19º SIMPÓSIO BRASILEIRO DE BANCO DE DADOS

III WORKSHOP DE TESES E DISSERTAÇÕES EM BANCO DE DADOS

21 de Outubro de 2004
Brasília – Distrito Federal – Brasil

Promoção

SBC – Sociedade Brasileira de Computação
Comissão Especial de Banco de Dados

ACM – Association for Computing Machinery
SIGMOD – Special Interest Group on Manangement of Data

Apoio

VLDB Endowment

Edição

Sandra de Amo (Universidade Federal de Uberlândia – UFU)

Organização

Murilo S. de Camargo (Universidade de Brasília – UnB)

Realização

Departamento de Ciência da Computação da Universidade de Brasília - UnB
Faculdade de Computação da Universidade Federal de Uberlândia - UFU

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Workshop Teses de Dissertações em Banco Dados(03.:2004 out. 21: Brasília).

Anais/Edição Sandra de Amo – Uberlândia: Faculdade de Computação da Universidade Federal de Uberlândia, 2004.
84 p.: il.

ISBN 85-7669-005-5

Conhecido também como WTDBD 2004.

1. Banco de Dados. I. Amo, Sandra de. II. WTDBD (03.: 2004: Brasília).

"Esta obra foi impressa a partir de originais entregues, já compostos pelos autores"

Capa: Fernando Ribeiro
Editoração: Diogo Rispoli, Hugo Lousa, Wantuil Firmiano Jr.

Apresentação

O Workshop de Teses e Dissertações em Bancos de Dados (WTDBD) ocorre no escopo das atividades do Simpósio Brasileiro de Bancos de Dados (SBBD), sendo promovido pela Sociedade Brasileira de Computação (SBC). Este evento constitui um fórum dedicado à apresentação e discussão de trabalhos de mestrado e doutorado em Banco de Dados sendo realizados no Brasil. O objetivo é propiciar um ambiente construtivo para discussões, onde não somente os alunos possam ter uma avaliação de seus trabalhos em fase de realização, mas igualmente tenham acesso a um panorama representativo da pesquisa em banco de dados no país, estimulando a integração e cooperação de pesquisadores nesta área, e dando maior visibilidade às pesquisas em andamento para a comunidade acadêmica e industrial.

O WTDBD deste ano, em sua terceira edição, teve um total de 24 submissões de trabalhos, sendo 21 de mestrado e 3 de doutorado. Destes trabalhos foram priorizados 12 para apresentação durante o workshop, dos quais 2 de doutorado e 10 de mestrado. Todos os trabalhos submetidos foram avaliados por pelo menos 3 pesquisadores qualificados. Procurou-se avaliar diferentes aspectos dos trabalhos, incluindo relevância, originalidade, clareza dos objetivos, metodologia, fundamentação teórica, apresentação e validação da proposta. Sabemos que a seleção de trabalhos para eventos desta natureza não é tarefa fácil, uma vez que implica na avaliação de pesquisa em andamento ou mesmo em estágio inicial. Desta forma, o objetivo maior não é aceitar ou rejeitar os trabalhos, mas sim criticar de forma construtiva os trabalhos submetidos, tentando apontar falhas e qualidades e dar sugestões que possam contribuir para o desenvolvimento da pesquisa.

O evento consiste na apresentação oral dos trabalhos selecionados, cada apresentação sendo seguida por questionamentos elaborados por uma banca de debatedores. As sessões de apresentações são intercaladas por palestras ministradas por pesquisadores abordando temas relacionados a metodologia científica em banco de dados, técnicas de redação a apresentação oral de artigos em ciência da computação. Esperamos que as atividades deste evento sejam frutíferas para os participantes sobretudo no que se refere às sugestões e críticas fornecidas pelos debatedores, que estas sirvam para melhorar e aprofundar aspectos de seus trabalhos. Também esperamos a participação ativa dos alunos durante a apresentação dos trabalhos, que ao final do evento possam identificar as principais áreas de pesquisa em andamento no país e delinear as áreas de atuação de suas pesquisas futuras.

Agradecemos a todos os orientandos e orientadores que submeteram trabalhos ao WTDBD, aos pesquisadores que se dispuseram a participar dos debates após cada apresentação, aos membros do Comitê de Programa, bem como o Comitê Organizador do SBBD.

Desejamos a todos os participantes um ótimo workshop e que de resto, aproveitem bem esta semana em Brasília.

Brasília, outubro de 2004.

Sandra de Amo
Coordenadora do III WTDBD

III WORKSHOP DE TESES E DISSERTAÇÕES EM BANCOS DE DADOS

Coordenadora do Comitê de Programa

Sandra de Amo (UFU)

Comitê Diretivo do SBBD

Carlos Alberto Heuser (UFRGS)
Alberto Laender (UFMG)
Sérgio Lifschitz (PUC-Rio)
Marta Mattoso (UFRJ)
Ana Carolina Salgado (UFPE)

Membros do Comitê de Programa

Altigran Soares da Silva (DCC-UFAM)
Astério Tanaka (UNIRIO)
Caetano Traina Jr. (USP - Sao Carlos)
Denise Guliato (UFU)
Ilmério Reis da Silva (UFU)
José Palazzo M. de Oliveira (UFRGS)
Karin Becker (PUCRS)
Maria Luiza Campos (UFRJ)
Marina T. Pires Vieira (UNIMEP/UFSCAR)
Rodolfo S. Ferreira Resende (UFMG)
Sandra de Amo (UFU) Coordenador do Workshop
Ulrich Schiel (UFCG)

Avaliadores

Thais Helena Chaves de Castro (DCC/UFAM)
Autran Macedo (FACOM/UFU)
Cláudio de Souza Baptista (DSC/UFCG)
Paulo Figueiredo Pires, (IM/NCE- UFRJ)

Membros do Comitê de Organização
Organizing Committee Members

Alba Cristina M. A. de Mello, UnB, Brasília DF
Célia Ghedini Ralha, UnB, Brasília DF
Daniel Arruda Santos Anjos, UnB, Brasília DF
Diogo de Carvalho Rispoli, UnB, Brasília DF
Hugo Antônio de Azevedo Lousa, UnB, Brasília DF
João José Costa Gondim, UnB, Brasília DF
José Carlos Ralha, UnB, Brasília DF
Maria Emília, UnB, Brasília DF
Murilo S. de Camargo, UnB, Brasília DF
Rafael de Timóteo de Souza, UnB, Brasília DF
Ricardo P. Jacobi, UnB, Brasília DF
Ricardo Puttini, UnB, Brasília DF
Robson de O. Albuquerque, UnB, Brasília DF
Soemes Castilho Dias, UnB, Brasília DF
Wantuil Firmiano Júnior, UnB, Brasília DF

Sumário WTDBD / *Contents WTDBD*

Palestras / *Talks*

Artigos / *Articles*

PROPOSTAS DE TESES DE DOUTORADO

PROPOSTAS DE TESES DE MESTRADO

Orientações para Orientandos

Marta Mattoso
Programa de Engenharia de Sistemas e Computação - COPPE
Universidade Federal do Rio de Janeiro, Brasil
e-mail: marta@cos.ufrj.br

Resumo

O objetivo desta palestra é fazer uma apresentação de diversos pontos importantes no ciclo de vida de um aluno de pós-graduação. Devido à subjetividade envolvida nos passos que levam à conclusão de um mestrado ou doutorado com êxito, não será apresentado um manual, regras ou receitas de bolo. Serão abordados aspectos que tentam evidenciar o que se espera de um candidato ao mestrado ou doutorado. Inicialmente, será apresentada uma caracterização do trabalho de dissertação e tese em bancos de dados usando uma perspectiva histórica no Brasil. A palestra se concentrará em três aspectos: (i) desenvolvimento de pesquisas em banco de dados, (ii) redação da dissertação/tese, e (iii) exposição oral e defesa da dissertação/tese.

Serão apresentadas algumas categorias de pesquisas em bancos de dados e que tipo de validação deve ser realizado para apresentar os resultados da investigação científica nessas categorias. Para a redação, serão analisados pontos que evidenciam que o trabalho possui contribuições, compreendendo um grau satisfatório de atividades de pesquisa. Finalmente, serão apresentadas sugestões para que o candidato possa mostrar o grau de conhecimento adquirido e a compreensão da área de pesquisa em questão ao defender sua solução durante a exposição oral da tese/dissertação.

O primeiro passo para concluir o mestrado/doutorado, com êxito, é reconhecer que as atividades de redação (de artigos, da tese/dissertação) e de exposição oral demandam muito cuidado e tempo em sua preparação. Essas atividades não são mera conseqüência de um trabalho de pesquisa bem desenvolvido. Nesse sentido, vários textos são encontrados para auxiliar a execução dessas tarefas com êxito. A seguir são listados alguns desses documentos e ponteiros (na área de computação) que foram utilizados como base desta palestra.

1. ACM Crossroads Student Magazine, http://www.acm.org/crossroads/
2. desJardins, M. How to Succeed in Graduate School http://www.csee.umbc.edu/~mariedj/
3. Kitchenham, B. Pickard, L. Pfleeger, S.L. Case Studies for Method and Tool Evaluation, In: IEEE Software v.12(4), pp.52-62, 1995
4. Levine, S.J. Writing and Presenting Your Thesis or Dissertation (com tradução para o português), http://www.learnerassociates.net/dissthes/
5. Mattoso, M. Thesis guidelines, http://www.cos.ufrj.br/~marta
6. OpenDirectory Project's, How to http://dmoz.org/Reference/Education/How_To_Study/Postgraduate_Research/
7. Valduriez, P. Some Hints to Improve Writing of Technical Papers, http://www.sciences.univ-nantes.fr/info/perso/permanents/valduriez/attaches/hints.pdf
8. Zobel, J., Writing for Computer Science, http://www.justinzobel.com/

Redigindo artigos de Ciência da Computação: uma visão geral para alunos de mestrado e doutorado

Vanessa P. Braganholo Carlos A. Heuser André Reis

Instituto de Informática
Universidade Federal do Rio Grande do Sul (UFRGS)

[vanessa,heuser,andreis]@inf.ufrgs.br

Resumo

Escrever artigos é uma forma importante de transmitir conhecimento descoberto, e também é importante para a avaliação dos cursos de pós-graduação. Existem algumas técnicas que ajudam autores a escrever artigos coerentes e de boa qualidade. Como um exemplo, neste texto apresenta-se o método de escrita através de scripts, além de várias dicas e procedimentos gerais que auxiliam na tarefa de escrever um artigo. Finalmente, o texto trata também da participação em congressos - o que fazer para aproveitar todas as oportunidades que um congresso oferece. Este artigo pretende ser uma fonte de conselhos e de material bibliográfico, muito mais do que um exemplo formal de escrita.

1. Introdução

Escrever é um processo que exige criatividade, conhecimento, e alguma técnica. Para muitos alunos de Ciência da Computação, escrever é uma tarefa árdua. Muito mais fácil programar, não? No entanto, alunos de mestrado e doutorado não podem escapar de escrever, e devem se esforçar para que o resultado final (o texto) seja o melhor possível.

Para muitos, a primeira experiência com um texto mais longo é o trabalho de conclusão de curso da faculdade. Uma dissertação ou uma tese, no entanto, são muito mais do que um trabalho de conclusão. Essas precisam de mais cuidado, mais argumentação, uma sequência de raciocínio cuidadosa, e até uma certa didática para que o texto flua de forma natural. O mesmo acontece com artigos. Artigos científicos também exigem um cuidado especial, uma vez que eles serão julgados e comparados com vários outros antes de serem aceitos.

O objetivo deste trabalho é mostrar a importância de se escrever artigos, e também apresentar algumas dicas para a produção de bons artigos científicos. De modo especial, enfatizamos o método de escrita de artigos através de *scripts* proposto por Hirsch [16]. Além disso, fazemos algumas considerações sobre como encarar um congresso, já que a grande maioria da produção em Ciência da Computação está publicada em congressos.

O restante do texto está organizado como segue. A Seção 2 apresenta uma motivação para a produção de artigos, apresentado o Qualis, que é o método de avaliação da Capes. A Seção 3 apresenta um apanhado de dicas e considerações sobre escrita de artigos, apresentando o método de escrita através de scripts na Subseção 3.1. Na seção 4 são apresentadas dicas de como se portar em um congresso. Finalmente, a Seção 5 conclui, apresentando uma lista de bibliografia adicional.

2. Por quê escrever artigos?

Agora que você é um aluno de mestrado ou doutorado, provavelmente sabe que antes de concluir o curso deverá publicar artigos. Mas você já se perguntou por quê? Alguns vão dizer que é só mais um dos

requisitos do curso, e aceitar o fato sem se questionar mais. No entanto, existe muito mais por trás disso.

Primeiro, publicações são a forma mais rápida de divulgar o conhecimento produzido pelo seu trabalho. A ciência avança através de pesquisas, e pesquisas são divulgadas através de publicações. De que adianta você descobrir a cura da AIDS ou um novo algoritmo de busca super rápido se ninguém (ou se só a banca da sua defesa) ficar sabendo disso?

Segundo, para se fazer pesquisa, é necessário dinheiro. Dinheiro para pagar bolsas, professores, papel para a impressora, equipamentos para o laboratório, limpeza das salas, sabonete no banheiro, etc. Sem dinheiro, não há pesquisa. No Brasil, a maior fonte de dinheiro para a pesquisa nas universidades é o Governo Federal, através de seus órgãos de fomento Capes [8] e CNPq [10]. Estes órgãos são os responsáveis pelo repasse de verbas às universidades federais, bem como pela concessão de bolsas de estudos (mestrado, doutorado, recém-doutor, etc.). Mas como saber que soma em dinheiro cabe a cada universidade? Diferentemente do que muitos pensam, o dinheiro disponível não é repassado em cotas iguais para cada universidade. Ao contrário, cada grupo de Pós-Graduação recebe uma cota de acordo com a sua *nota* na avaliação da Capes. Essa nota é dada para todas as instituições de pesquisa que possuem cursos de Pós-Graduação, inclusive para as universidades particulares. Isso explica o fato algumas de universidades particulares também disporem de bolsas de estudo Capes e CNPq para serem distribuídas para seus alunos.

A cada 3 anos, uma comissão da Capes avalia os cursos de pós-graduação do país, atribuindo a elas uma nota de 1 a 7. Os critérios de avaliação são definidos por um documento chamado *Documento de Área* [9]. De acordo com o documento de área da Computação, no cálculo da nota de cada curso entram os seguintes itens:

1. (15%) Corpo Docente (qualificação, dedicação, etc);
2. (10%) Pesquisa (distribuição de docentes por área de pesquisa, adequação dos projetos de pesquisa, infra-estrutura disponível para pesquisa, etc);
3. (15%) Formação dos alunos (currículo, adequação de carga horária, quantidade de orientadores, etc)
4. (10%) Corpo discente (número de alunos por orientador, número de titulados, número de abandonos, etc)
5. (20%) Teses e Dissertações (tempo médio de titulação, número de publicações, etc)
6. (30%) Produção Intelectual (quantidade e regularidade das publicações, qualidade dos congressos e revistas, etc)

Através desses itens, vê-se que as publicações contam bastante na nota final dos cursos (itens 5 e 6). Mais importante ainda é o fato de que esses dois itens somados valem 50% da nota final do curso [9]. Portanto, as publicações influenciam diretamente na qualidade e continuidade dos cursos de pós-graduação do país.

Analisando os itens 5 e 6, pode-se pensar que o item *qualidade dos congressos e revistas* (no item 6) é bem subjetivo. Como medir a qualidade de uma conferência? O Comitê de Computação da Capes mede a qualidade das conferências através de vários critérios. Um deles é o índice de impacto do *CiteSeer* (CS) (`http://citeseer.nj.nec.com/impact.html`). Para os periódicos são usados o índice de impacto do ISI/JCR (*Journal Citation Records*) e o do CiteSeer. Para os congressos ou periódicos que não estão listados no CiteSeer ou JCR, a comissão da Capes atribui uma nota. Uma lista de tais notas está disponível em [2].

A classificação funciona da seguinte maneira. Congressos e revistas são divididos em duas categorias: *nacional* e *internacional*. Dentro de cada uma dessas categorias, uma publicação pode se enquadrar nos níveis A, B ou C. Como um exemplo, de acordo com [9], os critérios para publicações em congressos

são os seguintes[1]:

- Tipo A: CS $\geq$ 0.57;
- Tipo B: 0.11 $\leq$ CS $\leq$ 0.56;
- Tipo C: CS $\leq$ 0.10

Usando esses índices, é possível saber que um artigo publicado no VLDB em 2003 é classificado como *internacional nível A* (CS = 1.52), e que um artigo publicado no SBBD em 2003 é classificado como *nacional nível A* (de acordo com [2]), enquanto que um artigo publicado no CLEI em 2003 é classificado como *nacional nível C* (de acordo com [2]). Veja que apesar do CLEI ser um congresso internacional, ele tem apenas impacto regional, e portanto, é considerado *nacional*.

Mesmo que esta avaliação seja questionável, é importante ressaltar dois pontos: (i) ela existe, e você será julgado por ela; (ii) a existência dela é muito melhor que a ausência de critérios.

Agora que você já sabe por que deve publicar, e que as publicações são classificadas de acordo com a qualidade do veículo onde foram publicadas, na próxima seção apresentaremos algumas dicas de *como publicar*.

3. Escrevendo um artigo

Para escrever um bom artigo, deve-se levar em conta uma série de questões. Por mais simples e banais que muitas dessas questões possam parecer, elas são importantes para evitar que um artigo seja recusado por motivos não técnicos. Portanto, ao escrever um artigo, certifique-se de que:

1. O artigo obedece à formatação exigida pela revista/conferência para o qual ele será submetido (incluindo margens, número máximo de páginas e tamanho e tipo de fonte);
2. Verifique se a revista/conferência para a qual o artigo será submetido é apropriada. Uma boa forma de fazer isso é verificar se algum dos trabalhos relacionados ao seu artigo foi publicado nessa conferência/revista. Se não houver nenhum artigo relacionado publicado lá, provavelmente você está submetendo para o lugar errado.
3. Seja formal, não use gírias. Como um antiexemplo, veja este artigo. Ele não é formal, pois está se dirigindo ao leitor por "*você*". Em artigos científicos deve-se evitar se dirigir ao leitor como fazemos aqui, e evitar sempre de se usar a primeira pessoa (eu, nós). Este artigo não segue essas regras pois não é um artigo técnico. Ele foi formulado para ser um guia para alunos onde o objetivo era justamente criar a sensação de que os autores estão falando diretamente com o leitor. Uma exceção a esta regra está em artigos redigidos em inglês, onde admite-se o uso da primeira pessoa (*In this paper, we propose...*).
4. Certifique-se de que todas as figuras e tabelas estão legíveis, são necessárias e são citadas no texto. Cuidado especial para capturas de tela – verifique se todas as "letrinhas" podem ser lidas com facilidade.
5. Cuide para que seu texto esteja claro, tanto na exposição de idéias quanto na redação propriamente dita. Aconselha-se também usar corretor gramatical e ortográfico.

Os itens acima se referem a questões não técnicas. Alguns são senso comum, outros foram retirados de [17]. Para os que escrevem artigos em inglês, Li apresenta em [22] uma lista de erros usualmente cometidos em artigos técnicos escritos nesta língua, que vale a pena verificar.

No entanto, não adianta apenas seguir esses itens para que um artigo seja aceito. É necessário também que ele tenha qualidade técnica. Neste ponto, pode-se enfatizar o seguinte:

[1]CS é o índice de impacto do *CiteSeer*

1. Certifique-se de que a bibliografia relacionada é citada e analisada no artigo. Não conhecer um trabalho relacionado importante conta pontos negativos, e contribui para a rejeição do artigo;
2. Tenha certeza de citar e enfatizar a contribuição do artigo. Não se deve deixar esse trabalho para o revisor. Se ele não encontrar essa informação facilmente no artigo, ele provavelmente desistirá e rejeitará o artigo por não estar convencido de que este apresenta uma contribuição para a ciência;
3. Argumente. Capriche na motivação e convença que sua abordagem é boa.
4. Cuidado para não cometer erros de lógica! Veja se não está concluindo algo que não pode ser derivado a partir das premissas disponíveis [27]. Por exemplo:
   Premissa: se choveu, então o chão está molhado
   Conclusão: se não choveu, então o chão não está molhado
   A conclusão acima está completamente errada, pois ela não pode ser tirada a partir da premissa! Se não choveu, então nada se sabe. E se alguém molhou o chão com um balde d'água?
5. Não manipule resultados em seus experimentos. Isso é completamente antiético! [4, 18] e [19] são artigos que tratam do tema ética em pesquisas ou na profissão acadêmica. Além disso, [28] é um texto que fala sobre manipulação de dados, o qual pode ser usado para o bem ou para o mal. Usem para o bem!
6. Também é antiético diminuir o trabalho de outro pesquisador de forma ofensiva, ou agressiva. Se o trabalho em questão tem limitações, fale sobre elas sem desmerecer o trabalho.
7. Não desafie conhecimento pré-estabelecido sem provas! Tentar publicar um artigo que diz que as teorias de Einstein estão erradas, se não estiver bem fundamentado, é certamente uma fria. Isso até pode ser feito, mas não é para qualquer um.

Além de seguir as dicas acima, uma forma segura de avaliar se você escreveu um bom artigo é saber como ele será avaliado (o que será levado em conta pelos revisores). Artigos sobre o tema estão disponíveis em [17, 32, 33, 31, 3]. De modo geral, são avaliados os seguintes tópicos: mérito científico; clareza; referências; equilíbrio; adequação ao escopo; originalidade; motivação; tamanho; título; resumo; diagramas; figuras, tabelas e legendas; capturas de tela e gráficos; matemática; trabalhos relacionados; conclusão.

Finalmente, ao receber os comentários dos revisores, leve-os em consideração quando preparar a versão final do artigo (caso ele tenha sido aceito), ou quando for reescrevê-lo para enviar para outra conferência/revista (caso ele tenha sido rejeitado). Apesar dos revisores não serem muito bons em expressar o que querem em suas revisões (eles são ocupados e têm muitos artigos para avaliar), na grande maioria das vezes os comentários são pertinentes [31] e com certeza contribuirão para a melhoria de seu trabalho. Portanto, ao invés de se zangar com os comentários, arregace as mangas e capriche no trabalho!

3.1. Método de escrita através de scripts

Muitos autores não adotam nenhum método para escrever seus artigos. Eles apenas vão colocando as idéias no papel de modo desordenado, e, de algum modo, produzem um texto final. Essa falta de metodologia pode acarretar sérios problemas no texto final. Por exemplo, pode-se desviar do tema, repetir a mesma idéia várias vezes, não apresentar as idéias de forma clara, etc.

Existe uma técnica de escrita de artigos que tenta evitar que tais problemas aconteçam, e se acontecerem, propicia que eles sejam detectados o quanto antes. Trata-se da técnica de escrita através de scripts

Um estudo sobre algoritmos de ordenação

Resumo
1. Introdução
2. Uma classificação para os métodos de ordenação
3. Ordenação por inserção
3.1. Método da inserção direta (*Insertion Sort*)
3.2. Método da inserção por incrementos decrescentes (*Shell Sort*)
3.3. Método da inserção direta binária (*Binary Insertion Sort*)
4. Ordenação por troca
4.1. Método da bolha (*Bubble Sort*)
4.2. Método da troca e partição (*Quick Sort*)
5. Ordenação por seleção
5.1. Método da seleção direta (*Selection Sort*)
5.2. Método da seleção em árvore (*Heap Sort*)
6. Outros métodos
6.1. Método de intercalação (*Merge Sort*)
6.2. Método da distribuição de chave (*Bucket Sort*)
7. Comparação e considerações finais
Referências

Figura 1: Outline

[16]. Esta técnica se caracteriza pela escrita em etapas. A cada etapa, o texto vai sendo aprimorado e vai tomando forma, até a produção do texto final. Basicamente, o método defende a adoção de quatro etapas:

Etapa 1: Outline O Outline nada mais é do que a estrutura do artigo que se pretende escrever. Isto é, as seções e subseções que se planeja colocar no artigo.

Etapa 2: Script 1 O Script 1 utiliza a idéia de *pontos*. Para cada seção do artigo, enumera-se uma sequência de pontos que devem ser abordados (P1, P2, ...). Os pontos devem ser frases curtas e claras. Um bom Script 1 deve garantir que pessoas da mesma área de pesquisa escrevam textos muito semelhantes a partir da mesma lista de pontos.

Etapa 3: Script 2 No script 2, cada ponto é detalhado. A idéia é já produzir o texto final que descreve cada um dos pontos. Ao final de cada ponto, coloca-se uma marcação entre chaves, indicando a qual ponto uma frase ou parágrafo se refere.

Etapa 4: Artigo Final Seguindo o processo de elaboração de artigos através de scripts, o artigo final nada mais é do que a remoção das marcas do script 2.

Para exemplificar, analise as Figuras 1, 2, 3 e 4. Elas apresentam trechos de cada uma das etapas de escrita de um artigo fictício sobre algoritmos de ordenação de dados. O artigo não apresenta uma inovação, uma vez que ele faz um apanhado das propostas existentes na literatura. Ele se caracteriza, portanto, como um tutorial (*survey*). A Figura 1 apresenta a estrutura do artigo, com todas as seções que o artigo irá conter. A Figura 2 apresenta a lista de pontos que serão abordados em cada uma das seções. Os pontos estão ordenados, conduzindo as idéias para a produção do texto final. Por simplicidade, a figura mostra apenas os pontos do resumo. Na Figura 3, cada um dos pontos foi desenvolvido de modo a obter um texto definitivo. Note as marcas de delimitação dos pontos na figura. Elas facilitam a revisão

Um estudo sobre algoritmos de ordenação

Resumo
P1: Por que ordenar dados
P2: Funcionamento dos métodos de ordenação - entrada e saída
P3: Objetivo do artigo - apresentar os métodos existentes e compará-los
P4: Classificação dos algoritmos
1. Introdução
...

Figura 2: Script 1

Um estudo sobre algoritmos de ordenação

Resumo
Várias aplicações práticas requerem que dados sejam ordenados. Exemplos são extratos de contas correntes, listas de aniversários, entre outros. Diante disso, vários métodos de ordenação foram propostos na literatura. [*P1, Resumo*] Um algoritmo de ordenação de dados é capaz de receber uma sequência de números como entrada, e devolvê-la ordenada em ordem crescente na saída. [*P2, Resumo*] O objetivo deste artigo é apresentar os métodos de ordenação existentes na literatura, classificá-los e compará-los. [*P3, Resumo*] A classificação aqui apresentada separa os algoritmos existentes em quatro categorias: métodos de ordenação por inserção, métodos de ordenação pro troca, métodos de ordenação por seleção e outros métodos. [*P4, Resumo*]

1. Introdução
...

Figura 3: Script 2

do texto, já que permitem checar se cada ponto foi desenvolvido corretamente. Já a Figura 4 apresenta o texto final, sem as marcações do script 2.

O uso dos scripts facilita a correção de caminhos logo no início. Assim fica fácil saber se as idéias estão sendo apresentadas de forma coerente. Como exemplo, veja as Figuras 5 e 6. A Figura 5 mostra como um bom artigo deve ser: um encadeamento de idéias coerente que leva suavemente de um ponto inicial até um ponto final. Na Figura 6, tem-se um exemplo de um artigo de má qualidade. Apesar dos pontos inicial e final serem os mesmos, o caminho tortuoso percorrido mostra que as idéias foram apresentadas de forma confusa e desordenada, o que contribui para um resultado final ruim.

É importante ressaltar que o método de escrita através de scripts pode ser empregado em qualquer tipo de texto, e não só na redação de artigos. Por exemplo, ele pode ser usado na redação de sua dissertação, ou mesmo de um trabalho de disciplina ou de um romance. Emile Zola, por exemplo, fazia maquetes das cidades onde se passariam seus romances, para melhor planejá-los. Érico Veríssimo fez a planta de Antares para o livro "Incidente em Antares".

Um estudo sobre algoritmos de ordenação

Resumo

Várias aplicações práticas requerem que dados sejam ordenados. Exemplos são extratos de contas correntes, listas de aniversários, entre outros. Diante disso, vários métodos de ordenação foram propostos na literatura. Um algoritmo de ordenação de dados é capaz de receber uma sequência de números como entrada, e devolvê-la ordenada em ordem crescente na saída. O objetivo deste artigo é apresentar os métodos de ordenação existentes na literatura, classificá-los e compará-los. A classificação aqui apresentada separa os algoritmos existentes em quatro categorias: métodos de ordenação por inserção, métodos de ordenação pro troca, métodos de ordenação por seleção e outros métodos.

1. Introdução

...

Figura 4: Texto Final

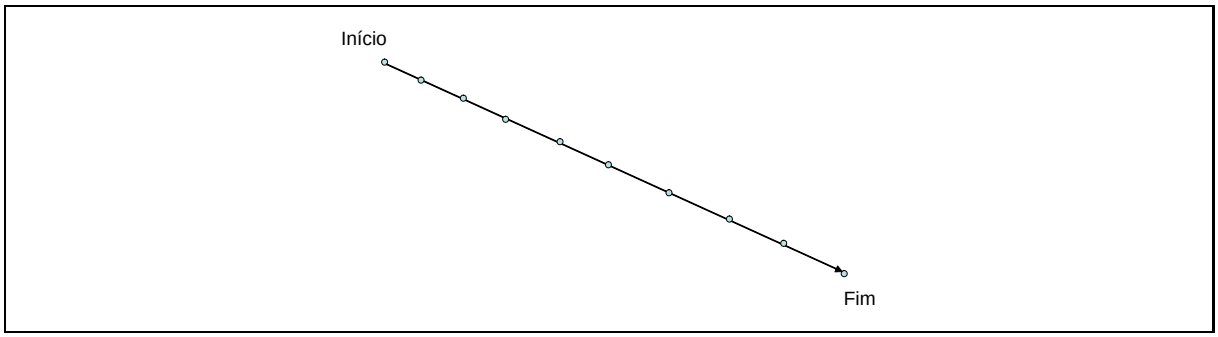


Figura 5: Exemplo de um bom artigo

4. Participando de conferências

Agora que seu artigo foi aceito, você deve ir ao congresso apresentá-lo. Aqui você encontra algumas dicas de como encarar esse desafio. Um texto mais completo sobre este e outros assuntos relacionados à vida de um mestrando ou doutorando pode ser encontrado em [26][2].

Um congresso não deve ser encarado como uma viagem de férias. Ao contrário, o congresso propicia um ambiente excelente para fazer contatos e conhecer as pesquisas mais atuais. Procure conversar com outros pesquisadores durante o congresso. Quando assistir a uma apresentação que seja interessante para o seu trabalho, procure o palestrante, faça perguntas, estabeleça um contato. Muitas parcerias de trabalho nascem assim.

Não tenha em mente assistir a todas as seções técnicas do congresso. Ficar fechado em uma sala o dia todo não vai te ajudar a fazer contatos. Assista aos trabalhos que mais lhe interessam, e aproveite o restante do tempo para conversar com outros pesquisadores. Mas cuidado, não exagere! Não queira ficar conhecido como "o mala caçador de contatos"!

Aproveite também tudo o que o congresso oferece. Vá ao coquetel e ao jantar. Em um ambiente mais descontraído é mais fácil iniciar uma conversa, conhecer pessoas novas. Esteja preparado para explicar

[2]Esta seção é apenas uma visão geral do conteúdo de uma das seções de [26]

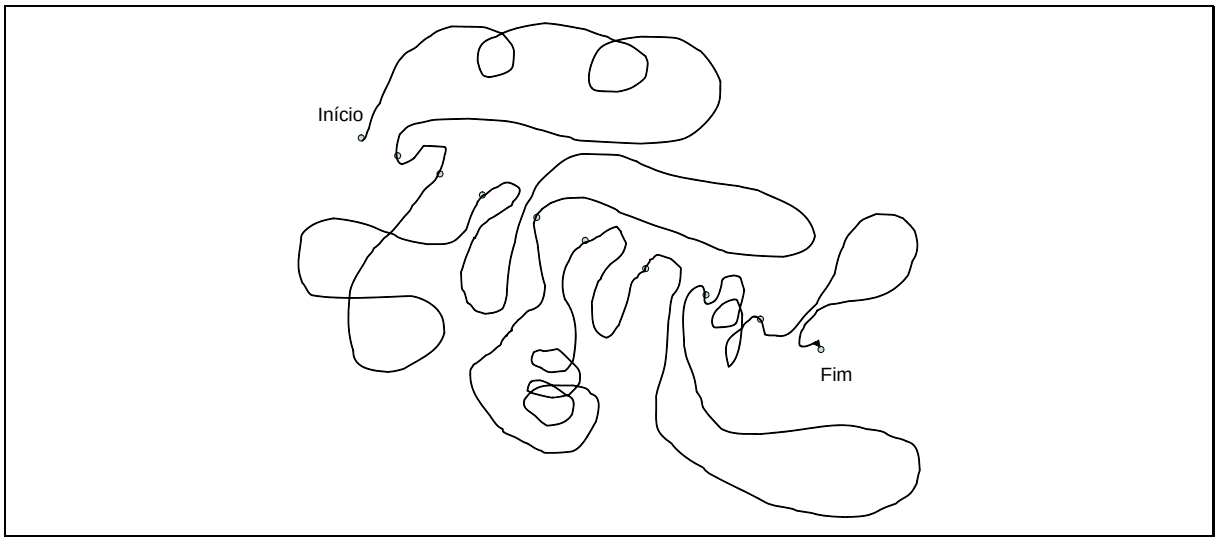


Figura 6: Exemplo de um artigo ruim

a contribuição principal do seu trabalho. Esse deve ser um resumo rápido, de uns 30 segundos, apenas para fazer com que a pessoa se interesse e volte a conversar com você mais tarde, ou leia seu artigo nos anais. É a famosa "apresentação de elevador". Use-a sempre que necessário.

