

VI ESCOLA REGIONAL DE BANCO DE DADOS
14 a 16 de Abril 2010
Joinville - SC

ANAIS

Editores

Denio Duarte (UFFS)
Fernando José Braz (UNIVILLE)
Ronaldo dos Santos Mello (UFSC)

Coordenação Geral

Denio Duarte (UFFS)
Fernando José Braz (UNIVILLE)

Coordenação do Comitê de Programa

Ronaldo dos Santos Mello (UFSC)

Realização

Universidade do Estados de Santa Catarina (UDESC)
Universidade da Região de Joinville (UNIVILLE)

Promoção

Sociedade Brasileira de Computação – SBC

Prefácio

A organização da sexta edição da Escola Regional de Banco de Dados (ERBD 2010) lhe dá as boas-vindas. A ERBD tem sido desde 2005 um fórum de discussão e integração entre alunos, professores e profissionais da Região Sul em torno do tema banco de dados. As cinco primeiras edições da escola ocorreram nas cidades de Porto Alegre – RS (organizada pela Universidade Federal do Rio Grande do Sul – UFRGS), Passo Fundo – RS (organizada pela Universidade de Passo Fundo – UPF), Caxias do Sul – RS (organizada pela Universidade de Caxias do Sul – UCS), Florianópolis – SC (organizada pela Universidade Federal de Santa Catarina – UFSC) e Ijuí – RS (organizada pela Universidade Regional do Noroeste do Estado do Rio Grande do Sul - UNIJUI), respectivamente. Este ano, a ERBD volta a ser realizada no Estado de Santa Catarina, na cidade de Joinville e sob a organizacao conjunta da Universidade do Estado de Santa Catarina (UDESC) e Universidade da Região de Joinville (UNIVILLE).

O tema da escola neste ano é *Mineração de Dados* que trata do problema de encontrar informações relevantes a partir de bases de dados volumosas, tendo a sua aplicação na indústria principalmente na área de Business Intelligence (BI – Inteligência do Negócio). Contamos, nesta edição, com a ilustre presença da professora e pesquisadora Dr. Sandra de Amo, da Universidade Federal de Uberlândia (UFU), responsável pela palestra principal sobre o tema. Além disso, outras partes da programação do evento, como palestras, minicursos e oficinas, também enfatizam o tema.

A ERBD 2010 não seria possível sem o esforço conjunto de um Comitê de Organização e de um Comitê de Programa. O Comitê de Organização, composto de Sub-Comissões e de um Comitê Local, foi fundamental para tornar realidade a programação e a logística do evento, respectivamente. Agradecimentos especiais aos membros da sub-comissão de Palestra/Painel, Profs. Guillermo Nudelman Hess (FEEVALE) e Cristiano Damiani Vasconcellos (UDESC); da sub-comissão de Minicursos, Profs. Carmem Hara (UFPR) e Rodrigo Ramos Dornel (UNIVILLE); da sub-comissão de Oficinas, Profs. Rebeca Schroeder (UDESC) e Luiz Melo Romão (UNIVILLE); e da sub-comissão de Demos, Profs. Deise de Brum Saccoll (UFSM) e Walter Silvestre Coan (UNIVILLE). Um agradecimento particular também ao Prof. Ronaldo dos Santos Mello, pelo zeloso trabalho na coordenação do Comitê de Programa, e a todos os integrantes deste Comitê, que permitiram a seleção de trabalhos relevantes e de qualidade para serem apresentados nas sessões técnicas.

À Sociedade Brasileira de Computação (SBC), pelo seu contínuo apoio na promoção do evento, aos nossos patrocinadores e apoiadores locais pelo suporte, à UNIVILLE pela infra-estrutura concedida, e ao Departamento da Ciência da Computação (DCC) da UDESC e de Informática da UNIVILLE, pela presteza constante no apoio a eventos locais. Também gostaríamos de agradecer a equipe de eventos da UDESC e UNIVILLE pela zelosa preocupação com os detalhes da VI ERBD. O nosso muito obrigado!

Um excelente evento a todos!

Denio Duarte (UFFS)
Fernando José Braz (UNIVILLE)
Coordenação Geral da VI ERBD

Carta da Coordenação do Comitê de Programa

A comunidade interessada na área de banco de dados é mais uma vez agraciada com os Anais da Escola Regional de Banco de Dados (ERBD), desta vez na sua sexta edição e realizada na cidade de Joinville, no Estado de Santa Catarina, nos dias 14, 15 e 16 de abril de 2010.

O programa técnico da ERBD 2010 apresenta tópicos atuais de pesquisa e desenvolvimento na área de banco de dados. Nesta edição, pela primeira vez, foram submetidos trabalhos em duas trilhas: *Pesquisa e Aplicações/Experiências*. A trilha Pesquisa considera trabalhos de investigação científica, surveys de tópicos atuais de pesquisa, resultados experimentais ou desenvolvimento, que apresentam contribuições científicas na área de banco de dados. A trilha Aplicações/Experiências considera aplicações, ferramentas ou trabalhos em desenvolvimento na área de banco de dados.

A ERBD 2010 possui 4 sessões técnicas para a apresentação de 8 artigos da trilha Pesquisa dentre 19 submetidos (taxa de aceitação em torno de 42%), bem como 2 dias do evento dedicados à apresentação e/ou demonstração de 10 trabalhos na trilha Aplicações/Experiências dentre 23 submetidos (taxa de aceitação em torno de 43%). A ERBD não seria possível sem o interesse dos autores que nos honraram com o envio de artigos em ambas as trilhas, divulgando trabalhos interessantes desenvolvidos junto a seus grupos e instituições que incluiram, além da região Sul, os Estados de São Paulo, Rio de Janeiro, Minas Gerais, Mato Grosso do Sul, Bahia e Piauí. Agradeço sinceramente a participação de todos!

A programação do evento conta ainda com palestras sobre mineração de dados, mini-cursos relacionados a *tuning* de banco de dados, mineração de dados, buscas semânticas e modelagem de dados XML, bem como oficinas sobre *tuning* de banco de dados, mineração de dados e serviços Web semânticos. Os participantes do evento são também contemplados com um painel, onde pesquisadores e profissionais debatem sobre o tema da escola.

Dedico aqui um agradecimento especial a todos os membros do Comitê de Programa, que contribuiram com avaliações dos artigos da ERBD 2010. Muitos deles, juntamente com os avaliadores externos por eles indicados, trabalharam em seus períodos de férias, tendo sido, mesmo assim, pontuais na entrega de suas revisões.

Cabe também um sincero agradecimento às sub-comissões da ERBD, que tiveram importante contribuição no fechamento da programação técnica, bem como aos Coordenadores Gerais da Organização, os profs. Denio Duarte (UDESC) e Fernando José Braz (UNIVILLE), cuja dedicação tornou possível a realização de mais uma ERBD.

Espero que todos vocês aproveitem a ERBD 2010, participando ao máximo de toda a programação que, com muito esforço, foi organizada. Não deixem também de conhecer a bela cidade de Joinville!

Bom evento e boa estadia a todos!

Ronaldo dos Santos Mello (UFSC)
Coordenador do Comitê de Programa

Coordenação

Coordenação Geral

Denio Duarte (UFFS)

Fernando José Braz (UNIVILLE)

Coordenação do Comitê de Programa

Ronaldo dos Santos Mello (UFSC)

Coordenação de Palestras/Painel

Guillermo Nudelman Hess (FEEVALE)

Cristiano Damiani Vasconcellos (UDESC)

Coordenação de Minicursos

Carmem Hara (UFPR)

Rodrigo Ramos Dornel (UNIVILLE)

Coordenação de Oficinas

Rebeca Schroeder (UDESC)

Luiz Melo Romão (UNIVILLE)

Coordenação de Demos

Deise de Brum Saccol (UFSM)

Walter Silvestre Coan (UNIVILLE)

Comitê de Programa

Adrovane Kade (UFRGS)
Angelo Augusto Fozza (UNIPLAC)
Carina Friedrich Dorneles (UFSC)
Carmem Hara (UFPR)
Cristiano Roberto Cervi (UPF)
Deise de Brum Saccol (UFSM)
Denio Duarte (UDESC)
Denise Bandeira (UNISINOS)
Eduardo Kroth (UNISC)
Guilhermo Nudelman Hess (FEEVALE)
Helena Graziottin Ribeiro (UCS)
Karin Becker (Quality Knowledge)
Raquel Kolitski Stasiu (PUC/PR)
Rebeca Schroeder (UDESC)
Renata Galante (UFRGS)
Renato Fileto (UFSC)
Ronaldo dos Santos Mello (UFSC) (Coordenador)
Vânia Bogorny (UFSC)
Vidal Martins (PUC/PR)
Viviane Moreira (UFRGS)

Revisores Externos

Fabiano Baldo (UDESC)
Edson Murakami (UDESC)

Comitê de Organização Local (Joinville)

Cristiano Damiani Vasconcellos(UDESC)
Denio Duarte (UDESC)
Edicarsia Barbiero Pillon (SOCIESC)
Edson Murakami (UDESC)
Fabiano Baldo (UDESC)
Fernando José Braz (UNIVILLE)
Luiz Mello Romão (UNIVILLE)
Roberto Rosso (UDESC)
Rodrigo Ramos Dornel (UNIVILLE)
Walter Silvestre Coan (UNIVILLE)

Comitê Diretivo da ERBD

Helena Graziottin Ribeiro (UCS)
Renato Fileto (UFSC)
Ronaldo dos Santos Mello (UFSC)

Um Modelo para Projetar e Implementar Bancos de Dados Analítico-Temporais para Apoio à Tomada de Decisões, Auditorias e Recuperação de Dados

Alex Sandro Romeo de Souza Poletto¹, Jorge Rady de Almeida Júnior²

¹Centro de Pesquisas em Informática - Instituto Municipal de Ensino Superior de Assis
Fundação Educacional do Município de Assis (FEMA)
Av. Getúlio Vargas, 1200 - CEP 19.807-634 – Assis - SP – Brasil

²Departamento Sistemas Digitais – Escola Politécnica – Universidade São Paulo (USP)
Av. Prof. Luciano Gualberto, 158. CEP: 05508-900 - São Paulo, SP - Brasil
apoletto@femanet.com.br, jorge.almeida@poli.usp.br

Abstract. This work describes a model whose main objective is to store historic data, resulting in the Analytic-Temporal Databases. This model can aid in the design and implementation of the Analytic-Temporal Databases that constitutes a very adequate foundation to help in the decision taking process, audits and data recovery. This work contains two stages. The first stage aims to manually help the modeling of Analytical-Temporal Database, based on models of data from Operational Databases. The second stage aims to provide automatic mechanisms, used in Database Management Systems, providing generation and storage of analytical-temporal data, using triggers and stored procedures.

Resumo. O presente trabalho descreve um modelo para, projetar e implementar Bancos de Dados Analítico-Temporais, cujo principal objetivo é o de armazenar históricos de dados para auxiliar no processo de tomada de decisões, auditorias e recuperação de dados. O trabalho contém duas etapas. A primeira etapa tem por objetivo auxiliar, manualmente, a modelagem do Banco de Dados Analítico-Temporal, com base nos modelos de dados dos Bancos de Dados Operacionais. Na segunda etapa objetivou-se disponibilizar mecanismos automáticos, explorados nos próprios Sistemas Gerenciadores de Banco de Dados, que possibilitem a geração e o armazenamento dos dados Analítico-Temporais, usando gatilhos e procedimentos armazenados.

1. Introdução

O objetivo deste trabalho é propor um modelo para projeto e implementação de Bancos de Dados Analítico-Temporais (BDAT), os quais, por sua vez, visam servir de alicerce aos processos de tomada de decisões, auditorias e recuperação de dados. Os dados a serem inseridos nesses bancos de dados serão originados exclusivamente dos Bancos de Dados Operacionais (BDO).

Normalmente, a finalidade dos BDO é armazenar somente o valor mais recente de um dado. A necessidade de possuir históricos de dados levou ao surgimento dos chamados Bancos de Dados Analítico-Temporais, que representam uma opção a ser utilizada para o armazenamento do estado evolutivo dos dados, no sentido também, de

evitar sobrecarga nas operações operacionais, quando da necessidade de recuperar históricos de dados [Cordeiro et al. 2004].

Recentemente surgiu a tecnologia Oracle Flashback, que tem a capacidade de consultar dados históricos, realizar análises, alterar e realizar a auto-reparação do serviço para recuperar corrupções lógicas, para monitoramento de alterações, bem como recuperação e retorno em períodos de tempo das exclusões de dados, mas todo esse processo é realizado diretamente nos BDO [Bryla e Loney 2008].

A criação de um BDAT é importante, já que além de oferecer um armazenamento mais completo e rico (especializado) em relação aos BDO, tem como propósito específico prestar auxílio aos responsáveis pelas tomadas de decisões, auditorias e recuperação de dados. Para se ter um melhor rendimento nessas atividades, e no sentido de não atrapalhar as operações operacionais, é necessário que esse banco de dados seja separado logicamente dos BDO [Sprague e Watson 1991].

Todo esse panorama, motivou o surgimento deste trabalho, cujo objetivo é possibilitar o desenvolvimento de mecanismos e técnicas que auxiliem na modelagem, geração e armazenamento de dados, resultando, assim, em um BDAT.

Este artigo está organizado da seguinte forma: na Seção 2 são apresentados alguns trabalhos correlatos; na Seção 3, é descrito o modelo proposto; na Seção 4, é apresentado um estudo de caso; e, finalmente, na última seção são relatadas as considerações finais.

2. Trabalhos Correlatos

Alguns trabalhos de proposta similar ao aqui apresentado são: “Bridging Relational Database History and the Web: the XML Approach” [Wang et al. 2006], e o “A Temporal Data Model and Management System for Normative Texts in XML Format”, de [Grandi et al. 2003], cujo interesse está em integrar sistemas gerenciadores de bancos de dados com o XML, no sentido de gerar históricos ou dados temporais. O trabalho “Conceptual Design of Data Warehouses from E/R Schemes” de [Moody e Kortnik 2000] e o “From enterprise models to dimensional models: a methodology for data warehouse and data mart design” de [Golfarelli et al. 1998], têm por finalidade encontrar formas de se converter um modelo entidade-relacionamento em um modelo multidimensional, bem como efetuar a consolidação e a agregação dos dados.

3. Descrição do Modelo Proposto

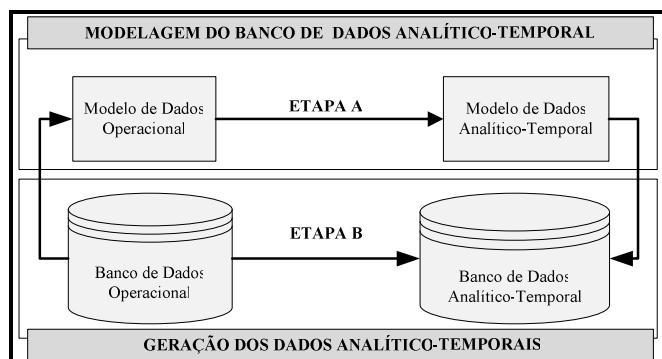


Figura 1. Diagrama geral do Modelo Proposto.

Para a realização do modelo proposto, dividiu-se o mesmo em duas etapas, conforme ilustrado na Figura 1.

3.1. Descrição da Etapa A: Modelagem do BDAT

Em se tratando de modelar dados analítico-temporais, um modelo de dados deve apresentar as características básicas de Bancos de Dados Convencionais (Operacionais), acrescentando a possibilidade de representar dados que se alterem ao longo do tempo. Em termos gerais, bancos de dados que mantêm dados correntes, bem como dados passados, são designados Banco de Dados Analítico-Temporais [Tansel 1997].

De maneira geral, esta etapa visa a preparação manual do Modelo de Dados Analítico-Temporal com base no Modelo de Dados Operacional. O objetivo deste último é criar um esquema que ofereça condições para o armazenamento de dados analítico-temporais, conforme apresentado a seguir.

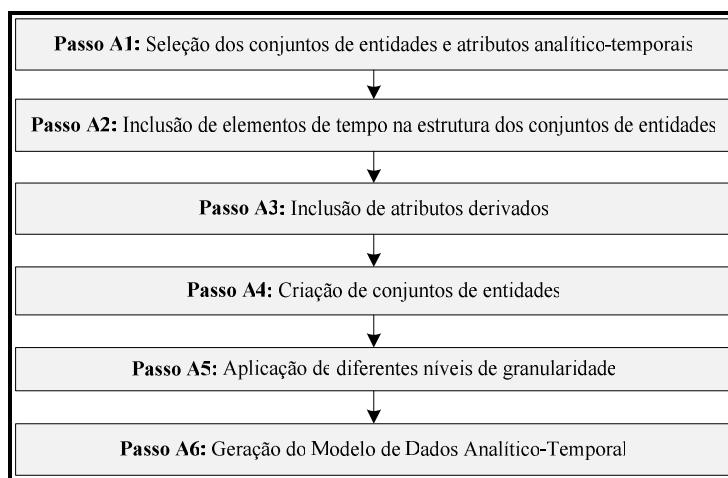


Figura 2. Passos da Etapa A.

Passo A1: Seleção dos conjuntos de entidades e atributos analítico-temporais

Este passo tem a função de identificar, no Modelo de Dados Operacional, quais são os conjuntos de entidades e atributos essenciais para a geração dos dados analítico-temporais, a fim de que os mesmos possam ser incluídos na elaboração do Modelo de Dados Analítico-Temporal. Além do que, será possível verificar quais conjuntos de entidades são puramente operacionais, uma vez que dados puramente operacionais não são incluídos no processo de geração de dados analítico-temporais.

Passo A2: Inclusão de “elementos tempo” na estrutura dos conjuntos de entidades

É primordial, em alguns casos, a inclusão de atributos que controlarão os períodos (intervalo de datas) nos quais os valores contidos nos conjuntos de entidades permanecerão válidos. Ou seja, em um período, determinado valor foi considerado o mais recente, ou válido. Para que isso seja possível, é necessária a inclusão de novos atributos “timestamp” para o controle dos períodos. Desta forma, a função deste passo é incluir, no BDAT, um ou mais atributos “timestamp”, na estrutura dos conjuntos de entidades “selecionados”, a fim de possibilitar o armazenamento exato dos dados analítico-temporais, podendo ou não, compor a chave primária existente. Pode ser necessária também, a adição de um novo conjunto de entidades, para que haja o armazenamento mais detalhado de todas as ocorrências de alteração de valores.

Passo A3: Inclusão de atributos derivados

Em alguns casos, é preciso tomar decisões em um curto espaço de tempo. Pode ser necessário realizar consultas muito complexas que impliquem acessar um grande volume de tuplas, com inúmeros cálculos. Isso poderá atrasar a recuperação das informações, daí a necessidade da adição de atributos derivados, já que tal procedimento pode eliminar a necessidade de realizar cálculos durante o processo de recuperação de valores.

Passo A4: Criação de conjuntos de entidades

Este passo é justificado pelo fato de que, em alguns casos, pode ocorrer a necessidade de armazenar “atributos multivalorados”, ou seja, atributos ou conjuntos de atributos que precisem armazenar muitos valores para uma mesma entidade proprietária. Portanto, a função deste passo é oferecer um meio para a representação dos vários relacionamentos entre os dados analítico-temporais.

Passo A5: Aplicação dos diferentes níveis de granularidade

A função deste passo é estabelecer, junto aos usuários, qual será o nível necessário de detalhe ou de sumarização para os dados a serem transportados ao BDAT, para que, futuramente, seja possível responder às diversas solicitações de consultas dos usuários.

Passo A6: Geração do Modelo de Dados Analítico-Temporal

A função deste passo é apresentar o Modelo de Dados Analítico-Temporal resultante desta etapa, e que culminará na criação do BDAT. No modelo aqui proposto, há uma tabela chamada Analítico-Temporal, que deverá conter os seguintes atributos: uma chave primária, atributos descritivos, atributos numéricos, atributos temporais e atributos derivados, conforme ilustrado na Figura 3.

TABELA ANALÍTICO-TEMPORAL	
Chave Primária	
Atributos Descritivos	
.....	
Atributos Numéricos	
.....	
Atributos Temporais	
.....	
Atributos Derivados	
.....	

Figura 3. Tabela Analítico-Temporal.

A intenção é que essa Tabela Analítico-Temporal seja utilizada como um padrão para a criação de toda a estrutura de dados do BDAT.

3.2. Descrição da Etapa B: Geração dos Dados Analítico-Temporais

Esta etapa tem por objetivo a geração, o transporte e o armazenamento de dados analítico-temporais, de forma dinâmica e automática.

Para a realização dinâmica e automática dessa geração, é necessário algum controle sobre o BDO, que permita, a qualquer momento, a aquisição e o transporte dos dados para o BDAT. Por essa razão é que são utilizados gatilhos e procedimentos armazenados, visto que são os recursos dos Sistemas Gerenciadores de Bancos de

Dados (SGBD) que permitem acompanhar as atualizações na base operacional e enviá-las para a base analítico-temporal [Italiano e Ferreira 2006].

Para auxiliar esse processo, são oferecidas, nesta etapa, duas especificações genéricas, uma de um gatilho e outra de um procedimento armazenado, que deverão ser utilizados como *templates* para as codificações dos gatilhos e procedimentos armazenados necessários para a geração, transporte e armazenamento dos dados junto ao BDAT [Poletto e Almeida Jr. 2007], conforme descrito a seguir:

3.2.1. Especificação Genérica do Gatilho

```

CREATE OR REPLACE TRIGGER TRIGGER_TEMPLATE
BEFORE/AFTER INSERT OR UPDATE OF BDO_Atributo_Y ON BDO_Tabela
FOR EACH ROW
[DECLARE
    Variavel_1 BDO_Tabela_X.BDO_Atributo_X%TYPE;
    Variavel_2 BDO_Tabela_X.BDO_Atributo_X%TYPE; ...;]
BEGIN
    IF INSERTING THEN
        [SELECT BDO_Atributo_1, BDO_Atributo_2, ... INTO Variavel_1, Variavel_2, ...
         FROM BDO_Tabela_X WHERE BDO_Chave_X = :NEW.BDO_Chave_2;]
        INSERT INTO BDAT_Tabela(BDAT_Chave_1, BDAT_Chave_2, ...
                               BDAT_Atributo_1, BDAT_Atributo_2, BDAT_Atributo_3, BDAT_Atributo_4, ...
                               BDAT_Atributo_Pre_1, BDAT_Atributo_Pre_2, BDAT_Atributo_Pre_3,
                               BDAT_Atributo_Pre_4, BDAT_Atributo_Data_1)
        VALUES(:NEW.BDO_Chave_1, NEW.BDO_Chave_2, ...
               :NEW.BDO_Atributo_1, :NEW.BDO_Atributo_2, ..., [Variavel_1, Variavel_2], ...
               :NEW.BDO_Atributo_Calc_1*:NEW.BDO_Atributo_Calc_2,
               :NEW.BDO_Atributo_Calc_1+:NEW.BDO_Atributo_Calc_2,
               :NEW.BDO_Atributo_Calc_1/:NEW.BDO_Atributo_Calc_2,
               :NEW.BDO_Atributo_Calc_1-:NEW.BDO_Atributo_Calc_2, Data_Dia);
    ELSE
        [IF :NEW.BDO_Chave_2 <> :OLD.BDO_Chave_2 then
            SELECT BDO_Atributo_1, BDO_Atributo_2, ... INTO Variavel_1, Variavel_2, ...
            FROM BDO_Tabela_X WHERE BDO_Chave_X = :NEW.BDO_Chave_2;
        ELSE
            SELECT BDO_Atributo_1, BDO_Atributo_2, ... INTO Variavel_1, Variavel_2, ...
            FROM BDO_Tabela_X WHERE BDO_Chave_X = :OLD.BDO_Chave_2;
        ENDIF;]
        UPDATE BDAT_Tabela SET BDAT_Atributo_Data_2=Data_Dia
        WHERE BDAT_Tabela.BDAT_Atributo_X=:OLD.BDAT_Atributo_X AND
              BDAT_Tabela.BDAT_Chave_1=:OLD.BDO_Chave_1 AND
              BDAT_Tabela.BDAT_Chave_2=:OLD.BDO_Chave_2 AND ...
              AND BDAT_Tabela.BDAT_Atributo_Data_2 is NULL;
        INSERT INTO BDAT_Tabela(BDAT_Chave_1, BDAT_Chave_2, ...
                               BDAT_Atributo_1, BDAT_Atributo_2, BDAT_Atributo_3, BDAT_Atributo_4, ...
                               BDAT_Atributo_Pre_1, BDAT_Atributo_Pre_2, BDAT_Atributo_Pre_3,
                               BDAT_Atributo_Pre_4, BDAT_Atributo_Data_1)
        VALUES(:OLD.BDO_Chave_1, OLD.BDO_Chave_2, ...
               :NEW.BDO_Atributo_1, :NEW.BDO_Atributo_2, ..., [Variavel_1, Variavel_2], ...
               :NEW.BDO_Atributo_Calc_1*:NEW.BDO_Atributo_Calc_2,
               :NEW.BDO_Atributo_Calc_1+:NEW.BDO_Atributo_Calc_2,
               :NEW.BDO_Atributo_Calc_1/:NEW.BDO_Atributo_Calc_2,
               :NEW.BDO_Atributo_Calc_1-:NEW.BDO_Atributo_Calc_2, Data_Dia);
    END IF;
END.
END.

```

Figura 4. Especificação Genérica do Gatilho.

A finalidade dessa especificação é oferecer um *template* genérico de um gatilho, que deverá servir de modelo para a codificação dos gatilhos junto ao BDO, que serão os responsáveis pela geração, transporte e armazenamento dos dados operacionais ao BDAT. Esses gatilhos deverão ser executados automaticamente pelo SGBD imediatamente após ocorrer alguma operação de inclusão ou alteração nos conjuntos de entidades e/ou atributos do BDO, selecionados para a geração dos dados analítico-temporais.

Para o armazenamento dos “elementos de tempo”, ou seja, dos períodos exatos em que o dado ficou válido no BDO, os atributos “BDAT_Atributo_Data_1” e “BDAT_Atributo_Data_2”, especificados no gatilho genérico devem ser utilizados, no sentido de controlar as épocas e/ou períodos em que ocorreram as mudanças nos valores dos dados operacionais. Vale destacar que durante as operações de inclusões (insert) dos

dados no BDAT, dos atributos que controlarão os períodos, somente o atributo “BDAT_Atributo_Data_1” será gerado e gravado. O atributo “BDAT_Atributo_Data_2” somente será gerado e gravado quando ocorrerem operações de alteração (update), no sentido de finalizar a validade do dado anteriormente incluído.

Para o armazenamento de valores pré-calculados, isto é, “atributos derivados”, por intermédio das operações de adição, multiplicação, subtração, divisão, e/ou combinações dessas operações matemáticas, provindos do BDO para o BDAT, devem ser utilizados os atributos “BDAT_Atributo_Pre_1”, “BDAT_Atributo_Pre_2”, “BDAT_Atributo_Pre_3” e “BDAT_Atributo_Pre_4”. Ver, na Figura 4, a especificação genérica do gatilho proposto como *template*.

3.2.2. Especificação Genérica do Procedimento

```

CREATE OR REPLACE PROCEDURE PROCEDURE_TEMPLATE
  (Variavel_Filtro_1 IN BDAT_Tabela.BDAT_Atributo_1%TYPE,
   Variavel_Filtro_2 IN BDAT_Tabela.BDAT_Atributo_2%TYPE, . . .)
IS
  Variavel_1 BDO_Tabela.BDO_Atributo_1%TYPE;
  Variavel_2 BDO_Tabela.BDO_Atributo_2%TYPE; . . .;
  CURSOR BDO_Tabela_CURSOR IS
    SELECT BDO_Chave_1, BDO_Chave_2, . . . , BDO_Atributo_1, BDO_Atributo_2, . . . ,
           Sum(BDO_Atributo_Calc_1*BDO_Atributo_Calc_2), Sum(BDO_Atributo_Calc_3, . . . )
      INTO Variavel_1,Variavel_2, . . . FROM BDO_Tabela
     WHERE BDO_Chave_1=Variavel_Filtro_1
       OR/AND . . . OR BDO_Atributo_1=Variavel_Filtro_2 OR/AND . . .
      GROUP BY BDO_Chave_1, BDO_Chave_2, . . . , BDO_Atributo_1, BDO_Atributo_2, . . .
BEGIN
  OPEN BDO_Tabela_CURSOR;
  LOOP
    FETCH BDO_Tabela_CURSOR INTO Variavel_1, Variavel_2, . . . ;
    EXIT WHEN BDO_Tabela_CURSOR%NOTFOUND;
    DELETE FROM BDAT_Tabela WHERE BDAT_Chave_1=Variavel_Filtro_1 OR/AND . . .
      OR BDAT_Atributo_1=Variavel_Filtro_2 OR/AND . . .
    INSERT INTO BDAT_Tabela (BDAT_Chave_1, BDAT_Chave_2, BDAT_Atributo_1,
                           BDAT_Atributo_2, BDAT_Atributo_3, . . . ) VALUES(Variavel_1,Variavel_2, . . . );
  END LOOP;
  COMMIT;
END.

```

Figura 5. Especificação Genérica do Procedimento Armazenado.

A finalidade dessa especificação é oferecer um *template* genérico de um procedimento armazenado que possibilitará a definição de “níveis de granularidade”, no sentido de totalizar dados operacionais e enviá-los ao BDAT, bem como gerar níveis maiores de agrupamento diretamente no BDAT, no sentido de diminuir o volume de dados com o passar do tempo. O procedimento genérico proposto para este passo está especificado na Figura 5.

Em suma, neste item foram especificados alguns subprogramas a fim de oferecer um padrão genérico para a realização prática da proposta.

4. Estudo de Caso

O BDO utilizado foi o do Sistema de Folha de Pagamento (BDO_FOLHA), do setor de RH da Fundação Educacional do Município de Assis, já que esse setor encontra muitas dificuldades quando lhe são solicitadas históricos para tomada de decisões e/ou auditorias, bem como uma simples recuperação de dados anteriormente modificados.

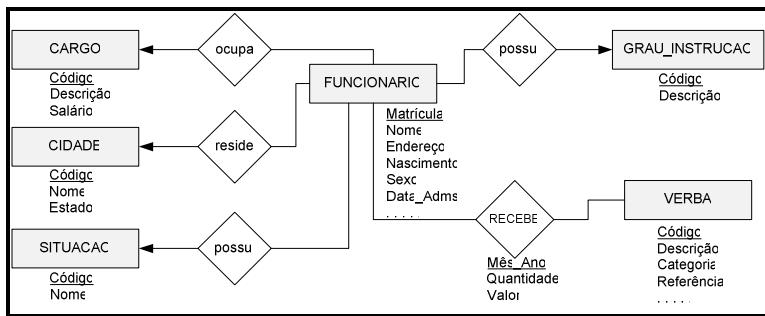


Figura 6. Parte do Modelo de Dados do Sistema de Folha de Pagamento (BDO_FOLHA)

4.1. Resultado da Aplicação da Etapa A

Após a aplicação dos passos da Etapa A, pôde-se chegar ao modelo de dados necessário para que se possa ter um controle razoável das modificações efetuadas junto ao BDO_FOLHA , conforme apresentado na Figura 7.

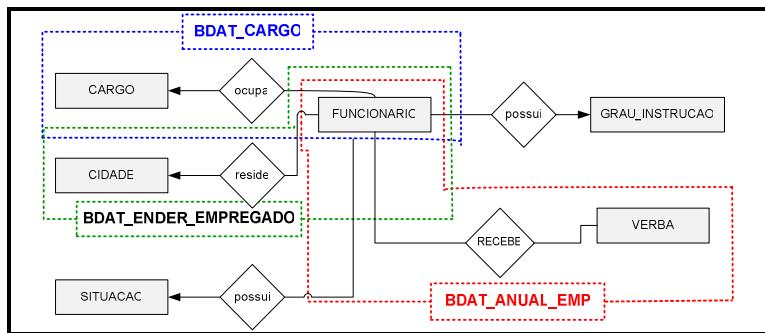


Figura 7. Modelo de Dados Analítico-Temporal

Os conjuntos de entidades (tabelas) BDAT_CARGO, BDAT_ANUAL_EMP e BDAT_ENDER_EMPREGADO são resultantes da modelagem, da aplicação da Etapa A. Essas, são as tabelas que formarão o Banco de Dados Analítico-Temporal.

4.2. Aplicação da Etapa B

Neste item, são apresentados dois exemplos, que foram testados e validados no SGBD Oracle 11g [Price 2009], porém essa proposta por ser aplicada em outros SGBD Relacionais Os construtores utilizados para ilustrar os modelos de dados apresentados na figuras foram retirados de [Silberschatz et al. 2006].

Exemplo 01: Aplicando o gatilho genérico.

A Figura 8 apresenta parte do Modelo de Dados Operacional, representado pelos conjuntos de entidades, FUNCIONARIO e CARGO, e parte do Modelo de Dados Analítico-Temporal, representado pelo conjunto de entidades BDTA_CARGO, obtido quando da aplicação da Etapa A .

Tendo como base esses conjuntos de entidades, foi elaborado o gatilho GAT_BDAT_CARGO, com base na especificação genérica da Figura 4, ilustrado na Figura 9, cujo principal objetivo é armazenar informações relacionadas aos cargos que um funcionário ocupou durante toda a sua vida na empresa, considerando-se também todos os períodos nos quais ocorreram essas mudanças. Os atributos TS_INICIAL e TS_FINAL destacam a adição dos elementos “tempo”. O atributo ID_USUARIO_BDO

tem por função, armazenar o usuário do banco de dados operacional que realizou a operação, para eventuais auditorias. Os atributos NOME_FUNC, DESCRIÇÃO_CARGO e VALOR_SALARIO, têm por função, retratar os valores exatos que um atributo assumiu ao longo do tempo, já que esses atributos podem sofrer alterações em seus valores operacionais. O gatilho ficará armazenado no BDO.

