H. Y. **Content Adaptation and Transmission Strategy of Spatial Information for WWW and Mobile Applications.** Lecture Notes in Computer Science (Springer), ISSN: 3-540-40456-2, Vol.: 2713, pp: 12-22. **2003.**

LI, Z., OPENSHAW, S. **Algorithms for automated line generalization based on a natural principle of objective generalization.** International Journal of Geographic Information Systems, v. 6, n. 5, p. 373-389, **1992**.

LIANG, C., LEE, C. H., LEE, J. D., BAE, H. Y. **Scale-Dependent transmission of spatial vector data on the Internet.** The 3rd International Conference on Information Integration and Web-based Applications & Services, Austria, **2001**.

McMASTER, R. B.; SHEA, K. S. **Generalization in digital cartography.** Association of American Geographers, **1992**.

MIRANDA, R. A. V., BAPTISTA, C. S., ALMEIDA, R. R., CATÃO, B., PAZINATTO, E. **iGIS: um Framework para Sistemas de Informações Geográficas em N-Camadas usando um SGBD Objeto-Relacional,** IV Simpósio Brasileiro de Geoinformática - Caxambú, MG, Brasil, **2002**.

MUSTAFA, N., KOUTSOFIOS, E. **Hardware-Assisted View-Dependent Map Simplification.** SCG'01, Medford, Massachusetts, USA, June 3-5, **2001**.

OGC, Open Geospatial Consortium. **The OGC Abstract Specification,** Disponível em <http://www.opengeospatial.org/standards/as>, acessado em 28 de setembro de 2006. **2005**.

OH, Y. H., BAE, H. Y. **Minimizing Up-Front Data Transmission on Web Based GIS.** IEICE Transf. Inf. & Syst., Vol E85-D. No. 10 October, **2002**.

ORACLE, Oracle Database, <http://www.oracle.com>

PAIVA, A. C., SILVA, E. R., LEITE JR., F. L., BAPTISTA, C. S. **A Multiresolution Approach for Internet GIS Applications.** Proceedings of the 15th International Workshop on Database and Expert Systems Applications (DEXA'04). **2004**.

PANATKOOL, A., LAOVEERAKUL, S.: **Decentralized GIS Web Services on Grid, Proceedings of the Open source GIS.** GRASS Users Conference 2002, Italy, September **2002**.

PAPAZOGLU, M. P.: **Service -Oriented Computing: Concepts, Characteristics and Directions.** wise, p. 3, Fourth International Conference on Web Information Systems Engineering (WISE'03), **2003**.

PENG, Z.R., **An Assessment of the Development of Internet GIS,** Proc. of the

1997 ESRI User Conference, **1997**.

PETERSON, M. P. **Trends in Internet Map Use**. In: Proceeding of the 20th ICA/ACI Conference, pp. 2306-2312. Beijing. **2001**.

POORTEN, P. JONES, C., **Customisable Line Generalisation using Delaunay Triangulation**, CD ROM Proceedings of the 19th ICA conference Ottawa, section 8. **1999**.

POORTEN, P., ZHOU, S., JONES, C. **Topologically-Consistent Map Generalisation Procedures and Multi-scale Spatial Databases**. GIScience 2002, LNCS 2478, pp. 209–227, **2002**.

POSTGRESQL: Open Source Relational Database System, Disponível em <http://www.postgresql.org/>. Acessado em 26/10/2006.

PUPPO, E., DETTORI, G. **Towards a formal model for multiresolution spatial maps**. In 4th International Symposium on Large Spatial Databases, pages 152–169, **1995**.

REICHENBACHER, T. **The World in Your Pocket - Towards a Mobile Cartography**. Proceedings of the 20th International Cartographic Conference (CD-ROM), Beijing (China), **2001**.

RUAS, A., LAGRANGE, J. P. **Etat de l'art en generalisation & survey on generalization**. Technical Report DT-93-0538, IGN-France, **1993**.

SAYOOD, K. **Introduction to Data Compression**. Morgan Kaufmann, **1996**.

SCHUENCK, M., SOARES, V. G. **Mecanismos para intercâmbio de dados geográficos através de Web Services**, VI Simpósio Brasileiro de Geoinformática - SP, Brasil, **2004**.

SALLEM, M. S. **Compressão de Mapas para Visualização na Web**. Trabalho de Conclusão de Curso. (Graduação em Bacharelado Em Ciência da Computação) - Universidade Federal do Maranhão. 56 f. **2003**.