Mais importante, se você vai apresentar seu artigo no congresso, ensaie antes! Esteja preparado! Procure respeitar o tempo de apresentação. Nada mais desagradável do que usar o tempo do colega seguinte! Mais desagradável ainda é deixar transparecer que você está "descobrindo" cada slide à medida que faz sua apresentação. Faça uma prévia com seu grupo de pesquisa antes da viagem, e siga as sugestões do grupo. Isso ajuda a melhorar a apresentação, e também a treinar o tempo de apresentação. Na preparação da apresentação, use letras grandes e muitas figuras. Procure não sobrecarregar os slides.

Enfim, saiba se portar durante o congresso, tente causar boa impressão. Os contatos feitos nos congressos com certeza serão muito úteis em sua carreira futura.

5. Considerações Finais

Neste artigo foi apresentada uma visão geral do processo de produção científica. Abordou-se desde a motivação da escrita de artigos, passando por dicas de como escrever um bom artigo, até a apresentação do artigo no congresso. Nesta última parte, no entanto, o foco não foi dado a como montar a apresentação em si, mas sim em como aproveitar o congresso para fazer contatos que serão importantes em sua carreira acadêmica futura. Existem vários artigos sobre como fazer uma boa apresentação que podem ser consultados caso necessário [22, 30, 7, 29].

Gostaríamos também de citar outras referências importantes, sobre as quais não temos espaço para falar. Grande parte dessa bibliografia está comentada em [5].

- Material sobre o curso de pós-graduação [13, 14, 26, 1, 24, 37, 21, 25, 15, 12];
- Material sobre como escrever [6, 20, 22, 23, 39, 35, 36, 38];
- Material sobre a carreia acadêmica [11, 34].

Esperamos que este material seja útil, e que lhe ajude a escrever artigos cada vez melhores. Boa sorte!

Agradecimentos

Gostaríamos de agradecer a Renata Galante, pelo fornecimento do material para a elaboração dos exemplos deste artigo. Gostaríamos de agradecer também ao CNPq por financiar parcialmente este trabalho.

Referências

[1] How to study: Postgraduate research. `http://dmoz.org/Reference/Education/How_To_Study/Postgraduate_Research`.

[2] Qualis. `http://www.dcc.ufmg.br/pos`, em Dia-a-Dia, Qualis Capes.

[3] A. B. Kahng. How to Write a DAC Paper Review, January 2004. `http://vlsicad.ucsd.edu/~abk/dacreviews.html`.

[4] ACM. ACM Code of Ethics and Professional Conduct, 1997. `http://www.acm.org/constitution/code.html`.

[5] André Inácio Reis. I Comment - Conferência de Mentirinha, 2004. `http://www.inf.ufrgs.br/~andreis/misc/coment1/coment1.html##8bib`.

[6] G. M. Blair. How to Write Right. *Engineering Management Journal*, June 1992.

[7] Bruce Randall Donald. How to give a talk. `http://www.cs.dartmouth.edu/~brd/Teaching/Giving-a-talk/giving-a-talk.html`.

[8] Capes. Coordenação de Aperfeiçoamento de Pessoal de Nível Superior. `http://www.capes.gov.br`.

[9] Capes. Critérios de avaliação 2004. `http://www.capes.gov.br`, em Avaliação, Critérios de Avaliação.

[10] CNPq. Conselho Nacional de Desenvolvimento Científico e Tecnológico. `http://www.cnpq.br`.

[11] D. A. Patterson. How to Have a Bad Career in Research/Academia, February 2002. `http://www.cs.berkeley.edu/~pattrsn/talks/BadCareer3.ppt`.

[12] Maria Ester de Freitas. *Viva a Tese: um Guia de Sobrevivência*. FGV, 2001.

[13] M. desJardins. How to Succeed in Graduate School: A Guide for Students and Advisors - Part I of II. *ACM Crossroads Student Magazine*, December 1994. `http://www.acm.org/crossroads/xrds1-2/advice1.html`.

[14] M. desJardins. How to Succeed in Graduate School: A Guide for Students and Advisors - Part II of II. *ACM Crossroads Student Magazine*, February 1995. `http://www.acm.org/crossroads/xrds1-3/advice2.html`.

[15] Umberto Eco. *Como se Faz uma Tese*. Perspectiva, 15a Ed., 2000.

[16] H. L. Hirsch. *Essential Communication Strategies for Scientists, Engineers and Technology professionals*. Wiley, 2003.

[17] IEE. What we look for in your paper. `http://www.iee.org/Publish/Support/Auth/authproc.cfm`, na seção *Checklist*.

[18] IEEE. IEEE Code of Ethics, 1990. `http://ewh.ieee.org/sb/sjce/ethics.htm`.

[19] M. Kremers. Teaching Ethical Thinking in a Technical Writing Course. *IEEE Transactions on Professional Communication*, 32(2), June 1989.

[20] G. D. Lapin. How to Write a Winning Scientific Paper. *IEEE Engineering in Medicine and Biology*, August 1994.

[21] H.C. Lauer. On phd thesis proposals in computer science. *Computer Journal*, 18(3), 1975.

[22] V. O. K. Li. Hints on Writing Technical Papers and Making Presentations. *IEEE Transactions on Education*, 42(2), May 1999.

[23] R. Manley, J. Graham, and R. Baxter. Some Guidance on Preparing Technical Articles for Publication. *IEEE Transactions on Professional Communication*, 32(1), March 2002.

[24] Silvia Miksch. Tips: How to do research. `http://www.ifs.tuwien.ac.at/~silvia/research-tips`.

[25] Edmund Miller. Choosing a grad school advisor. *IEEE Potentials*, 21(3), 2002.

[26] Mirella Moro, Vanessa Braganholo, André Nácul, and Miguel Fornari. Rumo ao título de doutor/mestre. *Journal of Theoretical and Applied Computing (RITA)*, 10(2), 2004.

[27] H. Parks, G. Musser, R.Burton, and W. Siebler. Critical thinking. In *Mathematics in Life, Society, & the World*, chapter 10. Prentice Hall, 2000.

[28] H. Parks, G. Musser, R.Burton, and W. Siebler. Descriptive statistics - data and patterns. In *Mathematics in Life, Society, & the World*, chapter 2. Prentice Hall, 2000.

[29] Patrick H. Winston. Some Lecturing Heuristics. `http://www.cs.dartmouth.edu/~brd/Teaching/Giving-a-talk/phw.html`.

[30] Roger Darlington. How to make a good presentation, 2004. `http://www.rogerdarlington.co.uk/Presentation.html`.

[31] S. D. Senturia. How to avoid the Reviewr's Axe: one editor's view. *Journal of Microelectromechanical Systems*, 12(3), 2003.

[32] B. D. Shriver. The Benefits of Quality Refereeing. *IEEE Computer*, 23(4), April 1990.

[33] A. J. Smith. The Task of the Referee. *IEEE Computer*, 23(4), April 1990.

[34] Richard T. Snodgrass. Why i like working in academia. *Sigmod Record*, 31(1), 2002.

[35] W. StrunkJr and E. B. White. *The Elements of Style*. Macmillan Publishing Company, 1999.

[36] B. Waite. Consequences of the Engineering Approach to Technical Writing. *ACM Journal of Computer Documentation*, 26(1), February 2002.

[37] Toby Walsh. Phd skills. `http://www-users.cs.york.ac.uk/~tw/phd`.

[38] E. H. Weiss. Egoless Writing: improving quality by replacing artistic impulse with engineering discipline. *ACM Journal of Computer Documentation*, 26(1), February 2002.

[39] Justin Zobel. *Writing for Computer Science: The Art of Effective Communication*. Springer-Verlag, 1998.

Managing Repositories for Digital Content Components *

André Santanchè[1] Claudia Bauzer Medeiros[1] (advisor)

[1]Institute of Computing – University of Campinas – UNICAMP
CP 6176, 13084-971 Campinas, SP, Brazil

{santanch,cmbm}@ic.unicamp.br

Level: Ph.D.
Computer Science Ph.D. Program
University of Campinas – UNICAMP
Admission: March 2003
Conclusion expected to: December 2006

Abstract

The Semantic Web pursues interoperability at syntactic and semantic levels, to face the proliferation of data files with different purposes and representation formats. One challenge is how to represent such data, to allow users and applications to easily find, use and combine them. The paper proposes an infrastructure to meet those goals. The basis of the proposal is the notion of digital content components *that extends the software engineering software component. The infrastructure offers tools to combine and extend these components, upon user request, managing them within repositories. It adopts XML and RDF standards to foster interoperability, composition, adaptation and documentation of content data. This work was motivated by reuse needs observed in two specific application domains: education and agro-environmental planning.*

Keywords: Digital Content Component, Semi-structured Databases, Metadata, Semantic Web, Information Integration and Interoperability

*This work was partially financed by UNIFACS, by grants from CNPq and FAPESP, and projects MCT-PRONEX SAI and CNPq WebMaps and AgroFlow.

1. Introduction

Different domains in Computer Science deal with reuse questions following parallel tracks to solve analogous problems. Reuse can be defined as the practice of using an existing object to build a new digital artifact using the object's content partial or totally [9]. We can distinguish two main currents in reuse research: first, reuse of program code, mainly in the software engineering context, and second, reuse of content, with distinct approaches depending on the application domain – e.g., content/document management or education. There are several obstacles to supporting reuse; perhaps the most serious is the problem of proliferation systems and standards for data representation.

The thesis proposes a model for reuse that integrates these two main currents in a general solution, combining them with database research on the Semantic Web and interoperability. We exploit some practices shared by these currents: (i) the program code or content is *decomposed in independent units* suitable for deployment and reuse; (ii) these units are indexed and *stored in a repository*; (iii) there is a Web-based *architecture to assemble* and connect the units.

The Semantic Web [5] foresees a new generation of Web-based systems, where semantic descriptions of data and services will booster interoperability. In parallel, software engineering has reached a high level of maturity concerning reuse units, by developing the technology of software components. Our idea is to extend these principles, to comprise any digital content. From now on, this extended notion of component will be called *digital content component*; the term will be used in this paper to denote any kind of data – e.g., pieces of software but also texts, audio, video, and so forth.

Like software components, our proposal for a digital component structure involves the encapsulation of specific data representations into a package with a standard format, and public interfaces that support relationships among components.

Though the advantages of such generalization are evident, there remains the problem of putting it into practice. Thus, the thesis treats three main issues, based in the practices shared by reuse approaches. The first issue concerns *establishing a model* to represent a digital content component, adopting interoperability standards of the Semantic Web. The second defines a *strategy to store and index* large volume of these components in a database. Finally it defines an *architecture to assemble* content components into a desired product.

The remainder of this text is organized as follows. Section 2 introduces related work. Section 3 presents the proposed digital content component model and storage and assembly considerations. Section 4 presents the adopted methodology and the present stage of this work. Section 5 presents concluding remarks.

2. Related Work

The presentation of related work is organized to analyze practices shared by reuse research currents, following the sequence component decompose $\rightarrow$ store/retrieve $\rightarrow$ compose.

2.1. Decomposing Content into Independent Units

At the content management context it is recognized that decomposing and storing the content in small units (assets) is a good choice to improve reuse. Nevertheless, there is no consensus on a standard way to represent those units – they may take the form of a file, or a record in a database, and so on.

A step ahead was taken in the educational domain, where content reuse was enhanced via assembly into a standard package for distribution. In our work, *package* is defined as a structure that delimitates,

organizes and describes one or more pieces of digital content suitable to reuse. The main example is the IMS Content Packaging Information Model [14], and based on that, the SCORM – Sharable Content Object Reference Model [3], a proposal to structure and distribute educational content in a package content form. They use XML to describe a hierarchical structure of each content package and their respective educational metadata.

The package structure has limited capabilities. An evolution of this concept is achieved with *components*. The component concept is often associated with the software component concept, which deals with software code. There is no agreement on the definition of the software component concept, though definitions are closely related [1]. Like a package, a component is designed to be a unit of independent deployment. There is a clear division between the content (encapsulated software implementation) and some external structure where this content is encapsulated. However, components have higher specialized external structures, which publish component functionality by explicit contractual interfaces. To interact and work together, component structure and interface follow a model that specifies design constraints.

2.2. Storing Units in Repositories

Many research initiatives concerning storage and retrieval of reuse units are developed on the software component context, especially on techniques to index components. The classification structure for component indexation can be borrowed from library science (e.g., enumerative method and faceted method [7]) or can based on ontologies [15] to represent domain knowledge associated with components. Domain specific ontologies will help the creation of homogeneous vocabularies, making easier to find shared reuse units. This is the situation of the educational domain, where the IEEE led an initiative to standardize educational metadata through LOM – Learning Object Metadata [4]. Nowadays, IMS Content Packaging Information Model and SCORM adopt LOM. LOM has been coded in XML [16], with subsequent studies to represent LOM in RDF [6].

Still another indexation solution relies on component signature match, but it can be refined by a formal specification of component behavior, used as a basis to behavior match [17].

2.3. Composing Units

The strategy to compose units varies depending on the kind of reused content. The content management context adopts a content centric development model. Composition is often represented by links between assets (e.g., a HTML page $\times$ image, or an XML file $\times$ XSL file) and the process takes a secondary role.

Software engineering, on the other hand, has a process centric development model and defines a strategy to assemble software components into applications. This induces a software architecture, which involves a configuration of components and connectors to bind components together. The approaches to configure and connect the components, and the role played by connectors can vary, and determine an architectural style [13].

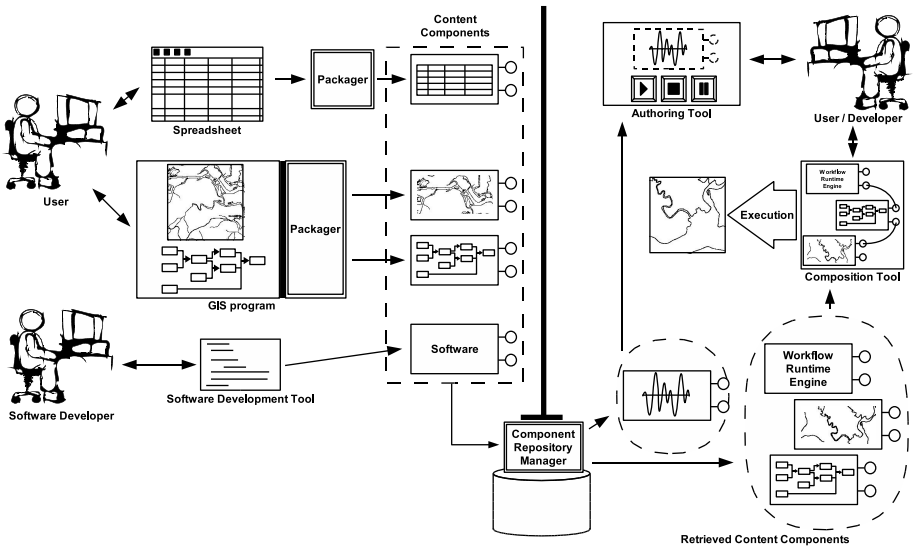


Figure 1: Diagram of content component cycle for production/storage/use.

3. A Proposal for Digital Content Components

3.1. Components' Life Cycles

Any digital component infrastructure must support the entire cycle of component production, storage, retrieval and use – see Fig. 1. Production comprises the well known software component production process, which we propose to extend to any piece of digital object the user wants to share. Such components can be built automatically, or guided by the user, through a tool named *packager*. The figure shows packager modules that encapsulate distinct kinds of content components – spreadsheets, workflows, maps, etc. A packager takes the form of a plugin attached to a software, which deals with the content, or an independent module specialized to process some file formats. Components are stored in a repository, and managed by a repository manager, which is accessed by users to retrieve and combine them. Again in the figure, a developer uses an authoring tool to compose distinct components – workflow, maps and workflow runtime engine – into an executable unit to produce a map. This specific example reproduces a scenario we deal with in UNICAMP, but without support of content components. In this scenario, experts specify workflows to generate maps for environmental planning based on combining several kinds of data. These workflows can then be run to produce distinct kinds of environmental plans.

The starting point for our work is the Anima project, an infrastructure for software components [10] developed by the author. Anima is being used to create educational tools in schools in Salvador. Anima comprises the complete cycle illustrated in Fig. 1, but restricted to software components and without database support. It supports building applications via component composition and uses RDF to represent component packages, including interface specification and component metadata. This allows applications to deal with software components implemented in different languages. An Anima application can be represented via a network of components whose configurations are stored in an XML file. Anima provides support to component execution and intercommunication via an XML based protocol.

With this background in mind, our research considers three aspects of content components: representation, storage/retrieval management and use.

3.2. Component Representation

Categories of Content Components

We differentiate between two kinds of component – process and passive components. A process component encapsulates any kind of process description (sequences of instructions or plans) that can be executed by a computer. Therefore, they usually define an input interface, and their results change according to different input values. Non-process components, named passive components, contain data that can be used by a process component. Passive component's interfaces match the interfaces of process components enabled to deal with their contents.

In order to illustrate content component categories, we will borrow an example from scientific applications. In this context, scientists are interested not only in reusing results, but in sharing the whole process of experiment development. This originated the notion of scientific workflows (e.g. [11]), to specify and record experiments; this allows, among others, experiment reproducibility and therefore reuse. The WOODSS system [11], developed at UNICAMP, follows this approach: it enables the capture of activities in agro-environmental planning to be stored as scientific workflows, which can be later edited, composed and re-executed. WOODSS' users manage two main kinds of file: maps and workflows. The same workflow can be executed using different input maps.

Workflows are an example of our process components, whereas maps are typically passive components. In our analogy, a "workflow component" can be linked to different passive "map components". Furthermore, one may envisage defining interfaces to workflows, in RDF, that will impose conditions on input files – e.g., indicating maps that can be acceptable as input.

Most process components cannot be executed directly, due to their need for interpreters or runtime modules, not embedded into their environment. In such cases, complementary attached components can perform this task. These complementary components are named *companion components*. Process components, moreover serve as companions to passive components, i.e., a passive component needs some sort of code to be processed.

The choice of the appropriate companion component for a passive or process component is determined by the application manager, and is sensitive to the context. This allows dynamic companion binding, transparent to end users, and a homogeneous treatment of passive and active components from the user's perspective.

Content Component Structure

A component's structure is composed of four distinct parts: (i) The content itself, in its original format; (ii) an XML specification of the internal structure used for component organization, based on SCORM [3]; (iii) an RDF specification of component interfaces; (iv) RDF metadata to describe functionality, applicability, use restrictions, etc. Components can be recursively constructed from composition of other components, each of which is structured by the same four parts.

Figure 2 shows a partial example of a component specification for encapsulating a set of satellite images of São Paulo state. This passive component can be used by WOODSS, as explained before. The structure part uses XML to arrange a set of images (content) coded in GIF. Metadata in RDF represents component's title and taxonomy. Specific terms employed – such as "Temporal Set of Satellite Images" – are extracted from domain-dependent ontologies (e.g., [8]).

The interface part shows three input messages ("access", "first", "next") and two outputs. All inputs are classified as simple messages without parameters ("single"). Outputs describe in RDF how the

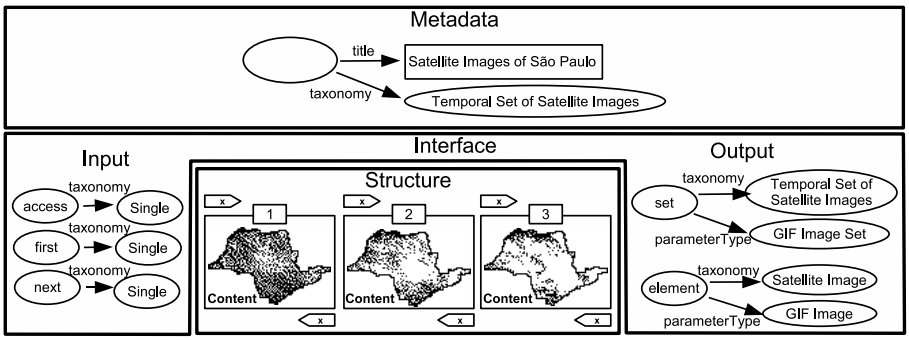


Figure 2: Diagram of content component structure.

contents are formatted (parameterType arrow) for a given ontological context within a domain ontology (taxonomy arrow). This shows output should always contemplate not only syntactic information (i.e., GIF format), but also semantics (ontology terms).

The “access” input will eventually produce a “set” output message, which contains the complete set of maps. The messages “first” and “next” induce the component to produce an “element” output message with the first or next image in the set. The implementation of this behavior will be provided by the companion component, transparently associated with the content component at execution time.

This four-part structure for component representation, is an important contribution of this work. It extends the software engineering concept of software component to any kind of content, and promotes interoperability. Related work deals with some of these aspects in an isolated form, without an integration perspective.

3.3. Component Storage and Retrieval Management

A digital content component is a combination of raw data, XML data and RDF data. The solution requires devising a database capable to interpret and manage each format, since XML and RDF data will be used for indexation and internal structure exploration purposes.

There are many possible solutions to this problem. They involve, among others, considerations and tradeoffs between constructing a native XML DBMS versus a relational DBMS that maps XML (and RDF) data to its structure and vice-versa. Shanmugasundaram [12] proposes a process to map XML to a relational database, which unifies many partial solutions to this problem. RDF, on the other hand, is based in a strong connected network of interdependent description elements. Therefore, besides the mapping, it is necessary to devise a model to retrieve coherent fragments of RDF data [2], preventing the transfer of large data volumes for each query.

Other possible solutions are the use of a native XML database, alone or combined with a relational database. Though XML data can be stored and indexed in a native way, there are limitations to current XML database implementations, such as the need for consistency mechanisms, transaction support, etc. To store RDF data in an XML database will require a costly process to code and decode the descriptions from XML data.

One of the purposes of this project will be to identify the best storage solution, and to combine it with a procedure to index and retrieve components. It will result in a component repository where that components may be dynamically combined. Indexation will be based on adapting and combining two techniques to components: use of ontologies [15] and behavior match [17]. The repository will have a storage structure divided in three main sections: (i) storage of passive and process components; (ii) ontologies shared by the RDF descriptions of the components; (iii) a history of component selections/combinations made by each user.

Another issue is finding adequate companion components, and configurations. Metadata associated with components and common ontologies shared by metadata can be used to help choose the adequate component configuration, e.g., for a given quality requirement. Another direction is to use the notion of DBMS monitoring to track the rate of use of each repository component, and the rate of adopted combinations, which are both stored in the history section. This will help guide the search for adequate component combination, based on previous experience, and can indicate the quality of component, based on its use rate.

3.4. Components Composition

Content components can be used and composed in many ways. Two forms of expected use are: component insertion and application construction. In the first form, components are inserted into documents or multimedia productions as content pieces. This use of components can be compared with the insertion of DDE/OLE objects into Windows documents, or insertion of embedded objects (such as Java applets) into Web pages. The components inserted are commonly passive components, or process components attached to passive components. For instance, a map component attached to a workflow component can be inserted in a scientific report.

In the second usage form – application construction – components are used as basic blocks to construct applications. Here, just like in software component composition, an application is built from a network of interconnected components, following an architectural style [13]. One difference is that the development can be content or process centric, depending on the kinds of components used, and the desired result. A content centric development can be achieved when some links are replaced by connections between components, like an XML component connected with an XSL component.

The configuration of components and connectors to form the application is stored in an XML document. This document contains the initial configuration of each component involved and the connectors configurations.

4. Methodology and Present Stage

(1) Definition of the content component model, extending Anima model to support any kind of content, using accepted standards and integrating major component approaches (finishing this stage).
(2) Design the database to support component storage/retrieval (finishing this stage).
(3) Design and construction of a reference implementation framework, based in the Anima infrastructure, to: provide an infrastructure for components execution and intercommunication; interact with a database to store/retrieve components; and provide support to component insertion, adaptation and reuse by applications (design finished).
(4) Identifying a test application domain in agriculture planning and construction of this application to validate the framework (under development).

5. Concluding Remarks

This project combines work in software engineering with content management, Semantic Web and database interoperability efforts. The main contribution is the formulation of an integrated view over these research areas, taking advantage from progress in the software components area to support development of applications in the Semantic Web. The second contribution is the four-part structure and the use of RDF to promote component specification interoperability. The concept of companion component

will allow a homogeneous treatment for any kind of component. A third contribution is the repository structured in three sections.

The work proposed is based on previous experience in construction component-based applications for the educational domain [10], and on the use of scientific workflows, stored in databases, for reuse and interoperability of environmental applications [11].

References

[1] F. Bachman et al. Volume II: Technical Concepts of Component-Based Software Engineering. Technical report, Carnegie Mellon University, July 2000.

[2] A. Barnell. RDF Objects, November 2002. `http://www.hpl.hp.com/techreports/2002/HPL-2002-315.pdf`, accessed on 10/2003.

[3] P. Dodds. Sharable Content Object Reference Model (SCORM) – Version 1.2, October 2001. `http://www.adlnet.org/screens/shares/dsp_displayfile.cfm?fileid=840`, accessed on 10/2003.

[4] IEEE L.T.S.C. Draft Standard for Learning Object Metadata – IEEE 1484.12.1-2002, July 2002. `http://ltsc.ieee.org/doc/wg12/LOM_1484_12_1_v1_Final_Draft.pdf`, accessed on 10/2003.

[5] R. Meersman and A. Sheth. Special Section on Semantic Web and Data Management – Guest editors introduction. *ACM SIGMOD Record*, 31(4):10–12, 2002.

[6] M. Nilsson. IEEE Learning Object Metadata RDF binding, August 2002. `http://kmr.nada.kth.se/el/ims/md-lomrdf.html`, accessed on 11/2003.

[7] R. Prieto-Díaz. Classification of reusable modules. In *Software reusability: vol. 1, concepts and models*, pages 99–123. ACM Press, 1989.

[8] R. Raskin. Semantic Web for Earth and Environmental Terminology (SWEET). In *Proc. of NASA Earth Science Technology Conference 2003*, 2003.

[9] A. Rockley. *Fundamental Concepts of Content Reuse*, chapter 2. New Riders, 2000.

[10] A. Santanchè and C. A. C. Teixeira. Anima: Promoting Component Integration in the Web. In *Proc. of 7th Brasilian Symp. on Multimedia and Hypermedia Systems*, pages 261–268, October 2001.

[11] L. A. Seffino, C. B. Medeiros, J. V. Rocha, and B. Yi. WOODSS – A spatial decision support system based on workflows. *Decision Support Systems*, 27(1-2):105–123, November 1999.

[12] J. Shanmugasundaram et al. A general technique for querying XML documents using a relational database system. *ACM SIGMOD Record*, 30(3), September 2001.

[13] M. Shaw and P. C. Clements. A Field Guide to Boxology: Preliminary Classification of Architectural Styles for Software Systems. In *Proc. of the 21st Int. Computer Software and Applications Conf.*, pages 6–13. IEEE Computer Society, 1997.

[14] C. Smythe. IMS Content Packaging Information Model. Specification, IMS Global Learning Consortium, Inc., June 2003. `http://www.imsglobal.org/content/packaging/`, accessed on 11/2003.

[15] V. Sugumaran and V. C. Storey. A Semantic-Based Approach to Component Retrieval. *SIGMIS Database*, 34(3):8–24, 2003.

[16] S. Thropp and M. McKell. IMS Learning Resource Meta-Data XML Binding Specification, September 2001. `http://www.imsglobal.org/metadata/imsmdv1p2p1/`, accessed on 10/2003.

[17] A. M. Zaremski and J. M. Wing. Specification Matching of Software Components. In *Proc. of 3rd ACM Sigsoft Symp. on the Foundations of Software Engineering*, October 1995.

Semantic-based Information Integration

Rosalie Barreto Belian[1], Ana Carolina Salgado[2] (advisor)
[1]Centro de Informática – Universidade Federal de Pernambuco
e-mail: {rbb, acs}@cin.ufpe.br

Nível: Doutorado
Programa de Pós-Graduação em Ciência da Computação
Centro de Informática, Universidade Federal de Pernambuco
Ano de Ingresso: Fevereiro de 2002
Previsão de Conclusão: Dezembro de 2005

Abstract

The Information Society growth has demanded the development of tools and systems that respond its requirements with efficiency, precision and objectiveness. Thus, systems have been designed for integrating data from multiple and heterogeneous Web sources. These systems have the main challenge of resolving information heterogeneity and additionally, present it to the users in a concise and uniform way, abstracting its syntactic, structural and semantic diversities. One of the biggest difficulties in developing those systems consists in establishing the semantic affinities among objects retrieved from the data sources and the subsequent employment of mechanisms for dealing with their structural and syntactic heterogeneity. This work intends to redesign an integration information system including semantic issues. To do this, the application of concepts such as contexts and ontologies in an information integration system was studied. The purpose is to establish correspondences and solve semantic conflicts among objects obtained from diverse heterogeneous data sources. A comprehensive process of information integration is proposed in this work, which incorporates semantic mechanisms to existent structural and syntactic processes.

Keywords: *Information integration and interoperability, Semantic of data, Semi-structured databases and XML, Web databases, Semantic Web.*

1. Introduction

The construction of systems designed for integrating data from multiple and heterogeneous Web sources has the challenge of resolving information heterogeneity and additionally, present it to the users in a concise and uniform way, abstracting its syntactic, structural and semantic diversities. One of the biggest difficulties in developing those systems consists in establishing the semantic affinities among objects retrieved from these data sources and the subsequent employment of mechanisms for dealing with their structural and syntactic heterogeneity.

In the last years, many research projects had tried to address the problem of integrating data from different and distributed sources and present it in a uniform way. These systems had initially aimed to dealing with structural and syntactic aspects of information integration. As example of these systems we can mention: TSIMMIS [1], IBIS [2], MIX [3], NIMBLE [4] and e-XML [5]. The focus given to the structural and syntactic heterogeneity in these systems aimed at the integration of semi-structured and structured data sources. Under another approach, several works looked for solutions seeking the information integration based on its semantic nature, such as: SIMS [6], MOMIS [7], InfoSleuth [8], The Information Manifold [9], DataFoundry [10], SHOE [11], INDUS [12], COIN [13], Buster [14] and Kashyap & Sheth [15].

The resolution for the semantic heterogeneity is essential for information integration. The identification of semantic affinities among objects consists in a critical task for the integration of information [16]. Moreover, the semantic alignment of the concepts found in the data sources makes possible the resolution of subsequent syntactic and structural heterogeneities.

There is a significant amount of works in the literature [15, 17, 18, 14, 13] discussing the application of context and ontology concepts with the purpose of integrating information, respecting its semantic nature. A context "contains metadata related to its meaning, properties (such as its source, quality, and precision), and organization" [14, 15]. A context, in an information integration system, can contain descriptions about the structural, organizational and conceptual nature of an object according to its knowledge domain. An ontology, as defined in [19], "is an explicit specification of a conceptualization." The ontology of the related knowledge domain can guarantee the terminological uniformity demanded in the determination of relationships among terms of the vocabulary shared by the data sources of the information integration system [14, 20].

This ongoing research intends to include in an information integration system features to deal with semantic issues. The system architecture is being redesigned applying context and ontology concepts with this purpose. Contexts are being considered to organize and contextualize the metadata used for information integration. The system metadata describes syntactic, structural and semantic information about the web data sources, and user preferences. A knowledge domain ontology is used giving the actual meaning for the data sources concepts and terms, allowing their correct interpretation by the system. A comprehensive process of information integration is being proposed, joining the semantic resolution to an existent process that had considered just structural and syntactic heterogeneities.