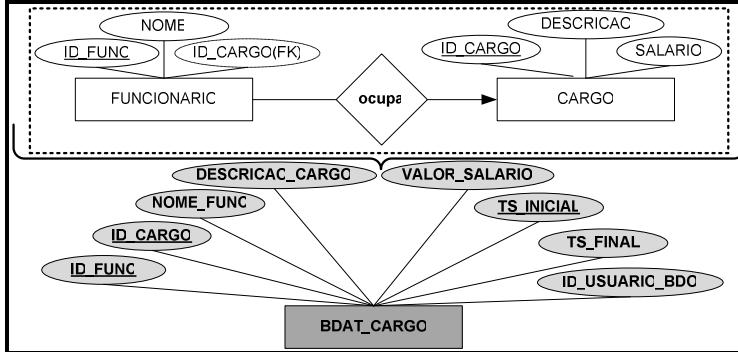


Figura 8. Conjuntos de Entidades utilizados para a criação de um Gatilho.

```

CREATE OR REPLACE TRIGGER GAT_BDAT_CARGO
BEFORE INSERT OR UPDATE OF ID_CARGO ON FUNCIONARIO
FOR EACH ROW
DECLARE
    V_DESCRICAO CARGO.DESCRIAO%TYPE;
    V_SALARIO CARGO.SALARIO%TYPE;
    V_USER_NAME USER_USERS.USERNAME%TYPE;
BEGIN
    SELECT DESCRICAO, SALARIO INTO V_DESCRICAO, V_SALARIO
    FROM CARGO WHERE ID_CARGO=:NEW.ID_CARGO;
    SELECT USERNAME INTO V_USER_NAME
    FROM USER_USERS;
    IF INSERTING THEN
        INSERT INTO BDAT_CARGO@BDAT_LINK(ID_FUNC, NOME_FUNC, ID_CARGO,
        DESCRICAO_CARGO, VAL_SALARIO, TS_INICIAL, ID_USUARIO_BDO)
        VALUES(:NEW.ID_FUNC, :NEW.NOME, :NEW.ID_CARGO, V_DESCRICAO,
        V_SALARIO, SYSDATE, V_USER_NAME);
    ELSE
        UPDATE BDAT_CARGO@BDAT_LINK SET TS_FINAL=SYSDATE
        WHERE BDAT_CARGO.ID_CARGO=:OLD.ID_CARGO AND
        BDAT_CARGO.ID_FUNC=:OLD.ID_FUNC AND BDAT_CARGO.TS_FINAL IS NULL;
        INSERT INTO BDAT_CARGO@BDAT_LINK (ID_FUNC, NOME_FUNC, ID_CARGO,
        DESCRICAO_CARGO, VAL_SALARIO, TS_INICIAL, ID_USUARIO_BDO,)
        VALUES(:OLD.ID_FUNC, :OLD.NOME, :NEW.ID_CARGO, V_DESCRICAO,
        V_SALARIO, SYSDATE, V_USER_NAME);
    END IF;
END;
OBS: As siglas :NEW e :OLD possibilitam obter os valores posteriores/anteriores de um atributo.

```

Figura 9. Ilustração Prática do Gatilho Genérico (GAT_BDAT_CARGO)

Exemplo 02: Aplicando o procedimento genérico.

A Figura 10 apresenta parte do Modelo de Dados Operacional, representado pelos conjuntos de entidades FUNCIONARIO e VERBA, bem como pelo conjunto de relacionamentos RECEBE, e parte do Modelo de Dados Analítico-Temporal, representado pelo conjunto de entidades BDAT_ANUAL_FUNC, ao qual demonstra a aplicação da granularidade, já que os recebimentos são agrupados e totalizados por funcionário, verba e ano. O atributo TOTAL se refere a soma do atributo VALOR, e o atributo QTDE a soma do atributo QUANTIDADE.

Tendo como base esses conjuntos de entidades e relacionamentos, foi elaborado o procedimento armazenado ilustrado na Figura 11. Assim, o procedimento armazenado PROC_BDAT_ANUAL_FUNC é um exemplo prático de como armazenar dados sumarizados, aplicando-se níveis de granularidade. Diferentemente dos gatilhos, sua

execução deverá ser agendada para ocorrer nos períodos de menor utilização desse ambiente, como, por exemplo, nas madrugadas ou em finais de semana.

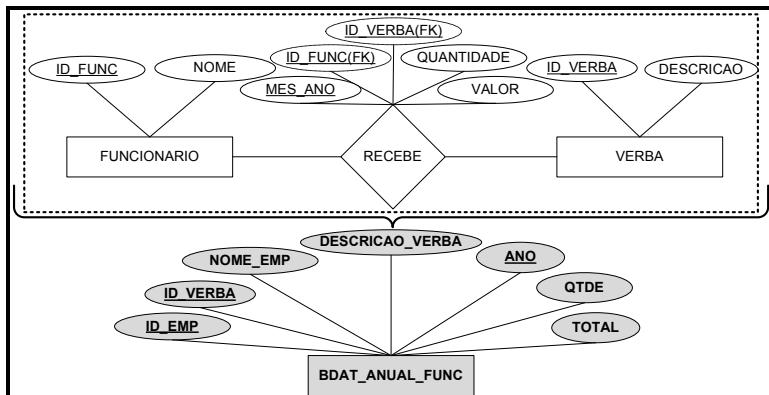


Figura 10. Conjuntos de Entidades utilizados para a criação de um Procedimento.

Os exemplos apresentados foram elaborados com base em um banco de dados que armazena informações sobre uma Folha de Pagamento, porém, esse modelo pode ser aplicado em outras áreas, tais como, telefonia, bancária, acadêmica, dentre outras, no sentido de armazenar históricos de dados para auxiliar no processo de tomada de decisões, auditorias e recuperação de dados.

```

CREATE OR REPLACE PROCEDURE PROC_BDAT_ANUAL_FUNC
(I_ANO IN CHAR)
IS
V_ID_FUNC RECEBE.MATRICULA%TYPE;
V_ID_VERBA RECEBE.VERBA%TYPE;
V_NOME_FUNC FUNCIONARIO.NOME%TYPE;
V_DESCRICAO_VERBA VERBA.DESCRICAO%TYPE;
V_ANO CHAR(4);
V_VALOR RECEBE.VALOR%TYPE;
V_QTDE RECEBE.QUANTIDADE%TYPE;
CURSOR BDO_ANUAL_FUNC_CURSOR IS
SELECT FUNCIONARIO.MATRICULA,FUNCIONARIO.NOME,VERBA.CODIGO,
VERBA.DESCRICAO, SUBSTR(TO_CHAR(MES_ANO),3,4),SUM(VALOR),
SUM(QUANTIDADE) INTO V_ID_FUNC,V_NOME_FUNC,V_ID_VERBA,
V_DESCRICAO_VERBA,V_ANO,V_VALOR,V_QTDE
FROM FUNCIONARIO,VERBA,RECEBE
WHERE FUNCIONARIO.MATRICULA=RECEBE.MATRICULA AND
VERBA.CODIGO=RECEBE.VERBA AND
SUBSTR(TO_CHAR(MES_ANO),3,4)=I_ANO
GROUP BY FUNCIONARIO.MATRICULA,FUNCIONARIO.NOME,VERBA.CODIGO,
VERBA.DESCRICAO, SUBSTR(TO_CHAR(MES_ANO),3,4);
BEGIN
DELETE FROM BDAT_ANUAL_FUNC@BDAT_LINK WHERE ANO=I_ANO;
OPEN BDO_ANUAL_FUNC_CURSOR;
LOOP
FETCH BDO_ANUAL_FUNC_CURSOR INTO V_ID_FUNC,V_NOME_FUNC,
V_ID_VERBA,V_DESCRICAO_VERBA,V_ANO,V_VALOR,V_QTDE;
EXIT WHEN BDO_ANUAL_FUNC_CURSOR%NOTFOUND;
INSERT INTO BDAT_ANUAL_FUNC@BDAT_LINK
VALUES(V_ID_FUNC,V_ID_VERBA,V_NOME_FUNC,V_DESCRICAO_VERBA,
V_ANO,V_VALOR,V_QTDE);
END LOOP;
COMMIT;
END;
/

```

Figura 11. Especificação Prática do Procedimento Genérico

5. Considerações Finais

Este trabalho possibilita a modelagem e a implementação de um BDAT, de grande valia para o processo de geração de históricos de dados, sem a necessidade de se adquirir onerosas ferramentas de *software*, já que as rotinas necessárias são implementadas por meio de gatilhos e procedimentos armazenados, recursos oferecidos pelos próprios SGBD. Das duas etapas elaboradas, acredita-se que a Etapa B é a que apresenta maior importância, visto que é nessa etapa que são especificados, genericamente, gatilhos e procedimentos armazenados. Praticamente, há na literatura disponível poucos estudos

acerca do que foi elaborado na referida etapa. Espera-se, portanto, que essas especificações possam auxiliar na implementação de BDAT. Vale reforçar, que a estrutura de dados dos BDO não é afetada, sendo necessária apenas a inserção de gatilhos e procedimentos armazenados.

Referências

- BRYLA, Bob; LONEY, Kevin. “Oracle Database 11g Manual do DBA”, 2008.
- CORDEIRO, Robson Leonardo Ferreira; SANTOS, Clesio Saraiva dos; EDELWEISS, Nina; GALANTE, Renata de Matos. Classificação de restrições de integridade em bancos de dados temporais de versões. Anais/Proceedings SBBD’2004. Brasilia, p. 336-337, 2004.
- GOLFARELLI, Mateo; MAIO, Dario; RIZZI, Stefano. Conceptual Design of Data Warehouses from E/R Schemes. In: Hawaii International Hierarquias Conference on Systems Sciences, 1998, Hawaii. Proceedings. Hawaii, 1998. 10 p.
- GRANDI, Fabio; MANDREOLI, Federica; TIBERIO, Paolo; BERGONZINI, Marco. A temporal data model and management system for normative texts in XML format. In: International Workshop on Web Information and Data Management, 1, 2003, New Orleans, USA. ACM - WIDM’03. New Orleans, 2003. p. 29-36.
- ITALIANO, Isabel C.; FERREIRA, João Eduardo. “Synchronization Options for Data Warehouse Designs”, Publicado na IEEE Computer Magazine, Revista de IEEE Computer Society, 2006.
- MOODY, Daniel L.; KORTINK, Mark A. R. From enterprise models to dimensional models: a methodology for data warehouse and data mart design. In: International Workshop on Design and Management of Data Warehouse, Stockholm, 28., p.2, 2000, p. 1-12.
- POLETTI, Alex S. R. de S.; ALMEIDA JUNIOR, Jorge Rady de. “Modeling of an Analytical Database System”, 9^a International Conference on Enterprise Information Systems - ICEIS’2007, Ilha da Madeira, Portugal, Funchal, 13 – 16 de Jun., 2007.
- PRICE, Jason. Oracle Database 11g SQL: Domine SQL e PL/SQL no banco de dados Oracle. Aborda as versões 11g, 10g, 9i e 8i. Porto Alegre. Editora Bookman, 2009.
- SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSCHAN, S. “Sistemas de Bancos de Dados”. 5^a edição - Rio de Janeiro. Editora: Elsevier, 2006.
- SPRAGUE, Ralph H. Jr.; WATSON, Hugh J. “Sistema de apoio à decisão: colocando a teoria em prática”, 2^a edição – Rio de Janeiro. Editora: Campus, 1991.
- TANSEL, Abdullah Uz. Temporal relational data model. Revista IEEE Computer Society (IEEE Transactions on Knowledge e Data Engineering), v.9, n.3, may/june, p. 464-479, 1997.
- WANG, Fusheng; ZHOU, Xin; ZANILOLO, Carlo. Bridging relational database history and the web: the XML approach. In: Workshop on Web Information and Data Management. Proceedings of the eighth ACM international workshop on Web information and data management - ACM - WIDM’06. Arlington, Virginia, USA, 2006. p. 3-10.

Mapeamento Automático de Modelos de Dados XML Temporais Ad-hoc para um Modelo de Dados XML Temporal Padrão

Edimar Manica¹, Renata Galante¹, Carina F. Dorneles²

¹Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

²Departamento de Informática e Estatística - Universidade Federal de Santa Catarina (UFSC)
Campus Universitário Trindade - 88049-900 - Florianópolis - SC - Brasil

{emanica,galante}@inf.ufrgs.br, dorneles@inf.ufsc.br

Resumo. Grande parte dos documentos XML, que possui informações históricas, não segue um modelo de dados temporal, ou seja, são documentos que refletem um modelo temporal ad-hoc que não possui controle das restrições de tempo nem suporta linguagens de consulta temporal. Este trabalho apresenta uma proposta de mapeamento automático de modelos de dados XML temporais ad-hoc para um modelo de dados XML temporal padrão, a fim de permitir a realização de consultas temporais. Dentre as contribuições, destacam-se o mapeamento automático e a implementação do protótipo AXPath para anotação de caminhos temporais. Experimentos iniciais mostram que o protótipo AXPath apresenta melhores resultados de precisão que os trabalhos relacionados.

Abstract. Most XML documents, which have historical information, do not follow a temporal data model, i.e., they are documents that reflect an ad-hoc temporal model that have neither time constraints control nor temporal query languages support. This paper presents a proposal for automatic mapping of XML ad-hoc temporal data models to a standard temporal data model, in order to allow temporal queries. Among the contributions, we highlight the automatic mapping and the implementation of a prototype, called AXPath, for annotating temporal paths. Our initial experiments show that the AXPath prototype produces better results of precision than the related works.

1. Introdução

A representação de aspectos temporais em sistemas de informação tem assumido um papel bastante significativo ao longo dos anos [Silva e Edelweiss 2001]. Vários modelos de dados temporais, com suas linguagens de consulta, têm sido propostos para banco de dados, tais como HRDM (*Historical Relational Data Model*) [Clifford e Croker 1987], TRDM (*Temporal Relational Data Model*) [Tansel 1997] e TF-ORM (*Temporal Functionality in Objects With Roles Model*) [Edelweiss et al. 1993].

Atualmente, a grande maioria das aplicações está migrando para a Web e o XML está sendo amplamente utilizado para integração dessas informações, tanto pela comunidade científica quanto pelo mercado empresarial. Com isso, surge a necessidade de controlar e acessar de maneira eficaz o aspecto temporal da informação, agora presente

em documentos XML. Para esta finalidade, foram definidos modelos de dados temporais para XML e suas respectivas linguagens de consulta, permitindo a representação de dados temporais e a realização de consultas, abstraindo a implementação da temporalidade [Gao e Snodgrass 2003, Rizzolo e Vaisman 2008]. Neste artigo, estes modelos são chamados de modelos de dados temporais padrão. Apesar de existirem na literatura modelos de dados temporais padrão para XML, uma grande parcela dos documentos XML que possui informações temporais não segue um modelo de dados temporal padrão. Isto significa que as características temporais são modeladas de forma *ad-hoc*, sem nenhum tratamento para a temporalidade. Neste artigo, esses modelos são denominados modelos de dados temporais *ad-hoc*. Cabe observar que os modelos temporais *ad-hoc* não oferecem suporte a linguagens de consulta temporal e o controle das restrições temporais é realizado exclusivamente pela aplicação.

Assim como em banco de dados temporais, a informação temporal presente nos documentos XML pode ser classificada em três tipos de tempo: (i) tempo de validade, tempo durante o qual um fato do banco de dados é verdadeiro na realidade modelada; (ii) tempo de transação, tempo em que o fato é armazenado; ou (iii) tempo definido pelo usuário, valor cujas propriedades temporais são definidas explicitamente pelos usuários e manipuladas pelos programas de aplicação, além disso, não possui suporte de uma linguagem de consulta especial, ao contrário do tempo de validade e de transação. Além disso, a informação temporal pode estar representada de três formas (rótulos temporais): (i) ponto no tempo, um determinado instante na linha de tempo; (ii) período temporal, um determinado intervalo na linha de tempo; ou elemento temporal, união finita de períodos temporais [Dyreson et al. 1994]. Uma descrição, em nível conceitual, das informações temporais (tais como, um ponto no tempo ou um período temporal) presentes no conteúdo de documentos (páginas Web, documentos XML, etc.) é conhecida como entidade temporal. Além disso, uma sequência de tokens que representa uma instância de uma entidade temporal é chamada de expressão temporal [Alonso et al. 2007].

Na Figura 1 são mostrados três exemplos de documentos XML que possuem características temporais modeladas de forma *ad-hoc*. O Doc A e o Doc B modelam informações sobre vacinação, contendo a cidade onde a vacina foi aplicada, o nome da vacina, o cidadão que foi vacinado, o agente que aplicou a vacina e a data da aplicação da vacina. O Doc C modela informações sobre internações, contendo o nome do paciente internado, a cidade onde ele foi internado e o período de internação. No Doc A, a informação temporal é um ponto no tempo, que é representado através do elemento `data` (linhas 9 e 15). No Doc B, a informação temporal também é um ponto no tempo, porém é representada utilizando três elementos (`dia` - linha 9, `mes` - linha 10 e `ano` - linha 11), ou seja, a data está fragmentada. No Doc C, a informação temporal é um período temporal, sendo seu início representado pelo elemento `baixa` (linha 6) e seu término representado pelo elemento `alta` (linha 7). Estes exemplos são utilizados ao longo do artigo para explicar características deste trabalho.

O objetivo deste trabalho é apresentar uma proposta de mapeamento automático de documentos XML que seguem modelos de dados temporais *ad-hoc* para documentos XML que seguem um modelo de dados temporal padrão. Este mapeamento possibilita a realização de consultas temporais que abstraiam a implementação da temporalidade. Para alcançar este objetivo, define-se, neste trabalho, o conceito de caminho temporal

<pre> 01. <?xml version="1.0"?> 02. <posto> 03. <cidade>Soledade</cidade> 05. <vacinas> 06. <vacina> 07. <nome>Febre Amarela</nome> 08. <pessoa>Edimar Manica</pessoa> 09. <data>12/03/1997</data> 10. <agente>Beltrano</agente> 11. </vacina> 12. <vacina> 13. <nome>Febre Amarela</nome> 14. <pessoa>Edimar Manica</pessoa> 15. <data>28/01/2009</data> 16. <agente>Ciclano</agente> 17. </vacina> 18. </vacinas> 19. </posto> </pre> <p style="text-align: center;">Doc A</p>	<pre> 01. <?xml version="1.0"?> 02. <posto cidade="Marau"> 03. <vacina> 04. <nome>Febre Amarela 05. </nome> 06. <pessoa>Edimar Manica 07. </pessoa> 08. <aplicacao> 09. <dia>12</dia> 10. <mes>01</mes> 11. <ano>1998</ano> 12. </aplicacao> 13. <agente>Ciclano 14. </agente> 15. </vacina> 16. </posto> </pre> <p style="text-align: center;">Doc B</p>	<pre> 01. <?xml version="1.0"?> 02. <posto cidade="Passo Fundo"> 03. <internacao> 04. <paciente>Edimar Manica 05. </paciente> 06. <baixa>23/11/2007</baixa> 07. <alta>24/11/2007</alta> 08. </internacao> 09. </posto> </pre> <p style="text-align: center;">Doc C</p>
--	---	--

Figura 1. Exemplos de documentos XML com características temporais modeladas de forma *ad-hoc*.

como um caminho (*path*) XML que vai da raiz até um nodo folha, ou atributo, que possui como valor uma expressão temporal. A fim de realizar a identificação destes caminhos no documento XML, foi implementado o protótipo AXPath (*Annotation XML Paths - Anotação de Caminhos XML*). As contribuições deste trabalho incluem:

1. especificação do mapeamento de documentos XML com características temporais modeladas de forma *ad-hoc* para documentos XML seguindo um modelo temporal padrão;
2. definição e implementação do protótipo AXPath;
3. realização de experimentos iniciais que mostram que o protótipo AXPath aplicado para anotação de caminhos temporais gera resultados melhores de precisão que os produzidos pelos trabalhos relacionados.

Este artigo está organizado como segue. A Seção 2 descreve os trabalhos relacionados. Na Seção 3, é apresentada uma visão geral da proposta de mapeamento, que é posteriormente detalhada na Seção 4. Os experimentos e resultados iniciais são apresentados na Seção 5. Finalmente, na Seção 6, são descritas as considerações finais e as direções futuras.

2. Trabalhos Relacionados

Existem trabalhos que visam identificar automaticamente informações temporais em documentos texto não estruturados, tais como [Alonso et al. 2007, Alonso e Gertz 2006]. Outros trabalhos definem modelos de dados temporais para XML [Rizzolo e Vaisman 2008, Di Vimercati 2002, Gao e Snodgrass 2003]. Diferente destes trabalhos, este artigo visa unificar estas duas linhas de pesquisa estendendo a identificação de informações temporais em documentos não estruturados para documentos XML. Esta identificação é necessária para possibilitar o mapeamento proposto.

Dentre os trabalhos que visam identificar automaticamente informações temporais em documentos não estruturados destacam-se as ferramentas GUTime [GUTime 2009] e ANNIE [ANNIE 2009]. GUTime é uma ferramenta de código aberto especificada para anotação temporal de documentos texto não estruturados. Esta ferramenta gera como resultado um documento XML com as informações temporais anotadas e normalizadas. ANNIE é uma ferramenta de código aberto para extração de informação em documentos texto não estruturados. Além de informação temporal, ANNIE também permite extrair

informações de localização, pessoas, esportes, etc. Esta ferramenta gera como resultado um arquivo XML com as anotações das informações extraídas. Porém, não normaliza as expressões temporais anotadas. O diferencial entre a identificação das informações temporais realizada por GUTime e ANNIE com a identificação realizada pelo trabalho descrito neste artigo está no fato de que a proposta apresentada no presente trabalho trata a anotação semântica dos dados XML, representada pelas *tags*, a fim de obter melhores resultados. Além disso, GUTime e ANNIE identificam e anotam cada expressão temporal isoladamente, enquanto a proposta apresentada neste artigo anota caminhos temporais.

Analisando os trabalhos que propõem modelos de dados temporais para documentos XML destaca-se [Rizzolo e Vaisman 2008], que adiciona a dimensão temporal para documentos XML considerando-os como grafos consistindo de arestas rotuladas com intervalos temporais. Além disso, apenas o tempo de transação é suportado, porém os autores citam que o tempo de validade poderia ser adicionado de forma análoga. Também, é descrita uma linguagem de consulta temporal para XML, denominada TXPath, que estende XPath 2.0, bem como é introduzida uma linguagem para alterações.

Destacam-se alguns outros trabalhos que abordam modelos de dados temporais em XML. Em [Di Vimercati 2002] é descrito um modelo de dados temporal que permite representar tanto o tempo de transação quanto o tempo de validade, além disso, descreve um modelo de autorização sobre o modelo temporal. Em [Gao e Snodgrass 2003] é apresentada uma linguagem de consulta temporal para XML, denominada TXQuery, que adiciona suporte ao tempo de validade para XQuery estendendo a sintaxe e a semântica de XQuery, bem como descreve o mapeamento de consultas em TXQuery para consultas em XQuery convencional.

3. TeXKeySearch: Visão Geral

O trabalho apresentado neste artigo faz parte de uma proposta de um método de recuperação de informação, denominado TeXKeySearch [Manica e Galante 2009]. O objetivo principal do TeXKeySearch é permitir a realização de consultas por palavras-chave em documentos XML considerando a anotação semântica dos dados e as características temporais da consulta e dos documentos. Este método apresenta as seguintes vantagens: **(i)** usuário não precisa conhecer o esquema do XML; **(ii)** usuário não precisa conhecer nenhuma linguagem de processamento de consulta XML; e **(iii)** características temporais da consulta e dos documentos são exploradas para melhorar a busca. Alguns cenários onde este método pode ser aplicado são descritos em [Manica e Galante 2009].

A Figura 2 apresenta uma visão geral do TeXKeySearch. O usuário informa uma consulta composta por palavras-chave e expressões temporais. São identificados os dados relevantes através das palavras-chave e do reconhecimento das expressões temporais presentes na consulta. Essa identificação é feita através do acesso a um dicionário, sendo os documentos XML mapeados para um modelo de dados temporal padrão. Por fim, os dados relevantes são retornados para o usuário e exibidos através de um agrupamento temporal. Este agrupamento temporal consiste em identificar quando um mesmo evento ocorreu várias vezes e mostrá-lo apenas uma vez, associando a ele seus instantes/ períodos de ocorrência. Por exemplo, considere que uma determinada pessoa fez a vacina da Gripe em 1987 e refez em 1999. Em vez de mostrar duas vezes a aplicação da vacina da Gripe nesta pessoa, mostra-se uma única vez indicando as duas datas.

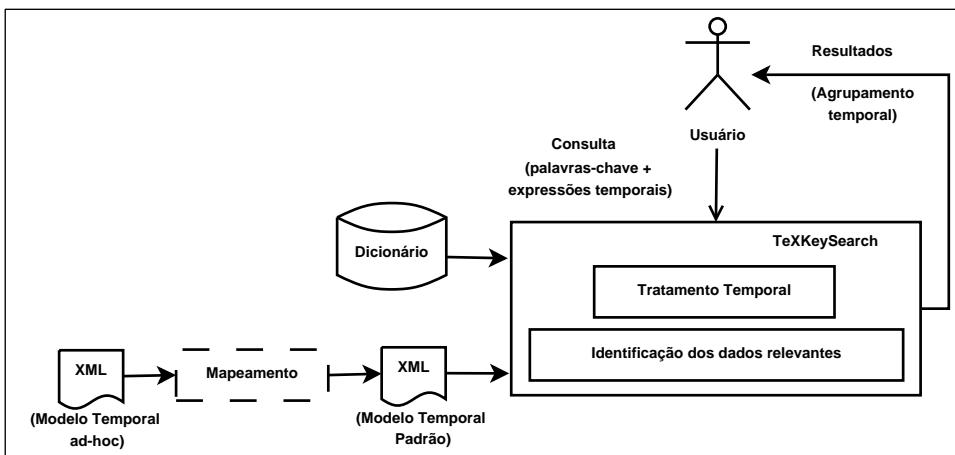


Figura 2. Visão Geral do TeXKeySearch.

O trabalho apresentado neste artigo foca no processo pontilhado observado na Figura 2, que descreve o mapeamento automático de documentos XML, onde o tempo é representado de forma *ad-hoc* para documentos XML que seguem um modelo temporal padrão. O mapeamento proposto possibilita a exploração temporal da consulta sobre os documentos XML pelo método TeXKeySearch. Além disso, o mapeamento permite a realização de consultas temporais através da linguagem de processamento de consulta temporal para XML definida pelo próprio modelo de dados temporal padrão.

4. Mapeamento Temporal

O objetivo do mapeamento temporal é transformar documentos XML que possuem dados temporais modelados de forma *ad-hoc* em documentos XML que seguem um modelo de dados temporal padrão. A Figura 3 mostra a arquitetura deste mapeamento temporal, que compreende 5 procedimentos.

O primeiro procedimento é a identificação de caminhos temporais. O conceito de caminho temporal é importante, pois permite agrupar as expressões temporais de acordo com o caminho dos nodos que as contém. Os caminhos temporais dos documentos de exemplo apresentados na Figura 1 são descritos na Tabela 1, onde se observa que para o Doc A são agrupadas duas expressões temporais em um mesmo caminho temporal. Dentre os benefícios da anotação de caminhos temporais em vez da anotação de expressões temporais isoladas destacam-se: **(i)** a possibilidade de desambiguação de formato. Por exemplo, anotando isoladamente a primeira expressão temporal do Doc A (12/03/1997) não haveria como descobrir se o formato é dia/mês/ano ou mês/dia/ano. No entanto, com a anotação do caminho temporal (/posto/vacinas/vacina/data) é possível descobrir que /posto/vacinas/vacina/data/'12/03/1997' está no formato dia/mês/ano, pois uma das expressões temporais do mesmo caminho temporal (28/01/2009) possui valor para o dia maior que 12, que é o maior valor para mês¹; **(ii)** o tamanho do arquivo

¹Ressalta-se que haverá casos onde não haverá nenhuma expressão temporal em um caminho temporal com valor do dia maior que 12, necessitando assim de outra regra para a desambiguação de formato deste caminho temporal. Porém, acredita-se que esta situação não ocorrerá na maioria dos documentos XML orientado a dados com um tamanho considerável que torne uma consulta temporal necessária.

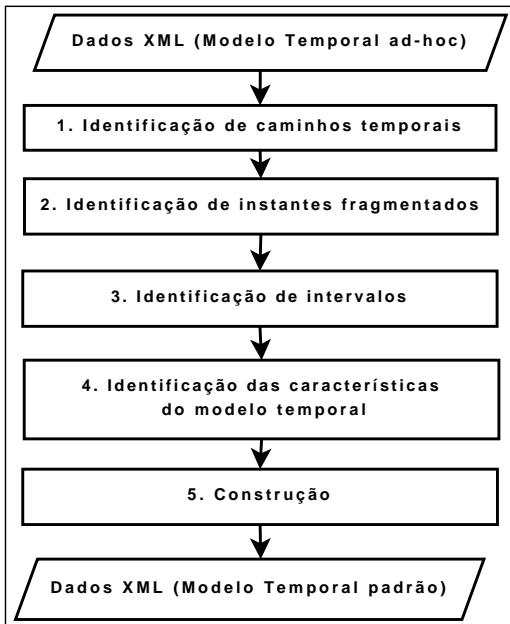


Figura 3. Arquitetura do processo de mapeamento temporal.

de anotação é menor (ou igual caso para cada caminho temporal haja apenas uma expressão temporal no documento); (iii) a adição de uma nova sub-árvore no XML seguindo a mesma estrutura das sub-árvores presentes no documento não exigirá nenhuma alteração na anotação, uma vez que seus caminhos temporais já estarão anotados. Ressalta-se que para cada caminho temporal é anotado seu formato e sua granularidade.

Tabela 1. Caminhos Temporais dos Documentos de Exemplo.

Doc	Caminhos Temporais	Expressões Temporais	Granularidade	Formato
A	/posto/vacinas/vacina/data	12/03/1997, 28/01/2009	data	dia/mês/ano
B	/posto/vacina/aplicacao/dia	12	dia	dia
B	/posto/vacina/aplicacao/mes	01	mês	mês
B	/posto/vacina/aplicacao/ano	1998	ano	ano
C	/posto/internacao/paciente/baixa	23/11/2007	data	dia/mês/ano
C	/posto/internacao/paciente/alta	24/11/2007	data	dia/mês/ano

O segundo procedimento é a identificação de instantes fragmentados. Caminhos temporais filhos de um mesmo nodo pai, sendo um no formato dia, outro no formato mês e outro no formato ano compõem um instante fragmentado. Por exemplo, os três caminhos temporais do Doc B (Tabela 1).

O terceiro procedimento é a identificação de intervalos temporais. Caminhos temporais filhos de um mesmo nodo pai e de mesmo formato compõem um intervalo temporal. Por exemplo, os dois caminhos temporais do Doc C (Tabela 1).

O quarto procedimento é a identificação de características do modelo temporal, tais como: tipo de tempo, rótulo temporal, etc. Se o nome do elemento folha que compõem o caminho temporal se referir ao documento XML (data_criacao, por exemplo) e não a realidade modelada no XML

(`data_contratacao`, por exemplo), então considera-se que o caminho temporal representa o tempo de transação, caso contrário o tempo de validade. Para isto, será criada uma lista de possíveis rótulos que se referem ao tempo de transação. Se o caminho temporal faz parte de um intervalo, então seu rótulo temporal é período temporal, caso contrário é instante. Elementos temporais não são tratados neste artigo devido a sua difícil aplicação.

O quinto procedimento é a construção que gera um novo documento XML seguindo o modelo de dados temporal padrão. Este procedimento é realizado através das informações identificadas nos procedimentos anteriores.

A próxima subseção descreve o procedimento de identificação de caminhos temporais que representa a principal contribuição deste artigo. Cabe ressaltar que o mapeamento proposto é independente do modelo temporal padrão escolhido, pois o importante é a identificação das características temporais no documento XML. O modelo apenas servirá para fins de padronização das informações temporais a fim de permitir consultas temporais que abstraiam a implementação da temporalidade. A utilização de um modelo ou outro apenas implicará no tipo de tempo (tais como, tempo de validade e tempo de transação) que poderá ser representado e na forma como o tempo é representado no XML (através de atributos ou elementos, por pontos no tempo, intervalos temporais ou elementos temporais).

4.1. Identificação de Caminhos Temporais

O procedimento de identificação de caminhos temporais é realizado através do protótipo AXPath (*Annotation XML Paths - Anotação de Caminhos XML*). A Figura 4 mostra a arquitetura do AXPath, onde o documento XML a ser anotado e um arquivo XML de configuração são entradas para uma consulta XQuery que gera um novo arquivo XML que contém os caminhos temporais e as informações sobre estes caminhos. O arquivo de configuração consiste em um documento XML que define regras genéricas sobre as tags, seus conteúdos e seus atributos para inferência dos caminhos a serem anotados. Este documento deve ser configurado de acordo com sua aplicação. Neste trabalho ele foi aplicado para anotação de caminhos temporais.

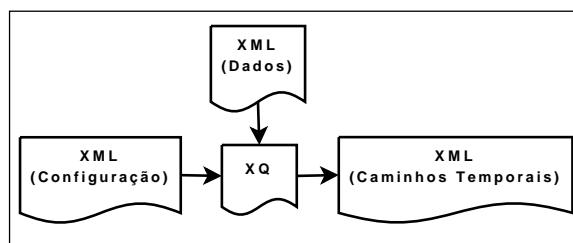


Figura 4. Arquitetura do protótipo AXPath.

A Figura 5 apresenta o DTD do arquivo de configuração, onde destacam-se as definições de candidatos (elemento `candidates` - linha 3), de eliminações (elemento `eliminations` - linha 7) e de anotações (elemento `annotations` - linha 8). Os candidatos definem expressões regulares que indicam possíveis caminhos temporais. As eliminações definem condições de eliminação de caminhos temporais candidatos. As anotações formam uma árvore de decisão tendo como nodo folha a previsão de anotação da expressão temporal (elemento `return` - linha 20) e os demais nodos sendo condições para a previsão (elemento `condition` - linha 11).

A previsão de anotação inclui a definição da granularidade do caminho temporal (elemento `gran` - linha 23), por exemplo: `data`; e do formato (elemento `format` - linha 22), por exemplo: `dia/mês/ano`. Para cada condição é necessário informar no atributo `test` (linha 13) se ela será realizada sobre o rótulo (nome do nodo folha ou do nodo atributo que é pai do nodo texto que representa uma expressão temporal), representado pela opção `tag`; ou sobre os valores (expressões temporais pertencentes ao caminho temporal), representados pela opção `values`. Caso a condição seja realizada sobre os valores é necessário informar a abrangência (atributo `abrag` - linha 18). Há três valores permitidos para a abrangência: **(i)** `some`, a condição é verdadeira pelo menos para uma expressão temporal do caminho temporal; **(ii)** `every`, a condição é verdadeira para todas as expressões temporais do caminho temporal; e **(iii)** `none`, a condição é falsa para todas as expressões temporais do caminho temporal. O predicado da condição é uma operação binária, onde o primeiro operando é definido por uma função e seus parâmetros (respectivamente os atributos `op1_f` e `op1_args` - linhas 14-15); o operador (atributo `operator` - linha 17) pode ser igual (`equals`), menor que (`lessthan`) ou maior que (`greaterthan`); e o segundo operando é um valor definido no atributo `op2` (linha 16).