SLIWINSKI, A. **Toward Perceived Value-based Pricing of Geographic Information Services**. Proceedings of the 7th AGILE Conference on Geographic Information Science, pp.541-549, **2004**.

SPACCAPIETRA, S., PARENT C., VANGENOT, C. **GIS Databases: From Multiscale to MultiRepresentation.** In B. Choueiry and T. Walsh, editors, Proceedings of the 4th International Symposium on Abstraction, Reformulation, and Approximation, Volume 1864 of Lecture Notes in Artificial Intelligence, pages 57-70. Springer, July 2000.

TAO, V. **Online GIServices.** Journal of Geospatial Engineering, Vol. 3, No. 2, pp. 135-143, December, 2001.

TIMPF, S. **Cartographic Objects In A Multi-Scale Data Structure.** In Geographic Information Research: Bridging the Atlantic, edited by M. Craglia and H. Couclelis. London: Taylor&Francis. 1997.

TO, D. S. P., LAU, R. W. H., GREEN, M. **A Method for Progressive and Selective Transmission of Multi-Resolution Models,** VRST 99 London UK, 1999.

TORUN, A., KÖBBEN, B., LEMMENS, R. **Processing Spatial Data on the Internet.** International Archives of Photogrammetry and Remote Sensing. Vol. XXXIII, Part B6. Amsterdam 2000.

TSOU, M.H., BUTTENFIELD, B.P. **Client/Server Components and Metadata Objects for Distributed Geographic Information Services.** In Proceedings of the GIS/LIS'98, pp. 590-599, 1998.

TU, S., HE, X., LI, X., RATCLIFF, J. **A Systematic Approach to Reduction of User-Perceived Response Time for GIS Web Services.** GIS'01, Atlanta, Georgia, USA. November 9-10, 2001.

VAN ELZAKKER, C.P.J.M. **Users of maps on the web.** In: Webcartography : developments and prospects / M.J. Kraak and A. Brown (eds.) 2001. pp. 37-52. 2001.

VAN OOSTEROM, P. **Reactive Data Structures for Geographic Information Systems.** In Oxford, Oxford University Press. 1993.

VAN OOSTEROM, P. **GAP-tree, An Approach to On-the-Fly of Map Generalization of An Area Partitioning.** In Muller J. C, Lagrange J P , Weibel R (eds) GIS and Generalisation: Methodology and Practice, Taylor & Francis, London, pp 120 - 132. 1995.

VAN OOSTEROM, P.; SCHENKELAARS, V. **The development of an interactive**

multi-scale GIS. International Journal of Geographical Information Systems, v. 9, n. 5, p. 489-507, **1995**.

VANGENOT, C. **Multi-representation in spatial databases using the MADS conceptual model.** ICA Workshop on Generalisation and Multiple representation – Leicester 20-21 August, **2004**.

W3C – World Wide Web Consortium, **Web Services**, Disponível em <http://www.w3.org/2002/ws>, acessado em 26/10/2006. **2002**.

W3C – World Wide Web Consortium. **Scalable Vector Graphics (SVG) 1.1 Specification**, Disponível em <http://www.w3.org/TR/SVG11>, Acessado em 26/10/2006. **2003**.

W3C – World Wide Web Consortium, **Web Services Architecture**, Disponível em <http://www.w3.org/TR/2004/NOTE-ws-arch-20040211>. Acessado em 26/10/2006. **2004**.

WALDÉN, M. **Towards the integration of vector map graphics in mobile environments.** Master of Science Thesis. **2003**.

WEIBEL, R. DUTTON, G. H. **Generalising Spatial Data and Dealing with Multiple Representations.** In P. Longley, M. Goodchild, D. Maguire, and D. Rhind, editors, Geographic Information Systems - Principles and Technical Issues, Volume 1, pages 125--155. John Wiley & Sons, 2 edition, **1999**.

YANG, B.S., PURVES, R. S., WEIBEL, R. **Implementation Of Progressive Transmission Algorithms For Vector Map Data In Web-Based Visualization.** The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 34, Part XXX. **2004**.

YUMIN, T., TIANHE, C. **Web-based GIS services in participatory forest management in China.** Geoscience and Remote Sensing Symposium, 2004. IGARSS '04. Proceedings. 2004 IEEE International Volume 7, p.4795 - 4798, **2004**.

ZHOU, S., JONES, C. B. **Design and Implementation of Multi-scale Databases.** C.S. Jensen et al. (Eds.): SSTD 2001, LNCS 2121, pp. 365–384, **2001**.