2. Methodology and Current State of the Research

The main objective of this work is to specify an integration information system involving semantic, syntactic and structural issues in its integration process. An information integration system was originally designed to solve syntactic and structural issues. To include semantic integration issues, the methodology adopted is based on the following steps:

- Acquire a complete understanding of the integration problem through the study and analysis of the referred system architecture and information integration related works. Some characteristics should be investigated such as mechanisms and technologies applied in these systems to solve semantic aspects;
- Make a detailed study of the existent technologies associated with semantic heterogeneities and how they can be used to solve information integration problems;
- Redesign the referred system architecture enriching it with new modules, processes, functionalities, patterns, templates and technologies due to the semantic approach;

- Validate the new system architecture and processes with a case study in health care area using a clinical ontology and related data sources. The integration information results produced by the system will be compared with and without the semantic enrichment.

Some of these steps are already being accomplished. The information integration problem was extensively studied. Relevant semantic issues related to information integration were investigated in the literature, and context and ontology concepts were pointed out as important mechanisms to solve the semantic heterogeneity. Furthermore, new modules and processes are being specified to be included in the integration system.

3. Information Integration System Architecture

An information integration system was originally designed to solve syntactic and structural integration issues [21]. The architecture proposed was based in a mediation system according to the GAV (Global as View) approach, allowing the isolation of user applications from the complexity of the data integration problems.

This information integration system uses a common data model for representing the content and structure of the data sources. The data sources and mediator schemas are captured and represented in XML Schema [22]. The system, however, converts the data sources and mediator schemas to an internal conceptual model, called X-Entity [23]. The X-Entity model was proposed to improve the data representation capacity of the system, once XML Schema does not allow the capture of some specific characteristics of the information. Entities in the X-Entity model are represented as individual concepts.

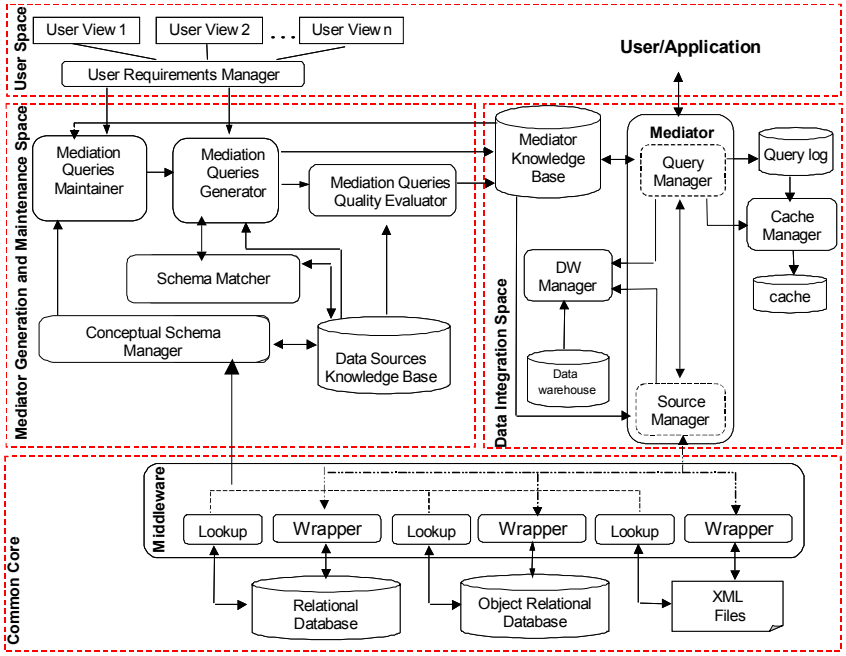


Figure 1. Original System Architecture

The original architecture of the data integration system is organized in four main modules as shown in Figure 1 [24]:

- **Common core**: this module feeds the generation and maintenance module of the mediator with information of local data sources schemas. It receives and answers queries coming from the data integration module addressed to these sources;
- **Data integration space**: this module is composed by the mediator, responsible by the restructuring and coalition of data of the autonomous data sources and by supplying a XML integrated view of the data. It receives user queries returning integrated answers;
- **Mediator Generation and maintenance space**: through semiautomatic processes this space executes the generation and maintenance of the mediator views;

- **User space**: this space is related to the definition of the user requirements, which are represented using a high level model allowing its translation to the system.

4. Including Semantic Issues

The system architecture described was adopted as the initial point to this work. In its initial specification, the information integration system, presented a metadata layer, containing intentional information of the data sources, used to abstract their structural details. This layer is implemented using XML Schema and the conceptual model X-Entity [23]. In order to analyze the information considering its semantic aspects, the architecture of the data integration system is being adapted to consider the concepts of contexts and ontologies.

A domain ontology is being used in the semantic description of the data sources and in the interpretation of their content. Moreover, the ontology of the system domain will be used as a shared vocabulary for the user application to establish the semantic affinities among local sources concepts and ontological terms. A semantic matching process was proposed with this purpose. The ontological term having the highest degree of semantic affinity with the data source concept identifier (usually representing an entity, attribute or relationship) is registered in its corresponding X-Entity concept. This process causes the enrichment of the X-Entity model with semantic information, i.e., it is the basis of a semantic clustering process that unifies semantically similar concepts. These unified concepts represent the mediator entities which will be considered by the users in the definition of their requirements.

At the current state of the research, the usage of contexts is being considered in two spaces of the system architecture: the mediator generation and maintenance space and the data integration space. In the mediator generation and maintenance space, contexts are being used to semantically enrich the data source and mediator metadata. Data source metadata describes relevant information about each concept extracted from the data sources, which includes information like correspondent ontological term, domain ontology, concept identifier in the local data source, data source identifier, concept type (entity, attribute or relationship), semantic similarity degree, data type, default value, constraints, precision, and so on. Similarly, to each concept selected by the user (in the user requirements definition), a mediator context is structured in order to map user concepts to data sources concepts. At the mediator level the contexts represent information like ontological term, ontology domain, concept type (class concept, property or relationship), data type, default value, precision, constraints, and corresponding mappings to each data source where this concept can be found. Finally, the system generates the mediation queries used by the data integration space.

The data integration space receives queries from users or their applications. In this space, mediator contexts, associated to each user selected concept, will be accessed to solve user queries decomposing them according to the corresponding data sources. The results from the sub-queries are integrated based on the data sources context information.

Some specific objectives are defined to include these semantic concepts in our information integration system:

- The definition of a formalism for the specification of contexts, as well as its manipulation by the system;
- The selection of ontology definition tools and its incorporation in the system. XML based technologies are being considered with this purpose such as OWL (Web Ontology Language). OWL is a markup language proposed by W3C with the purpose of publishing and sharing ontologies on the Internet. OWL was designed based on DAML+OIL Web Ontology Language (DARPA Agent Markup Language, Ontology Inference Layer) [25];
- The specification of the semantic matching process which compares ontology terms and data sources schema entities defining their semantic similarity and then, enriching the X-Entity model;
- The specification of the clustering process which produces the mediator entities representing the integrated collection of data source concepts (based on the domain ontology);
- The implementation of a prototype to validate the proposed system architecture using a health care application.

5. Related Works

The work of Kahyap and Sheth [15] presents a study on the use of contexts and ontologies in the resolution of semantic heterogeneity in information integration systems. In this system, contexts are classified as: m-contexts (contexts containing metadata used for abstraction of the details of data representation in the remote sources) and c-contexts (used in the maintenance of descriptions about the knowledge domain of the system). The representation of contexts in the system is accomplished through a language based in descriptive logic. Ontologies supply an extra semantic knowledge that cannot be captured from the schema of the data sources. The system uses mappings among object definitions and its real location in the data sources.
The approach used in Buster [14] is similar to the one presented in the work of Kashyap & Sheth. Individual contexts for the data sources are defined, and the interoperability is obtained through transformations among the contexts of the involved data sources using their terminological vocabularies. In the mentioned work two types of semantic conflicts are handled: conflicts of name and scale. A descriptive logic language is used to construct the ontologies. Templates are used to accomplish the mappings of the concepts (semantics) to the data sources.
The COIN project [13] presents an architecture for semantic interoperability in which the semantics of the data is represented through "enriched" schemas, contexts and domain models. A domain model consists of a conceptual model that establishes the terminological base through which the meaning of the data sources concepts can be explained. Through the information obtained from the context, the schemas of the data sources are "enriched" (mapping among relational schemas and the conceptual model of the domain). The execution of mediation queries is accomplished through an execution plan that determines which data source can contribute to the computation of a certain query and what transformations are necessary to solve the semantic conflicts that can happen in the query. The context mediator accomplishes the detection and resolution of semantic conflicts. COIN uses a frame-based language for representing the domain model.
The integration system described in this work presents convergent architectural aspects with the three systems discussed previously. The main contribution of our work is concerned with the specification and validation of processes, context templates and ontology use in an information integration system based on XML approaches. The essential objective of developing an information integration system on the Web is pursued. The main data models and processes include: a semantic enriched XML conceptual model (X-Entity) and, a semantic matching approach comparing ontology terms and data sources schemas. Besides, the system mediator schema is generated considering schema semantic conflicts earlier solved.

6. Expected results

One of the contributions of this work remains in the practical use of up-to-date technologies to address a complex problem as semantic information integration. Due to the fact that this information integration system is being designed to operate on the Internet environment, the problem complexity is increased once we need to deal with autonomous, heterogeneous and semi-structured data sources. Besides, the practical application of contexts and ontologies in a problem of the database world is a challenging task. Finally, this work that aim to address the semantic information integration problem, fits in the semantic web scenario which demands for complete application interoperability [26]. This does not happen without expressing meaning and integrating information. We expect this work presents the following general results:

- Validation of using context and ontology concepts in the semantic approach in information integration processes. Proposition of a context schema organizing all system metadata;
- Definition of a matching approach to identify semantic similarities among concepts and terms from local data sources and the associated domain ontology. Generation of mediator schema with semantically unified concepts;
- Evolution of the system architecture incorporating semantic features. Development and validation of an information integration process that considers structural, syntactic and semantic issues;

- Definition of a more realistic process of information integration considering user interventions and elucidating semantic conflicts when the system is not able to solve them.

References

1. Chawathe, S., Garcia-Molina, H., Hammer, J., Ireland, K., Papakonstantinou, Y., Ullman, J., Widom, J.: The TSIMMIS Project: Integration of Heterogeneous Information Sources, Proceedings. of The ISPJ Conference, pages 7-18, Tokyo, Japan, 1994.
2. Calvanese, D., De Giacomo, G., Lenzerini, M., Naggar, P., Vernacotola, F.: IBIS: Semantic Data Integration at Work, Proc. of the 15th International Conference Advanced Information Systems Engineering (CAiSE), 2003.
3. Baru, C., Gupta, A., Ludascher, B., Marciano, R., Papakonstantinou, Y., Velikhov, P., Chu, V.: XML-based information mediation with mix, ACM SIGMOD Conference on Management of Data, Proceedings p. 597-599, 1999.
4. Draper, D., HaLevy, A., Weld, D.: The Nimble XML Data Integration System, 17th International Conference on Data Engineering, Heidelberg, Germany, 2001.
5. Gardarin, G., Mensch, A., Tomasic, A.: An Introduction to the e-XML Data Integration Suite, 8th International Conference on Extending Database Technology, Prague, Czech Republic, 2002.
6. Arens, Y., Chee, C., HSU, C., Knoblock, C.: Retrieving and integrating data from multiple information sources, The International Journal on Intelligent and Cooperative Information Systems, v.2, n.2, p. 127-158, 1993.
7. Bergamaschi, S., Castano, S., De Capitani di Vimercati, S., Montanari, S., Vincini, M.:A semantic approach to information integration: the MOMIS Project, Sesto Convegno della Associazione Italiana per L'Intelligenza Artificiale, 1998.
8. Bayardo, R., Bohrer, W., Brice, R., Cichocki, A., Fowler, J., Helal, A., Kashyap, V., Ksiezyk, T., Martin, G., Nodine, M., Rashid, M., Rusinkiewicz, M., Shea, R., Unnikrishnan, C., Unruh, A., Woelk, D.: InfoSleuth: Agent-Based Semantic Integration of Information in Open and Dynamic Environments, Proceedings of The ACM SIGMOD International Conference on Management of Data, p.195-206, Tucson, Arizona, 1997.
9. Kirk, T., Levy, A., Sagiv, Y., Srivastava, D.: The Information Manifold, The AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments, Proceedings p 85-91, 1995.
10. Critchlow, T., Ganesh, M., Musick, R.: Meta-Data Based Mediator Generation, 3rd International Conference of Cooperative Information Systems, New York, USA, 1998.
11. Heflin, J., Hendler, J., Luke, S.: Applying Ontologies to the Web: A Case Study, IWANN'99, International Work-Conference on Artificial and Natural Neural Networks, Alicante, Spain, 1999.
12. Reinoso-Castillo, J., Silvescu, A., Caragea, D., Pathak, J., Honavar, V.: Information Extraction and Integration from Heterogeneous, Distributed, Autonomous Information Sources – A Federated Ontology-Driven Query-Centric Approach, IEEE International Conference on Information Integration and Reuse, 2003.
13. Goh, C., Madnik, S., Siegel, M.: Semantic Interoperability through Context Interchange: Representing and Reasoning about Data Conflicts in Heterogeneous and Autonomous Systems, Sloan School of Management, MIT, 1997.
14. Wache, H., Stuckenschmidt, H.: Practical Context Transformation for Information System Interoperability, Proceedings of the 3rd International Conference on Modeling and Using Context, Lecture Notes in AI, Springer-Verlag, 2001.
15. Kashyap, V., Sheth, A.: Semantic Heterogeneity in Global Information Systems: The Role of Metadata, Context and Ontologies, Chapter in Cooperative Information Systems: Current Trends and Directions, M. Papazoglou and G. Schlageter Editors, 1996.
16. Sheth, A., Kashyap, V.: So Far (Schematically), yet So Near (Semantically), Proceedings of the IFIP TC2/WG2.6 Conference on Semantics of Interoperable Database Systems, DS-5. In IFIP Transactions A-25, North Holland, 1992.

17. Ouksel, A., Sheth, A.: Semantic Interoperability in Global Information Systems, A brief introduction to the research area and the special section. SIGMOD Record, Vol. 28, No.1, 1999.
18. Kashyap, V., Sheth, A.: Semantic and schematic similarities between database objects: a context-based approach, The VLDB Journal 5: 276-304, 1996.
19. Gruber, T.: A Translation Approach to Portable Ontologies, Knowledge Acquisition, V.5, n.2, p.199-200, 1993.
20. Guarino, N.: Formal Ontology and Information Systems, Proceedings of OFIS'89, Trento, Italy, 1998.
21. Lóscio, B., Salgado, A., Vidal, V.: Using Agents for Generation and Maintenance of Mediators in a Data Integration System on the Web, Proceedings of the XVI Simpósio Brasileiro de Banco de Dados, Rio de Janeiro, Brazil, 2001.
22. Fallside, D.: XML Schema Part 0: Primer, W3C Recommendation, http://www.w3.org/TR/xmlschema-0/, 2001. Access date: 06/08/2004.
23. Lóscio, B., Salgado, A., Galvão, L.: Conceptual Modeling of XML Schemas, WIDM-International Conference on Conceptual Modeling ER, New Orleans, USA, 2003.
24. Lóscio, B.: Managing the Evolution of XML-based Mediation Queries, PHD Thesis, Federal University of Pernambuco, Brazil, 2003.
25. Dean, M., Connolly, D., Harmelen, F., Hendler, J., Horrocks, I., McGuiness, D., Patel-Schneider, P., Stein, L.: OWL WEB Ontology Language 1.0 Reference, W3C Working Draft. http://www.w3.org/TR/owl-ref/#Ontology-def. 2002.
26. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web, Scientific American, 284 (5), vol. 184, no. 5, pp. 34-43, 2001.

Slim$^+$-tree: Um Método de Acesso Métrico Baseado em Medidas de Dispersão

Carla Elena D. Martins, Denise Guliato (orientadora)
Laboratório de Computação Científica – Universidade Federal de Uberlândia
{carla,denise}@lcc.ufu.br

Nível: Mestrado
Programa de Mestrado em Ciência da Computação
Universidade Federal de Uberlândia
Ingresso: Setembro 2002
Previsão de Conclusão: Março de 2005

Resumo

Métodos de acesso métrico são utilizados para realizar consultas por similaridade em bases cujos dados podem ser representados no espaço métrico, isto é, onde a similaridade dos objetos é definida por uma função de distância, como seqüências de DNA ou vetores características que descrevem atributos visuais das imagens. Para garantir um bom desempenho na etapa de consultas em base de dados de natureza métrica, uma estrutura de índice deve manter agrupados objetos próximos entre si e garantir uma sobreposição mínima entre os nós da árvore. Neste trabalho é proposto a Slim$^+$-tree, uma variação do método de acesso métrico Slim-tree. Na Slim$^+$-tree o nó escolhido para conter um novo objeto é aquele que mantém o melhor agrupamento entre os nós candidatos, baseando-se num critério de dispersão mínima. Este critério favorece a redução de sobreposição entre nós e conseqüentemente uma recuperação mais eficiente da informação. Modificações para o algoritmo de divisão do nó também são propostas.

Palavras-chave: *método de acesso métrico, recuperação de imagens por conteúdo, indexação, espaço métrico.*

Abstract

Metrics access methods are used to realize similarity queries in databases where the data can be represented in the metric space, i.e. where objects similarity are defined by a distance function, like DNA sequences or vectors of features that describe visual attributes of the images. To guarantee a good performance for answering similarity queries in database of metric nature, an index structure must keep grouped similarity objects and to guarantee a minimum overlapping between the nodes of the tree. In this paper we proposed the Slim$^+$-tree, a variation of the method of metric access Slim-tree. In the Slim$^+$-tree the node chosen to contain a new object is that one that keeps the best grouping between the nodes candidates based on a criterion of minimum dispersion. This criterion consequently favors the reduction of overlapping between nodes and a more efficient recovery of the information. Modifications in the split algorithm are also proposals.

Key Words*: metric access method, content-based image retrieval, indexing, metric space.*

1. Temas e questões de interesse da pesquisa

Recuperação de imagens baseada em conteúdo (Content Based Image Retrieval) é um desafio que vêm ganhando interesse por parte dos pesquisadores nas áreas de banco de dados, processamento digital de imagens e visão computacional. Vários sistemas para recuperação de imagens baseada em conteúdo têm sido desenvolvidos [1,7,8]. Nestes sistemas a imagem é modelada por características que a representam como cor, textura ou forma. Consultas, em bases de dados de imagens, baseadas em abrangência ou nos k-vizinhos mais próximos são freqüentemente utilizados para recuperar imagens utilizando métodos de acessos espaciais como da família R-tree (R-tree[6] e R^*-tree [2]). No entanto, os métodos de acesso espacial não tratam o conteúdo da imagem diretamente. Neste caso, o vetor característica corresponde à localização espacial do objeto sendo indexado (os objetos são aproximados pelos limites dos seus retângulos mínimos). Os métodos de acesso espaciais assumem que os dados pertencem a um espaço vetorial multidimensional. Estes métodos não são aplicáveis em situações onde os dados pertencem a um espaço vetorial métrico, como é o caso de cadeia de caracteres ou de imagens médicas.

Existem vários trabalhos na literatura de Métodos de Acesso Métrico, sendo um dos mais eficientes a Slim-tree [9,10]. Nossa pesquisa consiste em desenvolver uma árvore de indexação métrica, denominada Slim$^+$-tree, que é uma variação da Slim-tree, cujo objetivo é organizar os objetos em nós de acordo com uma medida de dispersão mínima para aumentar a eficiência na recuperação da informação.

2. Trabalhos e Iniciativas Similares

Vários métodos de acesso métrico (MAM) foram desenvolvidos, entre eles M-tree [3] e Slim-tree [9,10]. Dentre os MAM, a M-tree foi a primeira estrutura métrica dinâmica e balanceada proposta na literatura e a Slim-tree foi a primeira a tratar explicitamente a sobreposição entre nós. Para garantir uma árvore com sobreposição reduzida entre os nós, a Slim-tree avalia periodicamente o grau de sobreposição entre os nós. Caso ocorra uma sobreposição indesejada, a árvore é modificada. Cada nó possui um representante, que define o centro do agrupamento e conseqüentemente o raio de cobertura de cada nó. Este representante só é modificado quando ocorre uma divisão do nó. A inserção de novos objetos pode ser realizada baseando-se em uma das três estratégias: mínima distância, mínima ocupação ou randômica. A estratégia baseada na mínima distância tem como objetivo obter o agrupamento com menor raio de cobertura. No entanto, como o representante é fixo, o raio de cobertura pode crescer de forma desnecessária. Com o objetivo de reduzir o grau de sobreposição entre os nós de uma árvore de estrutura métrica dinâmica e balanceada, propomos neste trabalho a Slim$^+$-tree,uma variação da Slim-tree, com modificações na estrutura do nó de tal forma a representá-lo de acordo com o conteúdo dos objetos que o compõem. Propomos também uma nova estratégia de inserção baseada no grau de dispersão que o novo objeto provoca no agrupamento representado pelo nó sendo analisado.

3. Formalização do Problema

Analisando a maneira que os objetos são inseridos na Slim-tree, percebemos que, após algumas inserções, o representante do nó não é mais centro do agrupamento e sim o centro do círculo. Este fato acarreta um aumento excessivo do raio de cobertura do nó, e maior probabilidade de sobreposição entre os nós e conseqüentemente maior ineficiência da recuperação da informação.

O objetivo deste trabalho é desenvolver um método de acesso métrico que permita uma recuperação eficiente dos dados. Primeiramente, a Slim$^+$-tree propõe uma estratégia de inserção em que a escolha do nó privilegia aquele nó em que o novo objeto causará a menor dispersão no agrupamento. A estratégia de menor dispersão garante que estarão armazenados no mesmo o nó objetos próximos entre si, o que resultará em nós com raios menores de cobertura e com menor sobreposição. Segundo, na Slim$^+$-tree o representante do nó será o centro efetivo do agrupamento, podendo ser alterado a cada inserção de um novo objeto. Na Slim-tree a alteração do representante do

nó só ocorre no momento da divisão de nós. A alteração constante do representante do nó representa um custo computacional maior para o algoritmo de inserção, no entanto garante uma recuperação mais rápida da informação, uma vez que objetos similares provavelmente estarão no mesmo nó. Por último, a Slim$^+$-tree utiliza a mesma estratégia para dividir um nó que a usada pela Slim-tree (algoritmo de agrupamento MST), no entanto introduzimos uma modificação que permite escolher, em caso de empate, aquela divisão que resulta em agrupamentos com a menor dispersão em relação as distâncias dos objetos.

O representante na Slim$^+$-tree é aquele objeto que está simultaneamente mais próximo de todos os objetos daquele nó. Como critério de escolha para direcionar a inserção de um objeto em um nó, a Slim$^+$-tree utiliza uma medida de dispersão das distâncias entre os objetos do nó sendo analisado. Esta medida é baseada na diferença entre os coeficientes de variação do nó, antes e depois da inserção provisória do novo objeto no nó candidato. Um nó é candidato se o objeto a ser inserido está dentro de seu raio de cobertura. Caso não exista nenhum nó cujo raio cubra o novo objeto, os nós candidatos são todas as entradas do nó de índice. É calculado, para cada nó candidato a suportar a inserção, o coeficiente de variação do nó antes da inserção e depois da inserção provisória do novo objeto. A diferença entre os dois coeficientes de variação nos dá uma medida do grau de perturbação que a inserção do novo objeto causa ao nó sendo analisado. O nó com o menor grau de perturbação é escolhido para abrigar o novo objeto. A idéia por trás desta estratégia é garantir nós cujos objetos mantenham alto grau de proximidade entre si. A cada inserção de um novo objeto, o representante do nó é reavaliado e modificado se necessário, de tal forma a continuar sendo o centro do agrupamento.

4. Estrutura da Slim$^+$-tree

A Slim$^+$-tree é uma árvore balanceada e dinâmica que cresce das folhas para o nó raiz (bottom-up) e os objetos são armazenados nas folhas assim como a Slim-tree. Os objetos são agrupados em páginas de disco de tamanho fixo, cada página correspondendo a um nó. Como as outras árvores métricas, a Slim$^+$-tree possui nós folha e nós índice que podem conter no máximo M. Os objetos são armazenados nas folhas da árvore e os nós índices correspondem a uma estrutura hierárquica que direciona as pesquisas.

A estrutura dos nós da Slim$^+$-tree é formada pelo cabeçalho do nó e por um conjunto de M entradas. O cabeçalho do nó não índice é formado pelos mesmos campos que o do nó folha.

O cabeçalho do nó é formado pelos campos:

NEntr: é o número de objetos armazenados no nó.
Var. (σ^2): é a variância das distâncias dos objetos entre si.
SArcos: é a soma dos pesos dos arcos.
A entrada de um nó folha possui os campos:

Obj_i: é o objeto
Id_i: identificador do objeto
$Mdist_{i:}$ maior distância entre o objeto i de todos objetos k do nó, para k = 1 até NEntr.

A entrada de um nó índice possui os campos:

$Objs_i$: é o representante do nó da subárvore cuja raiz é i.
RCs_i: raio de cobertura da subárvore, cuja raiz é $Ptrs_i$
$Ptrs_i$: aponta para a subárvore cujo representante é $Objs_i$.

A Figura 1 apresenta um exemplo da estrutura da Slim$^+$-tree. No exemplo, os objetos inseridos são o139, o140, o141, o144, o145,o146, o147. A capacidade de cada nó é de M=4.

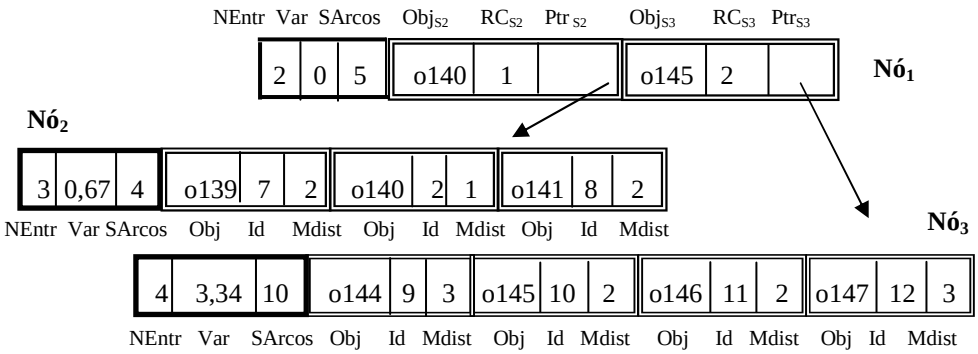


Figure 1. Exemplo da estrutura da Slim$^+$-tree

5. Metodologia Utilizada e Estado Atual da Pesquisa

Iniciamos o trabalho com o levantamento bibliográfico dos métodos de indexação existentes (SAM, PAM e MAM) que foi de fundamental importância para o método proposto. Realizamos seminários para discussões dos métodos até então propostos, de onde surgiram idéias que colaboraram para a idealização da Slim$^+$-tree. Identificamos os principais problemas dos métodos de acesso métricos apresentados na literatura e propusemos como projeto uma estrutura de indexação que os resolva. Estamos em processo de implementação da árvore Slim$^+$-tree utilizando a linguagem C. Realizaremos testes comparativos para que seja verificada a eficiência da estrutura sendo proposta. Simultaneamente aos testes será feita a redação da dissertação.

6. Resultados Esperados, Relevância e Aplicabilidade das Contribuições

Com a validação da nova técnica de indexação esperamos a redução no grau de sobreposição entre os nós o que resulta numa recuperação mais rápida da informação. Isso poderá ser conseguido, pois adotamos medidas estatísticas para auxiliar na escolha de um nó para a inserção de um novo objeto. A estratégia de inserção da estrutura proposta garante que os nós mantêm objetos agrupados de acordo com a similaridade entre si (que é medido pelas distâncias entre si).

A estrutura está sendo implementada de forma modular podendo ser incluída em sistema de gerenciamento de banco de dados que permitem a inclusão de novos recursos, por parte do usuário, como é o caso do PostgreSQL. A estrutura Slim$^+$-tree suporta a implementação de consultas por abrangência, por k-vizinhos mais próximos e por abrangência difusa [5]. A Slim$^+$-tree será incorporada ao sistema AMDI – Atlas Indexado de Mamografias Digitais, que gerencia uma base de dados de mamografias digitais [4], como estrutura para indexar consultas por conteúdo.

Referências

1. Antani, S., Long, R., Thoma, G. R., Lee, D.J., “Storage and Retrieval for Media Databases”. *Proceedings of IS&T/SPIE Electronic Imaging Science and Technology*, Vol.5021, pp. 405-416, 2003.
2. Beckmann, N., Kriegel, H.-P., Schneider, R.; Seeger, B., “The R*-tree: An efficient and robust access method for points and rectangles". *ACM-SIGMOD (1990)*, pp. 322–331.
3. Ciaccia, P.; Patella, M.; Zezula, P., “M-Tree: An Efficient Access Method for Similarity Search in Metric Spaces”. *VLDB (*1997) 426-435.

4. Guliato, D., Caetano, M., Rangayyan, R.M., De Azevedo Marques, P.M., Rodrigues, J.A.H. "Database Architecture and Queries Strategies for Content-based Retrieval of Mamograms".*Proceedings of the 7th International Workshop on Digital Mamography*. 2004.To appear.
5. Guliato, D., Caetano, M., Janones, F.R., Deus, V., De Azevedo Marques, P.M., Rangayyan, R.M., Lima, S.C.L., Medeiros,T.A., Rodrigues, J.A.H., "AMDI : An Indexed Atlas of Digital Mammograms Available via the Web ".*Proceedings of the 7th International Workshop on Digital Mamography*. 2004. To appear.
6. Guttman, A., "R-trees: A dynamic index structure for spatial searching". *ACM-SIGMOD (1984)* 47–54.
7. Kurniawati, R., Jin, S., Shepherd,J.A. "Techniques for Supporting Efficient Content-based Retrieval in Multimedia Databases". *Astralian Computer Journal*, 29(4) (1997) 122-130.
8. Petrakis,E.G. M, Faloutsos,C, "Similarity Searching in Medical Image Databases". *IEEE Transaction on Knowledge and Data Engineering*, 9(3) (1997) 435-447.
9. Traina Jr., Caetano; Traina, Agma; Seeger, Bernhard; Faloutsos, Christos. "Slim-trees: High Performance Metric Trees Minimizing Overlap Between Nodes". *Proceedings of the 8th International Conference Extending Database Technology (EDBT'00)*, 41-65, Konstanz,Germany, 2000.
10. Traina Jr., Caetano; Traina, Agma; Faloutsos, Christos; Seeger, Bernhard, "Fast Indexing and Visualization of Metric Data Sets using Slim-Trees". *IEEE Transactions on Knowledge and Data Engineering*, 14(2) (2002) 244-260.

Mineração de Padrões Seqüenciais Múltiplos*

Daniel A. Furtado[1†] Sandra de Amo[1]

[1]Laboratório de Computação Científica - Universidade Federal de Uberlândia

daniel@lcc.ufu.br, deamo@ufu.br

Nível: Mestrado
Programa de Mestrado em Ciência da Computação
Universidade Federal de Uberlândia
Ingresso: Março de 2003
Previsão de Conclusão: Março de 2005

Resumo

*Os estudos já realizados em mineração de padrões seqüenciais têm focalizado em padrões temporais que, de alguma maneira, podem ser especificados na Lógica Temporal **Proposicional**. No entanto, existem alguns padrões seqüenciais interessantes cuja especificação necessita de um formalismo mais expressivo, o formalismo da Lógica Temporal de **Primeira Ordem**. Neste trabalho, apresentamos o problema de mineração de padrões seqüenciais múltiplos (**psm**), que são padrões temporais de primeira ordem, e propomos três algoritmos (baseados na idéia Apriori) para minerar tais padrões. Dois desses algoritmos foram projetados para minerar todos os padrões considerados freqüentes no banco de dados, enquanto o terceiro foi desenvolvido para realizar a mineração considerando restrições impostas pelo usuário. O primeiro deles, o algoritmo PM (**Projection Miner**), adapta a idéia do algoritmo GSP para mineração de seqüências. PM projeta o padrão de primeira ordem em duas componentes proposicionais durante as fases de geração e poda. O segundo algoritmo, o algoritmo SM (**Simultaneous Miner**), executa as fases de geração e poda sem decompor o padrão, isto é, o processo de mineração não reduz o padrão a componentes proposicionais. Nossos experimentos mostram que SM executa mais rápido que PM. Por fim, propomos também o algoritmo PM-REC para minerar **psms** com restrições impostas pelo usuário.*

Abstract

Previous studies on mining sequential patterns have focused on temporal patterns specified by some form of propositional temporal logic. However, there are some interesting sequential patterns whose specification needs a more expressive formalism, the first-order temporal logic. In this work, we present the problem of mining multi-sequential patterns which are first-order temporal patterns and we propose three Apriori-based algorithms to perform this mining task. Two of these algorithms find all the frequent patterns in database and the third algorithm executes the mining process for consider constraints informed by the user. The first one, the PM (Projection Miner) Algorithm adapts the key idea of the classical GSP algorithm for propositional sequential pattern mining by projecting the first-order pattern in two propositional components during the candidate generation and pruning phases. The second algorithm, the SM (Simultaneous Miner) Algorithm, executes the candidate generation and pruning phases without decomposing the pattern, that is, the mining process, in some extent, does not reduce itself to its propositional counterpart. Our extensive experiments shows that SM scales up far better than PM. Finally, we also propose the PM-REC (Projection Miner with Regular Expression Constraint) algorithm to discover patterns satisfying constraints related to users specific interest.