1. <!ELEMENT cfg ((candidates, eliminations, annotations))> 2. 3. <!ELEMENT candidates ((candidate+)> 4. <!ELEMENT candidate EMPTY> 5. <!ATTLIST candidate regex CDATA #IMPLIED> 6. 7. <!ELEMENT eliminations ((condition+)> 8. <!ELEMENT annotations ((conditions))> 9. <!ELEMENT conditions ((condition+))> 10. 11. <!ELEMENT condition ((condition+ return)?> 12. <!ATTLIST condition	13. test (values tag) #REQUIRED 14. op1_f CDATA #REQUIRED 15. op1_args CDATA #IMPLIED 16. op2 CDATA #IMPLIED 17. operator (equals lessthan greaterthan) #IMPLIED 18. abrag (some none every) #IMPLIED> 19. 20. <!ELEMENT return EMPTY> 21. <!ATTLIST return 22. format CDATA #REQUIRED 23. gran CDATA #REQUIRED>
---	---

Figura 5. DTD do Arquivo de Configuração.

5. Experimentos

Para avaliar a qualidade da anotação de caminhos temporais realizados pelo AXPath, foram comparados seus resultados com os resultados de ANNIE e GUTime. Foram testadas duas métricas para comparar essas três técnicas: revocação e precisão. A revocação mede o percentual de caminhos temporais relevantes que foram identificados e a precisão mede o percentual de caminhos temporais identificados que são relevantes [Baeza-Yates e Ribeiro-Neto 1999]. Um caminho temporal relevante é um caminho XML que foi manualmente identificado como um caminho XML que possui alguma informação temporal. Nos testes, foram usadas duas bases de dados XML: Mondial e WSU². Mondial é uma base de dados geográficos resultante da integração das bases do CIA World Factbook, do International Atlas e do Terra, entre outras fontes. WSU é uma base de dados que descreve cursos. Cabe ressaltar que não foram utilizadas as bases inteiras, apenas 80,8 KB da base de dados Mondial e 370 KB da base de dados WSU. Na base Mondial foram selecionados os primeiros 20 elementos filhos do elemento raiz com seus descendentes. Como todos os elementos filhos da raiz eram `course` e tinham a mesma estrutura, esta seleção não implicou na perda de nenhum caminho temporal. Na base WSU foram selecionados os primeiros 855 elementos filhos do nodo raiz. Com esta seleção ficaram de fora

²Disponíveis em <http://www.cs.washington.edu/research/xmldatasets/>.

7 diferentes elementos (`organization`, `mountain`, `desert`, `island`, `river`, `sea` e `lake`), dos quais apenas `organization` possui informações temporais. No entanto, `organization` possui apenas um caminho temporal e seu formato é o mesmo de um caminho temporal incluído nos dados selecionados. Com isso, a não inclusão do elemento `organization` nos dados selecionados não afeta o experimento. As duas bases geradas após a seleção possuíam três caminhos temporais que foram identificados manualmente.

Também, vale observar que pelo fato de GUTime e ANNIE não terem sido projetadas para documentos XML, antes de submeter os documentos XML a estas ferramentas foi necessário transformá-los em documentos texto, no qual os nomes de elementos, os nomes de atributos, os valores de elementos e os valores de atributos foram apresentados como simples palavras de um texto. Por exemplo, o fragmento XML “`d`” seria substituído por “`a b c d a`”. Isto foi necessário, pois GUTime não aceita como entrada um documento XML e ANNIE remove as *tags* quando um documento no formato XML é submetido. Além disso, como GUTime e ANNIE não possuem o conceito de caminho temporal, considerou-se como identificação de um caminho temporal se eles identificaram pelo menos uma expressão temporal que pertencesse ao caminho temporal.

A Tabela 2 mostra os resultados dos experimentos onde é possível notar que AX-Path apresentou em ambas as bases uma melhor precisão. Para a base WSU todos os caminhos identificados como temporais pelo AXPath realmente eram caminhos temporais (precisão 1,00). Porém, ele apresentou uma revocação baixa (0,33), uma vez que no arquivo de configuração não foi considerado anotação de horário e os caminhos temporais não identificados para esta base foram justamente caminhos temporais com granularidade hora. Para a base Mondial, a revocação (0,67) obtida pelo AXPath superou os trabalhos relacionados, porém percebe-se uma queda na precisão (0,50), ainda que seja superior aos trabalhos relacionados. Isto ocorreu, pois o arquivo continha um formado de data não previsto (`dia mes ano`, separados por espaço). Os problemas de anotação na base Mondial e WSU foram resolvidos alterando o arquivo de configuração.

Tabela 2. Revocação e Precisão obtidas nos experimentos.

	Revocação Mondial	Precisão Mondial	Revocação WSU	Precisão WSU
AXPath	0,67	0,50	0,33	1,00
GUTime	0,33	0,17	0,00	0,00
ANNIE	0,33	0,10	0,67	0,33

6. Considerações Finais

Este artigo descreve a proposta de mapeamento automático de modelos de dados XML temporais *ad-hoc* para um modelo de dados temporal padrão, a fim de permitir consultas temporais sobre estes documentos. Este mapeamento possui cinco procedimentos: **(i)** identificação de caminhos temporais; **(ii)** identificação de instantes fragmentados, **(iii)** identificação de intervalos; **(iv)** identificação das características temporais do modelo; e **(v)** construção. Para atender ao primeiro procedimento foi desenvolvido um protótipo de anotação de caminhos XML, denominado AXPath. Resultados iniciais mostram que a aplicação do AXPath para anotação de caminhos temporais apresenta melhor precisão que os trabalhos relacionados (ferramentas GUTime e ANNIE).

Como trabalhos futuros destacam-se: **(i)** implementação dos demais procedimentos da proposta; **(ii)** realização de experimentos com várias bases de dados XML para: avaliar a qualidade dos resultados através das métricas revocação, precisão e f-measure, e medir o tempo gasto para a realização do processo; e **(iii)** extensão do AXPath para permitir uma maior expressividade na linguagem do arquivo de configuração.

Referências

- Alonso, O. e Gertz, M. (2006). Clustering of search results using temporal attributes. In *SIGIR '06*, pages 597–598, New York, NY, USA. ACM.
- Alonso, O.; Gertz, M. e Baeza-Yates, R. (2007). On the value of temporal information in information retrieval. *SIGIR Forum*, 41(2):35–41.
- ANNIE (2009). Annie - open source information extraction. Disponível em: <<http://www.aktors.org/technologies/annie/>>. Último acesso em: dezembro 2009.
- Baeza-Yates, R. A. e Ribeiro-Neto, B. A. (1999). *Modern Information Retrieval*. ACM Press / Addison-Wesley.
- Clifford, J. e Croker, A. (1987). The historical relational data model (hrdm) and algebra based on lifespans. In *ICDE'1987*, Washington, DC, USA. IEEE Computer Society.
- Di Vimercati, S. D. C. (2002). An authorization model for temporal xml documents. In *SAC '02*, pages 1088–1093, New York, NY, USA. ACM.
- Dyreson, C. et al. (1994). A consensus glossary of temporal database concepts. *SIGMOD Rec.*, 23(1):52–64.
- Edelweiss, N.; Oliveira, J. P. M. d. e Pernici, B. (1993). An object-oriented temporal model. In *CAiSE '93*, pages 397–415, London, UK. Springer-Verlag.
- Gao, D. e Snodgrass, R. T. (2003). Temporal slicing in the evaluation of xml queries. In *VLDB '2003*, pages 632–643. VLDB Endowment.
- GUTime (2009). Gutime - adding timex3 tags. Disponível em: <<http://www.timeml.org/site/tarsqi/modules/gutime/index.html>>. Último acesso em: dezembro 2009.
- Manica, E. e Galante, R. (2009). Suporte a consultas temporais por palavras-chave em xml. In *SBBB 2009 - Posterres*.
- Rizzolo, F. e Vaisman, A. A. (2008). Temporal xml: modeling, indexing, and query processing. *The VLDB Journal*, 17(5):1179–1212.
- Silva, R. G. e Edelweiss, N. (2001). Uma proposta de extensão temporal de xml e a rea- lização de consultas. In *Conferência Latino-Americana de Informática (CLEI 2001)*.
- Tansel, A. U. (1997). Temporal relational data model. *IEEE Trans. on Knowl. and Data Eng.*, 9(3):464–479.

Da importância de modelos preditivos comprehensíveis: Um estudo de caso em saúde bucal

Rodrigo C. Barros¹, Luciano C. Blomberg¹, José A. P. Figueiredo², Duncan D. Ruiz¹

¹Faculdade de Informática – Pontifícia Universidade Católica do Rio Grande do Sul (PUC-RS)
Caixa Postal 90619–900 – Porto Alegre – RS – Brazil

²Faculdade de Odontologia – Pontifícia Universidade Católica do Rio Grande do Sul (PUC-RS)
Caixa Postal 90619–900 – Porto Alegre – RS – Brazil

Abstract. Even though the oral health literature presents many works related to the generation and application of models for predicting pathologies, most of them concentrate on maximizing predictive accuracy, ignoring important issues such as result interpretation and validation. The goal of this work is to discuss the advantages of generating comprehensible models that can be interpreted by the domain specialist. For this purpose, it is presented a comparison of different techniques applied to an oral health problem in order to validate the concepts under discussion.

Resumo. Embora a literatura de saúde bucal disponibilize uma série de trabalhos relacionados à exploração de modelos preditivos para as mais diversas patologias bucais, boa parte destes trabalhos está eminentemente orientada a maximização da acurácia preditiva, desprezando aspectos importantes como a validação e interpretação dos resultados obtidos. Este trabalho tem como objetivo discutir em detalhes as vantagens de se gerar modelos comprehensíveis para o especialista de domínio e, para tanto, apresenta uma comparação de diferentes abordagens preditivas sobre um problema de saúde bucal, de forma a validar os conceitos discutidos.

1. Introdução

A tendência da maioria dos trabalhos realizados para geração de modelos preditivos nas áreas da saúde bucal é a de buscar àqueles algoritmos ou técnicas que proporcionam o melhor desempenho preditivo. Esta ocorrência é provavelmente resultante da importância que se dá ao desempenho preditivo nas áreas de aprendizagem de máquina e mineração de dados [Freitas et al. 2008]. Trabalhos como [Gansky 2003] e [Montenegro et al. 2008] que se utilizam de métodos preditivos como redes neurais e *support vector machine*, comprovam a tendência de utilização de técnicas que possuem um desempenho preditivo superior às demais, mensurando este fato através de determinada medida (por exemplo a acurácia para problemas de classificação, ou alguma medida de erro para problemas de regressão).

No entanto, este tipo de tendência parece fugir do conceito fundamental de mineração de dados, o de descobrir informações previamente desconhecidas e potencialmente úteis ao domínio em questão [Tan et al. 2006], [Han and Kamber 2006]. Uma vez que os modelos utilizados visam exclusivamente à capacidade de predição de novas instâncias, ignora-se o fato de que a identificação de padrões ou perfis que explicam o

comportamento dos dados é igualmente importante (e muitas vezes mais importante) que a predição de novas instâncias.

Este trabalho tem como contribuições eliciar as razões que justificam a adoção de modelos comprehensíveis na área de saúde bucal - generalizáveis para outros domínios de aplicação - e motivar uma nova frente de trabalhos em mineração de dados que não se limite à simples análise de desempenho preditivo, mas que busque realmente contribuir para identificação de padrões previamente desconhecidos. Para tanto, primeiramente são apresentados trabalhos de mineração de dados para saúde bucal, identificando a preferência dos autores por técnicas que privilegiam o desempenho preditivo em detrimento à comprehensibilidade. Em um segundo momento, aponta-se as vantagens em se utilizar técnicas ditas comprehensíveis, apresentando exemplos de uso. Por fim, executa-se um estudo de caso onde diversas técnicas preditivas são aplicadas de forma a validar os conceitos discutidos ao longo deste artigo.

2. Mineração de dados na área de saúde bucal

Discute-se nesta seção vários trabalhos que utilizam técnicas para predição de variáveis na área da saúde bucal, apresentando para tanto, trabalhos realizados para predição de cárie dentária.

Em [Baldani et al. 2002], modelos de regressão linear múltipla foram desenvolvidos para associação da variação do índice CPO-D (número de dentes perdidos, cariados ou obturados) com indicadores sociais e socioeconômicos de renda, habitação, escolaridade, oferta de serviços odontológicos e fluoração das águas no estado do Paraná. Os resultados deste trabalho corroboram com a literatura, apontando a cárie dentária como uma patologia diretamente associada a populações de baixa renda. Modelos de regressão linear múltipla tipicamente adotam a forma $y = \varpi_0 + \sum_{i=1}^{\alpha} w_i x_i$, sendo y o atributo alvo da predição e x_i um atributo preditivo pertencente ao conjunto de α atributos. Tais modelos são de difícil comprehensão, pois embora seja possível inferir o grau de importância de determinados atributos através dos pesos calculados pela técnica de mínimos quadrados [Björck 1996], o relacionamento direto entre variáveis preditivas permanece implícito, uma vez que apenas a influência dos atributos preditivos sobre o atributo alvo da predição é explicitada.

No estudo de [Tagliaferro et al. 2006], foi aplicada regressão logística para a identificação de fatores de risco associados ao aumento da cárie dentária em crianças de 6 a 8 anos de idade no estado de São Paulo. Como principal resultado deste trabalho, constatou-se uma forte relação do aumento do índice de cárie com o baixo nível escolar da mãe. Da mesma forma, incidência prévia de cárie na dentição decídua foram consideradas significantes preditoras para futuras cárries. Regressão logística difere-se da regressão linear por ser utilizada na predição de variáveis categóricas, porém com os mesmos problemas de comprehensibilidade já citados. Outro trabalho que utiliza a técnica de regressão logística é [Celeste et al. 2007], onde foram utilizados dados de 4033 jovens gaúchos entre 15 e 19 anos para avaliar a associação entre atividades de prevenção da cárie dentária e a prevalência do índice CPO-D. Entre as conclusões, ratificou-se a importância dos procedimentos preventivos como um fator inibidor da cárie dentária.

Em [Gansky 2003] foram analisados os dados de 466 crianças de até vinte e quatro meses de idade para predição do risco de cárries. Foram utilizadas e comparadas as

técnicas de regressão logística, árvores de regressão e classificação, e redes neurais. Embora tenha introduzido conceitos fundamentais do processo de KDD, este ainda é um trabalho eminentemente orientado a comparação da acurácia preditiva de diferentes técnicas de mineração de dados, desprezando a importância do caráter comprehensível dos modelos gerados.

Em [Montenegro et al. 2008] foi realizado um estudo experimental a partir dos dados coletados em entrevistas com mães de 3864 crianças abaixo de 5 anos de idade. O estudo consistiu na avaliação e comparação das técnicas de árvores de decisão, redes neurais, KNN (*K-nearest neighbors*) e *support vector machine* aplicadas à predição de cárie dentária. Conforme já verificado em outros estudos, constatou-se uma provável influência das condições financeiras e das experiências anteriores de cárie com a incidência de novos casos da doença. O trabalho foca nos bons resultados obtidos com as técnicas de KNN e redes neurais, embora reconheça, ainda que sucintamente, a importância dos modelos em árvore para interpretação dos resultados por parte do especialista de domínio.

Considerando os últimos trabalhos analisados, é importante destacar que embora seus resultados apontem para redes neurais como melhor técnica preditiva (em termos de acurácia), seus modelos ainda são pouco comprehensíveis, dificultando assim um melhor entendimento do problema investigado. Da mesma forma, os trabalhos que utilizam abordagens estatísticas como regressão linear e regressão logística [Baldani et al. 2002], [Tagliaferro et al. 2006], [Celeste et al. 2007], tendem a demandar um grande esforço, à medida que tipicamente apóiam-se na validação de hipóteses e na comprehensão de resultados exclusivamente numéricos que ignoram os relacionamentos entre atributos preditivos.

3. Da importância de modelos comprehensíveis

Esta seção está dividida nos aspectos julgados como determinantes na aplicação de modelos comprehensíveis na área de saúde bucal: (i) descoberta de novos padrões através da análise visual do modelo (Seção 3.2); e (ii) identificação de erros nos dados e detecção de problemas durante coleta ou processamento dos mesmos (Seção 3.3). Através da exploração destes dois aspectos, procura-se motivar uma nova frente de trabalhos em mineração de dados que priorize a geração de modelos comprehensíveis, capazes de realmente descobrir padrões previamente desconhecidos e potencialmente úteis na área de aplicação. Para auxiliar na argumentação destes aspectos, utilizou-se resultados de pesquisa em saúde bucal descrita no cenário da Seção 3.1. Por fim, é abordada a utilização de métodos híbridos capazes de prover modelos comprehensíveis a técnicas do tipo caixa-preta (Seção 3.4).

3.1. Cenário de Pesquisa

O cenário utilizado para ilustrar a importância da geração de modelos preditivos comprehensíveis está inserido no estudo interdisciplinar realizado entre as faculdades de odontologia e informática da PUC-RS no qual desenvolveu-se inicialmente um ambiente para armazenamento e recuperação de dados [Blomberg et al. 2009]. Por meio desta parceria, teve-se acesso aos dados de 598 pacientes atendidos junto ao CEU (Centro de Extensão Universitária) Vila Fátima, unidade administrada pela PUCRS e vinculada ao SUS (Sistema Único de Saúde).

É importante destacar a relevância social do serviço prestado por tal unidade, uma vez que esta presta assistência social e odontológica a uma população de 8 mil habitantes

de baixa renda, situada na zona leste de Porto Alegre-RS. Os dados coletados dizem respeito à anamnese, diagnóstico clínico e odontogramas registrados em prontuários odontológicos.

3.2. Descoberta de padrões através da análise visual

A etapa de mineração de dados dentro de um processo de KDD não se resume à produção de modelos que obtenham melhor desempenho com relação à determinada medida. Pode-se exemplificar tal cenário com a geração de dois modelos preditivos para um problema de determinação do uso de creme dental pelos pacientes da rede pública da região sul do país. Um destes modelos foi gerado pelo algoritmo de indução de árvores de decisão C4.5 [Quinlan 1993], ao passo que o segundo foi gerado por uma rede neural de retropropagação de Perceptron multicamadas.

Em termos de acurácia preditiva, a rede neural apresentou um percentual de instâncias corretamente classificadas de 97% enquanto a árvore de decisão mostrou um percentual menor, 94%. Embora a maioria dos trabalhos da área aponte para o sucesso do uso de redes neurais no processo de descoberta de conhecimento, levanta-se aqui a seguinte questão: qual novo conhecimento foi descoberto pela rede neural? O que diz a análise dos pesos dos nós *sigmoid* da rede neural? Existe a possibilidade de se descobrir algum padrão nos dados que aponte o motivo de determinada instância ter sido classificada de uma forma ou outra?

Propõe-se, agora, a análise da árvore gerada pelo algoritmo C4.5 para este mesmo problema (Figura 1). Como este modelo pode contribuir para o processo de descoberta de conhecimento? Em uma análise rápida, pode-se notar que o fator preponderante na determinação do uso de creme dental é a visitação prévia ao dentista. Dentre aqueles habitantes que nunca foram ao dentista, o índice CPO-D divide os habitantes conforme determinado ponto de corte para que, finalmente, sejam avaliados de acordo com suas idades. Conforme a idade, é determinado se certo habitante usa ou não creme dental.

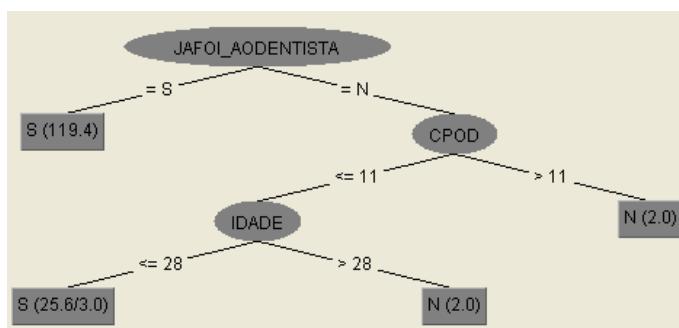


Figure 1. Árvore de decisão gerada pelo algoritmo C4.5 para um problema de classificação odontológica. O atributo classe indica a utilização, ou não, de creme dental.

É possível perceber que o modelo preditivo representado pela árvore permite a identificação de algumas evidências relacionadas ao perfil dos habitantes que não utilizam creme dental, embora nem todas sejam relevantes à investigação do problema. Ainda sim, tais modelos são extremamente importantes, à medida que contribuem para formulação de novas hipóteses, assim como para sinalização da necessidade de ações preventivas direcionadas à grupos de risco melhor caracterizados.

Os benefícios da análise de modelos comprehensíveis são potencialmente maiores do que a simples previsão, instância por instância, de determinado problema preditivo. Além disso, a acurácia preditiva resultante dos modelos é diretamente relacionada aos dados com os quais os modelos foram treinados. Portanto, a mineração de pequenas quantidades de dados, ou dados com distorções originadas no momento de coleta ou processamento, pode ser prejudicada a ponto de inutilizar os modelos para previsão de novos casos. Tal aspecto é considerado na seção seguinte.

3.3. Detecção de anomalias nos dados

Considere agora um problema onde deseja-se prever o indicador de cárie dentária em pacientes da rede pública do sul do país, considerando para tanto dois níveis de severidade: alto e baixo. Dois modelos preditivos distintos são gerados, o primeiro através do algoritmo de indução de árvores CART (*Classification and Regression Trees*) [Breiman et al. 1984] e o segundo através do algoritmo de *support vector machine* SMO (*Sequential Minimal Optimization*) [Platt 1998]. Considerando a acurácia preditiva, o algoritmo CART produz um modelo com 55% de taxa de acerto, enquanto o algoritmo SMO produz um modelo com 57% de acerto. Em uma análise simplista, poderia-se apontar a utilização de *support vector machine* como a melhor opção para este caso.

Entretanto, analisando a árvore gerada pelo CART (Figura 2), pode-se facilmente perceber certas incoerências nos dados, como por exemplo a percepção de que pessoas que utilizam fluoretos como complemento à escovação dentária estão mais propensas a altos indicadores de cárie dentária.

Tal percepção, embora equivocada, aponta para possíveis inconsistências durante o processo de anamnese, como por exemplo a subjetividade das respostas do paciente. Em outras palavras, poderia-se cogitar que por razões de constrangimento, pacientes não estariam sendo totalmente francos em suas respostas. Em contrapartida, este tipo de constatação não seria detectável no modelo de *support vector machine* gerado pelo algoritmo SMO, o que vem a ressaltar a importância de modelos comprehensíveis.

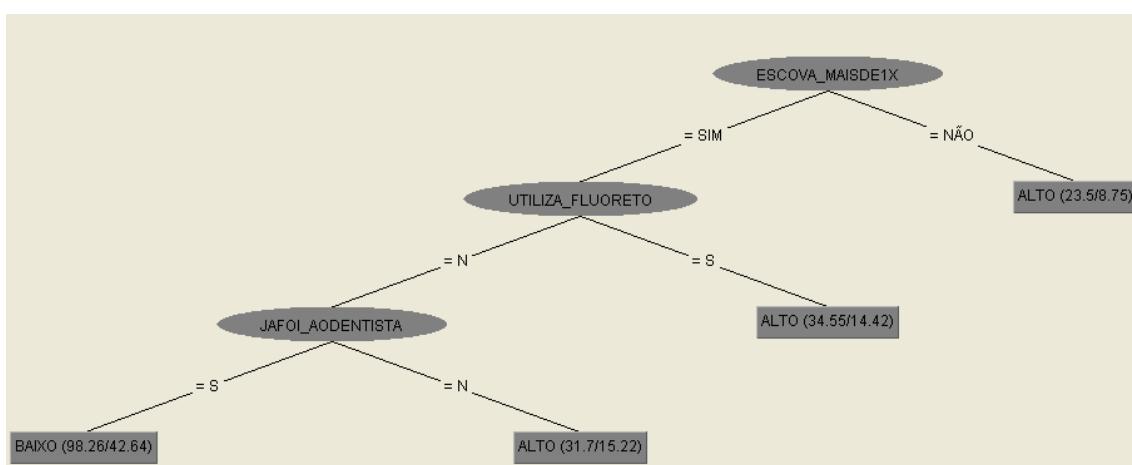


Figure 2. Árvore de decisão gerada pelo algoritmo CART para problema de classificação de nível de cáries.

3.4. Métodos híbridos - Combinando Acurácia com Compreensibilidade

Considerando que o foco deste artigo é motivar a utilização de técnicas que forneçam modelos mais compreensíveis aos especialistas de domínio, são exemplificados a seguir, trabalhos que procuraram modificar técnicas tipicamente caixa-preta de forma a fazê-las produzir modelos passíveis de interpretação. Tais trabalhos, aqui chamados de modelos híbridos, reconhecem a importância de combinar desempenho preditivo com geração de modelos comprehensíveis.

Um exemplo bem-sucedido de modelo híbrido é o trabalho de Setiono [Setiono et al. 2002] que tem por objetivo realizar a extração de regras de redes neurais para tarefas de regressão não-linear. Através das regras, o especialista de domínio é então capaz de realizar a análise dos dados em busca de novos padrões e detecção de erros, sem deixar de lado a alta capacidade preditiva das redes neurais.

Outro exemplo de combinação de desempenho preditivo e modelos interpretáveis é [Ratsch et al. 2006], onde os autores propõem um novo algoritmo de *support vector machine* que seja capaz de gerar modelos interpretáveis para classificação de sequências biológicas. O algoritmo computa pesos esparsos de substrings, destacando as partes da sequência que são importantes para discriminação.

Johansson et al. [Johansson et al. 2005] propõem o uso de um *ensemble* de classificadores, técnica que busca utilizar diferentes classificadores básicos combinados através de um sistema de votação. Tal técnica é reconhecidamente caixa-preta, pois o uso de diferentes classificadores inviabiliza a formação de hipóteses e descoberta de padrões consistentes entre si. Como forma de resolver o problema, os autores propõem a aplicação de um algoritmo de extração de regras, chamado G-REX, sobre o *ensemble* de forma a produzir um formato comprehensível que possibilite a formação de hipóteses e descoberta de conhecimento.

Os trabalhos supracitados adicionam uma camada de pós-processamento no processo de mineração de dados. Esta camada, que age como tradutora do modelo não-comprehensível gerado para algum formato passível de interpretação, como regras ou grafos, pode inserir distorções ou inconsistências nos dados, comprometendo o processo de descoberta de conhecimento.

4. Um estudo de caso na área de saúde bucal

Para demonstrar a importância da geração de modelos comprehensíveis ao especialista de domínio, descreve-se nesta seção o estudo de caso realizado junto ao CEU Vila Fátima, no qual se buscou comparar o desempenho de quatro técnicas classificadoras, aplicadas à predição de doenças periodontais (ex: gengivite, periodontite). Para tanto, obteve-se uma amostra, inicialmente composta pelos dados de 642 prontuários odontológicos.

Uma vez que o número de atributos que caracteriza as instâncias da base de dados é superior a 100, utilizou-se técnicas de *feature selection* para identificação dos atributos mais relacionados à incidência de doenças periodontais, chegando-se desta forma ao seguinte conjunto: idade, escolaridade, frequência_consumo_acucar2, escova_sozinho, usacremedental, escova_maisde1x, doencaperiodont_nafamilia, toma_mamadeira e roe_unhas.

Da mesma forma, buscou-se também equilibrar a frequência do atributo classe

(S ou N), de modo a evitar distorções na geração dos modelos preditivos. Realizadas as devidas preparações no conjunto de dados, utilizou-se a ferramenta de BI Weka 3.7.0 [Witten and Frank 2005] para extração de modelos preditivos. Foram aplicadas as técnicas de SVM, Redes Neurais, K-NN e árvores de decisões, respectivamente implementadas pelos algoritmos SMO, MULTILAYERPERCEPTON, IBK e J48.

Na Tabela 1 são apresentados os resultados obtidos para cada técnica aplicada, as quais foram avaliadas pelas métricas de desempenho "acurácia preditiva" e MAE (Mean Absolute Error, erro absoluto médio).

TÉCNICA	ALGORITMO	ACURÁCIA	MAE
SVM	SMO	69,1176%	0,3088%
REDES NEURAIS	MULTILAYERPERCEPTON	69,1176%	0,361%
K-NN	IBK	66,1765%	0,3336%
ÁRVORES DE DECISÃO	J48	66,1765%	0,4126%

Table 1. Resultados obtidos por 4 diferentes classificadores.

Embora os resultados apontem para uma pequena superioridade das técnicas de SVM, Redes Neurais e K-NN em relação às árvores de decisão, essas ainda produzem modelos pouco comprehensíveis ao especialista de domínio. Em outras palavras, pode-se dizer que suas representações prioritariamente numéricas não favorecem ao entendimento do modelo, bem como a detecção de possíveis inconsistências a partir deste.

Diferentemente destas técnicas, árvore de decisão destaca-se por oferecer uma representação gráfica que facilita o entendimento do especialista de domínio, oferecendo-lhe melhores condições para interpretação dos resultados e embasamento para tomadas de decisão.

A Figura 3 ilustra o modelo gerado pelo algoritmo J48 aplicado à predição de doenças periodontais, conforme descrito no início desta seção.

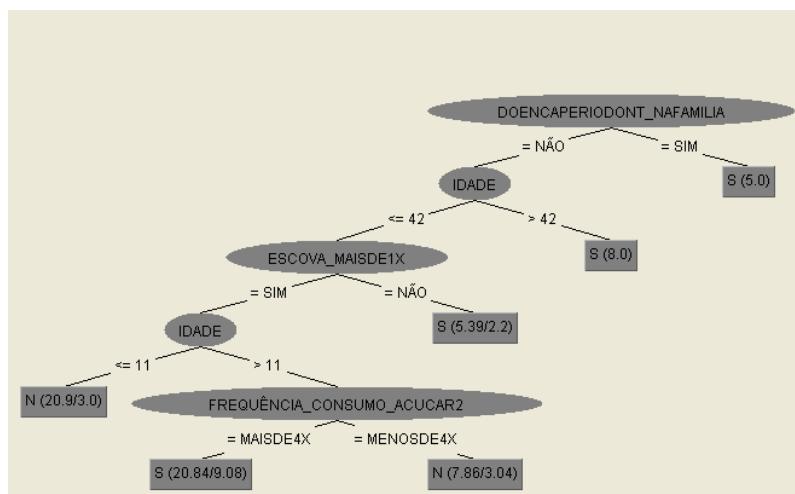


Figure 3. Árvore de decisão gerada pelo algoritmo J48.

Com base nos resultados sugeridos pelo modelo, pode-se constatar prováveis relações entre os atributos, como por exemplo, a relevância da idade e do fator genético

para a incidência de novos casos de doenças periodontais. Levando-se em conta a avaliação do especialista, chegou-se a conclusão que os resultados apontados pela árvore de decisão corroboram com os estudos encontrados na literatura. Assim, conclui-se que diferentemente da cárie dentária que é ocasionada por uma ação bacteriana, doenças periodontais estão diretamente associadas a pessoas com idade superior a 40 anos e com experiências prévias da doença na família.

Em contrapartida, a avaliação dos modelos gerados pelas técnicas de SVM, Redes neurais e K-NN, mostra-se relativamente complexa, à medida que seus resultados tipicamente dependem da aplicação de técnicas de pós-processamento para obtenção de maior comprehensibilidade, não garantindo a consistência dos resultados produzidos.

Desta forma, frente às considerações apontadas neste artigo, entende-se que para o cenário analisado, maiores contribuições podem ser alcançadas com o uso de técnicas que privilegiam a comprehensibilidade dos modelos gerados, e não apenas a acurácia preditiva.

5. Conclusões e Trabalhos Futuros

Este artigo apresentou uma avaliação crítica das abordagens tradicionais para medir o desempenho de modelos preditivos voltados à saúde bucal. Tais abordagens geralmente são baseadas em uma única medida de qualidade (geralmente acurácia preditiva), fazendo da indução de modelos preditivos "caixa-preta" a mais utilizada.

Modelos do tipo caixa-preta, ainda que apresentem alto desempenho preditivo, não podem ser interpretados pelos especialistas de domínio em questão, inviabilizando o processo de descoberta de conhecimento. Tais modelos impedem a formação de novas hipóteses e detecção de anomalias nos dados coletados e processados.

Como alternativa à esta abordagem, este artigo apresentou a importância da geração de modelos comprehensíveis no domínio de saúde bucal, utilizando para tanto, exemplos de uso baseados em uma pesquisa interdisciplinar entre as faculdades de odontologia e informática da PUC-RS. Complementarmente, foi apresentado um estudo de caso onde se aplicou diferentes técnicas para predição de doenças periodontais, fornecendo uma visão prática da importância da geração de modelos comprehensíveis dentro de um processo de descoberta de conhecimento.

É importante ressaltar, que este trabalho não tem a intenção de defender o uso exclusivo de técnicas que produzam modelos mais comprehensíveis, mas sim, motivar a utilização de tais modelos como complemento ao uso de técnicas preditivas mais robustas em termos de desempenho preditivo.

Como trabalhos futuros pretende-se dar continuidade ao estudo realizado junto ao CEU Vila Fátima, bem como explorar novas áreas de atuação junto a Faculdade de Odontologia, como por exemplo o reconhecimento de padrões a partir da análise de imagens médicas.

References

- Baldani, M. H., Narvai, P. C., and Antunes, J. L. F. (2002). Cárie dentária e condições socioeconômicas no estado do paraná, brasil, 1996. *Caderno de Saúde Pública*, 18:143–152.

- Björck, Å. (1996). *Numerical Methods for Least Squares Problems*. SIAM, Philadelphia.
- Blomberg, L. C., Mota, E., antônio Poli de Figueiredo, J., and Ruiz, D. D. A. (2009). Database in oral health for management of clinical records. *Journal of Dental Science*, 24:249–253.
- Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA.
- Celeste, R. K., Nadanovsky, P., and Leon, A. P. D. (2007). Associação entre procedimentos preventivos no serviço público de odontologia e a prevalência de cárie dentária. *Revista de Saúde Pública*, 41:830–838.
- Freitas, A., Wieser, D., and Apweiler, R. (2008). On the importance of comprehensible classification models for protein function prediction. *To appear in IEEE/ACM Transactions on Computational Biology and Bioinformatics*.
- Gansky, S. A. (2003). Dental data mining: Potential pitfalls and practical issues. *Advances in Dental Research*, 17:109–114.
- Han, J. and Kamber, M. (2006). *Data Mining Concepts and Techniques*. Elsevier.
- Johansson, U., Konig, R., and Niklasson, L. (2005). Automatically balancing accuracy and comprehensibility in predictive modeling. In *Information Fusion, 2005 8th International Conference on*, volume 2, pages 7 pp.–.
- Montenegro, R. D., Oliveira, A. L. I., Cabral, G. G., Katz, C. R. T., and Rosenblatt, A. (2008). A comparative study of machine learning techniques for caries prediction. In *ICTAI '08: Proceedings of the 2008 20th IEEE International Conference on Tools with Artificial Intelligence*, pages 477–481, Washington, DC, USA. IEEE Computer Society.
- Platt, J. C. (1998). Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. Technical report, Microsoft Research.
- Quinlan, J. R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- Ratsch, G., Sonnenburg, S., and Schafer, C. (2006). Learning interpretable svms for biological sequence classification. *BMC Bioinformatics*, 7(Suppl 1):S1–S9.
- Setiono, R., Leow, W., and Zurada, J. (2002). Extraction of rules from artificial neural networks for nonlinear regression. *IEEE Transactions on Neural Networks*, 13(3):564–577.
- Tagliaferro, E. P., Pereira, A. C., de Castro Meneghim, M., and Ambrosano, G. M. (2006). Assessment of dental caries predictors in a seven-year longitudinal study. *Journal of Public Health Dentistry*, 66:169–173.
- Tan, P., Steinbach, M., and Kumar, V. (2006). *Introduction to Data Mining*. Pearson Education.
- Witten, I. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco.