*Trabalho desenvolvido com apoio financeiro do CNPq.

†Bolsista de mestrado CNPq.

1. Introdução

O problema de descobrir padrões seqüenciais em dados temporais tem sido bastante estudado em vários artigos [2, 3, 6, **?**, 7] e sua importância é altamente justificada pelo grande número de setores onde a mineração de padrões seqüenciais pode ser aplicada com êxito, tais como no mercado financeiro (evolução de cotações de ações), no varejo (evolução de compras de clientes), na medicina (evolução dos sintomas dos pacientes), etc. Diferentes tipos de padrões seqüenciais e algoritmos para mineração dos mesmos já foram propostos e a maioria desses padrões pode ser formalmente especificada através da Lógica Temporal Proposicional.

Neste trabalho, propomos um novo padrão temporal, denominado *padrão seqüencial múltiplo (psm)*, que não pode ser expresso na Lógica Temporal Proposicional. Tal padrão possui várias aplicações, como no mercado financeiro, em vendas a varejo e de alguma forma, tem como objetivo representar o perfil de indivíduos relacionados entre si por algum critério, ao longo do tempo. As seguintes situações são exemplos de padrões seqüenciais múltiplos: (a) as cotações das ações x, y de uma mesma empresa freqüentemente apresentam o seguinte comportamento: um incremento de n pontos de x é seguido de um aumento de m pontos de y e um posterior decremento de k pontos de x. (b) os clientes x e y trabalhando em um mesmo local geralmente possuem o seguinte perfil de compra: quando um cliente x compra um computador M, algum tempo depois seu (sua) colega y compra o mesmo computador M, em seguida o cliente x compra uma impressora P e seu (sua) colega novamente compra a mesma impressora P. Ao contrário dos padrões seqüenciais (proposicionais) estudados até aqui [2, 3, 6, **?**], os padrões seqüenciais múltiplos não podem ser representados na Lógica Temporal Proposicional e necessitam do poder de expressividade da Lógica Temporal de Primeira Ordem para sua especificação. Por exemplo, o padrão expressando o comportamento das ações no mercado de valores de uma mesma empresa pode ser especificado pela sequinte fórmula temporal de primeira ordem: $\exists x_1 \exists x_2 (grupo(x_1, x_2) \wedge sobe(x_1, n) \wedge \Diamond(\ sobe(x_2, m) \wedge \Diamond\ desce(x_1, k)))$. O operador temporal $\Diamond$ significa "*em algum momento no futuro*".

Além de propor um novo tipo de padrão, apresentamos também três algoritmos para minerá-lo. Dois deles, os algoritmos PM (*Projection Miner*) e SM (*Simultaneous Miner*), executam a tarefa de mineração sem considerar qualquer tipo de restrição imposta pelo usuário nos padrões. O terceiro algoritmo, o algoritmo PM-REC (*Projection Miner with Regular Expression Constraints*), é proposto com o intuito de incorporar no processo de mineração um mecanismo o qual possibilita que somente os padrões satisfazendo restrições informadas pelo usuário sejam obtidos.

O restante deste documento está organizado como se segue. Na Seção 2, descrevemos alguns trabalhos e iniciativas similares. Na Seção 3, apresentamos o trabalho proposto na dissertação, descrevendo o novo padrão e os algoritmos para minerá-lo. Na Seção 4, apresentamos a metodologia utilizada e o estado atual da pesquisa e por fim as conclusões e resultados esperados são discutidos na Seção 5.

2. Trabalhos Relacionados

O problema de minerar padrões sequenciais simples de um banco de dados de transações de clientes foi originalmente introduzido por Agrawal e Srikant em [2]. Este artigo apresenta, entre outros, o algoritmo Apriori-All para minerar tais padrões. Um banco de dados D a ser minerado é composto por transações de clientes, onde cada transação consiste de um *cliente-id*, de um *tempo* correspondendo ao instante em que a transação ocorre e dos *itens* comprados na transação. Apriori-All encontra todos os padrões freqüentes em D com relação a um parâmetro α, denominado de *nível mínimo de suporte*, que informa o quão freqüente os padrões devem ser.

Posteriormente, Agrawal e Srikant propuseram o algoritmo GSP [3] para mineração de padrões seqüenciais. Eles incorporam restrições nos padrões de modo a atender a interesses específicos do

usuário. Tanto em Apriori-All quanto GSP, a tarefa de mineração é realizada em três fases: (1) *fase da geração das seqüências candidatas*, (2) *fase da poda* e (3) *fase do cálculo do suporte*. Na n-ésima iteração, Apriori-All e GSP geram todas as seqüências candidatas de comprimento n potencialmente freqüentes. Tais seqüências candidatas são geradas a partir da combinação de seqüências freqüentes de tamanho $n-1$, geradas na iteração anterior. Depois que as seqüências candidatas são geradas, são podadas aquelas que contém subsequências de tamanho $n-1$ que não ocorrem entre as sequências frequentes da iteração anterior $n-1$. Por fim, o banco de dados é varrido e somente as seqüências candidatas que nele aparecem com freqüencia superior a um limite mínimo fixado são mantidas. A diferença principal entre GSP e Apriori-All está na maneira como são especializados os padrões seqüenciais de uma iteração para outra. Em Apriori-All, um padrão de tamanho n é uma sequência com n *itemsets* (conjuntos de itens). Já em GSP, um padrão de tamanho n é uma sequência com n *items*. Assim, em GSP, a passagem entre as iterações é feita de forma mais paulatina do que em Apriori-All.

Em [7], a mineração de padrões seqüênciais é tratada considerando outros atributos, além daquele referente aos itens comprados na transação. Por exemplo, o padrão seqüencial multi-dimensional de [8], *s = (São Paulo, 'Alta',* $\{$*Vectra, BMW*$\}$*)* diz que "clientes que moram em São Paulo com renda alta compram Vectra seguido de BMW".

3. Trabalho Proposto na Dissertação

Nesta seção, descreveremos o padrão seqüencial sendo proposto e em seguida apresentaremos a idéia dos três algoritmos para minerá-lo.

A maioria dos trabalhos já realizados em mineração de seqüencias tem focado na descoberta de padrões seqüenciais correspondendo ao perfil de compra de *algum* cliente. No entanto, nosso interesse é descobrir padrões que correspondam ao perfil de compra de um *grupo* de clientes *relacionados um com o outro por algum critério*. O objetivo é descobrir como um dado relacionamento entre clientes pode influenciar nos seus perfis de compra. Veja o exemplo a seguir:

Exemplo 3.1 Suponha que estamos interessados em descobrir como o ambiente de trabalho pode influenciar no perfil de compra dos clientes. Clientes trabalhando em um mesmo local são agrupados e estes grupos são armazenados na tabela *G*. As transações desses clientes durante um certo período são armazenadas na tabela *Tr*. Nesta tabela, os atributos Gr e T indicam, respectivamente, a qual grupo o cliente pertence e o instante em que a transação é efetuada. Para simplificar a representação, consideramos as transações contendo apenas um item, ao invés de um conjunto de itens.

Tr

Gr	*Cliente*	*T*	*Item*	*Gr*	*Cliente*	*T*	*Item*
1	Paul	1	Comp MX	2	Charles	9	DVD
1	Mary	7	Comp MX	2	Susan	10	DVD
1	Sally	8	VCR	3	John	10	TV
1	Paul	9	Imp MZ	3	Gina	11	TV
1	Mary	10	Imp MZ	4	Gloria	12	TV
1	Charles	7	TV	4	Bill	13	TV
2	Susan	8	TV	4	Frank	14	VCR

G

IdGr	*Grupo*
1	{Paul,Mary,Sally}
2	{Charles,Susan}
3	{John,Gina}
4	{Gloria,Bill,Frank}

Considere a seguinte seqüência de transações efetuadas por dois clientes: (1) o primeiro cliente compra um computador MX e em seguida (2) o segundo cliente também compra um computador MX. Após isso, (3) o primeiro cliente compra uma impressora MZ e finalmente, em um momento posterior, (4) o segunto cliente também compra uma impressora MZ. Esta seqüência de transações é um exemplo de um padrão seqüencial múltiplo que é suportado somente pelo grupo de trabalho 1 da tabela G acima. Se

consideramos um suporte mínimo de 50%, então tal padrão não é tido como freqüente, pois é suportado por somente 25% dos grupos (grupo 1 dos quatro existentes).

Um *psm* pode ser representado na forma de uma matriz, onde as colunas estão associadas aos clientes e as linhas correspondem às transações desses clientes (analisadas de baixo para cima). Abaixo temos a representação, em forma de matriz, de dois padrões. A matriz (a) representa o padrão citado no exemplo anterior.

$$\begin{pmatrix} \perp & impressoraMZ \\ impressoraMZ & \perp \\ \perp & computadorMX \\ computadorMX & \perp \end{pmatrix} \quad \begin{pmatrix} \perp & d \\ \perp & c \\ b & \perp \\ a & \perp \end{pmatrix}$$

(a) (b)

O símbolo $\perp$ nessas matrizes indica que não estamos interessados nas compras do cliente em questão naquele determinado instante. O número de colunas (clientes) da matriz que representa o padrão é denominado de *rank* do *psm* e o número de linhas da mesma corresponde ao *comprimento* desse padrão.

Nosso objetivo é encontrar todos os *psms* interessantes freqüentes de um banco de dados D de grupos de clientes e transações desses clientes, com relação a um suporte mínimo α. Se o número de clientes em cada grupo de D for exatamente 1, tal problema de mineração recai à mineração convencional de padrões seqüenciais. Nesse aspecto, podemos dizer que o padrão sendo proposto é uma generalização daqueles já conhecidos.

A idéia geral dos algoritmos PM e SM propostos é descrita a seguir. Primeiramente são gerados os *psms* de rank 1 (e comprimento 1, 2, 3, etc) utilizando para isto um algoritmo de mineração de padrões seqüenciais simples (por exemplo o algoritmo GSP [3]). Em seguida, são gerados os *psms* de rank n e comprimento k iterativamente. Na iteração 2, por exemplo, são gerados os *psms* de rank 2 e comprimento 2 inicialmente, depois aqueles de rank 2 e comprimento 3, até que sejam gerados os *psms* de rank 2 de maior comprimento possível. Na iteração 3 são gerados os *psms* de rank 3 e comprimento 3, depois os de rank 3 e comprimento 4, e assim por diante. A cada iteração, o conjuntos dos novos *psms* de rank n e comprimento k são gerados a partir de conjuntos de *psms* freqüentes de comprimento e/ou rank menores. É importante deixar claro que o algoritmo GSP ou correspondente é utilizado apenas na primeira iteração dos algoritmos PM e SM e que o mesmo não pode ser utilizado para realizar toda a tarefa de mineração.

O algoritmo PM, como já mencionado anteriormente, realiza a *geração* e *poda* dos *psms* candidatos decompondo primeiramente cada *psm* em duas componentes. Considere como exemplo, os *psms* (a) e (b) ilustrados nas matrizes acima. O *psm* (a) pode ser totalmente caracterizado pela seqüência simples <computador MX, computador MX, impressora MZ, impressora MZ> informando a seqüência de produtos comprados e pela seqüencia $< 1, 2, 1, 2 >$, indicando a ordem das compras feitas pelos dois clientes. Da mesma forma, o *psm* (b) pode ser decomposto nas seqüências simples: $< a, b, c, d >$ e $< 1, 1, 2, 2 >$. As seqüências <computador MX, computador MX, impressora MZ, impressora MZ> e $< a, b, c, d >$ são denominadas de *seqüências de itens* dos *psms* e as seqüências $< 1, 2, 1, 2 >$ e $< 1, 1, 2, 2 >$ são chamadas de *formas* dos *psms*. Nas fases de geração e poda dos *psms* candidatos, o algoritmo PM trabalha com essas duas componentes de forma independente. Na obtenção dos *psms* de rank n e comprimento k, por exemplo, PM decompõe os *psms* de comprimento e rank inferior em suas componentes (obtendo as seqüencias de itens e formas desses *psms*) e gera a partir delas, novas seqüências de itens e formas de comprimento maior. Em seguida, o algoritmo realiza uma poda daquelas seqüências de itens e formas que não podem ser freqüentes. Para tal, o algoritmo se basea no conjunto de seqüências de tamanhos inferiores geradas até o momento para eliminar as seqüências candidatas que possuem sub-seqüências não presentes naquele conjunto. Por fim, as seqüências de itens são combinadas às formas (é feito o processo inverso àquele realizado na decomposição dos padrões) formando os *psms* completos. Só então, PM varre o banco de dados afim de obter somente os *psms* freqüentes.

O algoritmo SM difere do algoritmo PM por não fazer a decomposição dos padrões em suas componentes. Todo o processo de geração e poda de novos padrões de comprimento e/ou rank maiores também é feito a partir de padrões freqüentes de comprimento menor, mas sem realizar a quebra desses *psms* em padrões simples. Aqui, os padrões inteiros são combinados adequadamente, gerando *psms* de comprimento e/ou rank maiores. Para detalhes, ver texto completo em [5]. Aplicamos ambos algoritmos em diversos bancos de dados sintéticos e analisamos os tempos de execução dos algoritmos variando o nível mínimo de suporte. Através de nossos experimentos, concluímos que SM executa em média três vezes mais rápido que PM. Assim, podemos concluir que embora seja possível decompor um psm em componentes proposicionais e obter um algoritmo similar àqueles tradicionais (GSP ou Apriori-All) nas fases de geração e poda para minerar esse padrão (que é o caso do PM), a performance desse algoritmo revela-se muito aquém da performance do algoritmo SM, que por sua vez realiza a mineração sem decompor o padrão em componentes proposicionais.

Estamos propondo também um terceiro algoritmo, o algoritmo PM-REC para mineração de *psms* com restrições impostas pelo usuário. O objetivo de considerar tais restrições é possibilitar maior controle por parte do usuário dos padrões que serão minerados, além de diminuir o espaço de busca na fase de geração de candidatos do algoritmo (há um aumento de performance). PM-REC incorpora duas restrições expressas através de expressões regulares no processo de geração dos *psms*. Uma restrição é utilizada na componente referente a *seqüência de itens* do *psm* e permite que o usuário informe características com relação a essas seqüências. Outra restrição é utilizada na componente do *psm* referente à sua *forma* e possibilita ao usuário informar as formas dos padrões em que está interessado. Somente aqueles padrões satisfazendo tais expressões regulares serão gerados pelo algoritmo. PM-REC se basea na idéia do algoritmo SPIRIT-V ([8]). Atualmente, o algoritmo está em fase de implementação.

4. Metodologia Utilizada e Estado Atual do Trabalho

A metodologia de pesquisa sendo utilizada neste trabalho basea-se em etapas, que podem ser descritas como segue:

1. Pesquisa sobre trabalhos relacionados à mineração de padrões seqüenciais.
2. Formalização do problema de mineração de *psms* sem restrições.
3. Proposta e desenvolvimento dos algoritmos PM e SM para mineração desses padrões.
4. Estudo do problema de mineração de padrões seqüenciais múltiplos considerando restrições nos formatos e nos itens das seqüências múltiplas.
5. Proposta e desenvolvimento do algoritmo PM-REC para mineração de padrões seqüenciais múltiplos com restrições.
6. Projeto e implementação dos algoritmos PM e SM. Uma vez implementados, ambos os algoritmos serão testados em bancos de dados sintéticos e uma análise comparativa de performance será realizada.
7. Projeto, implementação e testes do algoritmo PM-REC.
8. Redação da dissertação e conclusão do trabalho.

O trabalho sendo proposto encontra-se em um estágio adiantado. As implementações dos algoritmos PM e SM foram concluídas e exaustivos testes em bancos de dados sintéticos foram realizados. Fizemos

uma análise detalhada com relação à performance e escalabilidade desses algoritmos para vários bancos de dados.

Atualmente estamos implementando o algoritmo PM-REC, destinado à mineração de padrões seqüenciais múltiplos com restrições. As fases de geração e poda de seqüências candidatas desse algoritmo são as mais complexas e demandam maior tempo para serem concluídas. A fase de geração de candidatos já está em estágio final de implementação. Já a fase de cálculo de suporte desse algoritmo é similar às fases de cálculo de suporte dos algoritmos PM e SM, o que nos permitirá reutilizar parte do código já escrito.

Depois de implementado, o algoritmo PM-REC também será testado em bancos de dados sintéticos. As restrições a serem utilizadas serão geradas através de um gerador de expressões regulares construído exclusivamente para este trabalho no Laboratório de Computação Científica (LCC) da Universidade Federal de Uberlândia.

5. Conclusão e Resultados Esperados

Neste trabalho estamos propondo um novo padrão seqüencial para mineração, denominado de padrão seqüencial múltiplo (*psm*) e três algoritmos para minerá-lo. Mostramos que um *psm*, diferente dos padrões seqüenciais comumente estudados, não pode ser especificado através da Lógica Temporal Proposicional e necessita da expressividade da Lógica Temporal de Primeira Ordem. Descrevemos brevemente os algoritmos PM e SM para minerar *psms* e nossos estudos e testes nos permitem concluir que, embora o algoritmo PM realize a geração desses padrões decompondo-os em padrões simples (proposicionais), sua performance é baixa quando comparada à performance do algoritmo SM. Também descrevemos rapidamente a idéia do algoritmo PM-REC, proposto para minerar *psms* com restrições de expressões regulares. Pretende-se, no final desde trabalho, desenvolver um software para mineração de *psms* com ou sem restrições. Será necessário construir um ambiente gráfico que possibilite ao usuário informar, de maneira fácil, os parâmetros e restrições necessários para a mineração.

Referências

[1] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo. *Fast Discovery of Association Rules.* In Fayyad, U.M., Piatetsky, G., Smyth, P; and Uthurusamy, R., eds. Advances in Knowledge Discovery and Data Mining, AAAI Press, 1995.

[2] R. Agrawal, R. Srikant. *Mining Sequential Patterns.* Proc. of the Int. Conference on Data Engineering (ICDE), Taipei, Taiwan, March 1995.

[3] R. Agrawal, R. Srikant. *Mining Sequential Patterns: Generalizations and Performance Improvements.* Proc. of the Fifth Int. Conference on Extending Database Technology (EDBT), Avignon, France, March 1996.

[4] Sau Dan Lee, Luc De Raedt: *Constraint Based Mining of First Order Sequences in SeqLog.* In Proc. of the Workshop on Multi-Relational Data Mining, ACM SIGKDD 2002, Edmonton, Alberta, Canada, July 2002.

[5] S. de Amo, Daniel A. Furtado, A. Giacometti, D. Laurent. *An Apriori-based Approach for First-Order Temporal Pattern Mining.* 19º Simpósio Brasileiro de Bancos de Dados, Outubro 2004, Brasilia, DF.

[6] Mohammed J. Zaki *SPADE: An Efficient Algorithm for Mining Frequent Sequences.* Machine Learning, 0, 1-31, 2000.

[7] H. Pinto, J. Han, J. Pei, K. Wang, Q. Chen, U. Dayal. *Multi-dimensional Sequential Pattern Mining.* CIKM 2001, pp. 81-88.

[8] M.N. Garofalakis, R. Rastogi, K. Shim. *SPIRIT: Sequential Pattern Mining with Regular Expression Constraints.* In Proc. of the 25th VLDB Conference, Edinburgh, Scotland, 1999.

GML Publisher: Um Framework para Publicação de Feições Geográficas como GML

Fábio Bezerra Feitosa[1], Vânia Maria Ponte Vidal[1]
[1]Departamento de Computação, Universidade Federal do Ceará
email: {fabio, vvidal}@lia.ufc.br

Nível: Mestrado
Programa de Mestrado em Ciência da Computação
Departamento de Computação, Universidade Federal do Ceará
Ingresso: Junho/2003
Previsão de conclusão: Junho/2005

Resumo

A missão do consórcio OGC (OpenGIS Consortium) é definir soluções padrões e recomendações que garantam a interoperabilidade entre SIGs. Entre as iniciativas do OpenGis estão a Geography Markup Language ou GML e a especificação Web Feature Service (WFS). O propósito da especificação WFS é descrever as operações de manipulação de dados no formato de instâncias de feições geográficas, codificadas em GML.
Neste trabalho, propomos GML Publisher, um framewok para Publicação de dados Geográficos Armazenados em Banco de dados Relacional ou Objeto-relacional como GML. O framework proposto deverá atender à especificação WFS publicadas pelo Open GIS Consortium (OGC). Além da implementação do Framework, será desenvolvido também um ambiente para facilitar publicação e manutenção de feições no GML Publisher.

Palavras-chave: Sistemas de Informações Geográficas; OpenGIS Consortium; WFS Specification; Geography Markup Language.

Abstract

The mission of the OpenGIS Consortium (OGC) is to promote the development and use of advanced open system standards and techniques in the area of geoprocessing and related information technologies. OGC manages a global consensus process that results in approved interfaces and encoding specifications that enable interoperability among diverse geospatial data stores, services, and applications [1]. Two important OGC's initiatives are: the Geography Markup Language (GML) and the Web Feature Service (WFS) specifications. The purpose of the WFS specifications is to describe the manipulation operations over geospatial data using GML.
In this work, we propose GML Publisher, a framework for publishing, as GML, geospatial data, which may be stored either in a Relational Database, or in an Object-Relational Database. XML publisher will attend the WFS specification proposed by OGC. We also propose an environment to support the process of feature publishing in our framework.

Key Words : Geographic Information System; OpenGIS Consortium; WFS Specification; Geography Markup Language.

1. Introdução

Servidores que implementam a especificação WFS proposta pelo Open GIS Consortium (OGC)[6] são chamados de Servidores WFS e objetivam proporcionar a consulta, atualização, troca e transporte de dados geoespaciais no formato de instâncias de feições geográficas, codificadas em GML [7]. GML é uma extensão de XML, que foi proposta pelo OGC como formato padrão para representação de dados geográficos na web.

Segundo o OGC, uma feição geográfica é "uma abstração de um fenômeno do mundo real que está associada com uma posição relativa à Terra". Pode-se descrever a forma e a localização de uma feição através de sua geometria, sendo que as demais propriedades da feição são representadas por atributos não geométricos (textuais, numéricos ou booleanos). Dado que os servidores WFS publicam visões GML de feições geográficas armazenadas em fontes de dados, usa-se dizer simplesmente que estes "publicam feições". Dessa forma, o usuário pode consultar e atualizar as fontes de dados através de uma visão/feição publicada.

A proposta desse trabalho é o desenvolvimento de um framewok para Publicação de Dados Geográficos Armazenados em Banco de Dados Relacional ou Objeto-Relacional como GML. O Framework deverá atender às especificações WFS publicadas pelo Open GIS Consortium (OGC). Uma requisição WFS consiste de uma descrição de operação de consulta ou transformação de dados e são aplicadas a uma ou mais feições. Existem cinco tipos de operações WFS. As operações GetFeature e Transaction permitem a consulta e atualização das feições publicadas.

2. Framework Proposto

A arquitetura do GML Publisher é ilustrada na Figura 1. O Módulo de Processamento de Consultas (MPC) é responsável pelo processamento de requisições WFS de consulta (GetFeature) e o Módulo de Processamento de Atualização (MPA) pelo processamento das requisições WFS de atualizações (Transaction).

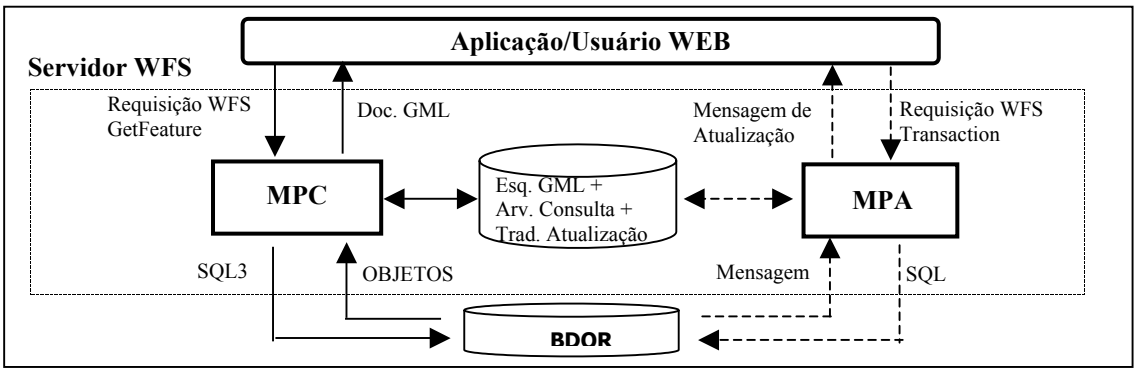


Figura 1 – Arquitetura do GML Publisher

A publicação de uma feição no GML Publisher é realizada em três passos: (i) o usuário define o esquema GML do tipo da feição. (ii) Em seguida, escolhe a tabela *master* e define as assertivas de correspondência [10,11,14] das propriedades do tipo da feição com atributos ou caminhos da tabela *master*. Essas assertivas especificam de forma axiomática como os valores das propriedades de uma instância do tipo da feição são derivados a partir dos atributos/caminhos de uma tupla da tabela *master*. O formalismo proposto permite especificar várias formas de correspondências, inclusive casos onde existe heterogeneidade estrutural [1,2]. (iii) Com base nas assertivas de correspondência do tipo da feição, o GML Publisher gera automaticamente a árvore de consultas e os tradutores para as operações de atualização da feição publicada.

O processamento de uma atualização WFS é realizado em 3 passos: (i) a atualização WFS é traduzida em uma seqüência de atualizações definidas sobre o banco. A tradução é feita de forma eficiente uma vez que os tradutores das operações de atualização foram definidos por ocasião da publicação da feição. (ii) A atualização SQL é processada pelo SGBD (iii) O resultado da operação (sucesso ou falha) é enviado ao WFS que o repassa ao usuário.

O processamento de uma consulta WFS é também realizado em 3 passos: (i) A consulta WFS é traduzida em uma única consulta SQL3 definida sobre o esquema do banco. A árvore de consultas é usada pelo MPC para traduzir de forma eficiente uma requisição WFS de consulta em padrão SQL3 (ii) A consulta SQL3 gerada é submetida ao SGBD. (iii) O resultado da consulta é transformado no documento GML correspondente. A seguir apresentamos um exemplo de processamento de consulta no Framework proposto.

Considere o esquema do Banco de Dados na Figura 2 . Suponha a feição F_Escola onde Geom_rel é a tabela *master*, e T[F-escola] é o tipo da feição publicada cuja estrutura é mostrada na Figura 3. Na Figura 4, para cada propriedade de F_Escola mostramos a assertiva que especifica a correspondência desta com um atributo ou caminho da tabela Geom_rel. Por exemplo, a assertiva $T[\text{F-escola}].\text{projeto} \equiv \text{Geom\_rel}.(FK_1)^{-1}.(FK_3)^{-1}.FK_2$, onde FK_1, FK_3 e FK_2 são chaves estrangeiras, especifica que dada uma instância $f de T[F-escola], a qual corresponde a tupla $t na tabela Geom_rel, então $f/projeto contém todos os projetos que estão relacionados com a tupla $t através do caminho $(FK_1)^{-1}.(FK_3)^{-1}.FK_2$. As assertivas são mapeadas nas templates das consultas como mostrado na Figura 3. A Figura 5 mostra um exemplo de requisição WFS de consulta para a feição "F_Escola". A seguir discutimos cada passo do processamento dessa consulta:

1. A consulta WFS é traduzida na consulta SQL3 da Figura 6. A cláusula **From** referencia a tabela *master* e tabelas referenciadas na condição de seleção. O elemento Filter é mapeado na cláusula **Where**. Para cada PropertyName na consulta WFS é gerada uma consulta parcial na cláusula **Select**, definida de acordo com a template correspondente na árvore de consultas;
2. A consulta é executada no banco. Suponha que o resultado da consulta contém os objetos mostrados na Figura 7.