Especificação de uma Taxonomia para Metadados Multifacetados em Ambientes de Integração de Dados na Web

Raquel Cruz¹, Carina F. Dorneles², Renata Galante³

¹Instituto de Ciências Exatas – Universidade de Passo Fundo (UPF)
Prédio B5 BR 285– Passo Fundo, RS – Brazil

²Departamento de Informática e Estatística – Universidade Federal de Santa Catarina
(UFSC)
Florianópolis, SC

³Instituto de Informática – Universidade Federal do Rio Grande do SUL (UFRGS)
Porto Alegre, RS –Brazil

raquel@upf.br, dorneles@inf.ufsc.br, galante@inf.ufrgs.br

Abstract. This paper presents a proposal of a taxonomy to be used for defining multi-faced metadata in a data integration environment. The main idea is to define this taxonomy in levels, where the more abstract one may be classified in: syntactic, structural and semantic. Our proposal is based on the idea of multi-faced classification, where a specific metadata must appear in more than one classification.

Resumo. Este artigo descreve a especificação de uma taxonomia a ser usada para definição de metadados multifacetados utilizados em ambientes de integração de dados na Web. A idéia principal é definir a taxonomia em níveis, onde o nível mais abstrato pode ser classificado em: sintático, estrutural e semântico. A proposta é baseada na idéia de classificação multifacetada, onde um mesmo metadado pode aparecer em mais de uma classificação.

1. Introdução

Devido à massificação da informação e à grande quantidade de dados disponíveis tanto na Web quanto em redes internas de organizações, faz-se necessário um esforço contínuo para assegurar que sistemas e processos sejam gerenciados para aumentar oportunidades de troca e reuso de informações internas ou externas de uma organização. Gerenciar a heterogeneidade semântica e estrutural das informações a fim de prover acesso integrado aos dados é um dos principais problemas a serem solucionados por sistemas de integração de dados [Siedler, 2004]. Assim, devido a essa grande quantidade de dados ser produzida em um curto período de tempo, as organizações se tornam potencialmente vulneráveis aos impactos da explosão de informação, podendo causar problemas em sua gestão. Nesse contexto, o desenvolvimento de metadados aparece como uma possível solução para os problemas da organização e da gestão de dados.

Em uma definição simplista, metadados seriam “dados que descrevem outros dados”, mas na verdade são marcos ou pontos de referência que permitem limitar a informação sob todas as formas. Metadados servem para descrever e estruturar, de maneira estável e uniforme os dados que são registrados sob diferentes suportes documentais [Taylor, 2003]. Metadados permitem acessar facilmente a informação, extraí-la e compreendê-la, fornecendo também um contexto, ou seja, para cada ambiente que produz ou gera uma massa documental, existem motivos particulares que justificam e explicam a elaboração deles. Metadados são, por assim dizer, resumos da informação sobre a forma ou o conteúdo de uma fonte. Mas, atualmente eles também vêm sendo usados pelos serviços de informação *on-line* em vários processos como busca de informação, autenticação, direitos de autor e arquivamentos.

No contexto de sistemas de integração de dados, os metadados devem conter informações sobre as fontes e prover suporte a diferentes níveis de informação desde sua estrutura até quais são as funcionalidades oferecidas por elas. Para cada participante do ambiente de integração, os metadados podem incluir (ou possibilitar a geração automática) o esquema da fonte (se houver), estatísticas, taxas de mudanças, capacidade de resposta a consultas, domínio a que pertence, e políticas de acesso e segurança. Relacionamentos existentes entre as fontes de dados podem estar armazenados como grafos, visões, ou até mesmo descrições textuais. Sempre que possível, os metadados devem fornecer informações sobre: identificador, tipo, data de criação, data de atualização e assim por diante. Deve servir, no mínimo, para responder sobre a presença ou ausência de determinados metadados, ou determinar quais participantes manipulam um determinado tipo de dado. Diante desse contexto, a grande dificuldade que existe é saber identificar quais são os metadados mais importantes para cada aplicação. Especificamente, no contexto deste trabalho, o grande desafio é identificar quais metadados são essenciais em aplicações inseridas em ambientes de integração de dados. Esta identificação é necessária para que seja possível acessar os dados integrados, através de um formato flexível que permita troca de informações – uma vez que na Web os dados podem possuir uma estrutura bem definida como também podem ser totalmente desestruturados - e executar consultas através de alguma.

Nesta linha de raciocínio, o presente trabalho tem por objetivo descrever uma proposta que apresenta uma taxonomia para definição de metadados, utilizados em ambientes de integração de dados. A taxonomia proposta possui, no nível mais alto de abstração, a seguinte classificação: (i) **metadados sintáticos**, descrevem informações não contextuais sobre o conteúdo, geralmente provendo informações de caráter geral (por exemplo, localização da fonte, data de criação, entre outros); (ii) **metadados estruturais**, provêm informações sobre a estrutura dos dados, independentes do conteúdo, e descrevem como os itens estão organizados na fonte de dados e regras para esta organização (por exemplo, estrutura seguida pelos dados, tais como XML, relacional, documentos e dados semi-estruturados, entre outros); e (iii) **metadados semânticos**, fornecem informações sobre o significado dos dados disponíveis e seus relacionamentos semânticos (por exemplo, dados que descrevem o conteúdo semântico de um valor de dado - como unidades de medida e escala), ou dados que fornecem informações adicionais sobre sua criação (algoritmo de cálculo ou derivação da fórmula usada), linhagem dos dados (fontes) e qualidade (atualidade e precisão). Uma discussão informal sobre a necessidade desta classificação pode ser encontrada em [Lines, 2008]. Quando o

mesmo metadado está inserido em mais de uma classificação, ele é chamado de **metadados multifacetados**. Por exemplo, o conceito “política de acesso” pode ser considerado ao mesmo tempo um metadado estrutural (que indica a estrutura computacional utilizada) e metadado sintático (indicando como proceder, sintaticamente, com os comandos de acesso).

Este artigo está organizado como segue. A Seção 2 apresenta os trabalhos relacionados enquanto a Seção 3 descreve os conceitos básicos e terminologias importantes no contexto de metadados e integração de dados. A Seção 4 descreve a taxonomia proposta através de uma classificação multifacetada de metadados para utilização em ambientes de integração de dados na Web. Por fim, a Seção 5 é dedicada às considerações finais e trabalhos futuros.

2. Trabalhos Relacionados

Bibliotecas digitais e páginas Web constituem importantes iniciativas de acesso à informação, entretanto, para oferecerem uma cobertura mais abrangente de recursos, utilizam serviços que fazem uso de metadados. Moura et. al. (2002) propõem uma estrutura formal para realizar tal tarefa, baseado em um modelo conceitual de metadados que explora as relações entre recursos de informação em diferentes níveis de granularidade.

Alguns trabalhos [Lourenço, 2007] se inserem em duas linhas de ação do Programa Brasileiro da Informação: conteúdos e identidade cultural e P&D. Na primeira linha de ação, o autor trata o problema da preservação e da disseminação da identidade cultural, com enfoque nos metadados. Na segunda linha, encontram-se estudos sobre a aplicação das tecnologias da informação de maneira apropriada às necessidades atuais da Web, através da utilização de metadados, a fim de que os conteúdos sejam descritos e estruturados com vistas a uma melhor recuperação pelas máquinas de buscas da Internet.

A questão sobre soluções para comércio eletrônico é levantada por alguns autores [Passos 2006], com ênfase ao grande número de mapeamentos e padronizações, propondo um método que utiliza ontologias para intermediar os domínios. O trabalho foca o problema de integrar dados originados de fontes distintas na comunidade de banco de dados, no ambiente Web, bases de conhecimento, planilhas, entre outros, com ênfase na interoperabilidade. O trabalho relata que duas características da Internet dificultam o acesso a informações específicas e relevantes: (i) a quantidade e a ausência de definição semântica precisa para as informações publicadas, para que sejam inteligíveis por programas e sistemas; e (ii) necessidade de agregar valor à informação disponível, tal que a mesma possa ser inferida tanto por humanos quanto por agentes inteligentes. Assim, ao descrever os desafios da integração de dados, o autor cita metadados como necessários aos esquemas mediadores como forma de descrição da estrutura das fontes envolvidas no ambiente de mediação.

Metadados não são usados apenas em catalogações bibliográficas [Baptista 2007], tradicionalmente provendo suporte às atividades de classificação, catalogação e indexação, mas também na identificação, localização e recuperação de informações na Web. O trabalho de Baptista 2007 foca o impacto dos metadados na representação descritiva, explora aspectos conceituais e a aplicação de metadados. O foco é caracterizar o impacto dos metadados na catalogação, entendida não só como atividade

bibliotecária, mas, sobretudo, como um conjunto de práticas. Tais práticas, baseadas em conhecimento especializado, passam a integrar novos conhecimentos no esforço multidisciplinar de se prover o acesso à informação da forma mais ágil e eficaz possível.

O que se observa na maioria dos trabalhos existentes na literatura é a grande importância dada ao uso de metadados. Esta característica reflete a real utilidade desta estrutura em sistemas heterogêneos e/ou distribuídos (integração de dados, bibliotecas digitais). O ponto fraco dos trabalhos existentes, no entanto, consiste na definição clara do conjunto de metadados que se considera adequado a sistemas de integração de dados. Como se pode observar, sistemas de bibliotecas digitais podem basear seus metadados em padrões consolidados como o Dublin Core, por exemplo; o que não acontece com ambientes distribuídos de integração de dados. Neste sentido, o presente trabalho apresenta um passo em direção à discussão e definição de metadados para este tipo de aplicação.

3. Metadados e Integração de dados na Web

Tomando como base os metadados que podem ser obtidos a partir de um recurso qualquer da Web, são diversos os fatores que justificam o seu emprego em sistemas de integração na Web, alguns dos quais são citados a seguir.

- **Desempenho:** registros de metadados são, geralmente, muito menores do que objetos que descrevem, requerendo menos recursos na sua transmissão, pesquisa e armazenamento.
- **Arquitetura:** a maioria dos protocolos de recuperação da rede (HTTP, por exemplo) não permite que partes ou sub-componentes de objetos sejam recuperados separadamente. No entanto, todo o conteúdo do documento pode representar mais informação do que o usuário necessita. Metadados podem ser usados para prover contexto a unidades específicas do documento, por ocasião de uma pesquisa, embora os sistemas de recuperação de informações convencionais não suportem de forma explícita a estrutura lógica de um documento.
- **Escopo:** metadados podem descrever recursos que não estão disponíveis no ambiente da rede. Esses recursos podem existir em alguma outra forma que não a digital (impresso, por exemplo) ou em algum dispositivo externo de armazenamento, tal como um CD.
- **Conteúdo:** alguns metadados não podem ser extraídos do conteúdo do objeto do qual descrevem. O assunto de um documento, por exemplo, é atribuído mediante uma análise intelectual de seu conteúdo. Da mesma forma, alguns metadados podem requerer métodos de extração complexos, sendo mais prático e menos custoso armazená-los logo após sua obtenção.
- **Privacidade:** metadados expressam de forma mais adequada os termos e condições que especificam o direito a propriedade intelectual por parte dos autores de recursos.

Neste ponto, uma importante constatação que se faz é em relação a ausência de um padrão para ambientes de integração de uma forma geral. Na Seção 2 foram apresentados diferentes trabalhos e padrões para definição de metadados, no entanto, nenhum deles é completamente aplicável a ambientes de integração. Desta forma, a

Seção 4 descreve uma proposta de taxonomia, composta por vários níveis, sendo um deles a definição de metadados para ambientes de integração de dados. Estes metadados podem ser descritos em XML, ou qualquer outra linguagem de representação de dados (RDF, OWL, entre outras)

4. MMID: Taxonomia para Metadados Multifacetados para Integração de Dados

Considerando um cenário de integração de dados com o uso de metadados, a proposta do presente trabalho é definir uma taxonomia para metadados multifacetados para integração de dados. Dentro desta taxonomia são definidos níveis que representam desde a forma mais abstrata da classificação até formas mais específicas que envolvem detalhes de implementação de um sistema/domínio presente no ambiente de integração. A Figura 1 mostra os níveis definidos para a taxonomia proposta. No nível mais alto de abstração define-se a classificação dos tipos de metadados a serem especificados (Sintáticos, Estruturais e Semânticos). No nível intermediário, estão os Metadados propriamente ditos e finalmente, no nível mais baixo é especificado o Projeto Físico, onde são especificadas ferramentas e detalhes de implementação usados na construção de cada fonte presente no ambiente de integração de dados (porta de acesso, driver do SGBD, entre outros). A ênfase descrita neste artigo é dada no nível mais alto, ficando os dois outros níveis (metadados e projeto físico) como trabalhos futuros.



Figura 1 – Níveis da taxonomia

A taxonomia proposta possui, no nível mais alto de abstração, a seguinte classificação:

- **metadados sintáticos:** descrevem informações não contextuais sobre o conteúdo, geralmente provendo informações de caráter geral (e.g. tamanho do documento, data de criação, etc.) [Matos,2008];
- **metadados estruturais:** provêm informações sobre a estrutura dos dados, independentes do conteúdo; descrevem como os itens estão organizados na fonte de dados e as regras para esta organização [Matos,2008].;
- **metadados semânticos:** descrevem informações sobre os dados, que são importantes em dado contexto ou domínio, permitindo certa interpretação; dados semânticos provêm um meio para pesquisas de alta precisão e possibilitam a interoperabilidade entre sistemas ou fonte de dados heterogêneos; estes dados

são usados para fornecer significado aos elementos descritos pelos metadados sintáticos ou estruturais [Madnick, 1995];

Além desta classificação, a taxonomia proposta trata conceitos multifacetados, ou seja, conceitos que podem aparecer em mais de uma classificação ao mesmo tempo. A Figura 2 descreve a taxonomia proposta, com alguns conceitos importantes a serem tratados em um ambiente de integração de dados. Por exemplo, o termo RIF (Relação Integração de Fontes), é usado para indicar que existe integração entre as fontes de dados que utilizam a taxonomia (ou seja, a taxonomia proposta não é usada simplesmente para gerar metadados de uma fonte individual, que não pertença a um ambiente de integração), e que cada fonte tem sua base formada pelos objetos descritos no modelo. A taxonomia é baseada em uma classificação que inclui informações sobre conteúdo, estrutura e semântica dos dados, onde cada classe possui um rótulo de conceito associado a ela, que define o objeto modelado e o objeto do mundo real que ela (classe) descreve. Alguns conceitos multifacetados podem ser percebidos nos seguintes casos: o conceito Esquema pode ser classificado tanto como Metadado Estrutural quanto Metadado Semântico, enquanto o conceito Abordagem pode ser classificado como Metadado Estrutural e Metadado Sintático. Cada categoria da taxonomia proposta é explicada em detalhes nas seções a seguir. A representação usada na Figura 2 é informal, onde as linhas contínuas representam relacionamento enquanto as setas representam generalização/especialização.

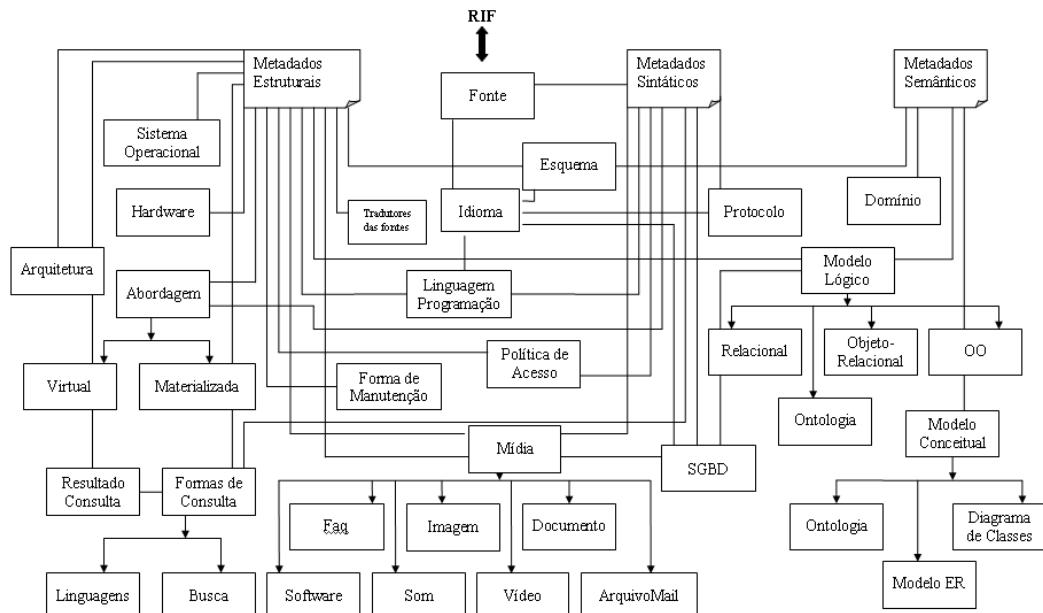


Figura 2 – Arranjo dos objetos

3.1 – Metadados Sintáticos

Os metadados sintáticos se referem àqueles conceitos que, em um ambiente de integração de dados, tem o papel de descrever as fontes de dados envolvidas no ambiente de integração. Através destes metadados é possível que outras fontes iniciem o acesso a ela, conseguindo identificá-la através de seu nome, protocolo de acesso, tipos de dados armazenados, formas de consultas possíveis, entre outros. Fazem parte desta classificação os metadados descritos a seguir.

- **Fonte:** é um dos principais conceitos da taxonomia proposta. É possível que outras fontes o utilizem para iniciarem seu acesso à fonte em questão, através da identificação de: nome, tipo e localização, principais tópicos cobertos nas informações disponíveis, estrutura dos dados, nomes das estruturas, nomes e tipos dos atributos.
- **Mídia:** é um metadado multifacetado, e foi criado para ilustrar que, devido à descentralização da Internet, os recursos de informação são heterogêneos, e incluem desde correspondência eletrônica, previsão de tempo, seções interativas de áudio e vídeo, agregações de informações (organizadas ou não de forma hierárquica), tais como banco de dados, arquivos acessíveis através do protocolo FTP até arquivos de listas de correspondência, grupos de notícias (“newsgroups”); devem-se armazenar metadados específicos ao tipo da mídia que a fonte possui, possibilitando a implementação de consultas baseadas em conteúdo e a integração de informações oriundas de mídias distintas.
- **Protocolo:** indica como as informações providas pelas fontes são acessadas, por exemplo, FTP, Telnet, Z39.50, HTTP, etc.
- **SGBD:** é um metadado multifacetado, e foi criado para indicar qual é o Sistema Gerenciador de Banco de Dados (SGBD’s) usado pela fonte. Diferentes aplicativos na mesma fonte de dados podem adotar diferentes SGBDs.
- **Linguagem de Programação:** cada fonte pode ter sido desenvolvida utilizando uma linguagem de programação diferente. Em um ambiente de integração de dados, muitas vezes é necessário saber qual, ou quais, são utilizadas.
- **Abordagem:** indica se é virtual ou materializada. É um dos pontos mais importantes em um ambiente de integração de dados, pois dependendo da abordagem, a forma de acesso às fontes pode ser feita de uma maneira ou outra.
- **Forma de consulta:** indica como a fonte pode ser acessada. Através deste metadado é possível identificar a sintaxe, ou forma, de acesso aos dados da fonte, que pode ser feita através de alguma linguagem de consulta (SQL, XQuery, entre outras) ou busca por palavra chave (estilo Google ou Yahoo!).
- **Política de acesso:** indica como a fonte de dados pode ser acessada, por exemplo, através de um WebService ou protocolo de segurança (HTTPS) entre outros, sendo fortemente vinculada ao tipo de abordagem utilizado.

3.2 – Metadados Estruturais

Os metadados estruturais se referem àqueles conceitos que desempenham o papel de permitir que as outras fontes conheçam a estrutura utilizada pela fonte em questão, para verificar a compatibilidade entre os componentes no que se refere não somente a interfaces, mas ao sistema operacional, formas de consulta, formas de manutenção, etc. Compõem esta classificação os metadados descritos a seguir.

- **Mídia:** é um metadado multifacetado. Como metadado estrutural, seu papel é apresentar as estruturas utilizadas nos diferentes tipos de mídia que uma fonte pode armazenar ou manipular.

- **Forma de consultas:** é um metadado multifacetado, e ao fazer o papel de metadado estrutural deve representar qual a estrutura utilizada pelas linguagens de consulta, ou *engines* de busca, para efetuar a recuperação dos dados da fonte.
- **Esquema:** é um metadado multifacetado, como metadados estruturais os esquemas das fontes de dados devem ser capturados, compreendidos e documentados de forma a permitir relacioná-los com objetivo de construir um esquema global. A documentação deve ser incorporada ao esquema global e deve conter uma completa descrição dos objetos, relacionamentos, atributos e métodos.
- **Sistema operacional:** descreve informações gerais sobre o(s) sistema(s) operacional(is) utilizado(s) pela fonte.
- **Hardware:** descreve quais requisitos são exigidos pelas fontes, quando houver necessidade de, por exemplo, identificar a quantidade de memória necessária para efetuar consultas, criar índices, executar aplicativos, informações sobre barramento, *cache*, etc.
- **Arquitetura:** tem seu contexto vinculado à abordagem, pois ela pode definir um esquema global dado pela integração dos esquemas das fontes locais, para acesso aos dados, ou um conjunto de banco de dados cooperantes e autônomos (que pode ser inapropriado no contexto da Web), ou baseada em uma abordagem multicamada (com o uso de mediadores, por exemplo).
- **Abordagem:** é um metadado multifacetado, e como metadado estrutural deve indicar a estrutura de implementação utilizada por cada abordagem (virtual e materializada).
- **Resultado de consulta:** representa como a fonte devolve os resultados das consultas realizadas, podendo ser nos formatos XML, documentos, *ranking* ou *resultSet* (quando for uma fonte relacional).
- **Forma de Manutenção:** serve para detectar quais técnicas são utilizadas para implementação das fontes, ou seja, caso uma fonte venha a sofrer alterações qual é a forma com que ela é armazenada, para ser rápida, flexível, confiável e com menor custo.
- **Tradutores das Fontes:** nos resultados de consulta as fontes podem ser retornadas nos mais diversos idiomas, e para que seja possível a integração deve haver tradutores de fontes.
- **SGBD:** é um metadado multifacetado e como metadado estrutural indica qual a estrutura lógica do SGBD usado pela fonte, ou seja, qual o modelo adotado por um SGBD específico, que pode ser relacional, orientado a objeto, objeto-relacional ou XML. Cada SGBD pode contar com interfaces de consulta próprias bem definidas.
- **Linguagem de Programação:** é um metadado multifacetado e como metadado estrutural indica por qual linguagem a fonte deverá ser acessada.

3.3 – Metadados Semânticos

Os metadados semânticos se referem àqueles conceitos que desempenham o papel de mostrar o significado das informações armazenadas em cada fonte de dado que faz parte do ambiente de integração. Esta pode ser a parte mais difícil da representação, pois a representação do conhecimento não é uma tarefa trivial. Fazem parte desta categoria, os metadados descritos a seguir.

- **Domínio:** apresenta uma breve descrição sobre qual o cenário da realidade é modelado na fonte.
- **Esquema:** é um metadado multifacetado, como metadado semântico seu papel é representar o esquema conceitual utilizado na fonte.
- **Modelo lógico:** indica o modelo utilizado pela fonte. Este modelo pode ser relacional, objeto-relacional, orientado a objetos, XML ou até mesmo semi-estruturado (neste caso, indicando que a fonte trabalha com dados semi-estruturados, tais como documentos HTML, PDFs, entre outros).
- **Modelo conceitual:** diz respeito a forma na qual o modelo conceitual do domínio é representado, podendo ser um modelo baseado em entidades e relacionamentos (ER), modelo baseado em classes (diagrama de classes) ou ontologia.

5. Conclusões e trabalhos futuros

Metadados são responsáveis por várias tarefas, tais como: organizar dados; manter controle sobre os níveis de atualização dos dados; documentar origem, formato, estrutura e sistemas de referência de dados; permitir intercâmbio entre diferentes sistemas; definir autoria, armazenamento, disponibilização e utilização dos dados. Com eles é possível criar moldes de entidades do mundo humano, tornando entidades desse mundo facilmente entendíveis pelo mundo computacional. Deste modo, a função básica e mais evidente dos metadados é a descriptividade, obedecendo a um padrão para obter interoperabilidade. Metadados devem ser produzidos e associados aos recursos da Internet para que os serviços de busca, por exemplo, tenham suporte à gestão, localização e recuperação e uma infra-estrutura que torne possível o intercâmbio de dados descritos nos serviços da Web..

Uma vez que em seu nível mais alto de abstração definiu-se a classificação dos tipos de objetos necessários no ambiente de integração, um trabalho futuro de grande importância para completar a taxonomia é a definição dos metadados envolvidos no projeto intermediário e físico de cada fonte de dados. Estes metadados dizem respeito aos atributos que farão parte de cada objeto e às ferramentas, configurações e gerenciamentos utilizados para a implementação completa do ambiente de cada fonte. A abordagem proposta traz uma novidade, pois é o primeiro passo em direção a discussões, ou definição, de metadados para aplicações que envolvem integração de fontes de dados.

Aparentemente, a construção de metadados, usando a taxonomia proposta, é praticamente inviável se feita manualmente, pois exigiria um enorme esforço pessoal na identificação dos valores a serem preenchidos em cada conceito. Assim, a eficácia das futuras ferramentas a serem desenvolvidas para a construção da taxonomia proposta depende diretamente da forma como os recursos são catalogados na Internet.

Como trabalhos futuros, pretende-se implementar um protótipo para validar a taxonomia proposta, através do teste da integração entre ambientes heterogêneas. Além disso, se planeja a implementação de um protótipo de um robô, que possa ser usado para descobrir os metadados (e seus valores) definidos, em cada fonte de dados (objeto) envolvida no ambiente de integração. Outra atividade planejada em um trabalho futuro, é um estudo mais aprofundado de trabalhos relacionados ao tema.

Referências

- Baptista, Dulce (2007). O Impacto dos Metadados na Representação Descritiva. Revista ACB: Biblioteconomia em SC, Florianópolis, v.12, n.2, p. 177-190, jul./dez
- Lines, Weibel. (2008) Metadata: Semantics; Structure; Syntax. February. Disponível em: <http://weibel-lines.typepad.com/weibelines/2008/02/metadata-semant.html>
- Lourenço, Cíntia de Azevedo (2005). Análise do Padrão Brasileiro de Metadados de Teses e Dissertações segundo o Modelo Entidade-Relacionamento. Tese (Doutorado na Escola de Ciência da Informação - UFMG), Belo Horizonte.
- Madmick, Stuart E. (1995) *From VLDB to VMLDB (Very MANY Large Data Base): Dealing with Large-Scale Semantic heterogeneity*, VLDB.
- Matos, Ely Edison da Silva (2008). CelOWS: Um Framework Baseado em Ontologias com Serviços Web para Modelagem Conceitual em Biologia Sistêmica. [dissertação]. Juiz de Fora (MG): Mestrado em Modelagem Computacional, UFJF.
- Moura, Ana Maria de C.; PEREIRA, Genelice da Costa and CAMPOS, María Luiza Machado (2002). A metadata approach to manage and organize electronic documents and collections on the web. J. Braz. Comp. Soc., vol.8, n.1, pp. 16-31.
- Passos, Rômulo Augusto Nogueira de O. (2006) Uma arquitetura para integração de dados baseada em ontologia. Centro de Informática – Universidade Federal de Pernambuco (UFPE).
- Siedler, Marcelo da Silveira. SOUZA. (2004) Fernando da Fonseca. *Sistema de Integração de Dados usando Técnicas de Web Semântica*. Centro de Informática – Universidade Federal de Pernambuco (UFPE).
- Taylor, Chris.(2003) An Introduction to Metadata. University of Queensland Library..

Interagindo com Data Warehouses Espaciais através de Descrições Semânticas¹

Renato Deggau^{1,2}, Renato Fileto¹

¹Programa de Pós-Graduação em Ciência da Computação (PPGCC)
Univ. Federal de Santa Catarina, Caixa Postal 476, 88.040-900, Florianópolis-SC

²Epagri – Empresa de Pesquisa Agropecuária e Extensão Rural de Santa Catarina
Rod. Admar Gonzaga 1347, Itacorubi, 88.034-901, Florianópolis-SC

{rdeggau, fileto}@inf.ufsc.br

Abstract. This paper describes the knowledge-based interface of S^2DW , a system that uses ontologies to support information analysis in spatial data warehouses (SDWs). S^2DW uses a domain ontology and an ontology of structures and resources for data manipulation in SDWs to semantically describe them. It supports searching for descriptions of SDWs related to some theme using domain-specific vocabulary, and allows the user to interact with such descriptions to specify SOLAP(Spatial OLAP) queries on the retrieved SDWs. The tables, graphs and maps generated by the system in response to such queries support additional SOLAP interactions.

Resumo. Este trabalho descreve a interface baseada em conhecimento do S^2DW , um sistema que usa ontologias para suportar análises de informação em data warehouses espaciais (SDWs). O S^2DW usa uma ontologia de domínio e uma ontologia de estruturas e recursos de manipulação de dados em SDWs para descrevê-los semanticamente. Ele suporta buscas por descrições de SDWs relacionados a algum tema, usando vocabulário específico de domínio, e permite ao usuário interagir com tais descrições para especificar consultas SOLAP sobre os SDWs correspondentes. As tabelas, gráficos e mapas gerados pelo sistema em atendimento a tais consultas suportam interações adicionais de OLAP espacial.

1. Introdução e Motivação

Data Warehouses Espaciais (*Spatial Data Warehouses - SDWs*) estendem data warehouses tradicionais com suporte a objetos geográficos [Rao et al. 2003; Rivest et al. 2005, Malinowsky and Zimányi 2007, Bimonte et al. 2007]. Objetos geográficos podem aparecer nas dimensões de um SDW (e.g., polígonos representando estados e cidades) ou como medidas na tabela fato (e.g., pontos onde ocorrem intoxicações por agrotóxicos). Assim, além de operadores OLAP tradicionais e funções de agregação de dados escalares, SDWs precisam suportar uma grande variedade de operadores e funções para a manipulação de objetos espaciais [Fidalgo 2005, Silva 2008].

Problemas em aberto na área de SDW incluem: (i) modelagem de SDWs [Malinowski and Zimányi 2007]; (ii) extração, transformação e carga de dados espaciais em SDWs [Skoutas and Simitsis 2006, Di Martino et al. 2009]; e (iii) interação do

¹ Este trabalho foi parcialmente financiado pela Fapesc (contrato 12552-2007-0) e pelo CNPq (contrato 48139212007-6), além de contar com o apoio da Epagri.

usuário com SDWs para busca e análise de informações [Rao et al. 2005, Sell et al. 2008, Xie et al. 2008].

Diversos autores reconhecem a necessidade de desenvolver recursos adequados para suportar a interação de usuários especialistas de domínio com data warehouses. Alguns trabalhos propõem o uso de semântica para esta finalidade [Sell et al. 2008, Xie et al. 2008]. Todavia, essas abordagens não contemplam aspectos espaciais.

Este trabalho foca na interação do usuário para encontrar informação relacionada a algum tema de interesse em SDWs e analisar a informação encontrada com recursos de OLAP espacial (*Spatial On-Line Analytical Processing – SOLAP*). O sistema proposto, denominado S²DW (*Semantic Spatial Data Warehouse*) [Deggau and Fileto 2009], descreve SDWs semanticamente usando mapeamentos entre uma ontologia de estruturas e recursos de manipulação de dados em SDW e uma ontologia de domínio. A ontologia de SDW é fixa. A ontologia de domínio varia de acordo com a aplicação, permitindo ao usuário interagir com o sistema segundo o conhecimento de domínio.

Na abordagem aqui proposta, o usuário especialista de domínio fornece palavras-chave ou navega no vocabulário específico de domínio para estipular suas buscas. O S²DW efetua buscas em sua base de conhecimento para retornar representações abstratas de SDWs relacionados ao tema buscado. Tais representações, na forma de grafos semanticamente enriquecidos para descrever estrutura e conteúdo de SDWs, descrevem componentes como tabelas fato, medidas, dimensões, níveis de dimensões, membros de níveis, operadores e funções de agregação de dados. Interagindo com tais representações abstratas de SDWs, o usuário pode efetuar consultas, mediante a seleção de medidas das tabelas fato, membros de diversos níveis das dimensões, além de operadores e funções para processamento da informação. O S²DW permite a visualização dos resultados de consultas em tabelas, gráficos e mapas. O usuário pode realizar interações SOLAP sobre os resultados retornados pelo S²DW, seja nas hierarquias das diversas dimensões das tabelas resultantes, ou clicando sobre os mapas e gráficos correspondentes.

O restante deste trabalho é organizado da seguinte maneira. A Seção 2 apresenta um SDW do domínio agrícola que servirá como estudo de caso para a apresentação das funcionalidades do sistema proposto. A Seção 3 descreve a arquitetura e o modelo conceitual do S²DW, isto é, a ontologia de SDW e uma ontologia do domínio agrícola, utilizada para ilustrar o funcionamento do sistema. A Seção 4 descreve a interação do usuário com o sistema, através de exemplos de análises de informação do setor agrícola. Finalmente, a Seção 5 discute os trabalhos relacionados e a Seção 6 apresenta algumas conclusões, a situação atual do trabalho e alguns temas para trabalhos futuros.

2. Um SDW do Domínio Agrícola

A Figura 1 apresenta um extrato do esquema dimensional de um SDW sobre propriedades agrícolas de Santa Catarina, produzido com dados da Epagri (Empresa de Pesquisa Agropecuária e Extensão Rural de Santa Catarina). A tabela fato deste SDW inclui as medidas escalares *ÁreaPlantada*, *ÁreaColhida*, *QuantidadeProduzida* e a medida geográfica *SedePropriedade* (coordenada geográfica da sede da propriedade). Tais medidas são agregadas segundo os níveis e membros das dimensões *Espaço* (com objetos geográficos representando o estado de Santa Catarina, suas regiões e municípios), *Tempo* e *Produto Agrícola*.