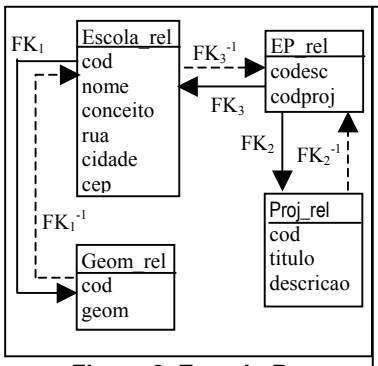


Figura 2–Esq. do Banco

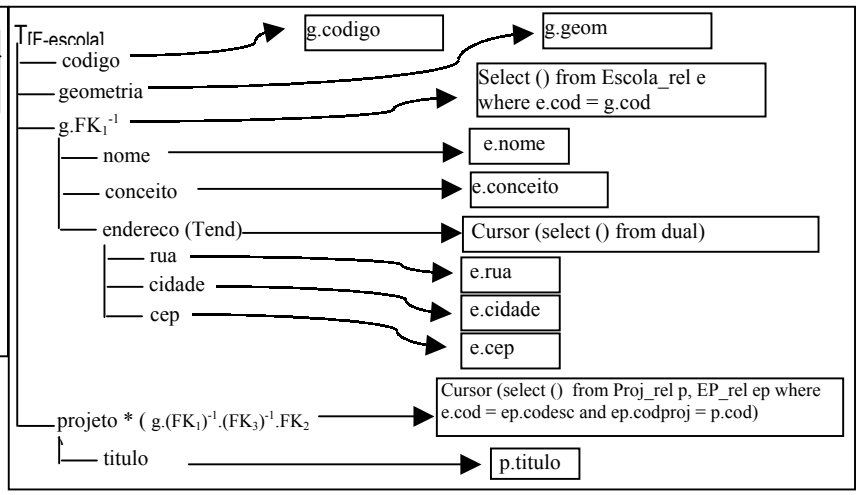


Figura 3 – Árvore de Consultas de T[F-escola]

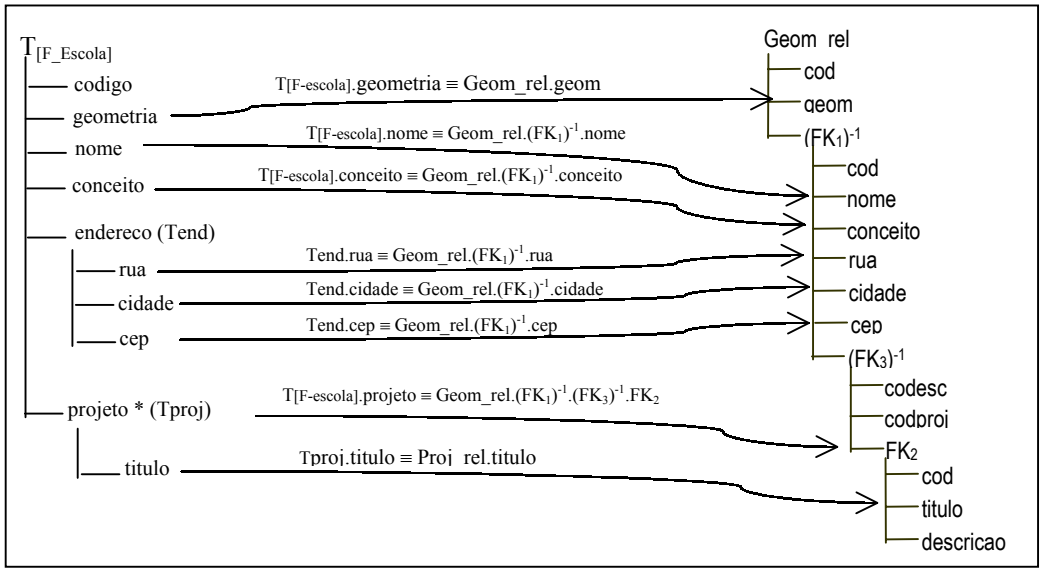


Figura 4 – Assertivas de Correspondência do Tipo da Feição

3. O resultado da consulta é convertido no Documento GML da Figura 8. Cada objeto do resultado é mapeada em um elemento featureMember e o elemento boundedBy é derivado dos limites geográficos das feições retornadas. Observe que o mapeamento de objeto em featureMember é direto, dado que estes têm a mesma estrutura.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<wfs:GetFeature outputFormat="GML2" …>
<wfs:Query typeName="F_Escola">
   <wfs:PropertyName>nome</wfs:PropertyName>
   <wfs:PropertyName>endereco</wfs:PropertyName>
 <wfs:PropertyName>projeto/titulo</wfs:PropertyName>
   <wfs:PropertyName>geometria</wfs:PropertyName>

  <ogc:Filter>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>conceito</ogc:PropertyName>
       <ogc:Literal>A</ogc:Literal>
    </ogc:PropertyIsEqualTo>
  </ogc:Filter>
</wfs:Query></wfs:GetFeature>
```

Figura 5 – Req. WFS de Consulta

```
Select
    e.nome,

cursor (select e.rua, e.cidade, e.cep from dual) endereco,
cursor (select p.titulo from Proj_rel p, EP_rel ep
                where e.cod = ep.codesc and
                ep.codproj = p.cod) projeto,
   g.geom geometria
From   Escola_rel e, Geom_rel g   <-- tabela master
Where  e.cod = g.cod and e.conceito = 'A'
```

Figura 6 – Consulta SQL submetida ao Banco

nome	endereco	projeto	geometria
CHRISTUS	rua: D. LUIS; cidade: FORTALEZA; cep: 60125070	titulo: HORA DA POESIA	15.0, 20.0
ARI DE SÁ	rua: D. PEDRO; cidade: FORTALEZA; cep: 60130040	titulo: BRACOS ABERTOS; BIBLIOTECA CIRCULANTE	30.0, 25.0

Figura 7 – Resultado da Consulta SQL