Este SDW possibilita análises de informação explorando componentes espaciais, isto é, que manipulam os objetos espaciais da dimensão *Espaço* e da medida *SedePropriedade*. Os objetos espaciais na forma de multi-polígonos, presentes na dimensão espaço, podem ser usados como parâmetros de operadores geográficos na seleção de informação e para plotar a informação em mapas temáticos. A agregação da medida espacial *SedePropriedade* pode ser feita pela montagem de coleções de sedes de propriedades ou aplicando funções de agregação espacial compatíveis com o tipo de dado coordenada geográfica e a semântica desta entidade geográfica (*geographic feature*). Os resultados das agregações desta medida geográfica também podem ser apresentados em mapas. Diferentes membros de dimensões e valores previamente agregados de medidas espaciais podem referenciar um mesmo objeto espacial com representação possivelmente extensa (e.g., perímetro de uma cidade, coleções de sedes de propriedades). Então a representação de cada objeto precisa ser armazenada em um único local e compartilhada através de referências, para evitar redundâncias.

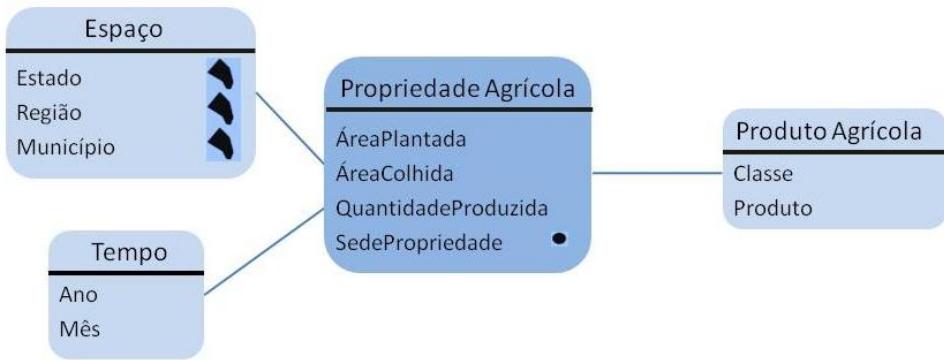


Figura 1. Esquema de um SDW sobre produção agrícola

Usuários especialistas de domínio, que frequentemente têm dificuldades ao interagir com data warehouses convencionais, manipulando somente dados escalares, encontram dificuldades adicionais ao se deparar com objetos espaciais e a grande variedade de recursos que podem ser usados para manipulá-los nas análises em SDWs.

3. O S²DW

O S²DW (*Spatial Semantic Data Warehouse*) é um sistema para suportar análises de informação em SDWs [Deggau e Fileto 2009]. Ele visa permitir ao usuário especialista de um domínio identificar SDWs relacionados a um tema de interesse e efetuar consultas nesses SDWs através de uma interface gráfica baseada em conhecimento. Interações adicionais com SOLAP podem ser realizadas sobre os resultados das consultas na forma de tabelas, gráficos ou mapas.

A Figura 2 ilustra a arquitetura do S²DW e destaca a função da sua interface gráfica. O S²DW utiliza a ontologia de SDW e a ontologia de domínio para descrever semanticamente a estrutura e o conteúdo das bases de dados convencionais e geográficos de SDWs. A ontologia de SDW é fixa e inclui definições conceituais de componentes estruturais de SDWs (dimensões, níveis, medidas) e seus relacionamentos, além da descrição de operadores e funções de manipulação de dados em SDWs. Ela foi criada pelos autores e é baseada na classificação de operadores e funções de agregação espacial proposta por [Silva, 2008]. A ontologia de domínio, que varia com a aplicação, permite descrever o conteúdo de SDWs utilizando vocabulário específico do domínio.

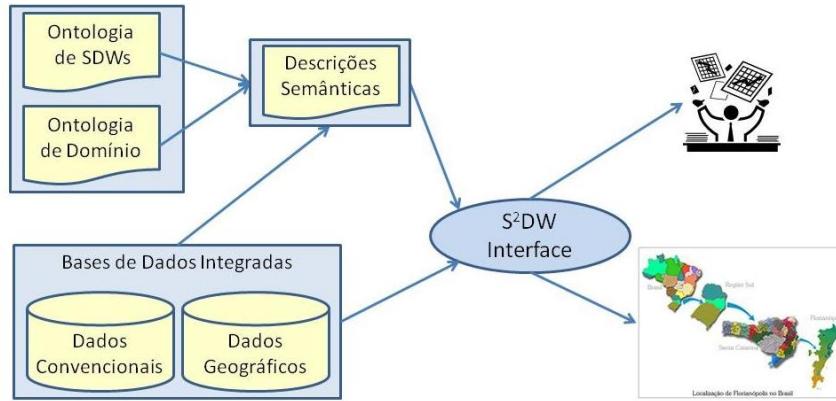


Figura 2. A arquitetura do S²DW e sua interface gráfica

3.1 Ontologia de SDW

A ontologia de SDW serve para descrever a estrutura de SDWs segundo o modelo dimensional com extensões espaciais. Ela inclui definições de conceitos como medidas, dimensões e níveis de dimensões, especificando como esses podem compor um SDW. Todos esses componentes podem ser escalares ou espaciais. Ela também classifica e descreve os operadores e funções de agregação de dados, tanto escalares como espaciais, indicando como eles podem ser usados, quais tipos de dados aceitam como parâmetro e que tipo de resultado geram.

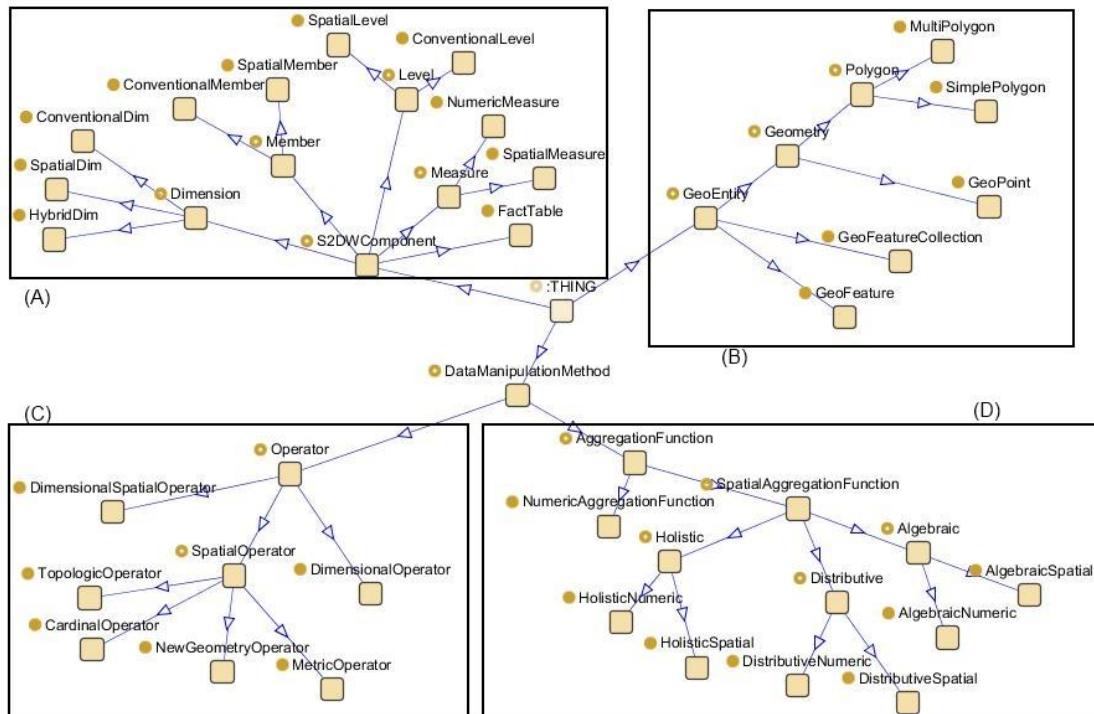


Figura 3. Ontologia de SDW

A figura 3 ilustra a hierarquia dos conceitos de nível mais alto desta ontologia, divididos em quatro porções: (A) os componentes estruturais de SDWs; (B) tipos de entidades espaciais; (C) operadores de manipulação de dados (usados para selecionar

informações) e (D) funções para agregação de dados (usados para agregar medidas de tabelas fato). Por questão de simplicidade e limitação de espaço, a figura 3 omite diversos conceitos dos níveis mais detalhados das hierarquias apresentadas e relacionamentos entre esses conceitos diferentes de IS_A (*subsumption*).

3.2 Ontologia de domínio

A ontologia de domínio descreve os conceitos existentes no domínio de aplicação em que o S²DW é usado. O S²DW é independente de domínio, permitindo adaptação mediante a substituição da ontologia de domínio. Como tal ontologia usa vocabulário específico de domínio, ela é usada para permitir ao usuário se expressar e interagir com o sistema numa linguagem que é do seu conhecimento. Isso facilita o processo de especificação das necessidades de informação pelo usuário. A figura 4 descreve uma porção de ontologia agrícola, utilizada neste trabalho, a título de ilustração.

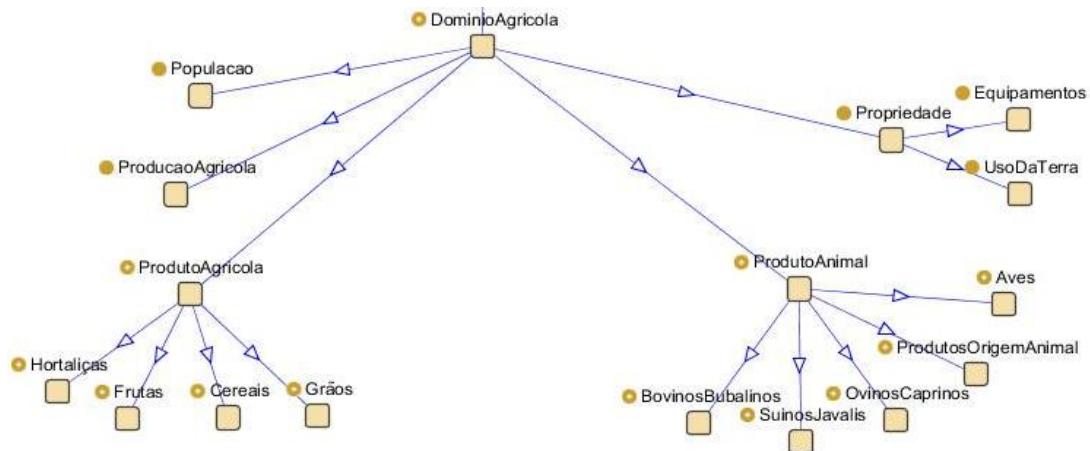


Figura 4. Trecho de ontologia do domínio agrícola

3.3 Descrições semânticas de SDWs

A descrição semântica de um SDW é baseada nas ontologias de domínio e de SDW. Um SDW pode ser descrito semanticamente através da definição de mapeamentos entre conceitos das duas ontologias [Xie *et al.* 2008]. Esses mapeamentos habilitam o sistema a informar e auxiliar o usuário no processo de busca, exploração e entendimento dos conteúdos do SDW e nos procedimentos de análise do conteúdo com operações SOLAP, promovendo uma interação mais efetiva com o sistema. A figura 5 apresenta algumas descrições baseadas em mapeamentos para o SDW descrito na figura 1.

Classe Ontologia Domínio	=	Classe Ontologia SDW
Área Plantada	=	Numeric Measure in SDW Agrícola
Área Colhida	=	Numeric Measure in SDW Agrícola
Quantidade Produzida	=	Numeric Measure in SDW Agrícola
Sede de Propriedade Agrícola	=	SpatialMeasure in SDW Agrícola GeoPoint
Produto Agrícola	=	ConventionalDim
Tempo	=	ConventionalDim
Espaço Geográfico	=	SpatialDim
Município	=	SpatialLevel of Espaço Geográfico SimplePolygon
Região	=	SpatialLevel of Espaço Geográfico MultiPolygon

Figura 5: Descrições semânticas de componentes de um SDW

4. Busca e análise de informação com o S²DW

A interface gráfica do S²DW permite a busca e a análise de informação de SDWs em três etapas:

1) Busca de SDWs com informação sobre determinado(s) assunto(s), através de consultas por palavras-chave ou navegação em uma visão da ontologia de domínio.

2) Especificação de consultas através da interação com uma descrição semântica da estrutura e conteúdo de um SDW em forma de grafo semanticamente enriquecido.

3) Apresentação de resultados e navegação adicional nas informações do SDW via *OLAP* dimensional.

Para a criação desta interface, os mapeamentos entre os conceitos das ontologias foram especificados manualmente.

O uso do S²DW é descrito a seguir, através de um estudo de caso do setor agrícola que demonstra os procedimentos executados por um usuário do sistema para efetuar busca e análises de informações.

4.1 Estudo de caso

A Figura 6 apresenta a tela inicial do S²DW customizada com uma ontologia agrícola. Esta tela inclui um campo para entrada de palavras-chave, uma visão das hierarquias presentes na ontologia de domínio e um quadro para apresentar descrições semânticas de conceitos ou instâncias presentes na base de conhecimento do S²DW. O usuário inicia o uso do sistema informando uma palavra-chave ou navegando na hierarquia de conceitos do domínio para encontrar um SDW que contenha informação sobre um determinado assunto.

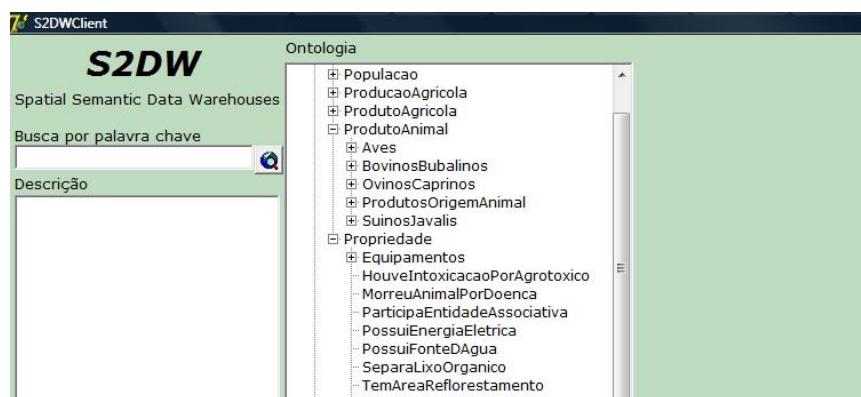


Figura 6. Busca e navegação na ontologia de domínio

Exemplo: Analisar a distribuição geográfica da produção de maçã, no estado de Santa Catarina, no ano de 2005, em uma tabela e um mapa.

Suponha que o usuário informe a palavra-chave “produção” ou selecione esta palavra na visão da ontologia de domínio. Quando ele solicita a execução da consulta com esta palavra chave-chave, ela é pesquisada usando inferência nas ontologias e descrições semânticas de SDWs. Então, cada descrição semântica de SDW que esteja relacionada semanticamente à palavra-chave informada aparece na forma de um grafo em uma aba que pode ser visualizada na porção direita da tela do S²DW. A Figura 7 ilustra a descrição do SDW *Produção Agrícola*, com a tabela fato *Propriedade*

Agrícola, ligada às dimensões *Produto Agrícola*, *Espaço* e *Tempo*. A palavra-chave “produção” não é mencionada diretamente nesta descrição de SDW, mas os termos “Área Plantada”, “Área Colhida” e “Quantidade Produzida”, usados para rotular medidas na tabela fato, estão relacionados a produção agrícola na ontologia de domínio, o que torna possível a recuperação deste SDW em atendimento à consulta.

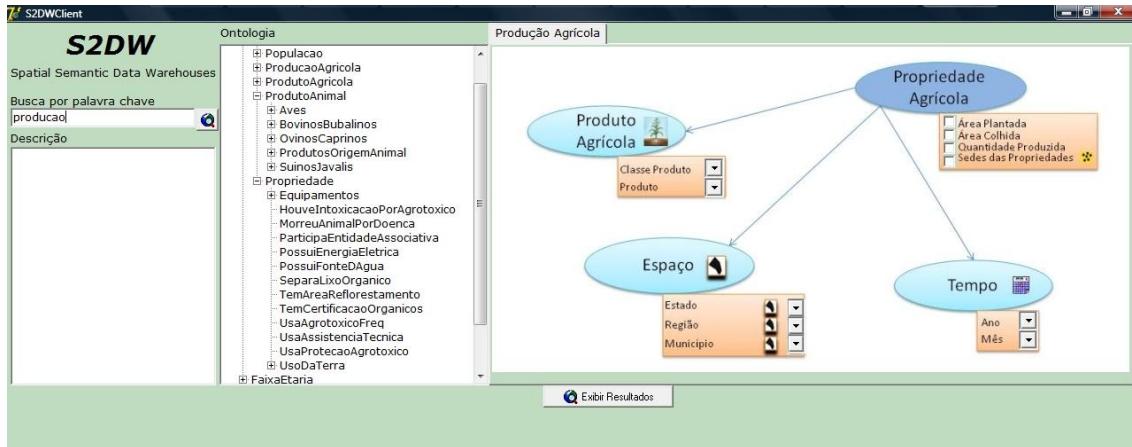


Figura 7: Apresentação da descrição semântica do SDW

Na próxima etapa, o usuário pode selecionar níveis e membros nas dimensões da SDW para especificar agrupamentos e seleções de informação, respectivamente. A Figura 8 ilustra a seleção da classe de produtos *Frutas* e produto *Maçã* para especificar o foco de interesse na dimensão *Produto Agrícola*.

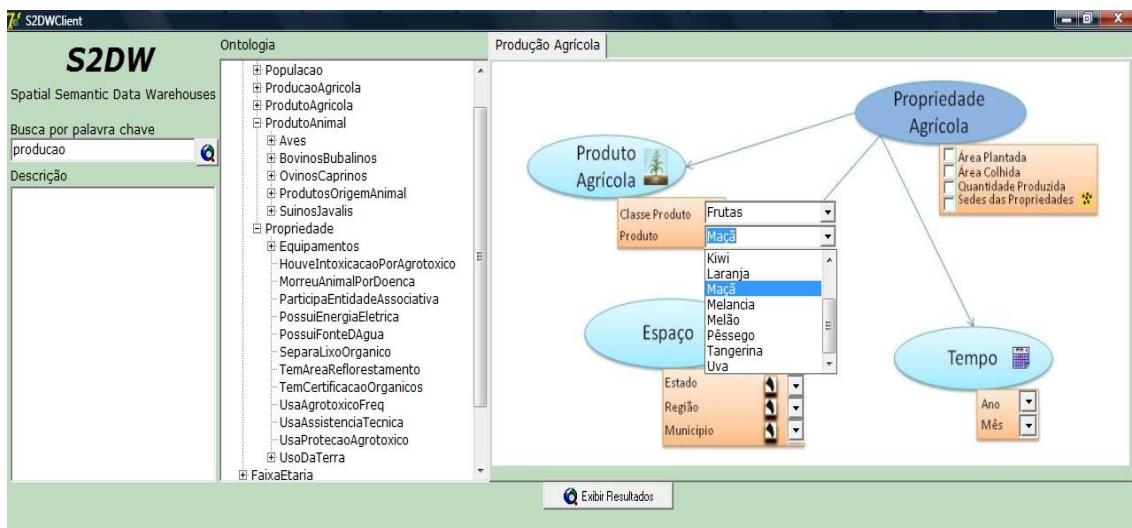


Figura 8. Seleção do produto

A seguir, o usuário pode selecionar o ano de 2005 na dimensão *Tempo*, o estado de *Santa Catarina* na dimensão *Espaço* e então selecionar a medida “Quantidade Produzida”. A ordem das seleções pode variar, em um processo no qual o usuário define uma consulta sobre a interface gráfica gradativamente. Concluídas as seleções desejadas, o usuário clica no botão *Exibir Resultados* para que o sistema efetue a consulta SOLAP correspondente e apresente as informações necessárias, como ilustrado na Figura 9. Os resultados são apresentados em uma tabela com valores numéricos e também em um mapa temático, que facilita a visualização e o entendimento rápido das informações. Tanto a tabela de resultados quanto o mapa, podem ser redimensionados

para melhor visualização do seu conteúdo. Eles também permitem interações adicionais de análise de informação, pela especificação de operações SOLAP sobre as informações apresentadas (i.e., clicando na tabela ou no mapa apresentados na tela).

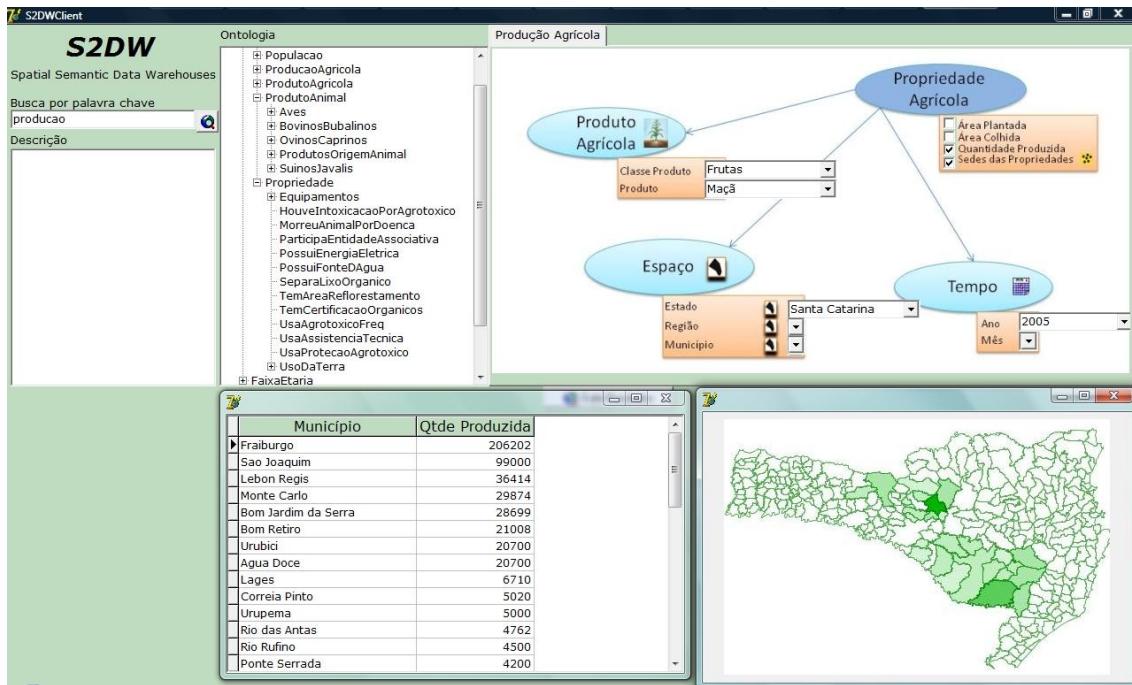


Figura 9. Produção de maçã no estado de Santa Catarina no ano de 2005

5. Trabalhos relacionados

Diversos trabalhos citam as dificuldades enfrentadas por usuários com pouco conhecimento de tecnologia da informação para efetuarem consultas sobre bancos de dados em geral. Eles propõem diferentes abordagens para os usuários visualizarem a estrutura e o conteúdo de bases de dados e especificarem consultas sobre as mesmas [Rishe *et al.* 2000, Terwilliger *et al.* 2007, Spahn *et al.* 2009]. Alguns outros trabalhos consideram especificamente o uso de semântica para suportar análise de informação em data warehouses [Sell *et al.* 2008, Xie *et al.* 2008, Diamantini and Potena 2008].

[Sell *et al.* 2008] usam ontologias e inferência para integrar a semântica do negócio com os dados dimensionais, de modo a suportar serviços de análise de informação. [Xie *et al.* 2008] usam uma extensão de OWL para especialistas em TI construírem mapeamentos entre o esquema de um data warehouse e termos utilizados no modelo de análise de informação, com o objetivo de permitir aos usuários especialistas de domínio especificar suas necessidades de análise e gerar automaticamente data marts para atendê-las. [Diamantini and Potena 2008] propõem um modelo para anotação semântica de data warehouses, que leva em consideração uma ontologia de domínio e uma ontologia matemática para descrever consultas. Entretanto, nenhum desses trabalhos apresenta interfaces visuais para o usuário efetuar consultas dimensionais, nem considera objetos espaciais e sua manipulação.

6. Conclusões e trabalhos futuros

O S²DW é um sistema baseado em ontologias para descrever a estrutura, o conteúdo e os recursos de análise de informação de SDWs. Este trabalho foca na interface provida

pelo S²DW para o usuário efetuar busca e a análise de informações sobre um tema de interesse. Tal interface inclui: (i) buscas por palavras-chaves ou navegação em uma visão da ontologia de domínio para identificar SDWs relacionados a um tema de interesse; (ii) uma representação de cada SDW de interesse em forma de grafo semanticamente enriquecido, que permite ao usuário especificar consultas sobre o SDW; e (iii) interações adicionais sobre os resultados de consultas, na forma de tabelas, gráficos e mapas, para o usuário realizar SOLAP sobre esses resultados.

O S²DW fornece abstrações baseadas em conhecimento em interfaces gráficas avançadas, para permitir o uso dos recursos de análise de informação do SDW. O principal benefício esperado com a proposta aqui descrita é facilitar a interação do usuário especialista de domínio com o SDW. Tais benefícios se aplicam a data warehouses convencionais, mas são ampliados em SDWs, nas quais objetos e recursos de manipulação de dados espaciais podem tornar as análises mais complexas.

Questões como a construção de mapeamentos entre a ontologia de SDW e a ontologia de domínio para descrever componentes de SDWs específicos, a geração de data marts, a tradução e a execução de consultas, além de detalhes do SOLAP realizado sobre os resultados de consultas, estão fora do escopo deste trabalho. Elas são contempladas em diversos outros trabalhos.

O próximo passo deste trabalho é o desenvolvimento de interfaces baseadas em conhecimento para apoiar o usuário no uso correto de operadores e funções de agregação de dados espaciais. Posteriormente, pretende-se validar a proposta em experimentos empíricos com usuários do setor agrícola tentando atender necessidades de análise deste domínio através da interface proposta. O objetivo desta validação, que deve se estender a outros domínios de aplicação, é coletar subsídios para aperfeiçoar a proposta, antes de partir para a implementação das diversas funcionalidades do S²DW em trabalhos futuros.

Referências

- Bimonte, S., Tchounikine, A., Miquel, M. (2007) GeWOLap: Spatial OLAP: Open Issues and a Web Based Prototype. In: *10th AGILE International Conference on Geographic Information Science*, Aalborg University, Denmark.
- Deggau, R. and Fileto, R. (2009). Enriquecendo Data Warehouses Espaciais com Descrições Semânticas. In: SBC - Workshop de Teses e Dissertações em Bancos de Dados (WTDBD). Fortaleza, Brasil, 61-66.
- Di Martino, S, Bimonte, S., Bertolotto, M., Ferrucci, F. (2009) Integrating Google Earth within OLAP Tools for Multidimensional Exploration and Analysis of Spatial Data. In: Intl. Conf. On Engineering of Information Systems (ICEIS), 940-951.
- Diamantini, C. and Potena, D. (2008) Semantic enrichment of strategic datacubes. In Proceeding of the ACM 11th international Workshop on Data Warehousing and OLAP (DOLAP). ACM, New York, 81-88.
- Fidalgo, R. (2005). Uma Infra-estrutura para Integração de Modelos, Esquemas e Serviços Multidimensionais e Geográficos. Tese de Doutorado. Centro de Informática – UFPE.

- Malinowski, E. and Zimányi, E. (2007). Logical Representation of a Conceptual Model for Spatial Data Warehouses. *Geoinformatica*. 11(4), 431-457.
- Rao, F., Zhang, L., Yu, X. L., Li, Y., and Chen, Y. (2003). Spatial hierarchy and OLAP-favored search in spatial data warehouse. In Proceedings of the 6th ACM international Workshop on Data Warehousing and OLAP (DOLAP). ACM, New York, NY, 48-55.
- Rishe, N., Yuan, J., Athauda, R., Chen, S., Lu, X., Ma, X., Vaschillo, A., Shaposhnikov, A., and Vasilevsky, D. (2000) Semantic Access: Semantic Interface for Querying Databases. In Proceedings of the 26th international Conference on Very Large Data Bases (VLDB). Morgan Kaufmann Publishers, San Francisco, CA, 591-594.
- Rivest, S., Bedard, Y., Proulx, M.J., Nadeau M., Hubert F., Pastor J. (2005) SOLAP technology: Merging business intelligence with geospatial technology for interactive spatio-temporal exploration and analysis of data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 60 (1), 17-33.
- Sell, D., da Silva, D. C., Beppler, F. D., Napoli, M., Ghisi, F. B., Pacheco, R. C., and Todesco, J. L. (2008). SBI: a semantic framework to support business intelligence. In Proceedings of the First international Workshop on ontology-Supported Business intelligence (OBI). ACM, New York.
- Silva, J. (2008). GEOMDQL: Uma linguagem de consulta geográfica e multidimensional. Tese de Doutorado. Centro de Informática – Universidade Federal de Pernambuco.
- Skoutas, D. and Simitsis, A. (2006). Designing ETL processes using semantic web technologies. In Proceedings of the 9th ACM international Workshop on Data Warehousing and OLAP (DOLAP). ACM, New York, 67-74.
- Spahn, M., Kleb, J., Grimm, S., Scheidl, S. (2008). Supporting business intelligence by providing ontology-based end-user information self-service. In Proceedings of the First international Workshop on ontology-Supported Business intelligence (OBI), ACM, New York.
- Terwilliger, J. F., Delcambre, L. M., Logan, J. (2007). Querying through a user interface. *Data and Knowledge Engineering*. 63 (3), 774-794.
- Xie, G., Yang, Y., Liu, S., Qiu, Z., Pan, Y., Zhou, X. (2008). EIAW: Towards a Business-Friendly Data Warehouse Using Semantic Web Technologies. LNCS 4825, The Semantic Web, 857-870.

RYLY - Query Analyzer: Ferramenta de Visualização e de Análise do Plano de Execução de Consultas ORACLE

Luzia A. Mendes¹, Rodrigo C. Barros², Bruno Visioli¹, Leandro Pompermaier¹

¹Faculdade de Informática - Pontifícia Universidade Católica do Rio Grande do Sul
Porto Alegre, RS - Brazil

²Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo
São Carlos, SP - Brazil

*leandro.pompermaier@pucrs.br, {luzia.mendes,bruno.visioli}@acad.pucrs.br
rcbarros@icmc.usp.br*

Abstract. *The Oracle database management system owns an internal query optimizer that generates an execution plan to each query that is executed. This execution plan is stored in a specific table (plan_table) and contains information on the operations that were executed, on the order they occur and on the operational cost of each of them. Properly understanding this execution plan enables the user to optimize each query once he knows a few optimization rules. The goal of this work is to facilitate both the visualization of the execution plan and its analysis.*

Resumo. *O Sistema de Gerenciamento de Banco de Dados Oracle possui um otimizador interno de consultas que gera um plano de execução para cada consulta executada. Esse plano é armazenado em uma tabela especial (plan_table) que contém informações quanto às operações executadas, à ordem em que elas acontecem e ao custo operacional de cada uma. O entendimento desse plano de execução de consultas auxilia o usuário a otimizá-las, mas, para isso, este precisa conhecer algumas regras de otimização. O objetivo deste trabalho é facilitar tanto a visualização do plano de execução de consultas quanto a análise deste.*

1. Introdução

As consultas em bancos de dados, assim como a criação e a manipulação de tabelas, são escritas utilizando a linguagem Structured Query Language (SQL) e possuem cláusulas de restrição e de junção. A rapidez e o baixo custo da resposta às consultas dependem diretamente de quão bem estruturadas estão suas cláusulas. Cada cláusula condicional ou de junção pode ser mapeada e organizada a fim de estruturar o plano de execução da consulta. Desta forma, ter a possibilidade de testar as diferentes formas de se estruturar uma consulta é uma necessidade de quem trabalha com bancos de dados.

Os Sistemas de Gerenciamento de Banco de Dados (SGBD) mais modernos organizam planos de execução de consultas antes de executá-las e calculam qual é o melhor plano. Esse processamento prévio dos planos de execução consome recursos do SGBD e segue regras definidas no otimizador de consultas do banco de dados [Watson 2009]. Por isso, é preciso que o responsável pelo código da consulta saiba como utilizar tais regras em

seu favor. Neste ponto, nota-se uma carência de ferramentas que auxiliem os profissionais - ou os estudantes - que utilizam SQL em bancos de dados Oracle a analisarem suas consultas de forma eficiente, uma vez que a função de visualização dos planos de execução de consultas disponibilizadas pela maioria das ferramentas de desenvolvimento apenas exibem textualmente as saída da tabela *plan_table* - utilizada para armazenar as operações executadas pelo otimizador. Sendo assim, para facilitar o entendimento das informações contidas no plano, é interessante que o profissional consiga visualizar sua consulta graficamente.

Com o objetivo de atender tal necessidade, foi construída a ferramenta RYLY, cujo *design* foi baseado em importantes preceitos da Interação Humano-Computador (IHC). A ferramenta permite que o usuário mapeie e identifique o comportamento do código escrito. Ele também pode avaliar quão satisfatório está o desempenho da consulta e testar modificações a fim de verificar se há melhoria em desempenho. Também é possível salvar a consulta que pretende-se executar. Por fim, a ferramenta RYLY apresenta orientações que seguem regras de *tuning* de consultas [Burleson and Danchenkov 2006] para que o usuário melhore a execução de suas consultas.

2. Ferramentas Similares

A opção de visualização de consultas é implementada em algumas ferramentas de desenvolvimento em bancos de dados. Porém, tais ferramentas não apresentam funcionalidades com o objetivo de facilitar o entendimento do plano de execução e de proporcionar explicações quanto às operações executadas.

Duas das ferramentas de desenvolvimento que implementam a visualização do plano de execução foram analisadas: o PL/SQL Developer, da allroundautomations [ALLROUNDAUTOMATIONS 2009] e o Aqua, da AquaFold [AquaFold 2009]. Ambas as análises utilizaram o mesmo banco de dados (Oracle 10g) e os softwares foram instalados no mesmo computador. Foram considerados seis fatores considerados importantes para esta análise, sendo que quatro são heurísticas consagradas de IHC.

2.1. Fatores analisados

Considerando as heurísticas estabelecidas por [Nielsen 1993], foram selecionadas as mais relevantes para o problema tratado.

- Correspondência entre o sistema e o mundo real - define que o sistema deve utilizar as mesmas palavras, expressões e conceitos da linguagem natural do usuário, em vez de utilizar termos orientados ao sistema. O projetista deve seguir as convenções do mundo real, fazendo com que a informação apareça em uma ordem natural e lógica.
- Visibilidade do estado do sistema - define que o sistema deve sempre manter os usuários informados sobre o que está acontecendo em termos de processamento através de *feedback* adequado e no tempo certo. Enquanto o sistema estiver processando alguma informação, o usuário deverá ser informado, inclusive, de quanto deverá aguardar até que receba algum retorno. O usuário também deve ser informado quanto a falhas no sistema.
- Reconhecimento em vez de lembrança - define que o projetista deve tornar os objetos, ações e opções visíveis. O usuário não deve ter de se lembrar de informações

relevantes de uma tela de diálogo para outra. As instruções de uso do sistema devem estar visíveis ou facilmente acessíveis sempre que necessário.

- Consistência e padronização, que define que os usuários não devem ter de se perguntar se palavras, situações ou ações diferentes significam a mesma coisa. O projetista deve seguir as convenções da plataforma ou do ambiente.

Além destas heurísticas, dois fatores foram considerados importantes para a análise das ferramentas similares.

- Utilização de imagens para representar os objetos e operadores - a árvore de consultas deve ser representada por imagens que representam os objetos e os operadores utilizados na consulta. Essas imagens devem ter fácil identificação com os objetos que representam.
- Tempo de processamento - o tempo de processamento deverá ser inferior a 3 segundos por tabela utilizada na consulta.

2.2. Tabela de Relevância

A Tabela 1 apresenta pesos que indicam a relevância do critério para a análise em questão, onde 5 é o mais relevante e 1 é o menos.

Tabela 1. Relevância dos fatores observados

Fator observado	Peso (1-5)
Correspondência	3
Visibilidade	2
Reconhecimento	5
Consistência	5
Uso de imagens	5
Tempo de Processamento	3

O uso de imagens, bem como as heurísticas de reconhecimento e de consistência, são os critérios mais relevantes desta análise, pois para se ter um bom entendimento do plano de execução de consultas, deve-se utilizar simbologias para representar os objetos manipulados bem como as manipulações em si.

2.3. Resultado da Análise

Para caracterizar os erros das heurísticas de IHC, foram utilizados os níveis de severidade definidos por [Nielsen 1993]. Para os demais fatores, foram utilizadas as classificações: *não possui*, *pouco implementado* e *satisfatório*. A Tabela 2 mostra o resultado da análise dos fatores importantes.

As ferramentas não possuem imagens que representem as informações presentes na tabela *plan_table*, o que faz com que as heurísticas de correspondência e reconhecimento não sejam atendidas satisfatoriamente. O Aqua utiliza algumas figuras para simbolizar as operações do otimizador, mas, por serem pequenas e repetitivas, o erro em IHC foi considerado grande.

Tabela 2. Resultado da análise dos fatores importantes, seguindo os níveis de severidade de [Nielsen 1993] para classificar as falhas encontradas.

Heurística/Fator	PL/SQL Developer	Aqua
Correspondência	Catastrófico	Grande
Visibilidade	Catastrófico	Catastrófico
Reconhecimento	Catastrófico	Catastrófico
Consistência	Não possui	Pequeno
Uso de imagens	Não possui	Pouco implementado
Tempo de processamento	Satisfatório	Satisfatório

Quanto à visibilidade do sistema, nenhuma das ferramentas possui mecanismos que ofereçam *feedback* ao usuário quanto ao processamento da requisição. No entanto, como essas requisições são atendidas imediatamente após sua execução em um ambiente normal, não caracteriza prejuízo para o usuário a inexistência deste feedback. Ainda assim, a falha foi considerada catastrófica, pois em um ambiente problemático onde a requisição não é atendida - ou demore a ser processada - o usuário não perceberá se o sistema está operante.

A falha de consistência no Aqua foi considerada pequena, pois a ferramenta utiliza uma mesma figura para representar diferentes operações. O PL/SQL Developer não utiliza imagens para representar o plano de execução, enquanto o Aqua utiliza um conjunto muito pequeno de figuras. Sendo assim, foram avaliados respectivamente como "não possui" e "pouco implementado" no fator "Uso de Imagens". Quanto ao tempo de processamento, ambos foram bem avaliados pois apresentam o resultado do plano no mesmo instante da execução.

Outro fator considerado ruim no modo de apresentação do plano de execução de consulta foi a forma como as informações da tabela são organizadas. Considerando que o plano pode ser representado por uma árvore de menus, conclui-se que o nodo pai é aquele mais à esquerda e que seus filhos estão identados um nível à direita. Esta forma de visualização dificulta a visualização global de como o plano está sendo executado, pois fica difícil perceber quais nodos estão em um mesmo nível.

3. Solução

A ferramenta RYLY exibe planos de execução de consultas para banco de dados Oracle. Nela, pode-se verificar todas as informações contidas na tabela *plan_table*, além de permitir a visualização de orientações relevantes para o processo de *tuning* de consultas. Essas informações foram selecionadas considerando-se, essencialmente, o livro de [Burleson and Danchenkov 2006] sobre otimização de consultas em bancos Oracle. Também é possível observar visões típicas da arquitetura Oracle que exibem informações sobre os dados armazenados no banco, tais como índices e sinônimos [Watson 2009].

O objetivo da ferramenta é permitir que o usuário analise o desempenho de sua consulta, aqui considerado como o menor custo de memória e de CPU. Além disso, este pode utilizar as orientações de *tuning* para melhorar o desempenho de suas consultas. Tais orientações não são executadas automaticamente, nem alteram o código da consulta.

As ferramentas que implementam a funcionalidade de exibir o plano de execução de consultas não trabalham as informações com imagens que facilitem o entendimento pelo usuário, sendo que apenas exibem em tela as informações textuais contidas na tabela *plan_table*. Assim, para entender o plano, o usuário deve conhecer os campos da tabela e entender o que os termos utilizados significam. O fato das informações estarem identadas para a direita de acordo com a ordem dos acontecimentos - como apresentado na Figura 1 - dificulta ainda mais o entendimento das informações do plano.

Description	Object owner	Object name	Cost	Cardinality	Bytes
SELECT STATEMENT, GOAL = ALL_ROWS			8	10	410
HASH GROUP BY			8	10	410
HASH JOIN			7	40	1640
TABLE ACCESS FULL	TESTE	ALUNOS	3	11	352
TABLE ACCESS FULL	TESTE	HISTORICO	3	40	360

Figura 1. Exemplo da exibição do plano de execução de consultas nas ferramentas similares analisadas

O diferencial da ferramenta RYLY é a forma com que as informações são preparadas antes de serem exibidas ao usuário. Um software analisador de consultas SQL sobre o banco de dados Oracle foi desenvolvido em Java para permitir tal funcionalidade.

3.1. Visualização das informações

O sistema apresenta uma árvore que representa o plano de execução seguido pelo otimizador e o custo de cada consulta. Essa árvore é ilustrativa, à medida que representa as tabelas e as operações executadas pelo otimizador com uma simbologia definida.

Para representar a seqüência das ações que serão executadas pelo otimizador do banco de dados para realizar uma consulta, foram utilizadas orientações de projeto de visualização de informações [Schneiderman 1998] para definição de que a árvore seria binária e invertida, de forma que o nodo pai represente o resultado da consulta e fique na parte inferior da imagem. Do mesmo modo, as tabelas acessadas são os nodos folhas e ficam na parte superior da imagem. Assim, os níveis representam a ordem em que as operações são executadas. Esta forma de visualizar a árvore do plano de execução de consultas facilita o entendimento da ordem na qual as operações são executadas (de cima para baixo), fazendo com que o usuário consiga ter um bom entendimento do plano gerado para sua consulta.

Quando o plano é apresentado, o usuário pode visualizar globalmente como a consulta foi estruturada, conforme a principal regra de visualização de informações de [Schneiderman 1998]. Após, é possível explorar as demais funcionalidades da ferramenta - como os *tooltips* que trazem informações sobre as operações ou sobre as tabelas representadas por cada nodo. Também é possível visualizar o resultado da consulta executada. Desta forma, o foco da ferramenta é conseguir exibir uma visão global do plano de execução sem sobrecarregar a imagem com informações, facilitando o entendimento do usuário sobre o plano.

A ferramenta utiliza, nos nodos, símbolos diferentes para representar as operações executadas pelo plano. Os símbolos foram concebidos representativamente, de forma

que seu desenho contribui para a identificação e para o entendimento do significado da operação representada pelo nodo.

Tendo em vista esses conceitos, ficam definidas as seguintes características para a visualização do plano de execução de consultas:

1. O plano é apresentado como uma árvore binária;
2. O plano é apresentado de forma completa e única;
3. O usuário pode detalhar o plano para verificar questões específicas;
4. Os nodos da árvore representam as operações executadas pelo otimizador ou as tabelas acessadas;
5. As conexões entre os nodos indicam a dependência entre estes e o custo da operação filho;
6. Os níveis da árvore representam a ordem de execução das operações, sendo o último nível (exibido na parte superior da imagem) as primeiras operações executadas;
7. A ordem de execução das operações está disposta de cima para baixo;
8. Os nodos possuem informações extras visíveis ao usuário conforme sua vontade;
9. Informações importantes relacionadas a configurações do banco de dados estão expostas de forma fixa na parte inferior da *interface*;
10. São utilizados termos técnicos para identificar as operações da execução;
11. São utilizadas imagens concebidas com o objetivo de aproximar o comportamento das operações do plano de execução à linguagem natural do usuário;
12. A visualização correta da imagem é independente das configurações de tela;
13. As informações sobre o plano são apresentadas de forma concisa em primeiro plano (apenas as tabelas, as junções entre elas e uma indicação da existência de informações específicas);
14. É permitida ao usuário a seleção das informações que desejar visualizar;
15. É possível, a qualquer momento, visualizar material de ajuda ao entendimento da simbologia e dos termos utilizados.

3.2. Regras de *tuning*

A execução de consultas segue regras matemáticas [Elmasri 2006] e depende da correta utilização das operações que o SGBD pode fazer. Para tanto, é preciso que se conheça quais são estas regras e como se comportam tais operações.

A atividade de melhoria do desempenho de consultas (*query tuning*) envolve a arquitetura do ambiente no qual o banco de dados está instalado, a modelagem das tabelas utilizadas e a forma como a consulta foi escrita. A ferramenta RYLY analisa esta última questão ao utilizar regras de *tuning* para avaliar se há a possibilidade de melhoria do desempenho da consulta.

O foco das regras de *tuning* implementadas pela ferramenta é a apresentação de orientações para que o usuário identifique possibilidades de melhoria do desempenho de sua consulta. Estas regras são definidas a partir de estratégias de *tuning* de consultas [Burleson and Danchenkov 2006] e apresentam orientações que se dividem em atividades e comandos a serem executados.

A ferramenta exibe um texto de ajuda ao usuário que indica, de acordo com a regra em questão, um conjunto de atividades e de comandos a serem executados com

a finalidade de melhorar o desempenho da consulta. Dessa forma, a ferramenta trará dicas para que o usuário melhore tanto sua consulta quanto a arquitetura ou a modelagem da tabela. Essas regras irão sugerir ao usuário que identifique, por exemplo, que uma consulta está percorrendo uma tabela a partir de uma coluna que não possui índice e irá sugerir que o usuário crie um. Para isso, irá indicar as atividades - consultas no banco de dados ou no sistema operacional - e os comandos a serem executados - também no banco de dados ou no sistema operacional.

As atividades têm como objetivo analisar tanto a arquitetura do ambiente em que o banco está instalado quanto a modelagem das tabelas acessadas na consulta. A arquitetura do ambiente é analisada, na ferramenta RYLY, à medida que se verifica alguns aspectos específicos - tais como a configuração do uso da memória. Já a análise da modelagem das tabelas acessadas é verificada com consultas às visões do dicionário de dados Oracle (o que pode ser feito na própria ferramenta).

Além das atividades, o usuário recebe a orientação de quais comandos deve executar - dependendo dos resultados obtidos com a realização das atividades. Esses comandos podem afetar tanto o banco de dados, quanto o ambiente em que está instalado.

4. Interface do RYLY

Para utilizar a ferramenta RYLY, o usuário precisa logar-se a um banco de dados através da *interface* da ferramenta - Figura 2. Assim, esta executa a operação *explain plan for* para a consulta submetida pelo usuário. Caso a tabela *plan_table* não exista no banco de dados, uma tabela similar é criada sob o usuário utilizado para login, e o usuário é informado desta criação.

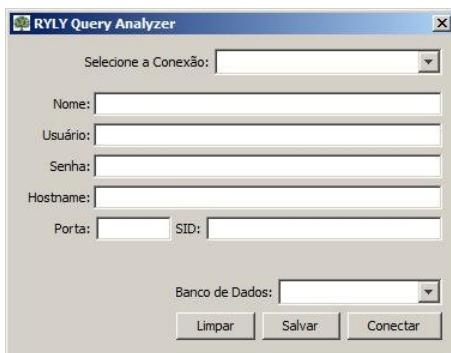


Figura 2. Tela de *login* do sistema

A ferramenta faz uma consulta sobre a tabela *plan_table*, referente ao plano de execução da consulta submetida, e trata as informações de forma a exibir um grafo que representa a execução das operações de consulta pelo otimizador - Figura 3. Antes da exibição do grafo do plano de execução, a RYLY processa informações relevantes às tabelas acessadas e as organiza em forma de *tooltips*, que são exibidas quando o usuário clica sobre o símbolo que representa a tabela em questão - Figura 4. Da mesma forma, são exibidas informações sobre as operações que compõem o plano.

Caso o usuário não fique satisfeito com as informações das *tooltips*, poderá visualizar a tabela *plan_table* completamente - Figura 5. Quando a ferramenta encontra uma regra de *tuning* para uma operação, marca tal operação com um círculo vermelho.

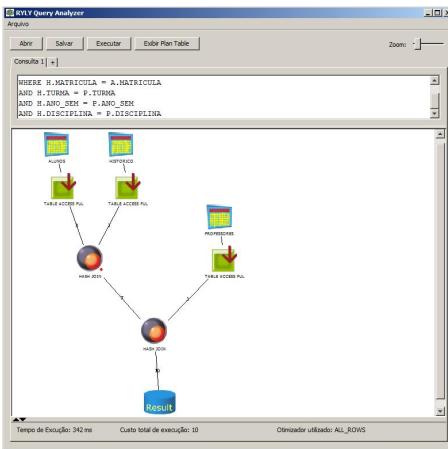


Figura 3. Árvore do plano de execução de consulta

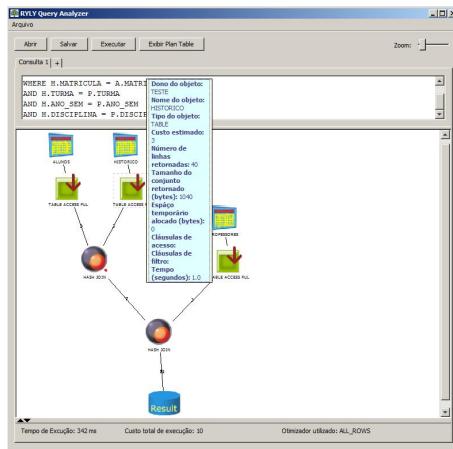


Figura 4. Exibição do *tooltip* do nodo "Table Access

O usuário visualiza esta regra ao clicar sobre a operação marcada - Figura 6. Para obter outras informações de uma tabela, o usuário deve clicar sobre o nodo que a representa e selecionar uma das visões Oracle disponíveis - Figura 7. Desta forma, poderá explorar a modelagem da tabela.

Também é possível visualizar o resultado da execução da consulta, o tempo de execução e o custo para executá-la - Figura 8. Assim, o usuário pode escrever uma consulta de diferentes formas e comparar seus planos de execução. Essa comparação se dá de maneira simplificada com os artifícios gráficos presentes na ferramenta.

5. Conclusões e Trabalhos Futuros

Os planos de execução de consultas são explorados por ferramentas de utilização de bancos de dados, mas não são exibidos graficamente. Para garantir um maior entendimento dos planos através de sua exibição gráfica, foram definidas heurísticas de IHC e fatores importantes para verificar a eficiência dessas ferramentas. Posteriormente, a ferramenta RLY foi submetida à mesma verificação.

O objetivo deste trabalho foi facilitar o entendimento do plano de execução de consultas utilizando, como estratégia, a exibição gráfica do plano de execução. A ferra-

Registros da Plan Table					
STATEMENT...	PLAN_ID	TIMESTAMP	REMARKS	OPERATION	OPTIONS
728	2009-11-29 ...		SELECT STA...		
728	2009-11-29 ...		HASH JOIN		
728	2009-11-29 ...		HASH JOIN		
728	2009-11-29 ...		TABLE ACCESS FULL		
728	2009-11-29 ...		TABLE ACCESS FULL		
728	2009-11-29 ...		TABLE ACCESS FULL		

Figura 5. Exibição completa da *plan_table*

menta segue padrões de IHC e facilita o entendimento exibindo, inicialmente, o plano de maneira geral e permitindo que o usuário explore informações relevantes do plano posteriormente. O usuário também pode visualizar o resultado de sua consulta no banco de dados e a tabela *plan_table* conforme esta está armazenada.

Referências

- ALLROUND AUTOMATIONS (2009). Real solutions for oracle developers.
- Aquafold (2009). [site organizacional].
- Burleson, D. and Danchenkov, A. B. (2006). *Oracle Tuning: The Definitive Reference*. Rampant TechPress.
- Elmasri, R. (2006). *Sistemas de banco de dados*. Pearson Education.
- Nielsen, J. (1993). *Usability engineering*. Boston: AP Professional.
- Schneiderman, B. (1998). *Designing the user interface: strategies for effective human-computer interaction*. Addison-Wesley.
- Watson, J. (2009). *Oracle Database 10, Certificação OCP: guia completo para o exame*. AltaBook, Jacaré, RJ.

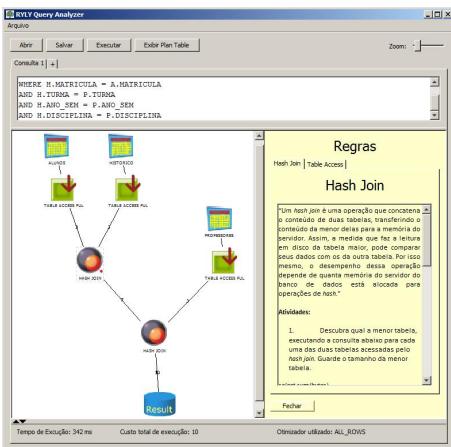


Figura 6. Exibição de uma regra de tuning

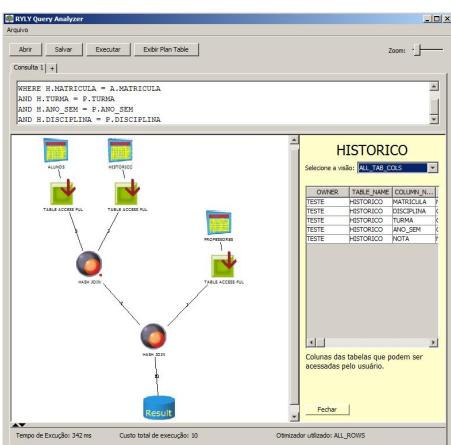


Figura 7. Exibição das views

The screenshot shows the RYLY Query Analyzer interface with the following query:

```
WHERE H.MATRICULA = A.MATRICULA
AND H.TURMA = P.TURMA
AND H.ANO_SEM = P.ANO_SEM
AND H.DISCiplina = P.DISCiplina
```

Below the query plan diagram, the results of the query are displayed in a table:

NOME	DISCiplina	PROFESSOR
Vanessa da Silva	46250-02	Ana Paula
Tatiane da Silva	46250-02	Ana Paula
Marcelo da Silva	46250-02	Ana Paula
Marcelo da Silva	46250-02	Ana Paula
Edna da Silva	46250-02	Ana Paula
Rogério da Silva	46250-02	Ana Paula
Jovana da Silva	46250-02	Ana Paula
João da Silva	46250-02	Ana Paula
Carla da Silva	46250-02	Ana Paula

Figura 8. Exibição da consulta

Similaridade entre Objetos Localizados em Fontes de Dados Heterogêneas

Rubens Guimarães¹, Gustavo Zanini Kantorski¹

¹Curso de Sistemas de Informação – Universidade Luterana do Brasil (ULBRA)
Campus Santa Maria – Santa Maria – RS – Brasil

rubens.poa@gmail.com, gustavoz@cpd.ufsm.br

Abstract. *The structured and semi-structured data sources integration is a major challenge for the database area. The objective of this paper is to present a tool capable to achieve integration and enable the identification of duplicates in structured and semi-structured data sources.*

Resumo. *A integração de fontes de dados estruturadas e semi-estruturadas é um dos grandes desafios para a área de banco de dados. O objetivo deste trabalho é apresentar uma proposta de ferramenta para realizar a integração e permitir a identificação de duplicatas em fontes de dados estruturadas e semi-estruturadas.*

1. Introdução

Atualmente com a expansão da internet, crescimento da disponibilidade e da demanda por informação, vem surgindo cada vez mais a necessidade de integrar dados de organizações distintas e permitir o acesso integrado a múltiplas fonte de dados. Estas fontes geralmente são heterogêneas, autônomas e distribuídas e que necessitam ser integradas para que a informação de diferentes setores de uma mesma organização, utilizando diferentes sistemas com grande redundância de dados e operações, torne-se algo limpo e transparente para o usuário.

Muitos problemas surgem quando são necessárias integrações de informações de várias fontes na web [Wiederhold 1993]. Um desses problemas é a existência de objetos em vários formatos, entre eles o XML. Dados XML são semi-estruturados e são organizados hierarquicamente. O formato XML torna complexa a tarefa de identificação de objetos, comparada com técnicas que tratam com fontes estruturadas tais como bancos de dados relacionais. Dados XML possuem estruturas diferentes e hierarquias que complicam a identificação dos objetos.

Este trabalho apresenta o desenvolvimento de uma ferramenta web, de código fonte aberto, cujo principal objetivo é realizar a identificação de similaridades de dados providos de documentos XML. A ferramenta proposta é parte do projeto denominado CORIDORA, desenvolvido em âmbito acadêmico na Universidade Luterana do Brasil, campus Santa Maria. O projeto CORIDORA tem como objetivo realizar o tratamento de inconsistências, e possíveis limpezas de dados, em bancos de dados, derivadas da representação de equivalências de um mesmo objeto do mundo real. O tratamento de inconsistência é realizado através do mapeamento de esquemas conceituais, identificando, consistindo e comparando divergências entre os objetos equivalentes, sem prejudicar a autonomia local das fontes de dados conforme proposta de [Ribeiro 1995].

A ferramenta que realiza o mapeamento de esquemas entre as fontes de dados heterogêneas, por meio da metodologia proposta por [Ribeiro 1995], está descrita nos trabalhos de [Meneghetti, Paes e Kantorski 2007a]. O acesso às fontes de dados e o resultado da consulta podem ser visualizados no trabalho de [Paes 2008]. Uma limitação na ferramenta desenvolvida por [Paes 2008] é a identificação no resultado da consulta de dados similares que existem em diferentes fontes. O objetivo deste artigo é apresentar uma solução para tratar o resultado da consulta realizada por [Paes 2008] através da identificação de similaridades entre documentos XML.

A próxima seção apresenta a ferramenta que realiza a consulta integrada nas fontes de dados heterogêneas. Na seção 3 é apresentada a proposta para resolver o problema resultante da consulta. Trabalhos relacionados são mostrados na seção 4. A seção 5 apresenta as considerações finais e trabalhos futuros.

2. Ferramenta de Consulta Integrada

Esta ferramenta baseia-se nos resultados obtidos durante os processos mapeamento de esquemas conceituais e identificação de equivalências semânticas, identificados nos trabalhos de [Meneghetti, Paes, Kantorski 2007a], [Meneghetti, Paes, Kantorski 2007b], [Meneghetti, Paes, Kantorski 2008], efetuados pelo ambiente Coridora para proporcionar a integração dos dados sem a necessidade da interação do usuário para realizar este processo.

O usuário deve escolher a equivalência que deseja consultar e então a ferramenta provê uma interface uniforme de acesso aos dados, de tal forma que abstrai a localização, conflitos semânticos ou até mesmo linguagem de consulta [Paes 2008]. Nesta interface o usuário deve informar os filtros que deseja fazer para sua consulta e a ferramenta analisa as informações adquiridas, onde novas consultas são geradas para, posteriormente, serem executadas nas diversas fontes de dados. A figura 1 ilustra o resultado da consulta.

Uma das dificuldades encontradas nessa etapa, é que a ferramenta tem a capacidade de determinar os objetos que são equivalentes, porém não é capaz de determinar quais objetos representam uma mesma entidade, retornando assim dados redundantes contidos nas diferentes fontes de dados selecionadas. Isto pode ser observado na figura 1 para a coluna “nome”. A consulta realizada para o nome “Rubens” pode retornar a mesma pessoa em fontes de dados diferentes.

3. Similaridade entre Objetos

Este trabalho tem por objetivo identificar objetos equivalentes providos do resultado da ferramenta de consulta integrada proposta por [Paes 2008] através da similaridade dos valores, calculada através da definição de pesos para os atributos e da utilização de algoritmos de similaridade. Os algoritmos são definidos por [Suder e Dornelles 2006] como funções pré-definidas que procuram identificar equivalências entre tipos de dados atômicos.

Identificador	Pessoa	Nome	Detalhes
38191	Null	RUBENS A. CARVALHO	i
220832	Null	RUBENS ALEX FIORIN	i
60449	Null	RUBENS ALEXANDRE TERRA QUESADA	i
142408	Null	RUBENS AMARAL SOUZA VIANA	i
348286	292320	RUBENS ANTONIO ANCHIETA CARNEIRO	i
158862	Null	RUBENS ARISTEU MOURA JAQUES	i
269568	236941	RUBENS AUGUSTO SANGOI	i
176736	Null	RUBENS BAIRET	i
21131	Null	RUBENS BARBOSA	i
77370	Null	RUBENS BONDARENKO GADEA	i
354974	276295	RUBENS BORBA DA SILVA	i
299480	250056	RUBENS CARDozo	i
100775	Null	RUBENS CARLOS DA SILVA	i
211737	217906	RUBENS CARLOS PEREIRA DOS SANTOS	i
113190	Null	RUBENS CARVALHO DIAS	i

Figura 1. Interface de Consulta Integrada.

Um padrão para estruturar documentos é o XML (*eXtensible Markup Language*), proposta pelo W3C como uma linguagem de marcação textual cuja tem sido aplicada para interoperabilidade, integração, estruturação e armazenamento de informações [W3C 2009]. Esta linguagem oferece uma abordagem para descrição, processamento e publicação de informações representadas por conteúdo, estrutura e apresentação. Desta forma, documentos XML são considerados coleções de documentos textuais com *tags* adicionais e relacionamentos entre as *tags*. A ferramenta proposta trabalha com fontes de dados XML validado pelo XSD (*XML Schema Definition*) descrito na Figura 2.

O arquivo XML (Figura 2) é composto por dois elementos que representam dois objetos providos de fontes de dados diferentes. Cada objeto contém um conjunto de elementos que representam seus atributos. Para cada atributo é necessário um identificador que será utilizado para relacionar os atributos equivalentes nos dois objetos, um nome que é utilizado como descrição no momento onde são exibidos os dados, um peso que é utilizado pela ferramenta para definir a relevância de cada atributo no processo de comparação e por fim um elemento que contém o conjunto dos valores de cada atributo.