```
<?xmlversion="1.0"?>
<wfs:FeatureCollection xmlns=http://www.server.com/myns ...>
<gml:boundedBy>
  <gml:Box srsName="http://www.opengis.net/./epsg.xml#4326">
     <gml:coordinates>15.0,20.0, 30.0,25.0</gml:coordinates>
  </gml:Box></gml:boundedBy>
<gml:featureMember>
  <F_ESCOLA fid="E1">
     <nome>CHRISTUS</nome>
     <endereco> <rua>D. LUIS</rua>
                <cidade>FORTALEZA</cidade>
                <cep>60125070</cep>
     </endereco>
     <projeto>
        <titulo>HORA DA POESIA</titulo>
     </projeto>
     <geometria>
       <gml:Point srsName="EPSG:4326">
          <gml:coordinates cs="," decimal="." ts=" ">15.0,20.0
          </gml:coordinates>
       </gml:Point>
     </geometria>
  </F_ESCOLA>
</gml:featureMember>
<gml:featureMember>
  <F_ESCOLA fid="E2">
     <nome>ARI DE SÁ</nome>
     <endereco> <rua>D. PEDRO</rua>
                <cidade>FORTALEZA</cidade>
                <cep>60130040</cep>
     </endereco>
     <projeto>
        <tituloo>BRACOS ABERTOS</titulo>
     </projeto>
     <projeto>
        <titulo>BIBLIOTECA CIRCULANTE</titulo>
     </projeto>
     <geometria>
       <gml:Point srsName="EPSG:4326">
          <gml:coordinates cs="," decimal="." ts="  ">30.0,25.0
          </gml:coordinates>
       </gml:Point>
     </geometria>
  </F_ESCOLA>
</gml:featureMember>
</wfs:FeatureCollection>
```

Figura 8 - Documento GML retornado

4. Trabalhos e Iniciativas similares

Já existem disponíveis no mercado algumas implementações da especificação WFS [3, 4, 5]. O Deegree [3] é um projeto de software livre (licença GNU LGPL) desenvolvido, em Java, pelo Departamento de Geografia da Universidade de Bonn, Alemanha, que implementa vários serviços especificados pelo OGC, entre eles o serviço WFS. O Deegree WFS utiliza um arquivo de mapeamento próprio para especificar as correspondências entre os elementos de uma feição publicada e os respectivos atributos da fonte de dados. A seguir, discutimos algumas das limitações do Deegree WFS.

Limitações de mapeamento: (i) não permite mapeamentos de tipos complexos. Assim, no caso da feição F_Escola não seria possível definir o mapeamento para a propriedade Endereço (ii) no caso de propriedades diferentes mapeadas na mesma tabela base, estas devem conter os mesmos elementos; (iii) não permite especificar mapeamentos envolvendo chaves estrangeiras compostas;

Limitações de consulta: (i) não permite definir expressões de caminho nos elementos <propertyName> de uma consulta WFS. Por exemplo, a expressão projeto/titulo na consulta da Figura 5 não é válida no Deegree. (ii) <propertyName> referencia nomes de atributos e chaves estrangeiras das tabelas no banco, ao invés de propriedade do tipo da feição. Por exemplo, na consulta da Figura 5, o elemento geometria deveria ser definido como geom, no Deegree.

Limitações de atualização:(i) operações de Inserção somente são traduzidas em uma única tabela (tabela master); (ii) Nos testes realizados, as operações modificação e remoção falharam para Oracle Spatial [9].

Outra desvantagem do Deegree WFS é que o processamento de uma consulta WFS pode ser reformulada em várias consultas SQL, ao invés de uma única consulta como no Framework proposto.

5. Metodologia e Estado Atual da Pesquisa

O trabalho será desenvolvido em etapas intercaladas de pesquisa bibliográfica e implementação de protótipos, sendo que cada etapa é seguida com a elaboração de relatório técnico e seminário para discussão dos resultados.

Já foi realizado um estudo aprofundado do código de implementação do Deegree WFS e identificadas suas limitações. Também já desenvolvemos uma ferramenta gráfica para edição de propriedades de tipos de feição e suas assertivas. Atualmente estamos trabalhando no desenvolvimento dos algoritmos GeraArvoreConsulta e GeraTradutores os quais geram, a partir das Assertivas do tipo da Feição, a árvore de consulta e os tradutores de atualização. Os passos seguintes são a implementação dos módulos MPC e MPA (vide Figura 1).

6. Resultados Esperados e relevância das contribuições

Neste trabalho propomos um Framewok para Publicação de visões GML de feições armazenadas em Banco de Dados Relacional ou Objeto-Relacional. No enfoque proposto uma visão de feição, ou simplesmente uma feição, é representada por uma tripla $F=<R_M, T_{[F]}, \mathbf{A}>$ onde R_M é a tabela master, $T_{[F]}$ é o esquema GML do tipo da feição, e $\mathbf{A}$ é o conjunto de assertivas de correspondência (ACs) de $T_{[F]}$. As Assertivas de correspondência são usadas para gerar a árvore de consulta e os tradutores de atualização da feição, os quais permitem que a tradução de consultas e atualizações WFS possam ser realizadas de forma eficiente no framework proposto. Espera-se que o framework proposto resolva as limitações do Deegree WFS.

Outra contribuição deste trabalho é desenvolvimento de uma ferramenta para publicação e manutenção de feições no GML Publisher. A ferramenta de publicação permite que o usuário defina, através de uma GUI, as propriedades da feição e as suas assertivas, e então, gera automaticamente a árvore de consultas e os tradutores de atualização da feição. A vantagem do nosso formalismo é que podemos provar formalmente que as consultas e tradutores gerados realizam corretamente o mapeamento definido pelas assertivas de correspondência. No caso de modificações no esquema do banco de dados, a manutenção das feições pode ser realizada de forma semi-automática, nos seguintes passos: (i) primeiramente, deve-se identificar as feições afetadas pelas modificações do esquema do

banco de dados. Isso pode ser feito de modo automático, baseado nas modificações do esquema do banco de dados e nas assertivas das feições. (ii) Em seguida, as assertivas das feições afetadas devem ser redefinidas de acordo com o novo esquema do banco de dados; e então, (iii) para cada feição afetada, é gerada automaticamente a nova árvore de consultas e os novos tradutores de atualização da feição.

A ferramenta de publicação é uma adaptação da ferramenta DFP (Deegree Feature Publisher) [13], que desenvolvemos para a publicação e manutenção de feições em servidores Deegree WFS. No caso da DFP, o processo de publicação é o mesmo, a única diferença é que a partir das assertivas da feição, a ferramenta gera de forma automática o arquivo de mapeamento próprio da Deegree WFS. A geração manual desse arquivo é tediosa e susceptível a erros, além de que requer que o usuário conheça profundamente a implementação do Degree WFS. Como podemos observar, a nossa ferramenta de publicação pode ser facilmente adaptada para publicar e manter feições para outras implementações do serviço WFS especificado pelo OpenGIS.

7. Referências

[1] BISHR, Y (1998).**Overcoming the Semantic and Other Barriers to GIS Interoperability**. International Journal of Geographical Information Science 12(4):299-314

[2] FONSECA F., EGENHOFER M. (2001). **Sistemas de Informações Geográficos Baseados em Ontologias**. Informática Pública 1 (2):47-65

[3] http://deegree.sourceforge.net/

[4] http://www.ordnancesurvey.co.uk/oswebsite/

[5] http://www.snowflakesoft.co.uk/products/goloader/

[6] OpenGIS Consortium, http://www.opengis.org

[7] OpenGIS Consortium, Schema for Geography Markup Language (GML) 2.0. http://www.opengis.org.

[8] OpenGIS Consortium, Web feature Service Implementation Specification. http://www.opengis.org

[9] Oracle Corporation. http://technet.oracle.com

[10] Popa, L., Velegrakis,Y., Miller, R.J., Hernandez, M.A., Fagin, R.: Translating Web Data. In VLDB, pages 598–609, August 2002.

[11] Rahm, E., Bernstein, P.A.: A Survey of Approaches to Automatic Schema Matching. VLDB Journal, 10(4):334–350, 2001.

[12] Rigaux, P., School, M., Voisard, A.: **Spatial Database With Application To GIS**. Morgan Kaufmann Publishers, 2002

[13] Teixeira, M.: Deegree Feature Publisher: Manual do Usuário, 2004. Disponível em http://www.lia.ufc.br/~teixeira/dfp

[14] Vidal, V.M.P., Vilas Boas, R.: **A Top-Down Approach for XML Schema Matching**. In Proceedings of the 17th Brazilian Symposium on Databases. Gramado, Brazil (2002).

Integration of Scientific Workflows on the Web *

Gilberto Zonta Pastorello Jr.[1] Claudia Bauzer Medeiros[1] (advisor)

[1] Laboratory of Information Systems – Institute of Computing
University of Campinas – CP6176, 13081-970 – Campinas, SP, Brazil
{gilberto,cmbm}@lis.ic.unicamp.br

Level: MSc.
Computer Science MSc. Program
University of Campinas – UNICAMP
Admission: March 2003
Conclusion expectation: March 2005

Abstract

Scientists have traditionally shared data, experiments and research results. Now, they continue to do this via electronic networks and the Internet, but often without an appropriate framework. One possible approach to this problem is coordinating cooperation via scientific workflows on the Web. Our research contributes to these efforts in two directions: proposal of a model compliant with Web standards to store workflow components in databases and publish them on the Web; and development of a set of Web-based tools to specify, edit and compose workflow components.

Keywords: *Workflow, Web services, Semantic Web, scientific information systems, scientific data integration, environmental planning, agricultural zoning.*

*This work was developed with financial support from CNPq, and partial support from the SAI project – Advanced Information Systems – of PRONEX-MCT, as well as WebMaps and AgroFlow CNPq projects.

1. Introduction

Workflow management systems can be used to improve scientific research – e.g. in as astronomy, physics or environmental studies. They make it easier to specify, develop and make use of a wide variety of processes that are often complex and need to be documented. One example of such processes appears in the use of Geographical Information Systems (GIS). The data managed by that kind of system usually pass through a series of operations. Workflows have proved to be a useful tool to specify this succession of operations, and support not only the execution but also the documentation of processes. This allows sharing and reuse of scientific procedures and data. Several scientific applications are currently being executed across the Internet in a distributed fashion, including data access. The coordination of this execution and effective sharing of data are among the problems in this context. There are projects that have developed dedicated workflow systems to allow scientific cooperative work on the Web [6, 8]. These systems are special purpose, with specialized data models.

This work proposes the use of scientific workflow technology to document scientific work on environmental planning and the publishing of this documentation over the Web. To do so, two goals must be attained. First, the workflow data model used must follow Web standards. Second, a set of tools that allow scientists to specify, compose and update workflows for subsequent execution on the Web must be developed. This will serve two goals: provide a Web-based framework for documenting scientific experiments by means of workflows; and contribute to Semantic Web interoperability efforts. Our proposal is based on the experience of the UNICAMP database research group in developing a system of scientific workflows for decision support in environmental planning. This software, called WOODSS, is a mono-user system that aids domain experts to specify and document their work via workflows. Our proposal will extend WOODSS' workflow repository and workflow specification and editing tools to be used on the Web.

The rest of this text is organized as follows. Section 2 reviews related work. Section 3 outlines the proposed solution and methodology. Section 4 lists the expected results and conclusions.

2. Related Work

2.1. Environmental Planning

For the present work, environmental planning is understood as the development of any plan that concerns any changes in some geographic region. It is often a complex and challenging task that can use a wide variety of knowledge gathered from many specialists of different domains. It is also heavily dependent on geographic data. An example of the development of an environmental plan can be drawn in agricultural zoning, which is a scientific process to determine lands suitability, in a geographic region, for a collection of crops [5]. We use an specific example that is the zoning process for determining land suitability for *Coffea arabica* in Paraná state [5]. The model of the overall process is depicted in Figure 1, where tasks embedding mathematical models are inside ovals. Given a set of heterogeneous geographic data, in shaded boxed, the output is a set of maps that show the portions of land suitable for *C. arabica*. This model can be extended and reused – e.g. for other crop in the state.

2.2. Workflow Systems, Scientific Workflows and WOODSS

A workflow is the automation of a process, in a whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules.

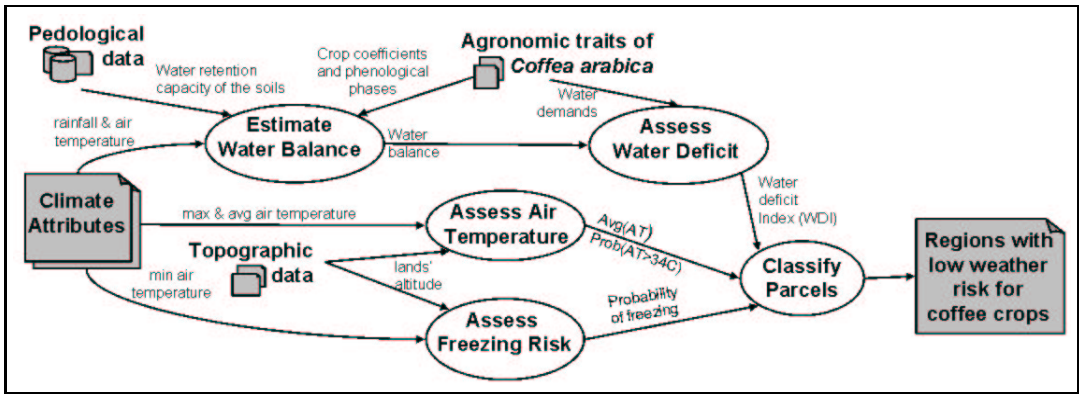


Figure 1: Determining land suitability for *C. arabica* in Brazil's Center-South [5].

From that, a workflow is a model of process which is subject to clearly defined rules. This model can be put in the run state, i.e., it can be instantiated and fed with data in order to generate concrete work results. A workflow management system (WMS) is a system that defines, creates and manages the execution of workflows through the use of software, running on one or more workflow engines.

We are concerned with a specific kind of workflow – a scientific workflow, which can be used to describe scientific experiments and processes. It differs from a usual workflow in having some additional characteristics mainly connected with high degree of flexibility, uncertainty and existence of exceptions. It can be dynamically modified and be defined on the fly. Figure 1 is an example of scientific workflow. Related works in this area include [4, 3], which, besides workflows, also make use of metadata frameworks. Also, an execution oriented project is [10].

Workflow technology has been growing recently and is being broadly used to describe both business and scientific processes. The Workflow Management Coalition (WfMC) [13] is the most important entity in establishing standards to workflow technology. Their reference model [13] is being adopted by most of the workflow product vendors and several researchers. Figure 2 gives a diagram of the workflow reference model from WfMC. It clearly divides responsibilities among its components and, subsequently, their communication interfaces. Interface 4 defines how different workflow enactment services will communicate. This interface is of major importance to our work since one of our goals is the effective representation of data on that interface.

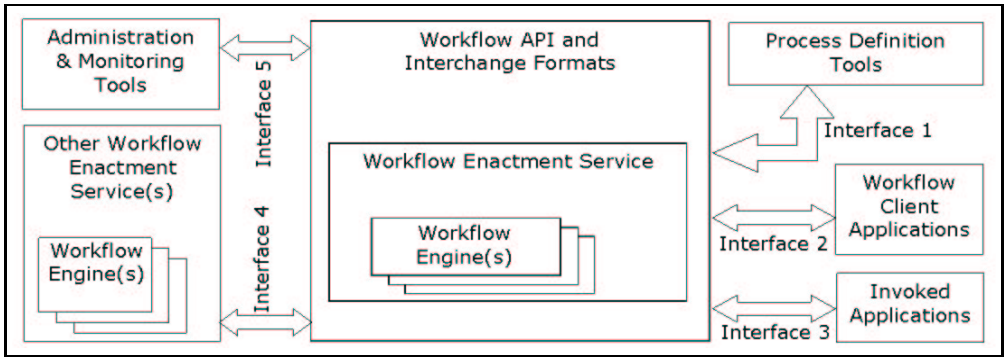


Figure 2: The Workflow Reference Model [13]

WOODSS (WOrkflOw-based spatial Decision Support System) [7, 11] is a software developed at UNICAMP on top of a commercial GIS (Idrisi [9]). It has been tested in several environmental planning efforts. WOODSS is centered on dynamically capturing user (expert planners) interactions with a GIS

in real time, and documenting these interactions by means of scientific workflows. This tool serves three purposes during environmental planning activities: (i) documentation of user procedures, for subsequent reuse and semantics enhancement; (ii) support for decision making; and (iii) construction of a database that describes solutions to planning processes. Our work is mainly concerned with documentation issues. Details on other aspects, including the data model, are covered elsewhere [7, 11].

2.3. Semantic Web and Web Services Efforts and Standards

Although data integration can be an important resource for scientific research, the availability of data itself does not suffice. Some degree of organization is necessary. The quantity of data on the Web that can actually be found and used by those who need them still is a tiny part of the available data that could be put into use. This is the main motivation that drives the emergence of concepts and technologies to describe, publish and use data on the Web – e.g. XML, RDF, ontologies, SOAP, Web services, among many others. These technologies involve standards, architectures, computational systems and try to solve the problem of organizing the integration of data, mainly on the Web. Nevertheless, they are relatively new and some are not yet mature enough to be used by "heavy-duty" applications. Even so, they are evolving fast and some of them are likely to become standards. The Semantic Web standards concentrate most of the efforts in this area.

In this work we consider two main dimensions of efforts: the Semantic Web that is primarily concerned with data representation standards; and Web services [1], which defines a set of implementation oriented standards. Both dimensions organize their standards in a stack of layers. The last layer of the Web services stack, service composition layer, is of interest to us, since it uses workflow structures and their Web representations. It is detailed in the next section.

2.4. Workflow interchange standards

Workflows play a major role in constructing applications across the Web, serving as a basis to coordinate services. Currently, there are two main approaches being used to represent workflows on the Web. The first is to directly use an XML-based specification. The other favors functionality, by proposing means of composing services. Since we will use workflows for documenting distributed scientific processes, we need to consider how to represent them for a distributed execution on the Web.

There are two major proposals of XML-based languages to represent workflows: XPDL (XML Process Definition Language) that is proposed by the Workflow Management Coalition (WfMC) [13], and BPEL4WS (Business Process Execution Language for Web Services) [2]. The first was created explicitly to represent workflows in an accessible language, so that different WMS can use the same process specifications. The latter was introduced to meet the requirements of service composition on the Web by representing service flow coordination, using workflow concepts to do so. It is based on the merge of two other coordination standards, namely IBM's WSFL and Microsoft's XLANG.

3. Methodology and Research Status

The main objective of this research project is to propose a framework for publishing and specifying distributed scientific workflows on the Web. This framework will be composed of a data model to interchange scientific workflows on the Web and a method to translate workflows stored in relational database system to/from this model. It will also provide Web-based tools to edit and compose workflow specifications.

The methodology and research status are as follows. The first step was to perform domain analysis for environmental planning applications and their mapping to distributed workflow execution on the Web. Figure 1 is a typical example of such execution. This was conducted in parallel with a study of behavioral and operational aspects of a workflow management system and workflow data models, especially those for scientific processes. This stage has been finished.

Since the present work follows Semantic Web standards, the language selected to specify the data interchange format must be based on XML. There are two main efforts in this direction, as shown in section 2.4. These efforts are presently being analyzed in order to determine which, if any, is best suited for scientific workflow data exchange. The idea is to decide whether we should adopt XPDL, BPEL4WS or a combination of both. This analysis uses the approach of comparing features of each standard, complemented with a study of patterns, as proposed in [12, 14], making them the building blocks of workflows and checking which language features represents each pattern.

The next step is to design and implement tools to translate this XML standard into database storage and vice-versa. In this phase WOODSS' data will be used to validate the translation method and the completeness of the standard. The idea is to propose a method easily adaptable to other workflow data models.

Finally, we will design and implement Web-based workflow editing and composing tools. One problem, in such context, is to choose architectural aspects of the implementation, including concurrency features to be provided. Once the system is at least partially functional, a Web user interface can be implemented. This requires the choice of interface features and design. As releases of the software are made available, it will be tested on environmental planning activities.

4. Conclusion and Contributions

The main contribution expected from the present research project is the introduction of a framework for scientific workflow specification and composition on the Web. This framework should be general enough to be applied in any WMS that complies with WfMCs Reference Model. Other contributions are the following: analysis of the XML compliant standards for representing workflow data, showing how much of a workflow model can be represented with them; reengineering of the WOODSS, including specific models for Web user interaction.

These contributions can be directly applied on environmental planning activities. More generally, it can also be applied in areas that use scientific processes to model activities.

References

[1] G. Alonso, F. Casati, H. Kuno, and V. Machiraju. *Web Services – Concepts, Architectures and Applications*. Springer-Verlag, 2004.

[2] BPEL4WS. Business Process Execution Language for Web Services Version 1.1. Technical report, BEA, IBM, Microsoft, Siebel, 2003. http://www-106.ibm.com/developerworks/webservices/library/ws-bpel/. (as of 2004-06-15).

[3] M. Cavalcanti, M. Mattoso, and M. Campos. Scientific resources management: Towards an in silico laboratory. Technical Report ES-605/03, COPPE – UFRJ, June 2003.

[4] M. Cavalcanti, M. Mattoso, M. Campos, F. Llirbat, and E. Simon. Sharing Scientific Models in Environmental Applications. In *Proc ACM Symposium Applied Computing - SAC*, pages 453–457, 2002.

[5] R. Fileto, L. Liu, C. Pu, E. D. Assad, and C. B. Medeiros. Poesia: An ontological workflow approach for composing web services in agriculture. *The VLDB Journal*, 12(4):352–367, 2003.

[6] I. Foster, J. Voeckler, M. Wilde, and Y. Zhao. Chimera: A virtual data system for representing, querying and automating data derivation. In *Proceedings of the 14th Conference on Scientific and Statistical Database Management*, 2002. http://www.globus.org/research/papers.html#VDS02. (as of 2004-06-15).

[7] D. Kaster, C. B. Medeiros, and H. Rocha. Supporting Modeling and Problem Solving from Precedent Experiences: The Role of Workflows and Case-Based Reasoning. *Environmental Modeling and Software*, 2004. Accepted for publication.

[8] Kepler: A system for scientific workflows. http://kepler.ecoinformatics.org/, 2004. (as of 2004-06-15).

[9] Clark Labs. Geographic analysis and image processing software. http://www.clarklabs.org/, 2004. (as of 2004-06-15).

[10] Mygrid – open source grid and grid middleware. http://mygrid.sourceforge.net/, 2004. (as of 2004-06-15).

[11] L. Seffino, C. B. Medeiros, J. Rocha, and B. Yi. WOODSS - A Spatial Decision Support System based on Workflows. *Decision Support Systems*, 27(1–2):125–123, 1999.

[12] W. M. P. van der Aalst. Dont go with the flow: Web services composition standards exposed. *IEEE Intelligent Systems*, 18(1):72–76, 2003.

[13] Workflow Management Coalition. The Workflow Management Coalition. http://www.wfmc.org/, 2004. (as of 2004-06-15).

[14] Workflow Patterns. http://tmitwww.tm.tue.nl/research/patterns/, 2004. (as of 2004-06-15).

OMT-G Temporal: Estendendo o Modelo OMT-G para Representação dos Aspectos Temporais dos Dados Geográficos

Giovani Volnei Meinerz[1*] Adilson Marques da Cunha (orientador)[1]

[1]Divisão de Ciência da Computação - IEC - Instituto Tecnológico de Aeronáutica - ITA
Praça Marechal Eduardo Gomes, 50 - Vila das Acácias
CEP 12228-900 - São José dos Campos - SP - Brasil

{giovani, cunha}@comp.ita.br

Nível: Mestrado
Programa de Pós-Graduação em Engenharia Eletrônica e Computação - Área de Informática - PG/EEC-I
Instituto Tecnológico de Aeronáutica - ITA
Ingresso: Agosto de 2003
Previsão de Conclusão: Junho de 2005

Resumo

Em aplicações geográficas, faz-se necessário levar em consideração, além dos fatores relacionados à representação de objetos espaciais, também os não-espaciais, bem como os objetos temporais. A necessidade de representar e gerenciar aspectos temporais em aplicações geográficas vem exigindo considerável esforço na área de geoprocessamento. A complexidade de manipular essas informações temporais vem demandando um esforço ainda maior para a criação de modelos de dados conceituais, que permitam realmente representar os fenômenos naturais e suas variações no tempo. O modelo de dados geográficos, abrangendo as Técnicas de Modelagem de Objetos para Aplicações Geográficas, OMT-G, propõe primitivas e oferece recursos utilizados para a modelagem de aplicações geográficas. Apesar de toda a sua expressividade, este modelo não dispõe de recursos para representação de aspectos temporais dos dados geográficos. O ***OMT-G Temporal*** *proposto neste tema de dissertação de mestrado objetiva estender a Técnica OMT-G, a fim de adicionar suporte à representação de versões, classes e relacionamentos temporais.*

Palavras Chave: Sistemas de Informação Geográficos, Modelos de Dados Geográficos, Bancos de Dados Geográficos, Técnicas de Modelagem Conceitual Temporal de Dados

Abstract

In geographic applications, it is necessary to take into account, besides factors related to the representation of spacial objects, also the non-spacial ones, as well as temporal objects. The needs of representing and managing temporal aspects in geographic applications is requiring considerable effort in the geoprocessing area. The complexity of manipulating those temporal information are still demanding a larger effort for models' creation of conceptual data, that really allow to represent natural phenomena and their on time variations. The model of geographical data, OMT-G, proposes primitives and offers resources used for modelling geographic applications. Despite of all your expressiveness, this model doesn't have resources for geographical data temporal aspect representations. The ***OMT-G Temporal*** *proposed in this master's degree dissertation aims to extend the OMT-G Technique, in order to add support to temporal representation of versions, classes and relationships.*

Key Words: Geographic Information Systems, Geographic Data Models, Geographic Databases, Techniques of Temporal Conceptual Data Modelling

*Trabalho financiado pela FINEP (Projeto FVA CMI-17).

1. Introdução

Uma forte relação entre espaço e tempo existe. Normalmente, informações referenciadas para espaço também são referenciadas para tempo [4]. Considerando os Sistemas de Informação Geográficos (SIGs), o aspecto temporal caracteriza um componente imprescindível para análise, previsão e conhecimento dos fenômenos geográficos. Um SIG pode ser definido sob diferentes visões. Segundo [4], dentre as principais características desse domínio de sistemas, cita-se a integração de informações espaciais numa única base de dados, e a oferta de mecanismos para combinação dessas informações através de algoritmos de manipulação e análise.

Os SIGs atualmente disponíveis consideram as entidades geográficas como se o mundo existisse no presente. Informações geográficas são incluídas e alteradas ao longo do tempo, mas o histórico dessas informações não permanece na base de dados, conforme [13]. Sendo assim, as aplicações de SIG deveriam oferecer condições para lidar com os dados espaço-temporais refletindo a natureza dos fenômenos geográficos. A erosão gradual dos solos, a formação de tempestades, o estudo da poluição ambiental, entre outros, caracterizam alguns exemplos.

A representação da evolução dos fenômenos geográficos, durante a modelagem conceitual de dados, implica no uso de diferentes técnicas. Formalizar a percepção do tempo através de modelos conceituais de dados, que permitam aos projetistas e usuários utilizarem a mesma linguagem, constitui-se num dos importantes desafios para a construção de SIGs na atualidade.

A extensão da Técnica de Modelagem de Objetos (*Object Modeling Technique - OMT*), denominada Geo-OMT, fornecendo primitivas para modelar a geometria e a topologia de dados geográficos, conforme [2], foi aprimorada para possibilitar a modelagem de restrições de integridade espaciais, passando a ser denominada OMT-G. Apesar de ser largamente utilizada por projetistas e usuários para a modelagem conceitual de SIGs, a OMT-G não oferece suporte para a modelagem dos aspectos temporais de dados geográficos. Desta forma, a temática de dissertação apresentada neste trabalho, tem por objetivo, estender a técnica de modelagem de dados geográficos OMT-G e permitir que esta suporte os modelos conceituais de aspectos temporais para aplicações geográficas. A seguir a Seção 2 apresenta uma abordagem dos principais trabalhos e iniciativas correlatas, e a Seção 3 considerações sobre a extensão proposta, o seu estado atual de pesquisa, bem como a metodologia utilizada. Finalmente, a Seção 4 apresenta os resultados esperados, a relevância e a aplicabilidade das contribuições.

2. Modelos Conceituais de Banco de Dados Geográficos

Nas fases iniciais de Projetos de Banco de Dados, Especificação de Requisitos, Análise e Projeto Conceitual, utiliza-se os Modelos de Dados Conceituais [9] independentes de Hardware, de Sistema Operacional ou de Sistema Gerenciador de Banco de Dados (SGBD).

2.1. A Técnica OMT-G

A Técnica OMT-G propõe a utilização de uma série de primitivas permitindo construir o esquema estático de aplicações geográficas, no qual são especificadas as classes envolvidas no problema, juntamente com as suas representações básicas e relacionamentos. A partir desse esquema estático, torna-se possível produzir um conjunto de restrições de integridade espacial que precisam ser implementadas pela aplicação ou pelo banco de dados geográfico utilizado [2].

Essa técnica parte das primitivas para o diagrama de classes da Linguagem de Modelagem Unificada *(Unified Modeling Language - UML)* [7], introduzindo primitivas geográficas com o objetivo de aumentar a capacidade de representação semântica do modelo, e portanto reduzir a distância entre o modelo mental do

espaço a ser modelado e o modelo de representação usual. A Técnica OMT-G baseia-se em três conceitos principais: *classes*, *relacionamentos* e *restrições de integridade espaciais*. *Classes* e *relacionamentos* definem as primitivas básicas usadas para criar esquemas estáticos de aplicação. A identificação de *restrições de integridade espacial* é uma atividade importante no projeto de uma aplicação, e consiste na identificação de condições que precisam ser garantidas para que o banco de dados permaneça íntegro. Essas *restrições de integridade espaciais* para a Técnica OMT-G foram detalhadas em [2]. As primitivas de *classes* e *relacionamentos* encontram-se a seguir.

2.1.1. Classes

As classes definidas pela Técnica OMT-G podem ser *geo-referenciadas* ou *convencionais*. Uma *classe geo-referenciada* descreve um conjunto de objetos com representação espacial associado a elementos do mundo real [4], assumindo a visão de campos e objetos proposta por Goodchild [5]. Uma *classe convencional* descreve um conjunto de objetos com propriedades, comportamentos, relacionamentos e semânticas semelhantes, podendo relacionar-se com objetos espaciais, mas não possui propriedades geográficas [2].

As classes geo-referenciadas especializam-se em *geo-campos* e *geo-objetos*. *Geo-campos* representam os objetos e fenômenos distribuídos continuamente no espaço, correspondendo a variáveis como tipo de solo, relevo e geologia [4]. *Geo-objetos* representam os objetos geográficos, particulares, individualizáveis associados a elementos do mundo real, como edificações, rios e árvores. As classes convencionais utilizam símbolos como na UML. As classes geo-referenciadas são simbolizadas como na Técnica de Modelagem OMT-G, incluindo no canto superior esquerdo, um retângulo utilizado para indicar a geometria da representação.

A Técnica OMT-G apresenta um conjunto fixo de alternativas de representação geométrica, usando uma simbologia que distingue geo-objetos e geo-campos. Ela é utilizada para definir cinco classes descendentes de geo-campo (*isolinhas*, *polígonos adjacentes*, *tesselação*, *amostragem* e *rede triangular irregular*) e duas classes descendentes de geo-objeto (*geo-objeto com geometria* e *geo-objeto com geometria e topologia*), sendo que da primeira descendem as classes *ponto*, *linha e polígono*, e da segunda descendem as classes *nó de rede*, *arco unidirecional* e *arco bidirecional*. Maiores detalhes encontram-se em [2].

2.1.2. Relacionamentos

Considerando a importância dos relacionamentos espaciais e não-espaciais para a compreensão do espaço modelado, a Técnica OMT-G apresenta três relacionamentos que podem ocorrer entre suas classes: *associações simples*, *relacionamentos topológicos em rede* e *relacionamentos espaciais*. *Associações simples* representam relacionamentos estruturais entre objetos de classes diferentes, convencionais ou geo-referenciadas. *Relacionamentos espaciais* representam relações topológicas, métricas, ordinais e *fuzzy*. *Relacionamentos de rede* ocorrem entre objetos conectados uns com os outros. Informações mais detalhadas encontram-se em [2].

2.2. Requisitos de um Modelo Conceitual Temporal de SIG

Segundo [6], a necessidade de dados geográficos qualificados com base no tempo, não se deve ao fato deles serem freqüentemente modificados, mas sim a necessidade de se registrar estados passados, de forma a possibilitar o estudo da evolução dos fenômenos geográficos. Para tanto, faz-se necessário adicionar aos SIGs as potencialidades dos sistemas de banco de dados temporais. Isso refere-se ao aspecto temporal no mundo real capturado num banco de dados. A modelagem conceitual dos dados deve ser capaz de capturar este aspecto para posterior implementação. Existem alguns trabalhos relacionados que definiram

os principais requisitos abaixo relacionados para a modelagem conceitual temporal de SIGs, dentre eles, [1], e mais especificamente [11]:

- Dimensão Temporal - temporalidade em Atributos, Objetos e Relacionamentos;
- Restrições de Integridade - possuem regras com validade apenas num determinado período;
- Temporalidade na Forma e Posição - registro das trocas quanto a sua forma e posição;
- Tempo Discreto - forma de tratamento que facilita o processo de implementação no SGBD;
- Tempo de Validade - considera o tempo em que a informação existe no mundo real;
- Instantes e Intervalo de Tempo - representa um ponto no tempo e o intervalo ocorrido;
- Granularidade do Tempo - permite maior flexibilidade na representação da realidade;
- Tempo Futuro - permite a representação de informações válidas no futuro;
- Tempo Ramificado - permite que se tenha múltiplos tempos futuros possíveis;
- Tempo Circular - representa a existência de fenômenos com ocorrência cíclica;
- Tempo Impreciso - caracteriza a informação que não se sabe exatamente quando ocorreu;
- Tempo Relativo - relação de tempo existente entre a ocorrência de um fenômeno e outro;
- Coexistência de Dados - dados com e sem informação temporal;
- Duração das Trocas de Estados - registra tempo de troca de entidades entre dois estados; e
- Restrições Temporais - define restrições de tempo de transação e tempo de validade.

2.3. Trabalhos Similares

Inúmeros modelos conceituais de dados tentam oferecer suporte à modelagem dos aspectos temporais. Conforme consta em [2], a seguir será apresentada uma comparação entre alguns modelos conceituais de dados orientados a objeto que contemplam o aspecto temporal da informação, apresentando um paralelo entre a diversidade dos modelos estudados:

- GeoOOA - conforme mostrado em [8];
- OO-TGIS - conforme mostrado em [12];
- Perceptory - conforme mostrado em [3];
- MADS - conforme mostrado em [10]; e
- GeoFrame-T - conforme mostrado em [13].

A Tabela 1 apresenta uma comparação entre os modelos conceituais de dados temporais orientados a objeto, em relação aos requisitos necessários para um modelo conceitual temporal de SIG apresentados na Seção 2.2 da página 3.

Levando-se em consideração os requisitos apresentados na Seção 2.2, a Tabela 1 mostra, em suas colunas, os modelos estudados, e em suas linhas, os requisitos temporais de SIG.

3. Considerações Sobre Inclusão Temporal na OMT-G

A proposta de extensão temporal da Técnica de Modelagem de Dados OMT-G, denominada *OMT-G Temporal* deverá ter a capacidade de representar os seguintes tipos de objetos:

- Convencionais - objetos que não possuem aspectos espaciais nem temporais;
- Espaciais - que capturam a forma e a posição geo-referenciada do objeto;
- Temporais - mantém a história do dado, ao longo do tempo; e
- Espaço-Temporais - dados espaciais cuja geometria altera-se com o passar do tempo.

O aspecto temporal a ser inserido na *OMT-G Temporal* deverá incluir o tempo em nível de atributos, objetos e relacionamentos, uma vez que essas três dimensões, conforme [13], expressam níveis diferentes de representação da realidade modelada e o tempo se manifesta em cada um deles. Para facilitar o processo de modelagem deverá ser definido um conjunto de estereótipos servindo para indicar explicitamente, nos elementos do modelo, qual o tipo de tempo utilizado e aplicado pelo mesmo.

Tabela 1: Comparação de Modelos Conceituais Temporais de SIG[13]

Requisitos	***GeoOOA***	***OO-TGIS***	***Perceptory***	***MADS***	***GeoFrame-T***
Dimensão Temporal	Sim	Sim	Sim	Sim	Sim
Restrições de Integridade	Sim	Sim	Sim	Sim	Não
Tempo na Forma e Posição	Sim	Sim	Sim	Não	Sim
Tempo Discreto	Sim	Sim	Sim	Sim	Sim
Tempo de Validade	Sim	Sim	Sim	Sim	Sim
Instantes e Intervalo de Tempo	Não	Não	Sim	Não	Sim
Granularidade do Tempo	Sim	Sim	Sim	Sim	Sim
Tempo Futuro	Não	Não	Não	Não	Não
Tempo Ramificado	Não	Não	Não	Não	Sim
Tempo Circular	Não	Não	Não	Não	Não
Tempo Impreciso	Não	Não	Não	Não	Não
Tempo Relativo	Não	Não	Não	Não	Não
Coexistência de Dados	Sim	Sim	Sim	Sim	Sim
Duração das Trocas de Estados	Não	Não	Não	Não	Não
Restrições Temporais	Sim	Sim	Sim	Sim	Sim

Tabela 2: Metodologia de Desenvolvimento da Pesquisa

Fases	***Descrição das Atividades***	***Início***	***Término***
I	Resumo dos principais conceitos atuais sobre a importância da representação de aspectos temporais em SIG	Ago/2003	Nov/2003
II	Identificar características temporais ideais que um modelo conceitual para SIG deve suportar	Dez/2003	Mar/2004
III	Estudo comparativo entre alguns dos modelos conceituais temporais para SIG existentes	Abr/2004	Jun/2004
IV	Investigar e analisar aspectos relevantes quanto ao tipo de solução a ser adotada para estender a OMT-G	Jul/2004	Dez/2004
V	Utilizar o *OMT-G Temporal* na modelagem de estudos de caso, como indicador de avaliação dos resultados	Jan/2005	Jun/2005

A metodologia utilizada para alcançar os objetivos propostos por este tema de dissertação de mestrado, conforme planejamento detalhado, é apresentada na Tabela 2.

A Tabela 2 descreve as atividades a serem realizadas em cada fase, bem como o respectivo tempo de início e término de cada uma. O estado atual da pesquisa encontra-se na Fase IV.

4. Conclusão

Como em qualquer modelagem conceitual de dados para o desenvolvimento de aplicações, sejam elas convencionais ou geográficas, faz-se necessário que modelos conceituais de dados ofereçam condições de representar graficamente, e da melhor maneira possível, o mundo real. Aplicações de SIG envolvem tempo e espaço, logo, um grande número de informações deve ser modelado, tornando as modelagens mais complexas.

Existem vários requisitos que um modelo conceitual temporal de SIG deve atender. A temática de

dissertação apresentada neste trabalho almeja, inicialmente, rever os principais conceitos necessários para a inclusão do aspecto temporal na Técnica de Modelagem OMT-G. Para isso está sendo desenvolvida uma extensão temporal dessa técnica, como uma nova técnica para modelagem conceitual temporal de dados geográficos, a *OMT-G Temporal*, que visa contemplar a maioria dos requisitos dimensionais e espaço-temporais abordados na Seção 2.2 Requisitos de um Modelo Conceitual Temporal de SIG, adicionando suporte de representação de versões, classes e relacionamentos temporais à OMT-G.

A utilização de estereótipos (símbolos) vem demonstrando ser um caminho interessante para introdução visual dessas dimensões, permitindo uma representação simbólica mais rica. Essa solução vem sendo adotada por propiciar maior simplicidade e melhor visualização, permitindo identificações mais apropriadas de objetos do mundo real. Uma das principais contribuições vislumbradas neste trabalho é a de possibilitar a identificação dos elementos da realidade geográfica que devem ser capazes de representar a sua temporalidade, definindo o tempo, e do que ele é formado.

Referências

[1] T. Abraham and J. F. Roddick. Survey of spatio-temporal databases. In *GeoInformatica*, volume 3, pages 61–69, Hingham, MA, USA, March 1999.

[2] Karla A. V. Borges, Alberto H. F. Laender, and Clodoveu A. Davis Jr. Omt-g: An object modeling technique for geographic applications. *GeoInformatica*, 2000.

[3] C. Caron and Y. Bédard. Extending the individual formalism for a more complete modeling of urban spatially referenced data. In *Computer, Environment and Urban Systems*, volume 17, pages 337–346. An International Journal, 1993.

[4] G. Câmara, Antonio M. Monteiro, and Clodoveu A. Davis Jr. Geoprocessamento: Teoria e aplicações. In *Bancos de Dados Geográficos*, volume III, chapter 5. 2001.

[5] M. F. Goodchild. Geographical data modeling. pages 401–408. Computers Geosciences, 1992.

[6] T. Hadzilacos and N. Tryfona. Logical data modelling for geographical applications. In *International Journal of Geographical Information Science*, volume 10, pages 179–203, London, 1996.

[7] Object Management Group Inc. The unified modeling language. Complete Specification, Version 1.5, 2003. Site: http://www.omg.org/technology/documents/formal/uml.htm. Last Access: Jun 2004.

[8] G. Kosters, B. U. Pagel, and H. W. Six. Gis-application development with geoooa. In *International Journal of Geographical Information Science*, volume 11, pages 307–335, London, June 1997. Taylor & Francis.

[9] S. B. Navathe. Evolution of data modeling for databases. In *Special issue on analysis and modeling in software development*, volume 35, pages 112–123, New York, NY, USA, September 1992. ACM Press.

[10] C. Parent, S. Spaccapietra, E. Zimanyi, P. Donini, C. Plazanet, and C. Vangenot. Modeling spatial data in the mads conceptual model. In *International Symposium on Spatial Data Handling*, Vancouver, Canada, July 11-15 1998. SDH 98. Site: http://lbdwww.epfl.ch/e/publications/articles.pdf/SDH98.pdf. Last Access: Jun 2004.

[11] D. Pfoser and N. Tryfona. Requirements, definitions, and notations for spatio-temporal application environments. In *ACM-GIS*, pages 124–130, 1998. Site: http://citeseer.ist.psu.edu/article/pfoser98requirements.html. Last Access: Jun 2004.

[12] A. Renolen. Conceptual modelling and spatiotemporal information systems: How to model the real world. In *Scandinavian Research Conference on Geographical Information Systems*, Stockholm, 1997. Site: http://citeseer.ist.psu.edu/renolen97conceptual.html. Last Access: Jun 2004.

[13] Luciana V. Rocha. Geoframe-t: um *framework conceitual temporal para aplicações de sistemas de informação geográfica. Master's thesis, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2001. Site: http://www.inf.ufrgs.br/ nina/Dissertacao/LucianaRocha.pdf. Last Access: Jun 2004.*

Extensão da arquitetura de banco de dados relacionais para suportar recuperação de imagens por conteúdo

Márcio dos Reis Caetano[1] , Denise Guliato[1] (orientadora)
[1] Laboratório de Computação Científica – Universidade Federal de Uberlândia
Av. João Naves de Ávila, 2160, sala 1B-61 – 38400-902, Uberlândia, MG
caetano@lcc.ufu.br, guliato@ufu.br

Nível : Mestrado
Programa de Mestrado em Ciência da Computação
Universidade Federal de Uberlândia
Ingresso: Março de 2003
Previsão de Conclusão: Dezembro de 2005

Resumo

Para suportar recuperação de imagens baseada em conteúdo, descritores das imagens precisam ser obtidos e combinados para formarem vetores-característica que as caracterizem. Os SGBDRs atuais armazenam imagens como um tipo especial de dados mas não apresentam flexibilidade para incluir novos descritores de imagens, para criar novos vetores-característica ou para realizar consultas por similaridade. Este trabalho propõe o desenvolvimento de uma extensão para a arquitetura do SGBD PostgreSQL e de um conjunto de comandos, SQL/IRPK, que estende os recursos da linguagem SQL, para suportar o desenvolvimento de sistemas de recuperação de imagens baseada em conteúdo.

Abstract

To support queries content-based in image databases, descriptors of all images must be assessed, yielding feature vectors to characterize them. The current relational database management systems (RDBMS) are able to store image as a special data type but do not present flexibility to include new image descriptors, to create new feature vectors and the retrieval technique is based on the exact match. To have an application-independent RDBMS to support CBIR and make use of the powerful tools available on the traditional relational database system, this paper proposes an extension for the architecture of RDBMS and for SQL (SQL/IRPK).

Palavras chave: recuperação de imagens por conteúdo, extensão de bancos de dados relacionais e consultas baseadas em similaridade.

1. Introdução

O conteúdo de uma imagem pode ser representado por um ponto num espaço de características multidimensionais. Este ponto é o *vetor-característica*, cujos elementos descrevem a imagem. Cada elemento do vetor-característica é denominado *descritor* e é obtido pela execução de uma função chamada *extrator de característica.* Sendo assim, o vetor-característica é formado por uma combinação de descritores, de acordo com a pesquisa a ser realizada.

A execução de consultas baseadas em conteúdo em um banco de dados de imagens exige a especificação de quais descritores serão utilizados na composição do vetor-característica. A consulta é realizada comparando o vetor-característica da imagem de referência com o vetor-característica de cada imagem armazenada no banco de dados. Os sistemas gerenciadores de bancos de dados (SGBD) atuais são capazes de armazenar imagens, mas, não oferecem flexibilidade para executar consultas baseadas em conteúdo e disponibilizam apenas técnicas de recuperação da informação baseadas em "matching" exato [1] e [7].

Para tornar disponível um SGBDR independente de aplicação que suporte recuperação de imagem por conteúdo e faça uso dos recursos já existentes nos SGBDRs tradicionais, este trabalho propõe especificar e implementar uma extensão para a arquitetura de um SGBDR e para a linguagem SQL (SQL/IRPK). Esta extensão está sendo projetada para ser flexível, portável, de fácil configuração e utilização. Estamos desenvolvendo a extensão proposta para o SGBD PostgreSQL e aplicando esta solução para construir o Atlas Indexado de Mamografias Digitais (AMDI) [3].

2. Proposta de trabalho

Para associar facilidade e flexibilidade às consultas por conteúdo, propomos uma extensão para a arquitetura do SGBD PostgreSQL. Esta extensão é composta pelos recursos já existentes no SGBD PostgreSQL associados a um pacote de comandos, denominado SQL/IRPK, que estende a linguagem SQL e a um catálogo estendido onde são mantidas informações para gerenciar os novos recursos acrescentados ao sistema, conforme Figura 1.

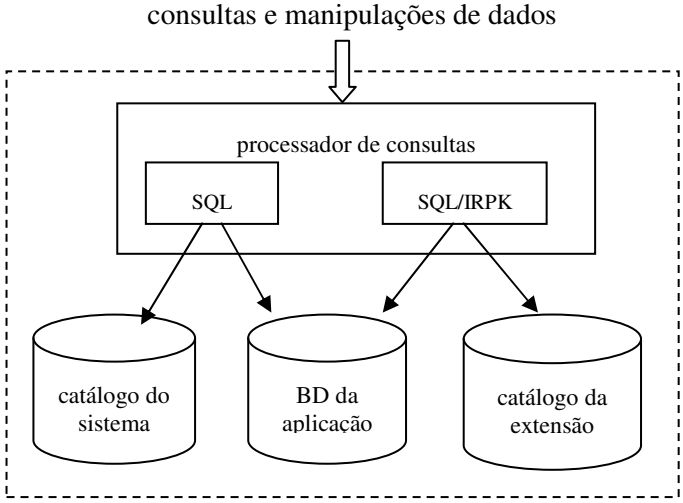


Figura 1: Arquitetura estendida de um SGBD para suportar recuperação de imagens por conteúdo

A SQL/IRPK é um conjunto de comandos que está sendo desenvolvido para permitir a inclusão de novos métodos para extrair características de imagens e de novos operadores de similaridade, assim como, a definição de novos vetores-característica para indexar a consulta por similaridade.

Os comandos da SQL/IRPK estão sendo desenvolvidos através da implementação de funções utilizando recursos das linguagens SQL , C ANSI e PL/PGSQL que já estão disponíveis no SGBD PostgreSQL. Os comandos que compõem a SQL/IRPK são de uso geral, independentes de aplicação e estão listados abaixo:

a) Create_Extractor(<N_Extractor>, <R_Data_Type>, <F_Path>, <P_List>): este comando permite incorporar ao SGBD PostgreSQL um novo extrator de características da imagem. Para tanto, devem ser fornecidos o nome da função <N_Extrator>, o tipo de dado que a função retorna <R_Data_Type>, o caminho da função no sistema <P_Path> e a lista de parâmetros para executar a função <P_List>. Uma vez incorporado ao SGBD, o novo extrator é armazenado no catálogo estendido e poderá ser utilizado por qualquer aplicação.

b) Create_Feature_Vector(<F_V_Name>, <E_List>, <I_Method>): este comando cria um vetor-característica no SGBD PostgreSQL que será representado por uma tabela. Para tanto, devem ser fornecidos o nome do vetor-característica <F_V_Name>, a lista dos extratores que formarão o vetor-característica <E_List> e o método de indexação que será utilizado para as consultas a serem realizadas neste vetor-característica.. O catálogo estendido é atualizado para suportar o novo vetor-característica.

c) Create_Operator(<O_Name>, (<P_List>, <O_Path>)): este comando permite incorporar um novo operador de similaridade no SGBD PostgreSQL. Para tanto, devem ser fornecidos o nome da função que implementa o operador <O_Name>, a lista de parâmetros necessários para a execução do operador <P_List> e o caminho da função no sistema <O_Path>. O novo operador será armazenado no catálogo estendido.

d) Create_Index(<I_Name>, <F_V_Name>, <I_Method>): este comando permite incorporar ao SGBD PostgreSQL um novo índice que será associado a um vetor-característica. Devem ser fornecidos o nome do índice <I_Name>, o nome do vetor-característica <F_V_Name> associado ao índice e o método de indexação multidimensional a ser utilizado <I_Method>.

e) Drop_Feature_Vector(<F_V_Name>): este comando exclui o vetor-característica F_V_Name do catálogo estendido.

f) Remove_Extractor(<N_Extractor>): este comando exclui o extrator N_Extractor do catálogo estendido.

g) Create_Linguist_Variable(<L_Name>, <(L_Term, F_Number) List>): este comando cria uma variável linguística <L_Name> associada a uma lista de termos lingüísticos <L_Term>, onde cada termo linguístico é associado a um número fuzzy <F_Number>. Um número fuzzy é definido por uma função trapezoidal representada pelo 4-tupla $(d_i-\delta, d_i, d_j, d_j+\delta)$, onde $\mu(d_i) = \mu(d_j) = 1$ and $\mu(d_i-\delta) = \mu(d_j+\delta) = 0$, [4]. Este comando incorpora ao SGBD PostgreSQL a possibilidade de se desenvolver operadores de similaridade baseados em variáveis lingüísticas. A inserção de novas variáveis lingüísticas é armazenada no catálogo estendido.

3. Metodologia e estado atual do trabalho

Inicialmente, pesquisamos trabalhos publicados no meio acadêmico que estavam relacionados à área de recuperação de imagens baseada em conteúdo. Após selecionarmos os artigos pertinentes à área, fizemos um estudo sobre os mesmos e identificamos suas relevâncias. Após este estudo, fizemos a especificação de nossa proposta de trabalho, detalhando a extensão da arquitetura do SGBD PostgreSQL por nós proposta. Definimos os comandos que serão disponibilizados na arquitetura proposta e suas respectivas sintaxes. Neste ponto, foi necessário avaliar, também, qual linguagem de programação seria utilizada na implementação. Decidimos, assim, utilizar uma combinação das linguagens SQL, C ANSI e pl/pgsql.

Atualmente, o nosso trabalho encontra-se na fase de implementação dos comandos especificados na SQL/IRPK. Após o término desta fase, iniciaremos os devidos testes de funcionalidade e performance de cada um destes comandos. É interessante ressaltar que a extensão da arquitetura do SGBD PostgreSQL não depende de uma aplicação específica e, sendo assim, os testes poderão ser realizados tanto através do *prompt* de comando do SGBD PostgreSQL quanto através de uma interface amigável a ser desenvolvida para o usuário final. Pretendemos aplicar o SGBD estendido ao AMDI, Atlas Indexado de Mamografias Digitais, [3], para tratar de incertezas nas consultas realizadas em bancos de dados mamográficos.

4. Trabalhos relacionados

Diversos trabalhos têm sido propostos oferecendo consultas por similaridade e recuperação baseada em conteúdo. No entanto, a maioria deles utiliza métodos específicos para uma determinada aplicação e não permitem que se sejam adicionados novos recursos ao SGBD em questão.

O trabalho apresentado em [6] propõe a linguagem SQL/MM, uma extensão da linguagem SQL, aplicada a consultas de dados multimídia e com recursos de mineração de dados. O trabalho apresentado em [8] propõe uma extensão da linguagem SQL, a WebSSQL, que permite consultas baseadas nas estruturas de links e recuperação de informações baseadas em similaridade para documentos multimídia da web. O trabalho apresentado em [5] propõem uma linguagem de consulta espacial chamada SQL/SDA (Spatial Data Analysis) para suportar a expressão de consultas espaciais complexas para pacotes GIS (Geographical Information System). O trabalho apresentado em [2] propõe uma extensão do SGBD Microsoft SQL Server, no entanto, esta extensão não é portável nem apresenta flexibilidade para a inserção de novos métodos.

5. Conclusão

O trabalho apresentado propõe a extensão da arquitetura do SGBD PostgreSQL, independente da aplicação, para suportar recuperação de imagens por conteúdo. Propõe também um pacote de comandos, SQL/IRPK, facilmente incorporado ao PostgreSQL, que facilita a inserção de novos extratores de características e de novos operadores de similaridade e ainda a criação de vetores-característica. Uma vez inseridos tais recursos no sistema, estes estarão disponíveis para o desenvolvimento de quaisquer aplicações, beneficiando assim a comunidade científica que trabalha com recuperação de imagens baseada em conteúdo e consultas por similaridade. Para os testes de desempenho nós aplicaremos a extensão sendo proposta ao projeto AMDI - Atlas Indexado de Mamografias Digitais proposto em [3].

Referências

1. Adler, D.W. IBM DB2 Spatial Extender - Spatial data within the RDBMS, Proceedings of the 27th VLDB Conference, Roma, Italy. Pp. 4p., 2001.
2. Araújo, M.R.B., C. Traina Jr., and A. Traina. Extending Relacional Database to support Content-based Retrieval of Medical Images, IEEE International Conference on Computer Based Medical Systems – CBMS, Maribor, Slovenia, 2002. Pp 303-308.
3. Guliato, D., Caetano, M. et. al. AMDI: An Indexed Atlas of Digital Mammograms Available via the Web. To appear in Proceedings of the 7th International Workshop on Digital Mammography. Durham, NC, 2004.
4. Klir, G. J. and Yuan, B. Fuzzy Sets and Fuzzy Logic: Theory and Applications. Prentice Hall, PTR, Upper Saddle River, NJ, 1995.
5. Lin, H. and Huang, B. SQL/SDA: A Query Language for Supporting Spatial Data Analysis and Its Web-Based Implementation. IEEE Trans. on Knowledge and Data Engineering, vol. 13, pp. 671-682, 2001.

6. Melton, J. and Eisenberg, A. SQL Multimedia and Application Packages (SQL/MM), SIGMOD Record, vol. 30, pp. 97-102, 2001.
7. Stonebraker, M., Rowel, L.A. and Hirohama, M. The implementation of POSTGRES, IEEE Transactions on Knowledge and Data Engineering, 1990. 2(1):125-42.
8. Zhang, C. Meng, W., Wu, Z., Zhang, Z. WebSSQL - A Query Language for MultimediaWeb Documents, IEEE Advances in Digital Libraries, 2000 - ADL2000, Washington, D.C.

Pré-seleção e pré-carga de dados para *Cache* em Bancos de Dados Móveis

Mariano Cravo Teixeira Neto[1] , Ana Carolina Salgado[1] (orientadora)
Sérgio Lifschitz[2] (co-orientador)

[1]Centro de Informárica - Universidade Federal de Pernambuco (UFPE)
Av. Professor Luis Freire, s/n, Cidade Universitária - 50740-540, Recife, PE

[2]Departamento de Informática - Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio)
Rua Marquês de São Vicente, 225 RDC, Gávea - 22453-900, Rio de Janeiro, RJ

[mctn,acs]@cin.ufpe.br, sergio@inf.puc-rio.br

Nível: Mestrado
Programa de Mestrado em Ciência da Computação
Universidade Federal de Pernambuco
Ingresso: Março de 2003
Previsão de conclusão: Março de 2005

Resumo

Um dos principais objetivos do estudo de acesso a dados móveis é atender aos requisitos de ubiqüidade inerente aos sistemas móveis: ter acesso à informação independentemente de local e hora. Devido a restrições de sistemas móveis como, por exemplo, memória limitada e largura de banda estreita, é natural que se pesquisem métodos para amenizar esses problemas. Este trabalho abordará questões relativas ao gerenciamento de cache *em bancos de dados móveis, com ênfase em técnicas para reduzir falhas à consulta de dados enquanto o dispositivo móvel esteja conectado, com pouca largura de banda ou fora de uma rede de dados. Dessa forma, espera-se reduzir o consumo de banda e aumentar, durante uma desconexão, a disponibilidade das informações armazenadas. Com esse objetivo, este trabalho propõe selecionar* a priori *o conteúdo da* cache *(pré-seleção de dados), através da mineração do histórico de acesso a dados do usuário do cliente móvel.*

Palavras-chave: banco de dados móveis, gerenciamento de *cache*, mineração de dados, operações desconectadas.

Abstract

One of the main goals of mobile data access study is to fulfill the ubiquity requirements inherent of mobile systems: to access information independently of time and location. Due to mobile systems constraints such as limited memory and bandwidth, it's only natural that researchers work on this area to soothe its constraints problems. This work will approach issues regarding the cache management of mobile databases, with emphasis on technics to reduce data access faults while disconnected to a network or connected with low bandwidth. Thus, we expect to reduce network usage and increase data availability. In order to achieve such goals, this work proposes to fetch, a priori, *cache content data by mining the trace data access history of mobile client user.*

Key Words: mobile databases, cache management, data mining, disconnected operations.

1. Introdução

A possibilidade de um indivíduo ter acesso à informação em qualquer lugar que ele esteja é uma vantagem adquirida através do uso de dispositivos móveis. Dispositivos, sejam eles telefones celulares, computadores de mão (*handheld computers*), ou até mesmo *notebooks*, estão cada vez mais acessíveis dado a avanços tecnológicos de *hardware* e de rede [Bar99]. Essa crescente necessidade é a motivação de pesquisas e trabalhos no desenvolvimento de mecanismos de acesso a dados móveis. À medida que usuários tornam-se mais dependentes deste tipo de acesso à informação, os repositórios destes dados deverão ser capazes de atender a esta nova demanda.

Tendo em vista o amadurecimento do uso de dispositivos móveis, os dados disponíveis em sistemas de arquivos, bancos de dados, e qualquer outro tipo de repositório de dados deverão oferecer uma maneira de serem acessados a partir de programas que estejam sendo executados em dispositivos móveis.

Apesar do avanço tecnológico dos dispositivos móveis, estes ainda têm como características restritivas, entre outras, tempo de bateria, tamanho de memória, capacidade de armazenamento e conectividade. A um determinado custo e tecnologia, dispositivos móveis sempre serão mais limitados se comparados a computadores de mesa [Sat96]. Conectividade de rede destes dispositivos, especialmente redes sem-fio que cobrem áreas grandes, tende a variar bastante sua largura de banda, latência, confiabilidade e custo em relação às redes físicas. A energia limitada por baterias requer que atividades do cliente móvel sejam negadas, evitadas ou diminuídas para minimizar o gasto de energia e prolongar o tempo de uso do dispositivo. Todas essas restrições são inerentes a dispositivos móveis, o que implica que mesmo avanços no *hardware* (p. ex., mais memória, aumento da duração de uma carga de bateria) não devem eliminá-las.

Um dos problemas que ainda está em aberto no campo de pesquisa em banco de dados móveis é a minimização de falhas no momento em que o dispositivo se encontra desconectado [BBHB$^+$04]. Este trabalho vem propor um mecanismo de gerência de *cache* em que seus dados são pré-selecionados através da mineração do histórico de uso dos dados do lado do cliente. Com a mineração desses dados, serão geradas regras de associação [AI93] para predizer itens de dados que têm mais probabilidade de serem usados em um futuro próximo pelo usuário do cliente móvel.

2. Definição do Problema

Um dos aspectos trabalhados em acesso a dados móveis é relativo a operações desconectadas. Quanto aos tipos de desconexões, são involuntárias aquelas que ocorrem em um ambiente de computação móvel quando há um impedimento temporário de comunicação. Esse tipo de desconexão pode ocorrer quando houver limitações de ambiente como, por exemplo, a inabilidade de o dispositivo móvel operar dentro de um túnel, no subterrâneo, ou estar fora do alcance da rede móvel. Uma desconexão é considerada voluntária quando o usuário deliberadamente não se conecta à rede. Isso pode acontecer para evitar custos de acesso à rede ou consumo de bateria, por exemplo.

O problema a ser abordado nesta pesquisa é a minimização de falhas em operações desconectadas. A natureza dessas falhas, neste trabalho, é relativa ao acesso a dados a partir de um dispositivo móvel com acesso a um banco de dados.

Em virtude da natureza dos dados (dados estruturados, dados semi-estruturados, arquivos), várias técnicas foram e estão sendo pesquisadas para tornar viáveis operações desconectadas: [SC04], [SUE00], [YD01], [BMR04], [PJFY04], [ZHH03], [Kue97] e [Kis93]. Essas técnicas têm o objetivo comum de escolher quais dados o usuário provavelmente usará e, sendo assim, disponibilizar estes dados em uma *cache* local no cliente móvel antes que ocorra uma desconexão. Duas técnicas utilizadas são:

- **Pré-carga** de dados (*pre-fetching*): processo de transferência de dados da máquina servi-

dora para o cliente móvel de forma transparente ao usuário.

- **Pré-seleção** de dados (*hoarding*): processo de seleção de dados baseado na análise do acesso aos dados pelo usuário no cliente móvel. A partir dessa análise, é possível realizar pré-carga com dados provavelmente relevantes para o usuário.

O problema também envolve a escolha da granularidade em que estes dados deverão ser avaliados para preencher a *cache*, que no caso de um banco de dados é mais complicado que no caso de arquivos, pois pode ser relativo a atributos [CSL98], páginas [CFZ94], tuplas [Fra96], região semântica [DFJ+96], entre outros [YD01].

3. Trabalhos Relacionados

O Coda [SKS87] é um sistema de arquivos distribuídos, desenvolvido no Departamento de Ciência da Computação da Carnegie Mellon University, EUA, desde 1987. No começo da década de 90, o projeto Coda criou o conceito de operações desconectadas e assim passou a tratar mobilidade no sistema de arquivos. Desde então, o projeto Coda passou a ser referência quanto a acesso a dados em um ambiente de computação móvel.

Em cada cliente do Coda existe um gerenciador de *cache* chamado Venus, que obtém dinamicamente os dados e os põe na *cache*. Para dar suporte à operação desconectada, o Venus trabalha em três estágios: estado de pré-seleção (*hoarding*), emulação e reintegração [SKM+93].

O Venus combina duas abordagens no gerenciamento de *cache* baseado em prioridade (seu estado de pré-seleção): ele combina informações implícitas e explícitas para avaliar quais dados devem estar na sua *cache*. Implicitamente, o Venus usa um algoritmo tradicional de gerenciamento de *cache* (no caso, o LRU) para avaliar o histórico de arquivos recentemente usados. Explicitamente, é utilizado um banco de dados de pré-seleção, o HDB, cujo conteúdo são os caminhos dos arquivos que o usuário tem interesse que o Venus mantenha na *cache*. Para manipular o HDB, o Venus usa um programa chamado *hoard profile* que atualiza diretamente ou via scripts.

O Venus periodicamente avalia a retenção de objetos da *cache* em um processo de pré-seleção chamado *hoard walking*. O *hoard walking* é necessário para que seja atingida a expectativa do usuário em relação à importância da permanência de seus objetos na *cache*, isto é, se aqueles dados que o usuário espera que estejam na *cache* de fato encontrem-se lá. Quando a *cache* atinge esta expectativa, é dito que ela está em equilíbrio.

O Seer [Kue97] é um sistema de predição de dados, onde a escolha dos dados que devem estar presentes na *cache* é feita de forma automática e transparente ao usuário. O Seer não executa a replicação e transferência de arquivos entre servidores e clientes, em vez disso ele roda sobre sistemas de arquivos que têm esta funcionalidade (como o Coda, por exemplo).

A abordagem do Seer em escolher quais arquivos devem ser mantidos na *cache* é baseada na observação do comportamento do usuário através do uso de seus arquivos, e faz inferências quanto à relação entre estes arquivos. O Seer é composto de dois módulos: o Observer, que acompanha o comportamento do usuário e seus acessos a arquivos, classificando cada acesso de acordo com um tipo e convertendo caminhos dos arquivos em um formato absoluto, alimentando assim o módulo Correlator. O Correlator avalia as referências dos arquivos, calculado a distância semântica entre eles. Essa distância semântica alimenta um algoritmo de *clustering* [Kue97], que atribui cada arquivo a um ou mais projetos.

A distância semântica é a métrica para relação entre arquivos, e assim grupos de arquivos necessários para se trabalhar em um determinado projeto sejam identificados. Desta forma, o Seer prediz os projetos que o usuário está trabalhando e carrega seus arquivos localmente no cliente móvel.

Em [SUE00], cujo escopo do trabalho era páginas *web*, foi sugerido o uso de uma técnica de mineração de dados [AS94] para trabalhar o problema de pré-seleção de dados (*hoarding*) em computação móvel. A extração de regras de associação representada pelo padrão de acessos do cliente pode ser usada para predizer futuras requisições do usuário. O conjunto de requisições preditas (*predicted request set*) é o que deve ser carregado antes de ocorrer a desconexão de forma que as requisições futuras do cliente sejam atendidas localmente, sem que haja necessidade de estar conectado ao servidor.

As regras de associação obtidas após a mineração são usadas para determinar o conjunto candidato e o conjunto real do usuário até a desconexão. O conjunto candidato é formado de todas as requisições de um usuário específico, ao passo que o conjunto real são aquelas requisições que de fato são carregadas antes que haja a desconexão. O conjunto candidato é construído a partir de inferências baseadas no conjunto de regras de associação extraídas analisando o comportamento do usuário. Para podar esse conjunto de regras com o intuito de obter o conjunto real, é usada uma heurística baseada em prioridades e tamanho das requisições.

[SC04] propôs realizar a pré-carga (*pre-fetching*) de dados através da pré-seleção de dados utilizando um algoritmo de geração de regras de associação similar ao proposto por [AS94] em cima de dois grupos de dados: (i) o grupo de itens de dados que estão freqüentemente na *cache*, e (ii) um grupo de itens de dados que foram selecionados pelo usuário mas não constavam na *cache*.

Esse esquema de pré-carga baseado em dois grupos de dados foi motivado pelas seguintes observações:

1. Deve-se sempre deixar disponível em *cache* os dados que são freqüentemente usados pelo cliente móvel, e assim poupar largura de banda de conexão e energia.
2. Uma procura por um item que não está na *cache* é sempre seguida por várias procuras a itens que não estão na *cache*.

Gerando regras de associação sobre os dados desses dois grupos, é gerado um grupo de itens de dados que tem boa probabilidade de ser usada em um futuro próximo. Dessa forma, minimiza-se a energia gasta com conexões ao servidor (quando há disponibilidade de uma rede) e minimizam-se as falhas de acesso quando fora de uma rede sem fio. Assim como em [SUE00], o domínio da aplicação do trabalho de [SC04] são páginas *web*, e portanto a granularidade dos dados minerados são arquivos html. Nessa abordagem, o processo de análise do histórico de acesso a páginas *web* do usuário e o processamento para a geração das regras de associação são feitas no cliente.

4. Metodologia

Para este trabalho, foram pesquisados temas relevantes como sistemas distribuídos, sistemas móveis, mineração de dados, sistemas de arquivos e gerenciamento de memória. Em seguida, foi realizado um levantamento bibliográfico sobre gerência de *cache* em sistemas de computação móvel.

Como base para o desenvolvimento deste trabalho será utilizado um *framework* de banco de dados móveis [Côr04]. Esse *framework* consiste no desenvolvimento de um modelo de computação que permita a integração de um sistema de gerência de banco de dados (SGBD) em um ambiente de computação móvel, de forma que a integração não crie inconsistências nos dados. Essa arquitetura leva em consideração características inerentes à computação móvel, tais como desconexões freqüentes, fraca conectividade na rede sem fio e movimentação dos clientes. O trabalho de [Côr04] também apresenta um novo modelo de transações e uma arquitetura para implantação desse modelo utilizando agentes de software. O gerenciador de *cache* integrará o *framework* no cliente móvel do banco de dados.

Para a pré-seleção de dados da *cache*, usaremos uma técnica de mineração de dados baseada na geração de regras de associação, que tem como propósito achar relações entre coleções de dados através da análise de um grande conjunto de dados [AI93]. Propomos aplicar esta técnica [AS94] para descobrir

regras de associação através do histórico de consulta a dados do usuário do cliente móvel. As regras geradas a partir da análise do histórico de consulta a dados do usuário serão utilizadas para pré-selecionar os dados que devem estar na *cache* do cliente móvel. Com base nos itens de dados pré-selecionados, o cliente móvel deve fazer a pré-carga destes dados no momento em que uma conexão estiver disponível.

Diferentemente de [SC04], nossa abordagem poderá repassar o histórico de acesso a dados do usuário para processamento no servidor, dependendo do tipo de *hardware* do cliente móvel. Isso deve-se ao fato que o cliente móvel, em muitos casos, tem baixo poder de processamento (p. ex., celulares e computadores de mão). Outro aspecto deste trabalho que difere das abordagens de [SUE00] e [SC04] diz respeito à estrutura utilizada, uma vez que trabalharemos com dados estruturados de um SGBD. Esse ponto nos leva a trabalhar com uma granularidade mais complexa em comparação a arquivos.

Em seguida, executaremos as seguintes tarefas:

- Definição formal do modelo de *cache* proposto, onde serão definidas estruturas de dados de representação dos itens de dados;
- Definição da granularidade de dados para a pré-seleção de dados;
- Implementação do algoritmo para gerar as regras de associação baseado em [AS94] e [SC04];
- Método de processamento deste algoritmo de acordo com o *framework* [Côr04];
- A pré-carga dos dados.

Será, então, implementado um protótipo com o objetivo de testar e validar o gerenciador de *cache* simulando um caso de uso. No lado do servidor, estudaremos o MySQL e o PostgreSQL uma vez que ambos têm código aberto, e escolheremos um para a implementação. Já para o lado do cliente, estudaremos a possibilidade de usar a arquitetura PocketPC ou Palm.

5. Conclusão

Preencher uma *cache* utilizando-se técnicas de pré-carga pode melhorar o desempenho de um sistema de banco de dados móveis. Entretanto, a computação necessária para preencher essa *cache* também consome recursos do sistema. Portanto, é importante fazer a pré-carga apenas de dados relevantes para o usuário do cliente móvel.

Para a pré-carga, é esperado que a utilização de técnicas de mineração de dados como regras de associação seja capaz de selecionar aqueles dados relevantes ao usuário. Dessa forma, no momento de uma desconexão, dados relevantes ao usuário já estariam em *cache* disponíveis para consulta.

Esperamos, então, que esta pesquisa tenha como principais contribuições:

- No cliente, construir um gerenciador de cache para bancos de dados móveis baseado em mineração de histórico de consultas;
- No servidor, implementar o algoritmo proposto por [AS94] para analisar esse histórico;
- Minimizar consultas ao servidor, e assim diminuir o uso da rede;
- Aumentar a disponibilidade de dados mesmo que o dispositivo móvel esteja sem acesso a uma rede.

Referências

[AI93] R. Agrawal and T. Imielinski. Mining association rules between sets of items in large databases. In *ACM Sigmod - Conference on Management of Data*, 1993.

[AS94] R. Agrawal and R. Srikant. Fast algorithm for mining association rules. In *International Conference on Very Large Databases - VLDB*, 1994.

[Bar99] D. Barbara. Mobile computing and databases: A survey. *IEEE Transactions on Knowledge and Data Engeneering*, 1999.

[BBHB+04] G. Bernard, J. Ben-Hotman, L. Bougamin, G. Canals, B. Defude, J. Ferrié, S. Gançarski, R. Guerraoui, P. Molli, P. Pucheral, C. Roncancio, P. Serrano-Alvarado, and P. Valduriez. Mobile databases: A selection of open issues and research directions. *ACM Sigmod Record*, To appear in 2004.

[BMR04] S. Bürklen, P. Márron, and K. Rothermel. An enhanced hoarding approach based on graph analisys. In *IEEE International Conference on Mobile Data Management - MDM2004*, 2004.

[CFZ94] M. Carey, M. Franklin, and M. Zaharioudakis. Fine-grained sharing in page server database systems. In *ACM Sigmod - Conference on Management of Data*, 1994.

[Côr04] S. Côrtes. *Um Modelo de Transações Para Integração de SGBD a Um Ambiente de Computação Móvel*. PhD thesis, Departamento de Informática - Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Julho de 2004.

[CSL98] B. Chan, A. Si, and H. Leong. Cache management for mobile databases: Design and evaluation. In *IEEE International Conference on Data Engineering*, 1998.

[DFJ+96] S. Dar, M. J. Franklin, B. T. Jonsson, D. Srivastava, and M. Tan. Semantic data caching and replacement. In *International Conference on Very Large Databases - VLDB*, 1996.

[Fra96] M. Franklin. *Client Data Caching: A Foundation For High Performance Object Database Systems*. Kluwer Academic Publishers, 1996.

[Kis93] J. Kistler. *Disconnected Operation in a Distributed File System*. PhD thesis, Department of Computer Science, Carnegie Mellon University - CMU, 1993.

[Kue97] G. Kuenning. *Seer: Predictive File Hoarding for Disconnected Mobile Operation*. PhD thesis, University of California at Los Angeles - UCLA, 1997.

[PJFY04] F. Perich, A. Joshi, T. Finin, and Y Yesha. On data management in pervasive computing environments. *IEEE Transactions on Knowledge and Data Engineering*, 2004.

[Sat96] M. Satyanarayanan. Mobile information access. *IEEE Personal Communications Magazine*, 1996.

[SC04] H. Song and G. Cao. Cache-miss-initiated prefetch in mobile environments. In *IEEE International Conference on Mobile Data Management - MDM2004*, 2004.

[SKM+93] M. Satyanarayanan, J. Kistler, L. Mummert, M. Ebling, K. Puneet, and Q. Lu. Experience with disconnected operation in a mobile computing environment. In *USENIX Symposium on Mobile and Location-Independent Computing*, 1993.

[SKS87] M. Satyanarayanan, J. Kistler, and E. Siegel. Coda: A resilient distributed file system. In *IEEE Workshop on Workstation Operating Systems*, 1987.

[SUE00] Y. Saygin, Ö. Ulusoy, and A Elmagarmid. Association rules for supporting hoarding in mobile computer environments. In *IEEE International Workshop on Research Issues in Data Engineering*, 2000.

[YD01] J. Yao and M. Dunham. Caching management of mobile DBMS. *Journal of Integrated Computer-Aided Engeneering*, 2001.

[ZHH03] J. Zhang, A. Helal, and J. Hammer. UbiData: Ubiquitous mobile file service. In *ACM Symposium on Applied Computing - SAC 2003*, 2003.

Integrity Constraints for Temporal Versions Model: Classification, Modeling and Verification

Robson Leonardo Ferreira Cordeiro[1], Clesio Saraiva dos Santos[1], Nina Edelweiss[1]
[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil
e-mail: {rlfcordeiro, clesio, nina}@inf.ufrgs.br

Level : Mestrado
Programa de Pós-Graduação em Computação
Universidade Federal do Rio Grande do Sul (UFRGS)
Admission : April 2003
Conclusion expected to: November 2004

Abstract

The integrity maintenance process of a database is usually based on constraints representing restrictions of the modeled reality. Without respecting these constraints, it is impossible to faithfully represent the real world. Several researches on the integrity maintenance for snapshot databases were made, but for temporal versions databases constraints still represent an almost unexplored research area. Considering this, the present text first describes a complete classification of integrity constraints for temporal versions databases in general. Then, a method for integrity maintenance of databases based on the Temporal Versions Model (TVM) is presented. The Temporal Versioned Query Language (TVQL) is used to express queries searching for inconsistent data. The Temporal and Versioning Language for Schema Evolution (TVL/SE) is used to specify possible repairing actions. Through the present method, constraints modeling and specification becomes very easy and intuitive. Moreover, it adds functionality without adding complexity to the TVM model.

Key Words: *Integrity Constraints, Temporal Versions Databases, Query and Data Manipulation Languages, Databases Modeling Methods.*

1. Introduction

The main issue of a Database Management System (DBMS) is to manage databases that represent some part of the real world. In order to make a faithful representation, all the data of a database must respect some integrity constraints that can come from the modeled reality or some other sources. In this way, it is possible to say that a constraint restricts the set of fair states and/or valid state transitions of a database [11]. Besides this, snapshot databases support is not fully adequate for many applications. Some of them need to register historical data and their valid times, others need to store data sets versions, managing their aggregations and representation forms. Through time and version concepts, supported by a Temporal Versions DBMS, these needs are better supplied. That is why this kind of DBMS is becoming more needed and common.

The integrity maintenance process for temporal versions databases can be considered more complex than the snapshot databases one. Traditionally, integrity maintenance considers only the current state (or state transition) of a single data version. In temporal versions databases, many database states and data versions must be considered. Several researches and implementations related to constraints for snapshot and temporal databases were made, but constraints for temporal versions ones represent an almost unexplored research area that must be deeply analyzed.

Considering this, the main goal of this research is to propose an improvement for Temporal Versions Model (TVM) [9] in order to provide a complete integrity constraints support, making it possible to represent and verify constraints considering time and version concepts. First, a detailed classification was defined to help their representation and verification. Then, based on the classification, on the query language Temporal Versioned Query Language (TVQL) [10] and on the data manipulation language Temporal and Versioning Language for Schema Evolution (TVL/SE) used in a TVM based approach to schema evolution [7], a method for constraints specification and verification is being created.

This text is organized as follows: section 2 comments important researches related to this work; In order to get the necessary conceptual base, section 3 shows the main characteristics of the TVM data model, and of the TVQL and TVL/SE languages; section 4 describes a detailed integrity constraints classification for temporal versions databases, while section 5 comments the proposed integrity maintenance method for TVM databases; Finally, the methodology, contributions and expected results are showed in section 6.

2. Related Works

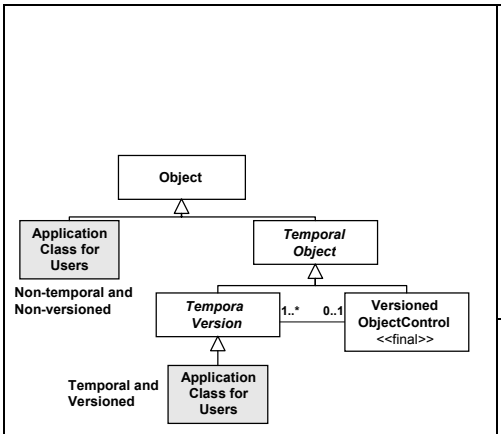
There are many researches on constraints for snapshot and temporal databases, but considering temporal versions databases, constraints are still almost unexplored. Papers on the integrity maintenance of these databases are rare and most of them consider just one of the involved concepts. Only in [5] a version model was used to implement temporal constraints, thus using both concepts. In this work, constraints scope and binding were considered. However, neither constraints involving both time and versions, their self-temporality, nor their self-versioning were considered.

In [4] a proposal for integrity maintenance for temporal constraints was made, considering their purpose, activation and deactivation type, declaration, temporal reference, precedence order and restricted and restrictive scope. In other works [1, 6], Böhlen and Escofet present good temporal constraints classifications and implementations based on their temporal reference, restricted and restrictive temporal type, besides temporal and state scope. Also, the paper [8] proposes a fine version constraints implementation, considering their declaration, origin, state scope and purpose. Moreover, Chomicki and Toman [2] analyze the state scope and purpose of temporal constraints, taking into account characteristics of real time.

3. Temporal Versions Model

The Temporal Versions Model (TVM) [9] is an object oriented data model that implements uniformly time and version concepts considering temporal versions objects together with traditional objects. These features can be seen in Figure 1a that shows the class hierarchy of TVM model. In this model, the time concept is used to control and store the historical evolution of data while the version

concept is applied to manage alternatives of projects. Besides this, time is represented through valid and transaction times associated to objects, version objects, attributes and relations.



a) Class Hierarchy of TVM