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema targetNamespace="http://tempuri.org/XMLSchema.xsd" elementFormDefault="qualified"
  xmlns="http://tempuri.org/XMLSchema.xsd" xmlns:mtns="http://tempuri.org/XMLSchema.xsd"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Teo" type="TeoType"/>
  <xs:complexType name="TeoType">
    <xs:sequence>
      <xs:element name="Objeto1" type="ObjectType" />
      <xs:element name="Objeto2" type="ObjectType" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ObjectType">
    <xs:sequence>
      <xs:element name="Atributo" type="AtributoType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="AtributoType">
    <xs:sequence>
      <xs:element name="id" type="xs:int" />
      <xs:element name="nome" type="xs:string" />
      <xs:element name="peso" type="xs:float" />
      <xs:element name="valores" type="xs:string" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>

```

Figura 2. XSD do arquivo XML.

A figura 3 mostra uma representação do documento XML através de uma árvore. O elemento TEO representa a Tabela de Equivalência de Objetos [Meneghetti, Paes, Kantorski 2007a] que contém quais objetos são equivalentes. O resultado da consulta apresentado na figura 1 somado às informações dos metadados registrados no ambiente CORIDORA resulta no documento XML que será utilizado para a identificação de similaridade entre os objetos.

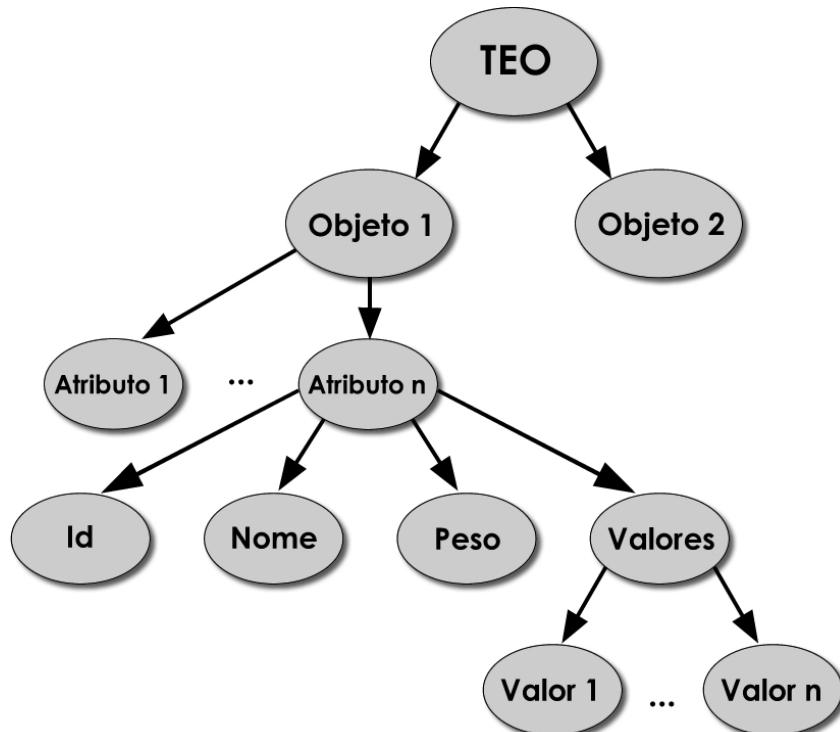


Figura 3. Arquivo XML representado em uma árvore.

O padrão para acesso e processamento de documentos XML é o XML DOM (*Document Object Model*). DOM representa elementos, atributos e textos como nós de uma árvore. Com a API DOM é possível processar um documento XML, iniciando pelo elemento raiz e navegando nas árvores nos demais elementos pais e filhos. Além da API DOM existe a API denominada SAX que permite a manipulação de documentos XML.

Com o documento XML criado, o usuário precisa informar apenas a similaridade referente à probabilidade com que deseja que os dados sejam equivalentes conforme a figura 4. Ao clicar em “Consultar” a ferramenta importa esses dados em formato de árvore através da API DOM e um *hashmap* de vetores é criado através do elemento ‘Objeto1’ onde cada posição contém um vetor com os dados de cada atributo. O vetor é acessado através do identificador definido no arquivo XML e a partir deste *hashmap* os objetos referentes ao elemento “Objeto1” são montados.

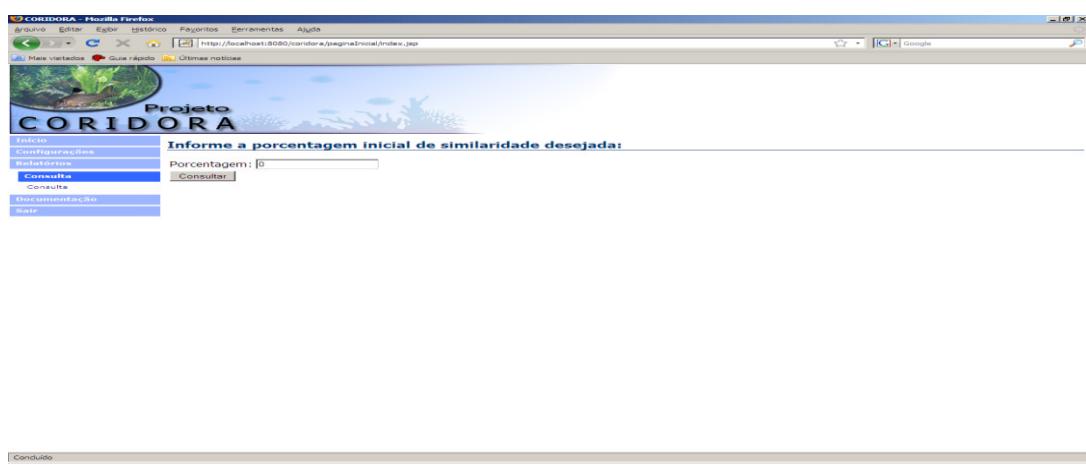


Figura 4. Interface onde deve ser informada a similaridade

Com os objetos criados, para cada valor contido nos atributos do elemento ‘Objeto2’ é montado um segundo objeto que é utilizado na execução de um processo de comparação que consiste em calcular uma similaridade utilizando-se algoritmos de similaridade definidos pela ferramenta. Foram escolhidos alguns algoritmos através de um estudo realizado levando em consideração aqueles mais citados na literatura [Chapman 2005].

Para atributos com valores numéricos e que possuem apenas um caractere a similaridade é 0 ou 1, onde 1 significa que são idênticos. Para os demais tipos atributos o algoritmo de similaridade calcula o valor v . Esse valor então é multiplicado pelo peso p definido para o atributo em questão. O processo é repetido até que todos os atributos estejam calculados, os valores obtidos são somados e a porcentagem de similaridade entre os dois objetos é calculada através da soma dos pesos. A fórmula na figura 5 descreve o cálculo da similaridade entre dois objetos:

$$sim(o_i, o_j) = \left(\frac{(\sum_{k=1}^n v_k * p_k)}{\sum_{k=1}^n p_k} \right)$$

Figura 5. Fórmula para cálculo da similaridade entre dois objetos

Onde n representa o número total de atributos equivalentes para os dois objetos, v representa o valor obtido com o cálculo de similaridade entre os dois valores e p representa o peso definido pelo projetista para o atributo k . O valor de p considera a

importância do atributo no conjunto de todos os atributos presentes no elemento *Atributo* do arquivo XML.

A tabela 1 descreve dois objetos equivalentes e provenientes de fontes de dados distintas que representam uma mesma pessoa, com seus atributos e dados. Para efeitos de comparação considere o objeto1 como Paciente e o objeto2 como Funcionário. O cálculo de similaridade entre os dois objetos é realizado da seguinte maneira: $((0*0) + (9*0.8) + (1*0.15) + (10*0,86)) / (0+9+1+10) = 0,798$.

Tabela 1. Objetos distintos com dados equivalentes

	Id	Data Nascimento	Profissão	Nome
Paciente	258	25/05/1975	Estudante	Adalberto C. Carvalho
Funcionário	325	25/05/75	Desenvolvedor	Adalberto Cruz Carvalho
Peso	0	9	1	10
Similaridade (<i>strings</i>)	0	0,8	0,15	0,86
Algoritmo	-	Levenshtein	Levenshtein	Smith-Waterman
$Sim(o_i, o_j) = 0,798$				

Pode ser observado que mesmo quando a maior parte dos atributos possui valores consideravelmente diferentes, ainda assim, com a utilização de pesos é possível identificar a equivalência dos dados, pois o atributo “Nome” juntamente com o atributo “Data Nascimento” com o maior peso dentre os demais, é mais conveniente para identificar uma mesma pessoa mesmo quando em contextos diferentes.

A interface de exibição dos dados apresentada na figura 6 mostra para cada processo de comparação os dados originais, o algoritmo de similaridade utilizado, o valor obtido através deste, o peso de cada atributo e o percentual de similaridade calculada para os dois objetos.

É importante verificar que a similaridade é calculada entre objetos e não entre atributos. Embora os algoritmos sejam aplicados nos atributos dos objetos, a similaridade considera o peso de cada atributo no objeto mais a similaridade entre os atributos para calcular a similaridade global entre os objetos.

O processo de seleção do algoritmo de similaridade, que é aplicado nos valores dos atributos textuais, atualmente utiliza aqueles contidos no pacote *SimMetrics* [Chapman 2005]:

- Levengshtein – Este algoritmo pode ser parafraseado como “o menor número de inserções, remoções e substituições para igualar duas strings” [Navarro 2001]. São definidos escores diferentes para cada possível operação: *match* (casamento, igualdade dos caracteres); *mismatches* (substituições); inserções, remoções. Onde são avaliadas todas as operações na tentativa de chegar ao maior escore. Este algoritmo demonstrou melhor resultado para comparações onde as strings possuem quantidades de caracteres semelhantes.
- Smith-Waterman – Este algoritmo é bastante utilizado para realizar alinhamentos locais de seqüências, isto é, determina regiões semelhantes entre as seqüências de caracteres existentes na string, e compara segmentos de todos os possíveis comprimentos e aperfeiçoa a semelhança medida para atingir o maior

escore. Este algoritmo demonstrou melhor resultado para as comparações quando as strings são compostas por mais de uma palavra.

- Jaro-Winkler – Este algoritmo variante do *Jaro Distance Metric* e é utilizado principalmente na área de *record linkage* (detecção de duplicidades). Esta extensão modifica os pesos dos pares identificados que partilham de um prefixo comum, porém não possuem um bom alinhamento. Demonstrou melhor resultado para as comparações quando a string é composta de uma palavra e um caractere, normalmente como acontece nas abreviações.

The screenshot shows a Mozilla Firefox browser window displaying the 'CORIDORA' project interface. The title bar reads 'CORIDORA - Mozilla Firefox'. The main content area shows a search result table for patient identification. The table has columns: 'Identificador', 'Identificador da Pessoa', and 'Nome do Paciente'. There are four rows of results:

	Identificador	Identificador da Pessoa	Nome do Paciente
Consulta	38191	0	RUBENS A. CARVALHO
	38191	0	RUBENS A. CARVALHO
	Peso: 0.19853073	Peso: 0.11915449	Peso: 0.16843599
	Algoritmo: não foi usado algoritmo	Algoritmo: não foi usado algoritmo	Algoritmo: não foi usado algoritmo
	Valor: 1.0	Valor: 1.0	Valor: 1.0
	Percentual: 100.0		
	38191	0	RUBENS A. CARVALHO
	220832	0	RUBENS ALEX FIORIN
	Peso: 0.19853073	Peso: 0.11915449	Peso: 0.16843599
	Algoritmo: não foi usado algoritmo	Algoritmo: não foi usado algoritmo	Algoritmo: Smith-Waterman
	Valor: 0.0	Valor: 1.0	Valor: 0.8
	Percentual: 52.0		
	38191	0	RUBENS A. CARVALHO
	60449	0	RUBENS ALEXANDRE TERRA QUESADA
	Peso: 0.19853073	Peso: 0.11915449	Peso: 0.16843599
	Algoritmo: não foi usado algoritmo	Algoritmo: não foi usado algoritmo	Algoritmo: Smith-Waterman
	Valor: 0.0	Valor: 1.0	Valor: 0.625
	Percentual: 46.0		
	38191	0	RUBENS A. CARVALHO

A status bar at the bottom left says 'Recebendo dados de localhost...'.

Figura 6. Interface de exibição das comparações

4. Trabalhos Relacionados

Vários trabalhos mostram o interesse da comunidade científica em explorar informações localizadas em fontes heterogêneas, sejam elas estruturadas, não estruturadas ou semi-estruturadas.

O trabalho *Duplicate Record Detection: A Survey* [Elmagarmid 2007] que consiste em uma pesquisa sobre algumas técnicas existentes para a busca de duplicatas em bancos de dados. Este trabalho parte da análise da heterogeneidade léxica, não se preocupando com a heterogeneidade estrutural, ou seja, analisa os dados partindo do princípio em que as estruturas são equivalentes. Neste artigo conforme descrito na seção 3, os dados provenientes de qualquer fonte de dados seja ela estruturada ou semi-estruturada, precisam estar disponibilizados em formato XML para que seja possível o cálculo da similaridade entre os objetos.

O trabalho de [Tejada 2001] apresenta um sistema de identificação de objetos chamado *Active Atlas* que aprende regras de mapeamento para um domínio específico de aplicação para determinar os mapeamentos dos objetos. O objetivo do trabalho proposto por [Tejada 2001] é aumentar a possibilidade de identificação de objetos com a participação mínima do usuário.

Trabalhos que envolvem a integração de documentos semi-estruturados e a sua heterogeneidade estrutural pode ser citado o *Structure-based inference of xml similarity for fuzzy duplicate detection* [Leitão 2007] onde baseado em conceitos de lógica *fuzzy*, propõe uma metodologia para identificar mesmas entidades com estruturas diferentes dentro de arquivos no padrão XML. Esta metodologia visa lidar com os dados em árvore e não somente identificar as duplicatas nos nós filhos, mas também calcular através de redes *Bayesianas*, as probabilidades dos nós descendentes também serem duplicados.

5. Considerações Finais e Trabalhos Futuros

Este trabalho apresentou uma forma de solução do problema de redundância de informações geradas no resultado do acesso integrado em fontes de dados heterogêneas. Desta forma, quando uma busca é realizada nas diversas fontes é possível unificar informações similares de fontes diferentes por meio da aplicação de algoritmos de similaridade. A similaridade de um objeto é calculada baseada em um peso, previamente definido, para cada atributo que o compõe e pelo valor assumido pelo atributo. Para um mesmo atributo (equivalente entre dois objetos) são comparados os seus respectivos valores e determinada a semelhança entre eles através de uma função. É importante salientar que a similaridade não é calculada entre os atributos de um objeto e, sim, entre os objetos. Isto é possível porque é realizada a avaliação de todos os atributos dos objetos.

Atualmente os pesos dos atributos dos objetos são definidos pelo projetista responsável pelo mapeamento das fontes no ambiente CORIDORA, ou seja, o projetista necessita de conhecimento sobre o esquema para aumentar a exatidão dos resultados. Técnicas como aprendizagem de máquina e descoberta de conhecimento podem ser aplicadas para verificar a possibilidade de determinar os pesos dos atributos de maneira semi-automática ou automática, diminuindo a participação do projetista.

Deve ser realizada uma avaliação da solução proposta considerando fontes com grande quantidade de dados para verificar questões relativas a desempenho, revocação e precisão nos resultados.

Referências

- Chapman, Sam. (2005) “String Similarity Metrics for Information Integration”, In: Natural Language Processing Group, Department of Computer Science, University of Sheffield, Sheffield, UK.
- Elmagarmid A. K., Ipeirotis, P. G., Verykios V. S. (2007) “Duplicate Record Detection: A Survey” The IEEE Transactions on knowledge and Data Engineering (TKDE) Vol. 19 No. 1 January 2007, pp. 1-16.

- Leitão, L., Pável, C., Weis M. (2007) “Structure-based inference of xml similarity for fuzzy duplicate detection”, In: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management - Lisboa, Portugal.
- Meneghetti, F. B., Paes, F. G., Kantorski, G. Z. (2007a) “CORIDORA Mapping: Uma Ferramenta Web para Mapeamento de Equivalências Semânticas em Bancos de Dados Heterogêneos”. In: Simpósio de Informática, 2007, Uruguaiana – RS. XII Simpósio de Informática, Nov.
- Meneghetti, F. B., Paes, F. G., Kantorski, G. Z. (2007b) “Ferramenta CORIDORA Mapping para Mapeamento de Esquemas em Bancos de Dados Heterogêneos”. In: Seminário de Informática, Torres – RS. VII Seminário de Informática, Nov.
- Meneghetti, F. B., Paes, F. G., Kantorski, G. Z. (2008) “Uma Interface Web para Identificação de Equivalências em Bancos de Dados Heterogêneos”. In: Escola Regional de Banco de Dados. Florianópolis –SC, 2008
- Navarro, G. (2001) “A Guided Tour to Approximate String Matching”. University of Chile. ACM Computing Surveys, Vol. 33, No. 1, Março 2001, pp. 31-88.
- Paes, F. G. (2008) “Consulta Integrada a Bancos de Dados Heterogêneos na Web”. In: Trabalho de Conclusão de Curso, ULBRA, 2008.
- Ribeiro, Cora Helena Francisconi Pinto. (1995) “Banco de Dados Heterogêneos: Mapeamento dos Esquemas Conceituais em um Modelo Orientado a Objetos (CPGCC)”. Porto Alegre: UFRGS, 1995. 165p.
- Suder, R. L., Dornelles, C. F. (2006) “Integração de Dados em Múltiplos Níveis”. In: Escola Regional de Banco de Dados. Passo Fundo – RS, 2006.
- Tejada, S., Knoblock, C.A., Minton, S. (2001) “Learning object identification rules for information integration”. Information Systems, Vol. 26, No 8, pp 607-633, 2001.
- W3C, (2009) “Extensible Markup Language (XML)”, <http://www.w3.org/XML> Dezembro 200.
- Wiederhold, G. (1993) “Intelligent integration of information” SIGMOD Record (1993), 434-437.

Um estudo de caso com análise comparativa entre ferramentas de BI livre e proprietária

**Manuele Ferreira, Robson Silva, Carina Guimarães
Juliana Carvalho, Vaninha Vieira**

¹Departamento de Ciência da Computação – Universidade Federal da Bahia (UFBA)
Salvador, BA – Brazil

{manuele052, robsonsilva, vaninha}@dcc.ufba.br

{carinapiauhy, juliana.esc}@gmail.com

Abstract. *The need for efficiency in the institutions' decision process requires the use of solutions that generate consistent information. BI (Business Intelligence) tools come to address this matter. An initial matter in any BI project is choosing support tools to be used in the development process. This paper presents a case study made at UFBA with comparative analysis between two BI tools: a free software solution (Pentaho) and a proprietary solution (Microsoft). The same project was developed in both suites, using a real database from UFBA. Also, various criteria for analysis were defined and categorized. These criteria assisted in the definition of user profiles that helped determine the most suitable solution for each institution.*

Resumo. *A necessidade de eficiência no processo decisório das instituições exige a utilização de soluções que gerem informações consistentes. É nesse contexto que se inserem as ferramentas de BI (Business Intelligence). Uma questão inicial em qualquer projeto de BI é a escolha do ferramental de apoio a ser usado no desenvolvimento. Este artigo apresenta um estudo de caso realizado na UFBA com análise comparativa entre duas ferramentas de BI: uma solução livre (Pentaho) e outra proprietária (Microsoft). Desenvolveu-se um mesmo projeto em ambas, usando dados reais da UFBA, e foram definidos e categorizados critérios para análise. Esses critérios permitiram definir perfis de usuários que apóiem a definição da ferramenta mais adequada para cada instituição.*

1. Introdução

Toda instituição de maior porte produz uma grande quantidade de dados que muitas vezes estão espalhados por diversas bases de dados heterogêneas. Estes dados, quando convertidos em informação, podem ser de grande valia para auxiliar o processo decisório da instituição favorecendo seu o desenvolvimento.

De acordo com [Petrini et al. 2006], *Business Intelligence* (BI) é um conjunto de tecnologias que tem como objetivo prover e oferecer suporte a um ambiente de informação. A necessidade de eficiência e agilidade no processo decisório nas instituições exige delas a utilização de soluções que gerem informações consistentes e ao mesmo tempo sejam flexíveis de modo a se enquadrar nas suas necessidades e limitações. Dessa forma, é necessário efetuar uma análise dessas instituições e das ferramentas do mercado de modo a verificar quais delas são compatíveis.

Existe uma grande quantidade de ferramentas de BI no mercado atualmente, com uma ampla variedade de funcionalidades e valores. Além disso, têm ocorrido um grande crescimento e amadurecimento das soluções livres. No entanto, como não existe um padrão estrutural e funcional seguido por todas, o processo de comparação entre essas ferramentas é dificultado, podendo levar a uma escolha demorada e não necessariamente correta da ferramenta. Nesse contexto, é inserido o risco decorrente da ferramenta não corresponder na realidade àquilo que está explicitado nos manuais.

Esse artigo tem como objetivo trazer os resultados de uma análise comparativa de ferramentas livres e proprietárias de BI baseado na experiência do desenvolvimento de um estudo de caso real na Universidade Federal da Bahia (UFBA). Este estudo de caso foi idealizado a partir de um projeto piloto de BI desenvolvido pela Universidade, em 2003, que foi descontinuado, dentre outras razões, pelo alto custo em desenvolver uma solução de BI, na época. Como forma de qualificar as ferramentas, foram definidos alguns critérios. Os critérios foram separados em "básicos", que são os considerados fundamentais para uma ferramenta de BI e os "desejáveis", que são aqueles que complementam a solução. Para esses foram definidas algumas subdivisões gerais relativas à arquitetura, ETL, relatórios, usabilidade, administração e produto. Assim, o principal objetivo desse estudo foi verificar a viabilidade de uso de uma ferramenta de BI livre, comparativamente com soluções proprietárias já conhecidas por técnicos e usuários da instituição.

O artigo está organizado da seguinte maneira. Na próxima seção são apresentados os trabalhos relacionados ao propósito desse artigo. Na Seção 3 é apresentada caracterização do cenário e ferramentas analisadas. A Seção 4 apresenta o estudo de caso realizado e analisa os resultados alcançados. Por fim, a Seção 5 apresenta a conclusão do trabalho e as perspectivas para trabalhos futuros.

2. Trabalhos relacionados

Para embasar a realização dessa pesquisa e identificar os critérios a serem utilizados para a análise comparativa das ferramentas, foram pesquisados trabalhos com temática correlata, os quais são descritos a seguir.

Em [Barreto 2003] é feita uma análise de ferramentas proprietárias de BI, *Oracle* e *Microsoft*. Foram definidas diversas características (e.g. preço da solução, suporte técnico e OLAP) e, para cada uma, foi dada uma pontuação baseada no conhecimento das ferramentas de dois analistas de negócio. A conclusão foi que não é possível definir qual a melhor opção, uma vez que as empresas seguem diferentes estratégias e políticas na elaboração de suas soluções de BI. É importante notar que uma característica analisada por estes analistas, provavelmente utilizando os seus conhecimentos baseados em contextos distintos, pode acarretar em opiniões divergentes. Assim, a utilização desses resultados para auxiliar a escolha da ferramenta mais adequada pode ser dificultada.

Em [Cacciapaglia 2008] são analisadas diversas ferramentas proprietárias e livres: *Microsoft* [Microsoft 2009], *BusinessObjects* [BusinessObject 2009], *Cognos* [Babik 2010], *Microstrategy* [Microstrategy 2009], *Pentaho* [Pentaho 2009], *SpagoBI* [SpagoBI 2009], *Cubeware* [Cubeware 2009] e *InetSoft* [InetSoft 2009]. O autor definiu diversas classes (e.g. tecnologia, funcionalidades e custos). Para cada critério foi definido um valor fixo. As ferramentas foram analisadas individualmente, sendo atribuído um percentual a cada critério analisado indicando o quanto a ferramenta consegue atender àquele

critério. Não é visto no trabalho argumentos que justifiquem as pontuações atribuídas, comprometendo assim a conclusão dos resultados obtidos na análise e a futura utilização desses por outros interessados.

Em [Holub 2009], o autor faz uma boa revisão do estado atual do mercado de BI livre, realizando comparações entre ferramentas utilizadas em soluções de BI livre. Essas ferramentas são comparadas verificando se elas são código aberto ou não e se possuem suporte comercial. Esse tipo de comparação, quando utilizado, é adequado para concluir se uma solução é mais custosa ou não em relação a outra, porém, não permite concluir se uma solução é mais adequada que a outra em uma situação específica.

3. Caracterização do Cenário e Ferramentas Analisadas

Em 2003, o CPD/UFBA iniciou um projeto piloto de BI cujo objetivo era construir um repositório com informações gerenciais relacionadas à área acadêmica. Esse projeto possuía informações sobre os docentes de nível superior, docentes de nível médio e alunos. A esse projeto deu-se o nome SIGDB (Sistema de Informações Gerenciais da Área Acadêmica). O projeto, entretanto, não pôde contar com o suporte de uma suite de BI, pois, na época, só foram pesquisadas soluções proprietárias e o CPD/UFBA possuía restrições financeiras para o projeto. O processo de ETL (Extração, Transformação e Carga) era feito por meio de consultas SQL escritas manualmente, enquanto que as análises OLAP e os relatórios eram gerados com o uso da ferramenta *Microsoft Office Excel*, a qual possui suporte limitado a consultas analíticas. Devido à grande dificuldade de manutenção das informações, o projeto foi descontinuado.

Em 2009, durante a disciplina Tópicos em Banco de Dados do curso de Ciência da Computação da UFBA, decidiu-se retomar o projeto do SIGDB, contando, desta vez, com o auxílio de uma suite de BI. Devido à restrição de tempo inerente a uma disciplina semestral, optou-se por reduzir o escopo do projeto, focando apenas no perfil de docente de nível superior. Para tanto, construiu-se um *Data Mart* contendo as mesmas informações desse perfil em 2003 para então seguir todo o processo de construção de um DW até a fase de análise dos dados consolidados. Como a carência de uma suite BI foi determinante para a descontinuação do projeto em 2003, para o projeto SIGBD 2009 foi definida a proposta de desenvolver duas versões do projeto, utilizando uma suite livre e uma suite proprietária objetivando, assim, levantar pontos fortes e fracos das mesmas.

3.1. Identificação das suites de BI Livre e Proprietária

A primeira etapa do projeto foi avaliar suites livres e proprietárias existentes para a implementação do SIGDB 2009.

As ferramentas pré-selecionadas foram: **Ferramentas Livres:** *Pentaho* [Pentaho 2009], *SpagoBI* [SpagoBI 2009] e *JasperSoft* [JasperSoft 2009]; **Ferramentas Proprietárias:** *BusinessObject* [BusinessObject 2009], *Cognos* [Babik 2010], *Microsoft* [Microsoft 2009], *Microstrategy* [Microstrategy 2009] e *SpotFire* [SpotFire 2009]. Um estudo sobre fatores como funcionalidades, componentes, custo e amadurecimento foi realizado sobre cada uma delas. A ferramenta livre selecionada foi a *Pentaho*, enquanto que a proprietária escolhida foi a *Microsoft*.

Dentre as proprietárias, a *Microsoft* foi escolhida, pois, além de possuir todas as funcionalidades fundamentais, essa solução não é complexa a ponto de exigir um

alto investimento inicial em treinamento dos desenvolvedores. Além disso, dentre as proprietárias, esta apresentou-se como uma das que possui menor custo financeiro. A disponibilização de uma versão de avaliação por 180 dias viabilizou sua utilização. Aliado a essas questões, a *Microsoft* é uma empresa bastante conceituada no mercado de tecnologia da informação e é a fornecedora do SGBD já em uso no CPD/UFBA.

No contexto livre, a escolha recaiu sobre a *Pentaho*, pois a mesma se mostrou a ferramenta com a maior quantidade de funcionalidades úteis ao projeto, com a melhor documentação, suporte técnico da comunidade, a mais madura e estável. Maiores detalhes sobre a avaliação das ferramentas podem ser encontradas em [MATB10 2009].

3.2. Arquitetura e ferramentas utilizadas

A arquitetura proposta por Kimball[Kimball 2002] sugere que os dados das bases operacionais sejam carregados por ETL nos *Data Marts* e posteriormente no *Data Warehouse*, para que então sejam disponibilizados para consultas analíticas e relatórios no servidor OLAP pelos usuários finais. Tanto a solução construída na suite *Pentaho* quanto na suite *Microsoft* seguiram as etapas propostas pela arquitetura de Kimball e cada uma delas possui ferramentas que permitiram fazer as devidas analogias. A seguir tem-se as ferramentas utilizadas para cada etapa da arquitetura citada.

A base de dados utilizada pelas duas soluções foi a do SIGDB 2003. Essa base foi definida através de um modelo multidimensional disponibilizados através de arquivos CSV, com a tabela fato e as de dimensão.

Com relação à etapa de ETL, na suíte *Pentaho* a ferramenta utilizada foi a *Pentaho Data Integration* (PDI), também conhecida como *Spoon* ou *Kettle*. O *Data Mart* produzido foi armazenado no SGBD livre MySQL. Quanto à *Microsoft*, o processo de ETL foi realizado usando a ferramenta *SQL Server 2008 Integration Services*. Para armazenamento do *Data Mart* foi utilizado o SGBD *Microsoft SQL Server 2008*.

Para tratamento de consultas OLAP, a geração de cubos no *Pentaho* deve ser através do mapeamento em um arquivo XML para que o seu servidor OLAP possa interpretá-los. A ferramenta utilizada para tal foi o *Pentaho Schema Workbench* (PSW) que também faz a publicação do arquivo XML gerado na pasta de soluções do *Pentaho Server*. Já a ferramenta utilizada para a análise do Cubo OLAP gerado foi a *Analysis View (Mondrian)*. Enquanto que para a suite *Microsoft*, os cubos não precisam ser mapeados em arquivo XML. O processo de análise e geração do cubo foi realizado usando a ferramenta *SQL Server 2008 Analysis Services*.

A criação de relatórios na suite *Pentaho* foi feita utilizando a ferramenta *Pentaho Report Designer* (PRD). Depois de criados, estes relatórios também foram publicados no diretório de soluções do *Pentaho Server*. Para a construção de relatórios personalizados foi utilizada a ferramenta *Pentaho Report Designer*. Já na suite *Microsoft*, foi utilizada a ferramenta *SQL Server 2008 Reporting Services* para a construção dos relatórios, a qual também permite a construção de relatórios personalizados.

Na criação de *dashboard* na suite *Pentaho* foram utilizadas as ferramentas: *Community Dashboard Framework* e *CDF Dashboard Editor*. A suite *Microsoft* indica que *dashboards* podem ser criados utilizando a ferramenta *Microsoft SharePoint*, a qual não faz parte da suite de BI *Microsoft*. Devido à ausência de versão de avaliação dessa ferra-

menta, essa funcionalidade não pode ser testada.

A versão utilizada da suite Microsoft foi a de avaliação por 180 dias da *SQL Server Business Intelligence Development Studio*. Ela é composta por todas ferramentas citadas acima com exceção da ferramenta *Microsoft SharePoint*, responsável pela administração da solução criada (perfis de usuário entre outras funções) e é também a ferramenta que possibilita a utilização da solução via internet. A versão utilizada da suite *Pentaho* foi a versão *Community* que é uma versão sem suporte técnico. O grupo *Pentaho* também disponibiliza uma versão paga, que não foi testada durante a elaboração e construção desse projeto.

4. Estudo de caso

4.1. Caracterização do Estudo de Caso

Para a análise comparativa entre as ferramentas *Pentaho* e *Microsoft*, foi conduzido um estudo de caso, baseado na metodologia proposta por Wohlin [Wohlin et al. 2000]. Como definição do estudo de caso, tem-se:

- **Objeto de estudo:** O objeto de estudo são duas suítes de desenvolvimento de soluções em BI: a *Pentaho* (solução livre) e a *Microsoft* (solução proprietária);
- **Propósito:** O propósito deste estudo é a comparação do uso das ferramentas durante etapas de: ETL, construção de cubos e análise OLAP, geração de relatórios e dashboards e, também, gerenciamento administrativo;
- **Perspectiva:** A perspectiva da análise é a dos alunos da disciplina Tópicos em Banco de Dados do curso de Ciência da Computação da UFBA, que manipularam as ferramentas durante a execução do projeto;
- **Foco:** O foco do estudo é a análise comparativa através do levantamento de pontos fortes e fracos de ambas as ferramentas;
- **Contexto:** As ferramentas serão utilizadas no desenvolvimento do projeto, referente a dados acadêmicos de docentes de ensino superior, disponibilizados pelo CPD/UFBA. O projeto será realizado durante a disciplina Tópicos em Banco de Dados, por alunos da graduação em Ciência da Computação da UFBA.

4.2. Critérios para Análise e Resultados Encontrados

Como forma de qualificar as ferramentas, foram definidos alguns critérios. Os critérios foram separados em 'Básicos', que foram os considerados fundamentais para uma ferramenta de BI e os 'Desejáveis', que são aqueles que complementam a solução.

4.2.1. Critérios Básicos

Neste contexto foram definidos: **modelo visual**, que permite ao usuário a modelagem visual do modelo de dados facilitando o seu gerenciamento e entendimento; **suporte ao SQL Server** que, por ser o SGBD utilizado pelo CPD, é essencial que a ferramenta suporte-o; **suporte a workflow no ETL**, que permite aos usuários definirem uma sequência lógica de execução de tarefas facilitando o processo do ETL; **consultas ad-hoc OLAP**, que oferece liberdade ao usuário de definir consultas que acredita ser melhor em um dado contexto; **relatórios ad-hoc**, que permite a geração padrão de relatórios, sem a

necessidade de customização; **suporte à geração de Gráficos**, que auxilia a visualização dos dados oriundos de consultas de uma forma comum para o usuário; **suporte à geração de Dashboards**, que auxilia, de uma forma diferenciada, à visualização de resultados; **suporte ao português**.

N	Critério	<i>Microsoft</i>	<i>Pentaho</i>
1.1	Modelagem Visual	Sim	Sim
1.2	Suporte SQL Server	Sim	Sim
1.3	Suporte workflow no ETL	Sim	Sim
1.4	Relatórios ad-hoc	Sim	Sim
1.5	Gráficos	Sim	Sim
1.6	Dashboards	Sim	Sim
1.7	Suporte português	Sim	Sim

Tabela 1. Avaliação das Ferramentas de acordo com os Critérios Básicos

Através da análise da Tabela 1 é possível observar que ambas as ferramentas cumprem com os requisitos básicos de uma ferramenta de BI para o contexto estabelecido.

4.2.2. Critérios Desejáveis

Para identificação dos critérios desejáveis, foram definidas algumas subdivisões gerais relativas a arquitetura, ETL, relatórios, usabilidade, administração e produto.

Arquitetura: os critérios de avaliação da arquitetura oferecida estão agrupados em: **multi-Plataforma**, que é a possibilidade do sistema ser executado nos sistemas operacionais Windows e Linux; **suporte a SGBDs livres**, que supõe que o sistema suporte, ao menos, os principais SGBDs livres: *PostgreSQL* e *MySQL*; **arquitetura escalável**, permitindo que a solução implementada na ferramenta seja escalável; **disponibilidade na internet/intranet**, permitindo a manipulação das ferramentas OLAP pelo usuário usando a internet, independente da sua localização geográfica ou arquitetura utilizada; **customização funcional de componentes**, de modo que o usuário possa efetuar modificações nas funcionalidades do sistema, adequando-as às suas necessidades.

N	Critério	<i>Microsoft</i>	<i>Pentaho</i>
2.1	Multi-Plataforma	Não	Sim
2.2	Suporte SGBDs livres	Sim	Sim
2.3	Arquitetura Escalável	Sim	Sim
2.4	Disponível na internet/intranet	Não	Sim
2.5	Customização funcional de componentes	Não	Sim

Tabela 2. Avaliação das Ferramentas de acordo com o Critério Desejável Arquitetura.

Em relação à arquitetura (Tabela 2), a ferramenta *Microsoft* mostrou-se, mesmo com suporte a SGBD's livres, não interoperável e fechada a customizações.

ETL: os critérios definidos quanto a essa categoria é possuir: **função de agrupamento**, facilitando o processo de agrupamento de tabelas e resultados; **função de**

extração de dados, permitindo de uma forma genérica extrair dados de diversas fontes; **função de ordenação**, cujo objetivo é ordenar os dados das tabelas e resultados extraídos; visando facilitar o processo de geração de informações.

N	Critério	Microsoft	Pentaho
3.1	Função agrupamento	Sim	Sim
3.2	Função extração de dados	Sim	Sim
3.3	Função Ordenação	Sim	Sim

Tabela 3. Avaliação das Ferramentas de acordo com o Critério Desejável ETL

Quanto ao suporte de ETL (Tabela 3), é possível observar que ambas tem suporte às mesmas funcionalidades.

Relatórios: é uma outra subdivisão, cujos critérios são: **relatórios personalizados**, o sistema oferece suporte para a geração de relatórios de forma customizada; **exportação para PDF; exportação para formato livre** (ODT), para permitir futuramente a integração com ferramentas livres.

N	Critério	Microsoft	Pentaho
4.1	Relatórios personalizados	Sim	Sim
4.2	Exportação à PDF	Sim	Sim
4.3	Exportação formato livre (ODT)	Não	Sim

Tabela 4. Avaliação das Ferramentas de acordo com o Critério Desejável Relatórios

O suporte a Relatórios de cada ferramenta (Tabela 4) evidencia que, mais uma vez, a *Microsoft* mostrou-se uma ferramenta limitada à seus formatos não sendo possível exportar os relatórios gerados para formatos livres tal como ODT.

Usabilidade: em relação a questões de usabilidade, os critérios definidos são: **facilidade de uso**, indica o quão fácil é para o usuário leigo identificar suas funcionalidades, onde encontrá-las e como executá-las; **atratividade**, avalia o grau em que a ferramenta possua uma interface amigável e atrativa; **interface personalizável**, identifica se a ferramenta permite customizações de interface para atender, por exemplo, a padrões gráficos e visuais do cliente; **suporte técnico/documentação**, avalia o nível de qualidade da documentação e o suporte técnico oferecido pela ferramenta.

N	Critério	Microsoft	Pentaho
5.1	Facilidade de uso	4.5	3
5.2	Atratividade	4.5	3.5
5.3	Interface personalizável	2.25	5
5.4	Suporte técnico/Documentação	4.5	2.25

Tabela 5. Avaliação das Ferramentas de acordo com o Critério Desejável Usabilidade

Para o critério de usabilidade (Tabela 5) foi utilizada uma avaliação diferente das demais, sendo atribuída uma escala de 1 a 5 na avaliação de cada critério. Tal processo foi

necessário uma vez que são critérios subjetivos. Foi possível observar desses resultados que a suite *Pentaho* não possui uma interface tão amigável quanto a da suite *Microsoft*, no entanto ela se destaca quando se trata de customização de interface.