```
query ::= SELECT [EVER][DISTINCT] targetC,targetC
          FROM identificC,identificC
          [WHERE[EVER]serchC]
          [GROUP BY groupC,groupC [HAVING logicalExpr]]
          [ORDER BY orderC,orderC [setOp query];
targetC        ::= (*|propertyName|aggregationFunctions|preDefInterval
                    |preDefInstant) [AS identifier]
identificC     ::= className [.VERSIONS] [aliasName]
searchC        ::= logicalExpr|tempExpr
groupC         ::= propertyName|preDefInterval|preDefInstant
logicalExpr    ::= AnyLogicalExpression
tempExpr       ::= logicalExpr|PRESENT (logicalExpr)
orderC         ::= groupC [ASC|DESC]
setOp          ::= UNION | INTERSECTION | DIFFERENCE
preDefInterval ::= [propertyName.](tInterval|vInterval)
preDefInstant  ::= [propertyName.](tiInstant|tfinstant|viInstant|vfInstant)
                   |[className.](iLifeTime)|(fLifeTime)
```

b) Simplified BNF Syntax of TVQL

```
query  ::= (insert | update | delete) identificSV
insert ::= INSERT INTO <className>
             [(<propertyName>=<expression> [VALIDITY INTERVAL definedInterval]
             [,<propertyName>=<expression>[VALIDITY INTERVAL definedInterval]]*)]
           VALUES (<constant>[,<constant>]*)
         [VALIDITY INTERVAL definedInterval]
delete ::= DELETE FROM <className> [WHERE<condition>][fValidity = instant]
update ::= UPDATE <className>
           SET <propertyName>=<expression> [VALIDITY INTERVAL definedInterval]
           [,<propertyName>=<expression> [VALIDITY INTERVAL definedInterval]]*
           WHERE <condition>
identificSV       ::= current | schemaVersionName=schemaVersionName
                      | VALIDITY INTERVAL definedInterval
definedInterval   ::= [iInstant..[fInstant]] | [.. fInstant]
schemaVersionName ::= identifier
instant           ::= tempValue | now | null
tempValue         ::= 'dateValue [,hourValue]'
```

c) Simplified BNF Syntax of TVL/SE

Figure 1. Class hierarchy of TVM and BNF syntaxes of TVQL and TVL/SE

The query language of TVM (TVQL) [10], is a SQL based query language with many time and version operators (before, into, after, intersect, overlap, equal, and others) to make possible a clear use of both concepts in queries. By default a TVQL query retrieves the current version data stored in the current database state. However, many combinations of historical and current values of data versions can be made in a query using the keywords ever (considering all data history), versions (considering all data versions) and present (considering only the current data). The select operation is defined in TVQL, but the operations insert, update and delete are not supported. According to this fact, the data manipulation language TVL/SE [7] was proposed to make possible data insertions, updates and deletions. The simplified BNF syntaxes of TVQL and TVL/SE can be seen in Figures 1b and 1c[1].

4. Integrity Constraints Classification

A complete constraints classification is essential for developing a good integrity maintenance process. Indeed, classifying all the possible constraints kinds is the best way to deeply analyze their characteristics, before the development of modeling and verifying tools. Besides constraints concepts, a classification must consider characteristics of their specification with respect to the used schema, DBMS and data model. Also, classes must be orthogonally defined wherever it does not affect the classification scope. Considering this and based on their temporality and versioning, constraints for temporal versions databases are analyzed and classified according to the following aspects: origin, substance, specification, application, temporality and versioning. A detailed description of this classification can be found in [3].

According to [11], a constraint specification must have the following components: (i) restrictive, consisting of a set with database or non-database components used to build restrictions on the database data; (ii) restricted, a set of database components whose contents are restricted by the constraint; (iii) restrictive condition, that consists in a generic logic expression relating the restricted and restrictive components; (iv) verifying points, that define the start points of the constraint validity verification; and (v) violation actions, that are the actions executed over the database, immediately after the constraint violation, in order to establish again the data integrity.

[1] Where brackets [] indicate optional segment, braces {} indicate optional repetitive segment (zero or more times), parentheses indicate set of options, | indicates "or" and emphasized (bold) words are terminal symbols.

The origin aspect is related to the kind of the origin agent of a constraint. Constraints can be originated from the environment, enterprise, data model or implementation. Another aspect considered is the substance, which shows the kind of restriction imposed on data. In this aspect the following criteria are considered: restrictive and restricted types, restrictive and restricted aspects, restrictive and restricted scope, state scope, restrictive and restricted structure, restrictive condition structure, completion and purpose. Through the aspect specification, constraints are analyzed based on the way they are declared to the DBMS. Its criteria are declaration, self-temporality type and scope, and self-versioning type. The distinct kinds of data integrity maintenance processes are considered in the application aspect and it is composed of the criteria: verification points, precedence order, treatment, activation form, activation and deactivation restriction, activation and deactivation type, inspection and vehicle.

One of the most important aspects of the classification is the temporality. According to it, a constraint is classified considering the temporal characteristics of the components of its restrictive and restricted sets. Constraints can be temporal or non-temporal. Temporal constraints are analyzed by the sub-criteria restrictive and restricted temporality type, temporal reference, restrictive and restricted temporal scope and temporal granularity. The versioning aspect is also important and classifies a constraint based on the versioning aspects of its restrictive and restricted sets. Considering this, a constraint can be versioned or non-versioned, and according to the sub-criteria restrictive and restricted versioning type and scope, the first ones are analyzed.

5. Modeling and Verifying Integrity Constraints

Considering the generic classes of constraints for any temporal versions DBMS, the next step of this research is to model and verify constraints for databases based on the TVM model. A constraint definition must be based on the TVQL query that searches for inconsistent data, a possible set of repairing actions specified by TVL/SE statements, and further controls related to its self-temporality and self-versioning.

The main idea of this modeling and verifying method is to assume hypothetically that all data modification is valid before a constraint verification and, consequently that the current database state virtually encloses these modifications. So, the constraint verification process consists in executing its TVQL query in order to look for data that violates it. If the query results an empty set, the new database state does not violate the constraint, otherwise a violation was detected. After the violation detection, the possible set of repairing actions (TVL/SE statements) can be executed in order to establish again the database integrity. If this is not possible, a rollback to the last valid database state must be made. Through this method, as an example, the traditional constraint "salaries can not decrease" could be specified as in Figure 2[2].

Inconsistent Database State
(Rollback Transaction)

←**Non-Empty** (**SELECT * FROM** EMPLOYEE E1, EMPLOYEE E2
WHERE EVER E1.CODEMP = E2.CODEMP **AND** E1.SALARY > E2.SALARY **AND**
E1.SALARY.VINTERVAL **BEFORE** E2.SALARY.VINTERVAL)

Figure 2. Constraint specification sample

Executing completely these queries for each constraint verification is extremely costing and possibly non-feasible. In spite of this, using the present method, constraints modeling and specification becomes very easy and intuitive. Also, it adds functionality without adding complexity to the TVM model, since almost all resources used were previously defined. So, in order to make possible its usage, optimization techniques are needed especially for the TVQL query processing process.

6. Methodology, Contributions and Expected Results

At the present moment constraints were completely classified, resulting in the paper [3]. This classification is being used as the base for developing the modeling method. Besides the modeling step, a verification method is being defined mapping the TVM definitions to relational ones, and TVQL and TVL/SE statements to SQL. Finally, in order to test the method, a prototype shall be implemented for the integrity maintenance of TVM databases. Also, some optimizations, mainly query

[2] Where SALARY.VINTERVAL represents the validity interval of an employee salary.

optimizations, shall be considered. During the whole work, papers shall be submitted to national and international conferences, in order to evidence the scientific production of the job.

The main contributions of this research, related to the proposed solutions, required activities, expected results and their current status, can be seen in Table 1.

Table 1. Main contributions of this research

Contribution	Proposed Solution	Required Activities	Expected Results	Status
Constraints Classification for Temporal Versions Databases	Constraints analysis over several aspects.	Aspects, criteria, and orthogonal classes definition.	To get a complete constraints classification for any Temporal Versions Database.	√
Constraints Modeling in TMV Databases	Modeling constraints through TVQL queries on inconsistent data and TVL/SE statements for repairing actions.	Defining a modeling method through a complete BNF syntax, based on the previous classification.	To get a generic method for modeling TMV constraints, considering the temporality and versioning of data and of the constraint itself.	♣
Constraints Verifying in TVM Databases	Implementing the integrity maintenance, considering constraints TVQL and TVL/SE statements.	Implementing TVQL and TVL/SE statements, besides some other controls.	To get a complete but non-optimized integrity maintenance prototype for TVM databases.	♣
Optimizations on the Integrity Maintenance of TMV databases	To use mainly query optimization techniques in order to optimize constraints verification.	Developing mainly TVQL query optimization techniques.	To get an optimized integrity maintenance prototype for TVM databases.	⊕

√ - Concluded task ♣ - In development task ⊕ - Future task

References

1. BÖHLEN, Michael H. Valid Time Integrity Constraints. Tucson, AZ: Department of Computer Science – University of Arizona, 1994. (Technical Report 94-30)
2. CHOMICKI, Jan; TOMAN, David. Implementing Temporal Integrity Constraints Using an Active DBMS. IEEE Transactions on Knowledge and Data Engineering, [S.l.], v.7, n.4, p. 566-582, Feb. 1995.
3. CORDEIRO, Robson L. F.; SANTOS, Clesio S.; EDELWEISS, Nina; GALANTE, Renata M. Classificação de Restrições de Integridade em Bancos de Dados Temporais de Versões. In: Brazilian Symposium on Databases, SBBD, 19., 2004. Proceedings… Brasília, Brazil, Oct. 2004. (In Portuguese)
4. DAYAL, Umeshwar. et al. The HiPAC Project: Combining Active Databases and Timing Constraints. ACM SIGMOD Record, [S.l.], v.17, n.1, p. 51-70, Mar. 1988.
5. DOUCET, Anne et al. Using Database Versions to Implement Temporal Integrity Constraints. In: Constraint Database and Applications, CDB, 2., 1997. Proceedings… Delphi, Greece:[s.n.], 1997. p. 219-233.
6. ESCOFET, Carme Martín. Analyzing Temporal Integrity Constraints to Obtain the Minimum Number of Transition Rules. Barcelona, Catalunya: Universitat Politècnica de Catalunya, 2001. (Technical Report LSI-01-52-R)
7. GALANTE, Renata M.; EDELWEISS, Nina; SANTOS, Clesio S. Data Modification Language for Full Support of Temporal Schema Versioning. In: Brazilian Symposium on Databases, SBBD, 18., 2003. Proceedings… Manaus, Brazil: p. 114-128 Oct. 2004.
8. MEDEIROS, Claudia Bauzer; JOMIER, Geneviève; CELLARY, Wojciech. Maintaining Integrity Constraints across Versions in a Database. Campinas, SP: DCC–Unicamp, 1992. (Relatório Técnico DCC-08/92)
9. MORO, M. M.; SAGGIONARO, S. M.; EDELWEISS, N; SANTOS, C. S. Adding time to an object oriented versions model. In: Intl. Conf. on Database and Expert Systems Applications, DEXA, 12., 2001. Proceedings… Munich, Germany: v. 2113 of LNCS, p. 805-814, Sep. 2001.

10. MORO, M. M.; EDELWEISS, N.; ZAUPA, A.; SANTOS, C. S. TVQL - Temporal Versioned Query Language. In: Intl. Conf. on Database and Expert Systems Applications, DEXA, 13., 2002. Proceedings… Aix-en-Provence, France: v. 2453 of LNCS, p. 618-627, Sep. 2002.
11. SANTOS, Clesio Saraiva dos. Caracterização Sistemática de Restrições de Integridade em Bancos de Dados. 1980. Tese de Doutorado – D.I-PUC, Rio de Janeiro. (In Portuguese)

Evolução de Documentos XML com Tempo e Versões*

Rodrigo Gasparoni Santos, Nina Edelweiss, Renata de Matos Galante
Computer Science Department – Federal University of Rio Grande do Sul (UFRGS)
Instituto de Informática - Universidade Federal do Rio Grande do Sul (UFRGS)
email: {rgsantos, nina, galante}@inf.ufrgs.br

Nível: Mestrado
Programa de Pós-Graduação em Computação
Universidade Federal do Rio Grande do Sul
Ano de Ingresso: 2003
Previsão de Conclusão: Dezembro de 2004

Resumo

O trabalho proposto explora a aplicação de técnicas de gerenciamento temporal sobre documentos XML. Estes foram escolhidos como alvo da proposta em função de sua rápida disseminação, nos últimos anos, como formato de representação e intercâmbio de dados via Web. Tais técnicas, já estudadas para bases relacionais e orientadas a objetos, ganham importância ainda maior quando associadas a documentos XML, dada sua forte ligação com a Internet, onde o conteúdo dos documentos evolui muito rapidamente.
As técnicas presentes nas propostas existentes possuem cada qual suas vantagens e suas limitações. O objetivo do modelo proposto é combinar estas técnicas em um único modelo; seu uso conjunto deve fornecer grande flexibilidade no tratamento temporal de documentos XML. Além da definição da organização lógica dos dados, a proposta inclui a definição de uma arquitetura para a utilização do modelo, incluindo linguagens de consulta e manipulação dos dados.

Palavras-chave: *XML, Temporalidade, Versionamento, Evolução de Documentos*

Abstract

The proposed work explores the application of temporal management techiniques over XML documents. These were chosen to be the target of the proposal because of their growing acceptance, over the past few years, as a format for representation and exchange of data through the World Wide Web. Such techniques, which have been extensively studied for relational and object-oriented databases, become even more important when associated to XML documents, given its srong connection with the Internet, where page contents evolve very quickly.
The existing techniques possess each one a set of advantages and limitations. The goal of the proposed model is to combine these techniques into a single model; its united use should supply great flexibility in the temporal treatment of XML documents. Besides the definition of the logical organization of the data, the proposal includes the definition of an architecture to be used algon with the model, which includes query and data manipulation languages.

Keywords: *XML, Temporality, Versioning, Document Evolution*

*Este trabalho está sendo financiado com recursos do CNPq.

1. Introdução

Recentemente, a linguagem XML vem se convertendo em um formato universal de meta-linguagem para descrição de dados eletrônicos. Estima-se que, em um futuro próximo, empresas e indivíduos estarão usando uma grande quantidade de documentos nesse formato em suas transações diárias. Em tais situações, bases de dados XML capazes de gerenciar grandes volumes de dados sob a forma de documentos XML ganham grande importância. A capacidade de manipular alterações em documentos XML é também importante, pois o conteúdo destes, em muitos casos, evolui conforme o tempo passa, e os usuários podem desejar utilizar não só as versões recentes como também as mais antigas. Ademais, a necessidade da associação de informação temporal, já muito estudada em bases relacionais e orientadas a objeto, é intensificada para documentos XML, em função de sua forte associação com a Web, onde os documentos evoluem mais freqüentemente e podem apresentar estruturas diferentes ainda que seguindo um mesmo esquema.

O interesse por unir os conceitos de modelagem temporal a documentos XML proporcionou o surgimento de várias propostas [1, 2, 5, 9, 10] nos últimos anos. Tais propostas concentram-se no controle das modificações ao conteúdo dos documentos; poucas chegam a esboçar mecanismos para lidar com a evolução dos esquemas. Apesar desse ponto em comum, as propostas diferem em muitos outros: algumas concentram-se em registrar a evolução através do armazenamento dos diversos estados que os dados assumem [9]; outras, através das operações que provocam as transformações [10]; e outras ainda através de uma mescla de informação propriamente dita e referências a estados anteriores [1, 2]. Diferem também na definição de que tipo de informação deve ser temporalizada, mas, principalmente, diferem no emprego de rótulos temporais: algumas trabalham com tempo de transação, outras com tempo de validade, e outras ainda apenas com a numeração seqüencial das diversas versões que o documento assume ao longo do tempo.

Embora a importância da evolução dos esquemas seja inegável, esta não será abordada no presente trabalho; pretende-se, contudo, estabelecer uma base para evolução do conteúdo, para que a mesma possa ser estendido em trabalhos futuros para agregar a evolução estrutural. O objetivo do trabalho proposto é definir um modelo para a evolução do conteúdo de documentos XML, unificando as dimensões de tempo de transação, tempo de validade e de definição de novas versões. A utilização conjunta desses conceitos, tratados isoladamente nas propostas estudadas, possibilita grande flexibilidade no tratamento da evolução dos documentos – o versionamento permite que haja linhas paralelas na evolução do documento; o tempo de validade, o armazenamento histórico e a projeção de informações futuras; e o tempo de transação, a recuperação de estados passados da base de dados. Além da organização lógica dos dados, o trabalho deve incluir uma proposta de arquitetura para aplicações que se valham do modelo, bem como um estudo de caso e um protótipo de ferramenta com suas principais características. O trabalho será desenvolvido com base na experiência adquirida com os modelos TVM [6] e TVSE [4], trabalhos similares desenvolvidos para bases relacionais e orientadas a objeto, respectivamente.

2. O Modelo XML Empregado

A definição de documento XML que será considerada para a construção do modelo é uma simplificação da proposta original [5], que exclui os conceitos de *namespaces*, instruções de processamento e comentários. Essas construções foram excluídas para simplificar o modelo e permitir que o mesmo se concentre no conteúdo propriamente dito. Deve-se ressaltar, contudo, que o mesmo pode ser estendido para incluir tais características. Dessa forma, os objetos da linguagem que serão abordados são: elementos, atributos e nodos de texto.

Este modelo será estendido para associar a cada um destes tipos de objetos uma série de rótulos temporais (*timestamps*), para registrar os tempos de validade inicial e final dos objetos – delimitando o período no qual modelam adequadamente a realidade – bem como os tempos de transação inicial e final – demarcando o período no qual informação foi registrada na base de dados. Nodos de elementos e de texto ganham também identificadores globais e persistentes, para registrar a correspondência entre os diferentes estados de um mesmo objeto ao longo da evolução do documento.

Em adição aos *timestamps* para controle de tempo de transação e tempo de validade, o modelo proposto inclui também a possibilidade de definição de versões do documento. A criação de uma versão é determinada pelo usuário, em função de uma mudança significativa no documento, e é realizada através da derivação dos valores de uma versão corrente. Um documento sempre possui uma versão raiz, a partir da qual pode ou não haver versões derivadas. A hierarquia de derivação de versões construída segundo essas regras toma a forma de uma árvore: cada versão possui exatamente uma versão pai (com exceção da raiz), e pode possuir uma ou mais versões derivadas, as quais, por sua vez, evoluem de forma independente umas das outras. Os identificadores globais registram a correspondência entre os trechos de informação entre uma versão e outra.

```
<schema
xmlns="http://www.w3.org/2001/XMLSchema">
 <element name="Document">
  <complexType>
   <sequence>
    <element name="Schema" type="string"
             minOccurs="0"/>
    <element name="Version"
             type="VersionType"/>
   </sequence>
  </complexType>
 </element>
 <complexType name="VersionType">
  <sequence>
   <element name="Element"
            type="ElementType"/>
   <element name="DerivedVersions">
    <complexType>
     <element name="Version"
              type="VersionType" minOccurs="0"
              maxOccurs="unbounded"/>
    </complexType>
   </element>
  </sequence>
 </complexType>
 <complexType name="ElementType">
  <sequence>
   <element name="Validity"
            type="ValidityType"/>
   <element name="Attributes">
    <complexType>
     <element name="Attribute"
              type="AttributeType"
              minOccurs="0"
              maxOccurs="unbounded"/>
    </complexType>
   </element>
   <element name="Content"
            type="ContentType"/>
  </sequence>
  <attribute name="Name" type="NMTOKEN"
             use="required"/>
  <attribute name="GlobalID" type="integer"
             use="required"/>
 </complexType>
 <complexType name="ValidityType">
  <element name="TimeStamps"
           type="TimeStampsType"
           maxOccurs="unbounded"/>
 </complexType>
 <complexType name="TimeStampsType">
  <attribute name="IVT" type="dateTime"
             use="required"/>
  <attribute name="FVT" type="dateTime"
             use="required"/>
  <attribute name="ITT" type="dateTime"
             use="required"/>
  <attribute name="FTT" type="dateTime"
             use="required"/>
 </complexType>
 <complexType name="AttributeType">
  <sequence>
   <element name="Validity"
            type="ValidityType"/>
   <element name="String" type="StringType"
            maxOccurs="unbounded"/>
  </sequence>
  <attribute name="Name" type="NMTOKEN"
             use="required"/>
 </complexType>
 <complexType name="StringType">
  <element name="TimeStamps"
           type="TimeStampsType"
           maxOccurs="unbounded"/>
  <attribute name="Value" type="string"
             use="required"/>
 </complexType>
 <complexType name="ContentType">
  <choice minOccurs="0" maxOccurs="unbounded">
   <element name="Element"
            type="ElementType"/>
   <element name="Text" type="TextType"/>
  </choice>
 </complexType>
 <complexType name="TextType">
  <sequence>
   <element name="Validity"
            type="ValidityType"/>
   <element name="String" type="StringType"
            maxOccurs="unbounded"/>
  </sequence>
  <attribute name="GlobalID" type="string"
             use="required"/>
 </complexType>
</schema>
```

FIGURA 1. Descrição em XML Schema do Modelo Proposto

O modelo completo pode ser descrito através da especificação em XML Schema vista na figura 1. Cada documento pode possuir um esquema associado; esse esquema, quando presente, deve ser definido no momento de criação do documento, e uma vez definido permanece inalterado ao longo de todo o intervalo de vida deste. O esquema é, portanto, imutável e único para todas as versões que vierem a ser associadas àquele documento. Cada alteração que vier a ocorrer deve manter todas as visões, de qualquer instante, coerentes com o esquema. O formato escolhido para representação do esquema é XML Schema, por três razões básicas: ampla utilização e aceitação, flexibilidade superior à de DTD's e principalmente por ser um dialeto XML, podendo ser incorporado diretamente ao documento.

Cada documento possui uma versão inicial, da qual é possível derivar outras versões, cujas estruturas serão idênticas à da primeira. Cada versão possui um elemento distinto, correspondente à raiz do documento XML que representa. Elementos são codificados em tags *Element*, com atributos para registrar o nome do elemento e o seu identificador. Dentro de *Element*, há um elemento *Validity*, que engloba os diversos intervalos de validade que definem o tempo de vida do elemento apontado por *Element*. Cada intervalo é definido através de uma tag *TimeStamps,* que apresenta os atributos *IVT*, *FVT*, *ITT* e *FTT*, os quais correspondem, respectivamente, aos tempos de validade inicial, validade final, transação inicial e transação final. No caso dos elementos raízes, seus rótulos temporais indicam também o tempo de vida da versão à qual estão subordinados.

Dentro de cada *Element*, há também um elemento *Attributes*, que contêm o conjunto (possivelmente vazio) de atributos que correspondem àquele elemento. Cada atributo é registrado através de um elemento *Attribute*, que representa seu tempo de vida também através de um elemento *Validity*. Dentro do seu tempo de vida, um atributo pode assumir diversos valores, cada qual com seu intervalo de validade específico, os quais são registrados através do elemento *String*. Cada *String* possui um atributo que identifica seu valor, bem como elementos *TimeStamps*, para marcar os intervalos temporais associados a cada um dos valores assumidos ao longo do tempo.

Por fim, cada elemento pode conter sub-elementos e nodos de texto; estes entram no modelo dentro do elemento *Content*. Sub-elementos possuem a mesma estrutura descrita anteriormente; nodos de texto possuem também rótulos temporais para o elemento em si e para cada um de seus valores, construídos nos mesmos moldes das demais propriedades. Cada um desses objetos apresenta a restrição de que seu intervalo de vida deve estar contido no intervalo de vida de seu superior imediato na hierarquia.

O modelo assume que o tempo avança linearmente em passos discretos, porém não estipula a granularidade dos rótulos temporais, permitindo que esta varie conforme a aplicação. Além de um dado ponto no tempo, cada rótulo pode conter também o termo *now*, indicando um intervalo em aberto, associado ao tempo presente. As figuras 2 e 3 mostram um exemplo de codificação com o modelo proposto de um documento XML em processo de evolução (os valores dos rótulos de tempo de transação e a especificação do esquema foram omitidos por simplicidade).

```
<Artigo>
 <Título> Evolução de XML </Título>
 <Autor Email = "rgsantos@inf.ufrgs.br">
  Rodrigo G. Santos
 </Autor>
</Artigo>
```

```
<Artigo>
 <Título> Evolução de XML </Título>
 <Autor>
  Rodrigo Gasparoni Santos
 </Autor>
 <Autor>
  Nina Edelweiss
 </Autor>
</Artigo>
```

FIGURA 2. Documento Original em 01/01/2004 (esquerda) e 15/06/2004 (direita)

```
<Document>
 <Schema> ... </Schema>
 <Version>
  <Element Name="Artigo" GlobalID="1">
   <Validity>
    <TimeStamps IVT="01/06/2004" FVT="now"
                ITT="..." FTT="..."/>
   </Validity>
   <Attributes/>
   <Content>
    <Element Name="Título" GlobalID="2">
     <Validity>
      <TimeStamps IVT="01/06/2004" FVT="now"
                  ITT="..." FTT="..."/>
     </Validity>
     <Attributes/>
     <Content>
      <Text GlobalID="3">
       <Validity>
        <TimeStamps IVT="01/06/2004" FVT="now"
                    ITT="..." FTT="..."/>
       </Validity>
       <String Value="Evolução de XML">
        <TimeStamps IVT="01/06/2004" FVT="now"
                    ITT="..." FTT="..."/>
       </String>
      </Text>
     </Content>
    </Element>
    <Element Name="Autor" GlobalID="4">
     <Validity>
      <TimeStamps IVT="01/06/2004" FVT="now"
                  ITT="..." FTT="..."/>
     </Validity>
     <Attributes>
      <Attribute Name="Email">
       <Validity>
        <TimeStamps IVT="01/06/2004"
                    FVT="14/06/2004"
                    ITT="..." FTT="..."/>
       </Validity>
       <String Value="rgsantos@inf.ufrgs.br">
        <TimeStamps IVT="01/06/2004"
                    FVT="14/06/2004"
                    ITT="..." FTT="..."/>
       </String>
      </Attribute>
     </Attributes>
     <Content>
      <Text GlobalID="5">
       <Validity>
        <TimeStamps IVT="01/06/2004" FVT="now"
                    ITT="..." FTT="..."/>
       </Validity>
       <String Value="Rodrigo G. Santos">
        <TimeStamps IVT="01/06/2004"
                    FVT="14/06/2004"
                    ITT="..." FTT="..."/>
       </String>
       <String Value="Rodrigo Gasparoni
                      Santos">
        <TimeStamps IVT="15/06/2004" FVT="now"
                    ITT="..." FTT="..."/>
       </String>
      </Text>
     </Content>
    </Element>
    <Element Name="Autor" GlobalID="6">
     <Validity>
      <TimeStamps IVT="15/06/2004" FVT="now"
                  ITT="..." FTT="..."/>
     </Validity>
     <Attributes/>
     <Content>
      <Text GlobalID="7">
       <Validity>
        <TimeStamps IVT="15/06/2004" FVT="now"
                    ITT="..." FTT="..."/>
       </Validity>
       <String Value="Nina Edelweiss">
        <TimeStamps IVT="15/06/2004" FVT="now"
                    ITT="..." FTT="..."/>
       </String>
      </Text>
     </Content>
    </Element>
   </Content>
  </Element>
 </Version>
</Document>
```

FIGURA 3. Documento Ajustado ao Modelo

3. Arquitetura

Para uma aplicação que empregue o modelo proposto, sugere-se a arquitetura vista na figura 4. Note que o modelo restringe-se à organização lógica dos dados, não fazendo quaisquer restrições quanto à organização física da base de dados que armazena os documentos XML temporalizados; esta pode ser uma base XML nativa, uma base relacional ou ainda algum tipo de organização projetado especificamente para o modelo em questão. A escolha do modelo físico de representação não altera, portanto, a lógica do gerenciamento temporal, embora certamente influencie no tempo do acesso aos dados.

A visão que o usuário tem dos dados corresponde ao esquema que definiu para os mesmos: o usuário entra com os comandos da linguagem de manipulação de dados e com expressões de consulta especificados sobre o esquema original; cabe à aplicação traduzi-los para a visão do modelo, e então repassá-los à base XML, que os executa. Já os resultados retornados vêm no formato do modelo, e devem ser convertidos de volta para a visão do usuário.

Na arquitetura proposta, há um módulo através do qual as modificações aos documentos devem ser informadas; é comum em cenários nos quais os dados são extraídos da Web, contudo, que o registro das operações que transformam o documento não estejam disponíveis, tendo em seu lugar apenas os estados atual e anterior. Esse caso, contudo, pode ser reduzido ao primeiro, através do uso de algoritmos de detecção de diferenças como os apresentados em [3, 8]. Tais algoritmos tomam por entrada dois documentos XML quaisquer, possivelmente representando estados consecutivos do mesmo documento, e produzem como saída um conjunto de operações que transforma um no outro.

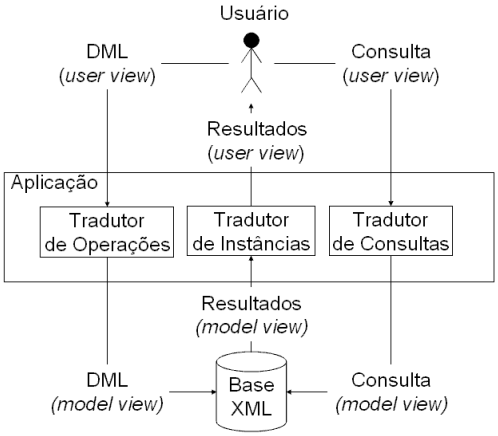


FIGURA 4. Arquitetura Proposta

4. Conclusões e Continuidade do Trabalho

O trabalho até aqui desenvolvido mostrou um modelo capaz de combinar características de diversas propostas presentes na literatura, reunindo as diversas possibilidades de representação temporal que as mesmas oferecem em um único modelo. A proposta centra-se na organização lógica dos dados, e não em sua implementação física, ao mesmo tempo em que mantém separadas a representação interna das informações e a visão do usuário.

A seguir, definir-se-á um conjunto de operações de manipulação dos documentos XML, associadas a regras para manutenção da consistência dos rótulos temporais. Da mesma forma, deve ser sugerida uma linguagem de consulta para acesso aos dados contidos na base. Uma vez consolidada a proposta do modelo e da arquitetura, o mesmo será submetido a uma verificação empírica, através de um estudo de caso e da implementação de um protótipo com as principais características do modelo.

5. Referências

[1] CHIEN, Shu-Yao; TSOTRAS, Vassilis J.; ZANIOLO, Carlo. **Copy-Based versus Edit-Based Version Management Schemes for Structured Documents.** RIDE-DM, 2001.

[2] CHIEN, Shu-Yao; TSOTRAS, Vassilis J.; ZANIOLO, Carlo. **Efficient Management of Multiversion Documents by Object Referencing.** Roma, Proceedings of VLDB, 2001.

[3] COBÉNA, Grégory; ABITEBOUL, Serge; MARIAN, Amélie. **Detecting Changes in XML Documents.** San Jose, International Conference on Data Engineering, 2002.

[4] GALANTE, Renata de Matos. **Modelo Temporal de Versionamento com Suporte à Evolução de Esquemas.** Porto Alegre, PPGC/UFRGS, 2004. Tese de Doutorado.

[5] MANUKYAN, M. G; KALINICHENKO, L. A. **Temporal XML.** Advances in Databases and Information Systems, 2001.

[6] MORO, Mirella Moura. **Modelo Temporal de Versões.** Porto Alegre, PPGC/UFRGS, 2001. Dissertação de Mestrado.

[7] World Wide Web Consortium. **Extensible Markup Language (XML).** Disponível por WWW em http://www.w3.org/XML/.

[8] WANG, Yuan; DeWITT, David J.; CAI, Jin-Yi. **X-Diff: An Effective Change Detection Algorithm for XML Documents.** Relatório Técnico, University of Wisconsin, 2001.

[9] WANG, Fushang; ZANIOLO, Carlo. **Publishing and Querying the Histories of Archived Relational Databases in XML.** Roma, Anais do WISE, 2003.

[10] WONG, Raymond K.; LAM, Nicole. **Efficient Re-construction of Document Versions Based on Adaptative Forward and Backward Change Deltas.** Praga, Anais do DEXA, 2003.

Uma Ferramenta em Software Livre para Modelagem e Projeto de Banco de Dados para Aplicações OLAP com Análise Espacial

Rodrigo Soares Manhães[1], Rogério Atem de Carvalho[1] (orientador),
Asterio Kiyoshi Tanaka[1] (co-orientador)
[1]Núcleo de Pesquisa e Desenvolvimento em Informática
Universidade Candido Mendes (UCAM-Campos)
e-mail: {rmanhaes, ratem}@cefetcampos.br, tanaka@uniriotec.br

Nível : Mestrado
Programa de Pós-Graduação em Informática
Núcleo de Pesquisa e Desenvolvimento em Informática
Universidade Candido Mendes (UCAM-Campos)
Ano de Ingresso : 2003
Previsão de Conclusão : Julho de 2005

Resumo

Muitos problemas no âmbito do Suporte à Decisão têm como característica a necessidade de processamento analítico de dados geográficos e não-geográficos de modo conjunto, de forma que sejam atendidos de maneira integrada todos os aspectos que influenciam uma instituição: o próprio negócio, o tempo e o espaço. A ferramenta PostGeoOLAP [1] integra componentes analíticos (Data Warehouses e OLAP) e geográficos (SIG), desde o nível conceitual até o nível de implementação. Trata-se de uma aplicação OLAP que trabalha sobre um SGBD objeto-relacional espacial, o PostGreSQL acrescido de uma extensão espacial, o PostGIS. A dissertação de mestrado que aqui se descreve pretende introduzir aprimoramentos e extensões à ferramenta, além de torná-la multiplataforma e fazê-la consonante com os conceitos de Software Livre. Entre os aprimoramentos e extensões pretendidos podem-se citar: a criação de ferramentas para a administração do data warehouse, principalmente no que diz respeito à carga de dados do sistema transacional para o analítico; otimização de consultas analíticas e geração das agregações pré-armazenadas; automatização da criação do banco de dados dimensional através de arquivos XMI gerados a partir do modelo UML por ferramentas CASE e outras. Para que a ferramenta, ora escrita em Visual Basic.Net, se torne multiplataforma, pretende-se porta-la para a linguagem Java. Além disso, o fato de a ferramenta ser livre viabiliza seu uso por instituições anteriormente limitadas pelo alto custo de licenciamento deste tipo de tecnologia.

Palavras Chave : Banco de Dados Espaciais, Data Warehouse e Aplicações OLAP, Sistemas de Informações Geográficas, Técnicas de Modelagem em Banco de Dados.

1. Tema da Dissertação

A dissertação de mestrado ora em andamento visa introduzir aprimoramentos e extensões à ferramenta PostGeoOLAP [1], que, além de fornecer uma técnica de desenvolvimento integrando, desde o nível conceitual, dados analíticos e geográficos, permite esta integração de forma direta, possibilitando a criação de aplicações que contemplem funcionalidades espaciais e de data warehousing.
A principal motivação do projeto é a possibilidade de tornar mais simples a modelagem e implementação de um sistema de suporte à decisão que integre características analíticas (de um DW) e geográficas (de um SIG), fazendo com que tais conceitos possam coexistir no mesmo modelo, tanto no nível conceitual quanto no nível da implementação, com mapeamento direto. No nível conceitual, são usados diagramas de classe UML estendidos com estereótipos geográficos na modelagem de um DW, em conformidade com padrões propostos pelo consórcio OpenGIS [7]. Na implementação, é usado um SGBD espacial (PostgreSQL [10] estendido com PostGIS [9]), e a aplicação OLAP é integrada com um componente de visualização geográfica.
PostGeoOLAP é, deste modo, uma aplicação ROLAP capaz de manipular dados espaciais e convencionais, sem a necessidade de módulos de integração entre aplicações desenvolvidas com softwares proprietários distintos.
A dissertação cujo desenvolvimento aqui se descreve trata de tópicos pendentes na implementação da citada ferramenta, visando portá-la para uma linguagem de uso corrente no ambiente de Software Livre (Java), introduzir melhoramentos em algumas características da ferramenta e incluir funcionalidades até então não contempladas.

2. Objetivos

A dissertação tem por objetivos:
- Portar a ferramenta PostGeoOLAP, atualmente escrita em Visual Basic.Net, para a plataforma Java. VB.Net, por ser um ambiente de desenvolvimento proprietário e de distribuição mediante pagamento de licenças, não goza da mesma aceitação e acessibilidade que a linguagem Java no ambiente de Software Livre.
- Automatizar, através de arquivos XMI exportados por ferramentas CASE para UML, a criação do banco de dados dimensional, tanto suas definições quanto manipulações de dados em SQL. Na versão corrente, a definição da estrutura do banco de dados analítico e de suas manipulações é feita manualmente.
- Criação de ferramentas para a administração do data warehouse, notadamente no que concerne ao auxílio à tarefa de carga de dados do banco de dados transacional para o analítico. Na versão atual, isto é feito de modo totalmente manual.
- Otimização da ferramenta PostGeoOLAP, com o objetivo do aprimoramento de seu desempenho, tanto na geração das agregações pré-armazenadas, quanto no processamento das consultas analíticas.
- Implementação da persistência de metadados em XML, a qual é feita na versão corrente no banco de dados Access.
- Substituir o componente de visualização geográfica ora utilizado no PostGeoOLAP, o PlanetGIS, visto que este é proprietário (componente COM), por um equivalente de distribuição livre e código aberto. Até o momento, testes bem sucedidos têm sido realizados com a ferramenta Java Unified Mapping Platform [5].

3. Iniciativas Similares

Diversos outros trabalhos foram conduzidos no sentido de buscar a integração entre sistemas analíticos e geográficos e, por sua relevância, são aqui apenas citados, para estabelecer um paralelo entre o que já foi produzido e o que está sendo realizado neste projeto.
- Projetos que abordam processamento analítico sobre dados geográficos: GeoMiner, um projeto de datamining espacial [12]; MapCube - uma extensão ao conceito de Cubo de Dados para o domínio espacial [11].

- Projetos que integram aplicações distintas (OLAP e GIS) através de módulo de integração: GOAL: Geographical Information On-Line Analysis [6]; GISOLAP – GIS/OLAP [2]; GOLAPA - Geographical On-Line Analytical Processing Architecture [3].
- Projetos que propõem OLAP Espacial, similar ao nosso projeto: trabalhos descritos em [4] e em [8].

Sob o aspecto conceitual, essas duas últimas propostas são as mais próximas do nosso projeto, com a diferença de que não propõem a parte de modelagem conceitual e geração de códigos de esquema do banco de dados espacial.
Sob o aspecto de plataforma computacional, o trabalho aqui descrito pretende ser realizado em consonância com os conceitos do Software Livre, a exemplo do que é feito no projeto GOLAPA, acima mencionado, que utiliza uma abordagem de módulo de integração. Não temos conhecimento de qualquer projeto com as características do PostGeoOLAP que seja gratuito e de código aberto. Não há, disponíveis no ambiente de Software Livre, sequer ferramentas que realizem apenas OLAP.
Com isto, o PostGeoOLAP e, por conseqüência este trabalho, já que nele se insere, não encontra similares com as mesmas características.

4. Restrições Iniciais de Projeto

Como o trabalho pretende estar em consonância com os conceitos de Software Livre, as definições de projeto têm, desde o início, que trabalhar com essa restrição. Assim, algumas das definições de projeto originais do PostGeoOLAP serão modificadas nesta dissertação. A versão inicial, criada no ambiente Visual Basic.Net, funciona apenas nos sistemas operacionais da família Windows de 32 bits. Propõe-se, aqui, uma migração do código para a linguagem Java, o que tornará o PostGeoOLAP efetivamente multiplataforma.
Do mesmo modo, o componente de visualização geográfica utilizado na versão corrente da ferramenta é um objeto COM denominado PlanetGIS, que é proprietário. Este trabalho propõe a sua substituição por um equivalente livre. Estuda-se, no momento, com bons resultados, a utilização de componentes de visualização que fazem parte do framework aberto JUMP (Java Unified Mapping Platform).
O banco de dados utilizado será o mesmo da versão inicial, o PostGreSQL, que é um SGBD relacional, gratuito e de código aberto. A escolha e a continuação com o uso deste banco se deram porque o citado SGBD possui uma extensão para o tratamento de dados geográficos, PostGIS, tornando-se, assim, único no universo do Software Livre.

5. Estado Atual da Pesquisa

A principal atividade, no momento, é a adaptação do código escrito em Visual Basic.Net para a linguagem Java. Espera-se que até o mês de julho de 2004 esta etapa esteja concluída, com o código inteiramente portado e funcionando, no mínimo, nas mesmas condições que a atual implementação.
Paralelamente, vêm sendo realizados testes para o substituto livre do componente proprietário de visualização geográfica ora utilizado. Os testes, bem sucedidos até o momento, alçam o JUMP à categoria de melhor candidato a uma efetiva utilização no trabalho.
Além disto, desenvolve-se um esforço de pesquisa bibliográfica, com o objetivo de se realizar um levantamento dos trabalhos realizados que tangenciem os problemas a serem abordados na dissertação. Simultaneamente, faz-se um estudo detalhado da ferramenta PostGeoOLAP e uma revisão das demais tecnologias e metodologias envolvidas no projeto, como ROLAP, sintonia em bancos de dados geográficos, Java, XML/XMI, SIG, padrões de projeto orientado a objeto, etc. Após uma análise mais aprofundada dos problemas a serem abordados, iniciar-se-á o processo de construção do modelo conceitual dos tópicos a solucionar, utilizando-se a metodologia orientada a objetos.

6. Relevância

Como se sabe que o custo da aquisição ou licenciamento de sistemas de suporte à decisão como aplicações OLAP ou SIG é proibitivo a pequenas e médias empresas, esta iniciativa torna acessíveis a um público muito maior estas tecnologias, normalmente restritas à esfera das grandes organizações. No setor público, cujos custos com tecnologia da informação em boa parcela são creditados a licenciamentos de software, também possui grande importância o uso do Software Livre no sentido da economia de recursos, da independência de fornecedores e, principalmente, da melhoria das condições sociais e da qualidade de vida das populações, seguramente resultantes da honesta utilização das citadas tecnologias de suporte à decisão.
Além disso, principalmente em países pobres ou "em desenvolvimento" como o Brasil, é ainda mais importante a adoção de software sem restrições de uso ou distribuição e de código aberto, como instrumento para a democratização da tecnologia e da informação.

7. Aplicabilidade

A configuração atual da economia mundial tem por característica a concentração da tecnologia na esfera das grandes empresas e corporações, até que esta se torne obsoleta e, só então, seja repassada às pequenas e médias empresas e aos países ditos "em desenvolvimento".
Neste sentido, o Software Livre é um instrumento de democratização tecnológica, indo na contramão destas tendências centralizadoras, na medida em que permite que o acesso às tecnologias mais recentes seja simultâneo para pequenas e grandes organizações, para países pobres e ricos.
Num contexto mais localizado, pode-se facilmente vislumbrar a imensa gama de projetos envolvendo suporte à decisão baseado em OLAP e SIG que empresas e instituições que antes não possuíam acesso a tais tecnologias, principalmente devido aos custos proibitivos de licenciamento, poderão desenvolver com o apoio do PostGeoOLAP.
A ferramenta vem sendo utilizada em uma empresa distribuidora de publicações de editoras nacionais, em pontos de venda no Norte Fluminense, e, no contexto de gestão municipal, em aplicações espaciais nos municípios de Macaé – RJ, e Cachoeiro do Itapemirim – ES. Outras aplicações estão sendo consideradas e propostas, inclusive internacionalmente, a partir da divulgação do projeto no SourceForge (http://sourceforge.net/projects/postgeoolap).

Acrônimos
CASE – Computer Aided Software Engineering
COM – Component Object Model
DDL – Data Definition Language
DML – Data Manipulation Language
OLAP – On-Line Analytical Processing
ROLAP – Relational On-Line Analytical Processing
SGBD – Sistema Gerenciador de Banco de Dados
SIG – Sistema de Informações Geográficas
UML – Unified Modeling Language
XMI – XML Metadata Interchange
XML – eXtended Markup Language

Referências
1. Colonese, G. Uma Ferramenta Aberta para Desenvolvimento Integrado de Sistemas de Informação para Processamento Analítico e Geográfico. Campos dos Goytacazes: Universidade Candido Mendes, 2004. (Dissertação de Mestrado) – Projeto registrado no SourceForge e disponível em http://sourceforge.net/projects/postgeoolap
2. Ferreira, A. C. F. Um Modelo para Suporte à Integração de Análises Multidimensionais e Espaciais. Rio de Janeiro: UFRJ/IM/NCE, 2002. (Dissertação de Mestrado).
3. Fidalgo, R.N.; Times, V. C.; Souza, F. F. GOLAPA: Uma Arquitetura Aberta e Extensível para Integração entre SIG e OLAP. GeoInfo 2001, III Workshop Brasileiro de

GeoInformática. p.111-118. Instituto Militar de Engenharia, Rio de Janeiro. 4 e 5 de outubro de 2001.

4. Han, J; Stefanovic, N; Koperski, K. Selective Materialization: An Efficient Method for Spatial Data Cube Construction. PAKDD, 1998.
5. JUMP. Java Unified Mapping Platform. Disponível em http://www.vividsolutions.com/jump/ Acesso em 21/06/2004.
6. Kouba, Z.; Matousek, K..; Miksovsky P. "On Data Warehouse and GIS Integration". In Proceedings of DEXA2000. Greenwich, Inglaterra: DEXA2000, 2000.
7. OGC (OPEN GIS CONSORTIUM). OpenGIS Simple Specifications for SQL Revision 1.1. OpenGis Project Document 99-049. Publicado em 05/05/1999.
8. Papadias, D.; Kalnis, P.; Zhang, J.; Tao, Y. Efficient OLAP Operations in Spatial Data Warehouses. Proceedings of the 7th International Symposium on Advances in Spatial and Temporal Databases. Pág.: 443 a 459. ACM Records, 2001.
9. POSTGIS. Documentação da extensão espacial PostGIS ao PostGreSQL, versão 0.8. Disponível em http://postgis.refractions.net. Acesso em 05/02/2004.
10. POSTGRESQL.ORG. Documentação do SGBD PostGreSQL 7.4. Disponível em: http://www.postgresql.org. Acesso em 05/02/2004.
11. Shekhar, S.; Lu, C. T.; Ta, X.; Chawla, S.; Vatsavai, R.R.. Map Cube: A visualization Tool for Spatial Data Warehouse. Disponível em http://www.cs.umn.edu/research/shashigroup/mapcube.htm. Acesso em 20/01/2004.
12. Stefanovic, N.. Design and Implementation of On-Line Analytical Processing (OLAP) of Spatial Data. Dissertação de Mestrado. Disponível em http://gunther.smeal.psu.edu/3070.html. Universidade de Belgrado, 1997. Acesso em 20/09/2003.

Índice por Autor / Author Index