Administração: identifica questões ligadas ao gerenciamento do uso da ferramenta e contém os seguintes critérios: **permitir agendamento de tarefas**, cujo objetivo é avaliar se a ferramenta possibilita o cadastro e gerenciamento de tarefas a serem efetuadas pelo sistema como, por exemplo, atualização da base; **permitir gerenciamento centralizado**, verifica se a ferramenta permite, em uma interface centralizada, o gerenciamento das tarefas administrativas da ferramenta, como controle de acesso e segurança; **perfil de usuário**, verifica se a ferramenta permite que o administrador defina níveis hierárquicos para os usuários do sistema

N	Critério	Microsoft	Pentaho
6.1	Permitir schedule de tarefas	Sim	Sim
6.2	Permitir gerenciamento centralizado	Sim	Sim
6.3	Perfil de usuário	Não	Sim

Tabela 6. Avaliação das Ferramentas de acordo com o Critério Desejável Administração

Quanto à parte de Administração (Tabela 6), devido a necessidade da ferramenta *Microsoft SharePoint* não foi possível analisar a administração de perfis de usuário na suite *Microsoft*, sendo esse critério avaliado como ausente.

Produto: é a ultima subdivisão e visa avaliar as ferramentas quanto à questões do produto em si, e possui os seguintes critérios: **custo**, cujo objetivo é efetuar uma comparação dos valores de compra das ferramentas; **amadurecimento do produto**, que visa analisar o nível de consolidação e estabilidade do sistema; **capacidade de integração**, onde avalia a possibilidade de integração do sistema com outras ferramentas que são utilizadas pelo usuário comumente como, por exemplo, ferramentas de planilhas eletrônicas e fontes de dados.

N	Critério	Microsoft	Pentaho
7.1	Custo	R\$ 25.000	R\$ 0
7.2	Amadurecimento	4.5	3.5
7.3	Integração	2	4.25

Tabela 7. Avaliação das Ferramentas de acordo com o Critério Desejável Produto

A Tabela 7 apresenta os resultados encontrados para caracterizar o produto. O critério custo é apresentado em valores absolutos e equivale apenas ao custo de aquisição da ferramenta. Não foram analisados custos com treinamento de pessoal ou aquisição de hardware necessário. Em relação aos critérios de Amadurecimento e Integração, a análise foi feita por meio de atribuição de notas numa escala de 1 a 5. Percebe-se que apesar do alto custo a ferramenta da *Microsoft* mostrou-se mais madura em relação à ferramenta *Pentaho*, apesar da facilidade de integração entre as ferramentas ser maior na ferramenta *Pentaho*.

4.3. Análise dos Resultados e Perfis de Uso

A partir da realização do estudo de caso e da avaliação das ferramentas de acordo com os diversos critérios estabelecidos, pode-se observar que ambas suites apresentam características positivas e negativas, e que a simples indicação de que uma suite é superior à outra, pode não atender a diferentes cenários e contextos. Dessa forma, buscou-se categorizar os critérios de acordo com perfis de uso, que evidenciassem características relevantes para os usuários ao buscar o suporte de uma ferramenta de BI. Esses perfis visam identificar em que situações a suite livre *Pentaho* é mais adequada e em que cenário a suite proprietária da *Microsoft* demonstra-se superior.

O primeiro perfil é caracterizado pelos critérios 2.1, 2.2, 2.3, 2.4, 2.5, 4.3, 7.1 e 7.3. Este perfil corresponde a instituições que possuam restrições financeiras necessidades de customização tanto funcional dos componentes quanto de interface, necessidade de utilização de vários sistemas operacionais e acesso remoto à ferramenta. Pela análise desses itens pode-se concluir que para o perfil descrito, a suite *Pentaho* se mostra mais adequada.

Os critérios 5.1, 5.2, 5.4, 6.2, 7.2 e 7.3 estão relacionados a instituições que possuam recursos disponíveis para investimento em BI, tenham baixo conhecimento na área, pouco tempo disponível para implantação da solução e exijam interfaces mais amigável e de mais fácil uso. Para esse perfil, observa-se que a suite da *Microsoft* mostra-se mais adequada.

5. Conclusões e Trabalhos Futuros

Este trabalho apresentou um estudo de caso com análise comparativa entre uma ferramenta livre e uma proprietária de BI, utilizando o contexto específico de uma instituição pública de ensino, e a realização de implementações nas duas ferramentas de um mesmo projeto, o que permitiu a análise das ferramentas em iguais condições. O estudo visava identificar pontos fortes e fracos das ferramentas de BI estudadas e verificar em que aspectos cada uma seria mais adequada do que a outra. Foram definidos diferentes critérios de avaliação das ferramentas, com base em requisitos básicos e desejáveis em uma ferramenta de BI para o contexto do CPD da UFBA. Os resultados alcançados foram obtidos a partir da experiência prática no desenvolvimento de projetos reais, executados numa parceria entre alunos da disciplina Tópicos em Banco de Dados do curso de Ciência da Computação e técnicos do Centro de Processamento de Dados da UFBA.

Baseado nos resultados obtidos, foi possível concluir que para determinados tipos de perfis das instituições, uma ferramenta pode se destacar em relação a outra. Foram exemplificados dois perfis a partir de conjuntos de critérios estabelecidos, que evidenciaram a ferramenta mais adequada para aqueles perfis. É importante ressaltar que esses conjuntos de critérios podem ser agrupados de diferentes maneiras de forma a produzir diferentes perfis de instituição possibilitando-as utilizar nossos resultados para auxiliar a escolha de sua ferramenta de BI. A realização de uma análise comparativa entre soluções proprietárias e livres foi de extrema importância para verificar a viabilidade de utilizar ferramentas livres em substituição a proprietárias. Os resultados se mostraram mais consistentes dado que os critérios foram analisados de forma prática.

Como trabalhos futuros, pode-se indicar a expansão da análise realizada com um número maior de critérios, em diferentes contextos, e incluir outras suítes de BI, pro-

prietárias e livres, permitindo auxiliar diferentes instituições em suas escolhas de BI. Alguns critérios podem ser avaliados com maior rigor, como é o caso da questão do custo, podendo ser incluídos outros elementos como estimativa de custos com treinamento, suporte e hardware.

Agradecimentos

Os autores agradecem ao Centro de Processamento de Dados da UFBA, em especial a Ana Cristina e André Andrade, por todo o apoio oferecido, e aos alunos da disciplina Tópicos em BD que contribuíram para a realização deste trabalho.

Referências

- Babik, M. (2010). Cognos business intelligence and financial performance management. Acessado em 28/01/2010.
- Barreto, D. G. (2003). Business intelligence: comparação de ferramentas. Master's thesis, Universidade Federal do Rio Grande do Sul.
- BusinessObject (2009). Business object. <http://www.sap.com/solutions/sapbusinessobjects> - Acessado em 28/01/2010.
- Cacciapaglia, A. (2008). Comparativa de suites de business intelligence. Master's thesis, Universitat Politècnica de Catalunya.
- Cubeware (2009). cubeware. <http://en.cubeware.de/> - Acessado em 28/01/2010.
- Holub, S. (2009). Open source bi: A market overview. In *Open Source Business Resource*, pages 17–21.
- InetSoft (2009). inetsoft. <http://www.inetsoft.com/> - Acessado em 28/01/2010.
- JasperSoft (2009). Jasper soft. <http://www.jaspersoft.com/> - Acessado em 28/01/2010.
- Kimball, R. (2002). *The Data Warehouse Toolkit*. Wiley.
- MATB10 (2009). Tópico de banco de dados. <https://disciplinas.dcc.ufba.br/MATB10/FerramentasBI> - Acessado em 28/01/2010.
- Microsoft (2009). Microsoft business intelligence. <http://www.microsoft.com/bi/> - Acessado em 28/01/2010.
- Microstrategy (2009). Microstrategy. <http://www.microstrategy.com.br/> - Acessado em 28/01/2010.
- Pentaho (2009). Pentaho bi. <http://www.pentaho.com/> - Acessado em 28/01/2010.
- Petrini, M., Freitas, M. T., and Pozzebon, M. (2006). Inteligência de negócios ou inteligência competitiva: noivo neurótico, noiva nervosa. *Encontro da Associação Nacional de Pós-Graduação e Pesquisa em Administração (EnANPAD)*.
- SpagoBI (2009). Spago bi. <http://www.spagoworld.org/xwiki/bin/view/SpagoBI/> - Acessado em 28/01/2010.
- SpotFire (2009). Spotfire. <http://spotfire.tibco.com/> - Acessado em 28/01/2010.
- Wohlin, C., Runeson, P., Host, M., Ohlsson, M. C., Regnell, B., and Wesslén, A. (2000). *Experimentation in Software Engineering - An Introduction*. Kluwer Academic Publishers.

Base de dados de gestão de ordens de serviço: extração de regras interessantes para apoio à decisão

Danilo S. da Cunha¹, Miriam L. Domingues¹

¹Faculdade de Computação – Instituto de Ciências Exatas e Naturais
Universidade Federal do Pará (UFPA) – Belém, PA – Brasil
danilocunha85@gmail.com, miriam@ufpa.br

Abstract. This paper describes the use of data mining in order to extract interesting rules from a database of service orders. The rules obtained are considered relevant to the strategic decisions that can improve the performance of maintenance services.

Resumo. Este artigo descreve o uso de mineração de dados com o objetivo de extrair regras interessantes de uma base de dados de ordens de serviço. As regras obtidas são consideradas relevantes para a tomada de decisões estratégicas que podem melhorar o desempenho dos serviços de manutenção.

1. Introdução

Utilizou-se a mineração de dados (MD) com o objetivo de adquirir conhecimento relevante de uma base de dados do WebOficina, *software* responsável pela gerência das ordens de serviço do Centro de Informática e Telecomunicação – CITEL, da Polícia Militar do Pará. Esse conhecimento permite aos responsáveis pela gestão do CITEL responder perguntas como: “Quais equipamentos apresentam mais problemas e qual a eficiência da equipe técnica em mantê-los?”, o que pode levar a ações que venham a contribuir para melhoria do serviço oferecido por esse Centro. A pesquisa contou com a ajuda de um dos responsáveis pela oficina do CITEL para compreender e validar as regras geradas na mineração.

Os modelos de MD extraídos são representados na forma de regras associativas geradas com o algoritmo *Apriori* da ferramenta Weka 3.7 [Witten e Frank 2005]. Como metodologia para realizar o estudo de caso, usou-se o modelo CRISP-DM [Chapman et al 2000].

Neste artigo, a Seção 2 apresenta o estudo de caso e a Seção 3 apresenta as considerações finais sobre o estudo.

2. Mineração dos Dados do CITEL

O estudo de caso foi realizado seguindo os passos do modelo CRISP-DM, o qual organiza o processo de MD em uma hierarquia que parte de tarefas mais gerais (fases) para as mais específicas (tarefas genéricas, tarefas especializadas e instâncias do processo). As seis fases desse modelo são: compreensão do domínio, compreensão de dados, preparação de dados, modelagem, avaliação e aplicação [Chapman et al 2000]. Existem vários laços entre as fases da metodologia, que permitem o retorno a tarefas preliminares sempre que houver a necessidade de serem refeitas.

2.1. Compreensão do Domínio

A base de dados do WebOficina tem como fonte o CITEL, da Polícia Militar do Pará, e armazena dados referentes à manutenção de equipamentos de informática. Nesse sistema, os técnicos ficam em fila e realizam um atendimento por vez. O serviço é passado a eles de acordo com a chegada das ordens de serviço. Neste estudo, são usados 1.206 registros do ano de 2008, com o objetivo de extrair regras associativas interessantes para melhorar o desempenho dos serviços de manutenção.

2.2. Compreensão dos Dados

Os dados foram carregados com o *software* Weka para serem analisados estatisticamente e para verificar a presença de erros ou de valores atípicos, o que não foi constatado. Com a observação estatística no Weka, percebe-se que a organização por **Turno** demonstra que o turno MATUTINO realizou a maior parte das ordens de serviço. Esta análise permite responder, rapidamente, perguntas como: “Qual turno tem mais trabalho e precisa de mais técnicos?”

2.3. Preparação de Dados

Esta fase envolveu a formatação dos dados para serem carregados no Weka, bem como a seleção e a transformação de dados para a extração dos modelos de MD.

Selecionou-se para esta pesquisa os campos do WebOficina: **Data_saida**, que se refere ao dia em que o serviço foi finalizado; **Equipamento**, que se refere a equipamentos mantidos pelo CITEL; **Tecnico**, ao nome do técnico responsável pela manutenção; **Estado**, ao andamento do serviço, que pode ser FECHADO, quando concluído, ou PENDENTE, quando o técnico está por terminar o serviço ou aguardando a chegada de uma peça que teve de ser solicitada à Diretoria de Apoio Logístico; e **Turno**, que é o turno em que o serviço foi solicitado, MATUTINO ou VESPERTINO.

As informações de **Data_saida** passaram por transformação, com a construção de um novo campo denominado **Dezena**, no qual as datas de saída foram mapeadas da seguinte forma: os dias de 01 a 10 foram substituídos por “1^a DEZENA”; os dias de 11 a 20 foram substituídos por “2^a DEZENA” e os dias de 21 a 31 foram substituídos por “3^a DEZENA”. Esta transformação fez-se necessária para facilitar a descoberta de padrões, aumentando o intervalo de tempo em que este será procurado, pois achar um padrão no 1º dia de cada mês, por exemplo, é mais difícil para o algoritmo.

Os valores de **Tecnico** também foram modificados para preservar os dados pessoais dos técnicos do CITEL. Optou-se por transformar os nomes dos técnicos para dois grupos: E. SUPERIOR, para os técnicos cursando o ensino superior e TEC. EM INFORMATICA, para os que estão no curso técnico.

2.4. Modelagem e Avaliação

Os modelos de MD foram obtidos com a descoberta de regras associativas. Uma regra associativa possui a forma $X \Rightarrow Y$, em que X (antecedente) e Y (consequente) são conjuntos de itens, significando que a presença de X implica a presença de Y na mesma transação T. São usadas métricas para avaliar a qualidade das regras, tais como o suporte de X (número de transações T, tal que $X \subseteq T$), o suporte da regra (número de

transações que contêm X e Y) e a confiança da regra (suporte da regra dividido pelo suporte de X). O usuário precisa informar à ferramenta de MD os valores de suporte e confiança mínimos desejados para as regras descobertas.

No Weka, foi selecionado o algoritmo *Apriori* para a extração de regras, com a configuração de parâmetros: suporte mínimo de 10%, confiança mínima de 70% e número de regras igual a 50. As regras mais interessantes desta primeira mineração são mostradas na Figura 1. Cada regra apresenta, no antecedente, um conjunto de itens seguido de seu valor de suporte e, no consequente, outro conjunto de itens seguido do valor de suporte da regra. Entre parênteses, é dada a confiança da regra que, na terceira regra, por exemplo, é o resultado da divisão de 393 por 395, aproximadamente 99%.

1. Técnico=E. Superior Turno=Matutino 404 => Estado=Fechado 404 conf:(1)
2. Equipamento=Computador Técnico=E. Superior 381 => Estado=Fechado 380 conf:(1)
3. Dezena=3^a dezena 395 => Estado=Fechado 393 conf:(0.99)
4. Dezena=1^a dezena 412 => Estado=Fechado 403 conf:(0.98)

Figura 1. Regras da 1^a MD feita com os dados do CITEL

Os técnicos do turno da manhã que estão cursando o ensino superior, apesar de estarem em menor quantidade do que os técnicos cursando o técnico em informática, apresentam melhor desempenho, pois finalizam todas as ordens de serviços que iniciaram. No ano, a média de atendimentos foi de 61,8 por técnico em informática e de 52,1 por técnico cursando o ensino superior. Percebe-se que dos 381 computadores com defeito atendidos por técnicos cursando o ensino superior, apenas 01 não foi finalizado.

Com relação ao tempo, as segundas e terceiras dezenas têm melhor desempenho em relação à primeira. Um dos responsáveis pela oficina do CITEL informou que esta queda nos primeiros dias é devido ao pagamento da bolsa (salário) ocorrer nesse intervalo. Os técnicos ficam menos concentrados no serviço e mais preocupados com as contas a pagar, justificando o baixo rendimento em relação às outras dezenas do mês. O valor da bolsa é o mesmo para todos os técnicos e corresponde ao salário mínimo.

Removeu-se alguns atributos para uma nova mineração. Desta vez, os atributos que permaneceram foram: **Equipamento**, **Estado** e **Turno**. As principais regras obtidas são mostradas na Figura 2. Observa-se que o desempenho dos técnicos é melhor com as impressoras do que com os computadores; apesar de somente três técnicos fazerem esse tipo de manutenção, todos os problemas foram resolvidos.

1. Equipamento=Impressora 190 => Estado=Fechado 190 conf:(1)
2. Equipamento=Impressora Turno=Matutino 159 => Estado=Fechado 159 conf:(1)
3. Equipamento=Computador Turno=Vespertino 150 => Estado=Fechado 148 conf:(0.99)

Figura 2. Regras da 2^a MD feita com os dados do CITEL

Com o conhecimento extraído, procurou-se facilitar o entendimento das regras pelos gestores dos dados. Tomou-se como base a solução dada por Albergaria et al (2008, p. 161), que cria abstrações textuais para as regras obtidas. Os itens de atributo-

valor e valores de confiança das regras do Weka foram mapeados para expressões textuais padronizadas, por exemplo: “Tecnico=E. Superior”, no antecedente da regra, foi substituído por “se o serviço for realizado por um técnico cursando o ensino superior”. Na Figura 3, é mostrado um exemplo de expressão gerada.

Expressão para regra da 1^a. MD:

1. Se o serviço for realizado por um técnico cursando o ensino superior e se o serviço for realizado no turno matutino, então o serviço será fechado em 100% dos casos.

Figura 3. Expressão para regra da 1^a MD feita com os dados do CITEL

2.5. Aplicação

O conhecimento extraído é relevante e suficiente para que as decisões estratégicas listadas na Figura 4 sejam adotadas pelo CITEL, de forma a proporcionar um melhor desempenho em relação ao atendimento das manutenções realizadas. Tais decisões se baseiam nas regras da primeira mineração e na análise estatística da etapa de compreensão de dados.

1. Quando forem substituir os técnicos atuais, dar preferência por técnicos que estejam cursando o ensino superior, pois estes têm melhor desempenho que os técnicos em informática.
2. Dividir o pagamento da bolsa dos técnicos em duas parcelas por mês, por exemplo, a cada 15 dias, assim a 1^a dezena não fica sobrecarregada de serviços pendentes.
3. O turno da tarde não precisa da mesma quantidade de técnicos que o turno da manhã, pois, durante a tarde, solicitam menos o serviço da oficina do CITEL do que durante a manhã. Aproximadamente, para cada nove solicitações durante a manhã, no turno da tarde são solicitadas apenas duas.

Figura 4. Decisões que permitem a melhoria do serviço prestado pelo CITEL

3. Considerações Finais

A mineração de dados com o uso da descoberta de regras associativas, do modelo CRISP-DM, do software Weka e da transformação das regras do Weka em expressões textuais na fase de avaliação, possibilitou a extração de conhecimento da base de dados de gestão de ordens de serviços. O conhecimento descoberto é considerado relevante para que o CITEL possa tomar decisões estratégicas que podem melhorar o desempenho em relação ao atendimento das manutenções realizadas.

4. Referências

- Albergaria, E., Mourão, F., Prates, R. e Meira Jr., W. (2008) “Modelo de Interface Extensível como Solução para Desafios de Interação em Sistemas de Mineração de Dados”. In: SEMISH. Anais do XXVIII Congresso da SBC. Belém: SBC, p.151-165.
- Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Wirth, R. and Shearer, C. (2000) “The CRISP-DM Process Model”. CRISP-DM consortium, 2000. Disponível em: <<http://www.crisp-dm.org/Process/index.htm>>.
- Witten, I.H and Frank, E. (2005). “Data Mining: Practical machine learning tools and techniques”, Morgan Kaufmann, 2nd Edition, San Francisco.

Um Modelo de Dados Temporal para Sistemas de Recomendação de Calagem e Adubação de Solos

Cristina Paludo Santos¹, Karine Barbosa de Oliveira¹

¹Curso de Ciência da Computação – Universidade Regional Integrada do Alto Uruguai e das Missões (URI) – Santo Ângelo – RS – Brazil

{paludo, karineb}@urisan.tche.br

Abstract. *The paper describes the query module of the CERES System. CERES is an Expert System based in ontology that automate the activities of interpretation of the soil properties and recommends the use of fertilizing, when necessary. Among the requirements of the application domain, includes to retrieve historical data about the soil conditions. In this way, a model capable of storing temporal data in the database has been defined. Also, an interface supports the querying process and allows the expert system to define selection expressions based on soil characteristics. The interface consists in a relevant tool to the system.*

Resumo. *Este artigo apresenta o módulo de consultas do sistema CERES. CERES é um sistema especialista baseado em ontologia que suporta o processo de interpretação e recomendação de adubação e calagem de solo. Dentre os requisitos da aplicação inclui-se a recuperação de dados históricos sobre as condições do solo. Desta forma, tem sido definido um modelo capaz de armazenar dados temporais. O processo de consulta proposto permite ao especialista definir expressões de seleção baseados em características das amostras de solo e consiste em uma relevante ferramenta para o sistema.*

1. Introdução

A maioria das aplicações que fazem uso de Bancos de Dados necessita manipular dados históricos que representam diferentes estados da aplicação. Para isto pode-se fazer uso da modelagem temporal de dados, que permite a representação de aspectos dinâmicos das aplicações, bem como a interação temporal entre processos [EDE98].

Mais especificamente, no que se refere a aplicações de recomendação de adubação e calagem de solos cujo principal objetivo é a utilização racional de insumos em quantidade, forma e época de aplicação, a persistência de dados temporais é de grande valia [COM04]. No entanto, apesar de haver demanda no sentido de armazenamento e estruturação temporal dos dados envolvidos neste tipo de aplicação, existe uma carência muito grande de ferramentas computacionais que sejam capazes de prover consultas detalhadas da sua evolução ao longo do tempo. Como exemplo, pode ser citado o sistema Agrissolos [FIL05] que apesar de implementar bases de dados e bases de conhecimento que contemplam aspectos bastante abrangentes a respeito do perfil dos solos, não mantém dados históricos sobre as condições do solo.

Este fato impulsionou o desenvolvimento de um modelo de dados temporal que provenha subsídios necessários para que as aplicações do domínio armazenem de forma organizada os dados históricos gerados pelos processos de adubação e calagem baseados na análise química do solo. Para validar o modelo proposto, o mesmo foi integrado ao CERES – um sistema especialista, baseado em ontologia, destinado à recomendação de calagem e adubação de solos [SIL09].

As próximas seções estão organizadas da seguinte forma. A seção 2 apresenta uma visão geral sobre o processo de modelagem temporal dos dados. A seção 3 descreve o processo de integração do modelo proposto ao CERES e as consultas temporais executadas sobre alguns casos de uso. As considerações finais são apresentadas na seção 4.

2. O Modelo de Dados Proposto

A construção do modelo baseou-se em informações obtidas a partir do Manual de adubação e calagem para os estados do Rio Grande do Sul e Santa Catarina [COM04] que é usado como referência na área. Além disso, para auxiliar no processo de coleta de dados contou-se também com o apoio de uma especialista em solos que proveu informações a respeito das necessidades de análise de dados históricos para uma tomada de decisão mais efetiva quanto ao uso adequado de adubos e calcários nas propriedades, o que permitiu uma análise criteriosa em relação ao comportamento das informações em relação ao tempo.

A análise realizada permitiu a construção de um modelo de dados com 17 entidades sendo uma delas bitemporal e 18 relacionamentos que contemplam características bitemporais. A notação utilizada para confecção do modelo, em nível conceitual, foi a TempER (*Temporal Entity-Relationship*) [ANT97], uma vez que a mesma incorpora dispositivos que permitem referenciar os objetos (entidades, relacionamentos ou valores de atributos) à dimensão temporal. Além disso, outras características que impulsionaram a escolha por esta abordagem foram: a abordagem permite representar a associação entre elementos temporalizados e não temporalizados e os atributos não são explicitados graficamente, mas através de um dicionário de dados associado ao diagrama ER (*Entidade-Relacionamento*), o que resulta em um modelo mais administrável visualmente. Já, para a representação do modelo proposto em nível lógico foi utilizado o modelo temporal TRM (*Temporal Relational Model*) [ELM05], que incorpora a semântica temporal do mundo real a um modelo de dados relacional. Juntamente com sua linguagem de consulta TSQL, o modelo permite a manipulação tanto de informações temporais quanto não temporais, de forma coerente e consistente. Além do aspecto de temporalidade presente no modelo, outra característica importante é a disponibilização de uma estrutura de armazenamento dinâmico para que possa adquirir a característica de um modelo temporal que possa ser utilizado em diferentes sistemas de recomendação de calagem e adubação de solos. A possibilidade de configuração do armazenamento permite que sejam inseridos dados utilizados por aplicações específicas, adequando o modelo ao sistema no qual estiver sendo aplicado. A Figura 1 apresenta uma parte do modelo proposto. Uma descrição mais detalhada sobre as entidades e relacionamentos presente no modelo pode ser obtida em [OLI09].

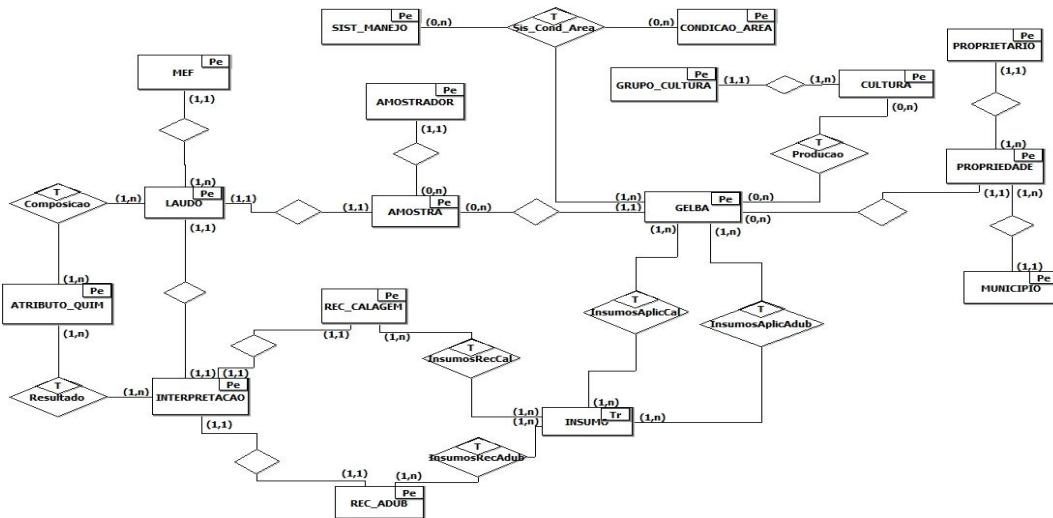


Figura 1: Modelo ER Temporal no TempER

Dentre as entidades, *insumo* necessita de armazenamento temporal, pois possui um atributo que representa o *preço* do mesmo, o qual varia em relação ao tempo. Além disso, alguns relacionamentos necessitam representação temporal tais como: *produção*, *composicao*, *resultados*, *InsumosRecAdub*, *InsumosRecCal*, *InsumosAplicCal*, *InsumosAplicAdub*, que armazenam informações relacionadas com a situação do solo. Em relação à fertilidade, os relacionamentos *composicao* e *resultados* armazenam o teor dos atributos do solo resultantes do laudo e sua

interpretação de acordo com as classes de valores em relação a uma determinada *gleba*. Outros relacionamentos como *InsumosRecCal* e *InsumosAplicCal* auxiliam a tomada de decisão por parte de especialistas da área agrícola, pois fornecem informações a respeito das práticas de calagem, ou seja, as quantidades de insumos recomendadas e aplicadas ao longo do tempo.

A validação do modelo proposto consiste na execução de consultas que resultem em dados significativos para análise. Para possibilitar essa validação, o modelo foi integrado ao CERES - um sistema de recomendação de calagem e adubação de solos baseado em Ontologia que recebe como entrada os dados dos laudos das amostras de solo, realiza consultas em uma base de conhecimento a apresenta os resultados de interpretação e recomendação baseados nessas consultas [SIL08]. O processo de implementação do banco de dados e do módulo de consultas temporais são apresentados a seguir.

3. Módulo de Consultas Temporais

O modelo proposto foi implementado utilizando o SGBD PostgreSQL e o emulador de bancos de dados bitemporais BtpgSql. Para que fosse possível a implementação de consultas temporais, foi necessário o mapeamento de TSQL2 para SQL utilizando a ferramenta EMap [MAN08], já que o BtpgSql não apresenta suporte a TSQL2. A consulta resultante em SQL convencional pode ser executada no módulo de consultas temporais, codificado na linguagem de programação Java. A Figura 2 apresenta a arquitetura do CERES, destacando o módulo de consultas temporais agregado ao sistema.

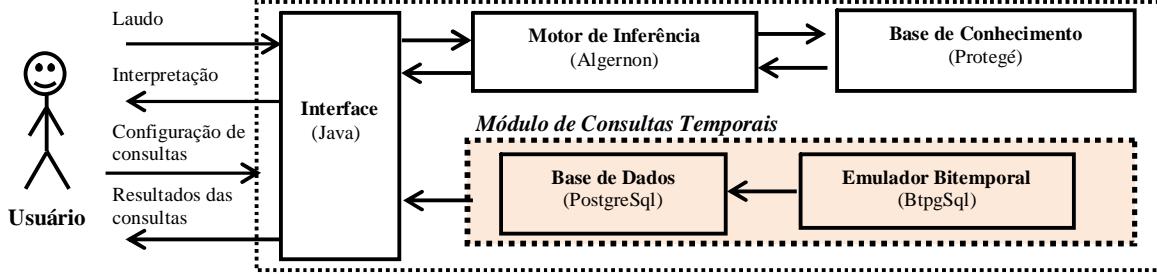


Figura 2. Arquitetura CERES com o Modulo de Consultas Temporais

A Figura 3(a) representa a interface que permite configurar uma consulta, ou seja, determinar os critérios de consulta e quais serão os atributos a serem exibidos no resultado. Também é possível consultar aspectos referentes à recomendação ou aplicação de insumos, tanto para calagem quanto para adubação, podendo especificar ou não, o insumo que se deseja analisar em relação ao tempo. A Figura 3(b) apresenta a interface que permite especificar este tipo de consulta.

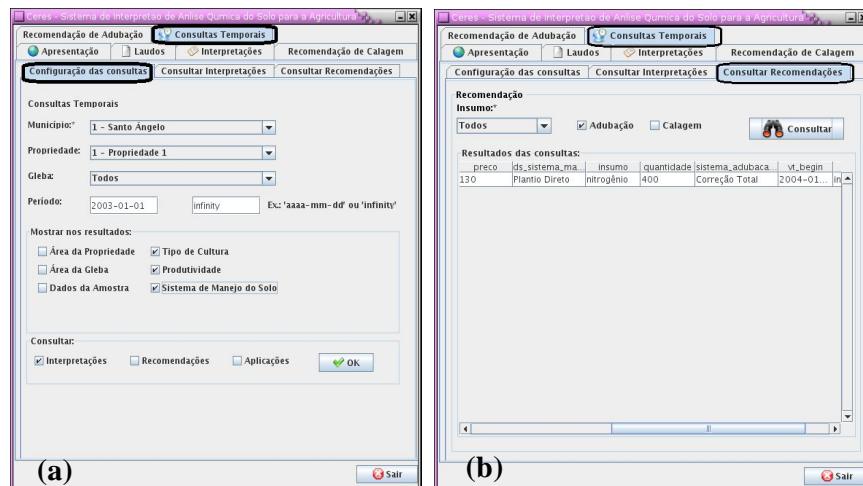


Figura 3 – Interfaces do Módulo de Consulta Temporal

As consultas executadas basearam-se em dados considerados significativos para análise de acordo com especialistas do domínio e estudos de fertilidade, obtendo-se assim informações úteis. Para isso foram implementadas interfaces amigáveis que possibilitam análise dos dados por especialistas do domínio, tanto de forma genérica como, por exemplo, de todas as propriedades e de todas as glebas dessas propriedades, como de forma específica como, por exemplo, o acompanhamento de dados referentes a uma determinada gleba ao longo do tempo. Com os resultados constata-se que o modelo proposto contempla os dados necessários referentes ao acompanhamento histórico no domínio da recomendação de calagem e adubação de solos. Além disso, proporciona a possibilidade de diferentes consultas aos dados, estruturados segundo os princípios da modelagem temporal, resultando em informações precisas.

5. Considerações Finais

A estrutura temporal de armazenamento de dados oferecida pelo modelo proposto permite definir inúmeras consultas históricas aos dados de todas as etapas do processo de calagem e adubação de solos. A análise dos dados dessas consultas por especialistas contribui para a realização de estudos de fertilidade, acompanhamento da evolução dos teores de atributos do solo, acompanhamento de recomendações de calagem e adubação para determinada propriedade, acompanhamento da aplicação de insumos em determinada propriedade, projeção de rendimentos, entre outras possibilidades de aplicação dos dados históricos extraídos. Desta forma o modelo promove contribuições para a área agrícola, sendo uma ferramenta computacional que auxilia o processo decisório relacionado com a calagem e adubação de solos. Além disso, o modelo também poderá ser utilizado para o desenvolvimento de diversas aplicações onde o uso desse tipo de informações seja pertinente.

6. Referências Bibliográficas

- [COM04] COMISSÃO DE QUÍMICA E FERTILIDADE DO SOLO – RS/SC. (2004). *Manual de adubação e calagem para os estados do Rio Grande do Sul e Santa Catarina*. 10. ed. Porto Alegre: Sociedade Brasileira de Ciência do Solo/Núcleo Regional Sul.
- [OLI09] OLIVEIRA, KARINE B., (2009) *Um Modelo de Dados Temporal para Sistemas de Recomendação de Calagem e Adubação de Solos*. Trabalho de Conclusão de Curso (Ciência da Computação) – Universidade Regional Integrada do Alto Uruguai e das Missões, Santo Ângelo, RS, 2009.
- [EDE98] EDELWEISS, N. *Banco de Dados Temporais: Teoria e Prática*. In: CONGRESSO NACIONAL DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 17, 1998, Anais... Recife: Sociedade Brasileira de Computação, 1998. p. 225-282.
- [SIL09] SILVA, G. A. C. ; SANTOS, C. P. ; SILVA, D. R. (2009) *ONIAQUIS – Uma Ontologia para a Interpretação de Análise Química de Solo*. Disciplinarum Scientia. Série Ciências Naturais e Tecnológicas^ , v. 6, p. 85-96.
- [GUB07] GUBIANI, IVONIR P.; SILVA, LEANDRO S.; REINERT, DALVAN J.; REICHERT, JOSÉ M., (2007) *CADUB GHF – um programa computacional para cálculo da quantidade de fertilizantes e corretivos da acidez do solo para culturas produtoras de grãos, hortaliças e forrageiras*. Em: **Ciência Rural, Santa Maria**, v. 37, n.4, p.1161-1165.
- [ANT97] ANTUNES, D. C.; HEUSER, Carlos A.; EDELWEISS, N. (1997). *TempER: Uma Proposta de Modelagem de Dados Temporal*. Em: **Revista de Informática Teórica e Aplicada**. Porto Alegre, v.4, n.1, p. 49-85.
- [ELM05] ELMASRI, R.; NAVATHE, S. B. *Sistemas de Banco de Dados*. 4. ed. São Paulo: Pearson Education, 2005.
- [FIL05] FILETO, Renato; ASSAD, Maria Leonor; SILVA, João Villa; SOARES, Amarindo Fausto; VENDRUSCULO, Laurimar Gonçalves . *Uma Arquitetura para Sistema de Informação sobre Solos para o Zoneamento Agrícola*. In: Congresso da Sociedade Brasileira de Informática Agropecuária, 2005, Londrina, 2005.

TerraER: Uma Ferramenta voltada ao Ensino do Modelo de Entidade-Relacionamento

Henrique Santos C. Rocha, Ricardo Terra

¹ Departamento de Ciéncia da Computação
Universidade Federal de Minas Gerais (UFMG) – Belo Horizonte, MG – Brasil

hscr@ufmg.br, terra@dcc.ufmg.br

Resumo. Grande parte das instituições de ensino superior ainda utilizam o modelo ER no ensino de modelagem de dados conceitual. Contudo, nota-se uma carência em relação a ferramentas que refletem exatamente o que é ensinado em sala de aula. A partir dessa motivação, este artigo descreve as funcionalidades e vantagens da utilização da ferramenta TerraER no ensino de disciplinas de Banco de Dados em cursos de graduação. Além disso, são relatados resultados demonstrando uma forte aceitação por parte dos professores e alunos.

1. Introdução

A modelagem de dados é o principal componente do projeto conceitual do banco de dados. Dentre as técnicas existentes para essa modelagem, a técnica entidade-relacionamento (ER) – apresentada em 1976 por Peter Chen [2] – é ainda largamente utilizada principalmente pela sua simplicidade e legibilidade, produzindo um modelo que seja inteligível tanto pelo projetista do banco de dados quanto pelo usuário final [6, 3, 5, 1].

É importante mencionar que várias empresas vem adotando o diagrama de classes da UML (*Unified Modeling Language*) como uma alternativa ao modelo ER. Mesmo que, por um lado, o diagrama de classes tenha tido inspiração no modelo ER e também consiga capturar os requisitos de dados do mundo real de uma maneira simples e significativa produzindo um modelo inteligível, essa não foi a motivação por trás da sua criação [4].

Em razão disso, grande parte das instituições de ensino superior aindam utilizam o modelo ER no ensino de modelagem de dados conceitual. Contudo, nota-se uma carência em relação a ferramentas que utilizem a notação de Chen estendida e que tenham foco no modelo conceitual. Em razão disso, professores vêm adotando ferramentas voltadas para o modelo lógico como DBDesigner¹, ERWin², entre outras.

A adoção dessas ferramentas, mesmo sendo estáveis e populares, não favorece ao aluno, uma vez que o aluno pratica o que lhe foi ensinado em uma ferramenta voltada a um outro modelo e que não possui fins acadêmicos. A partir dessa motivação, foi desenvolvida a ferramenta TerraER com o intuito de cobrir essa carência acadêmica. O objetivo principal da ferramenta é prover aos professores uma ferramenta mais voltada ao conteúdo lecionado e prover aos alunos uma ferramenta que estimule o seu aprendizado.

O restante deste artigo está organizado conforme descrito a seguir. A Seção 2 apresenta a ferramenta TerraER. A Seção 3 relata os resultados percebidos nas classes

¹<http://www.fabforce.net/dbdesigner4>

²<http://www.ca.com/us/data-modeling.aspx>

que vêm adotando a ferramenta. A Seção 4 apresenta os trabalhos relacionados. E, por fim, a Seção 5 apresenta as considerações finais e os trabalhos futuros.

2. Ferramenta

TerraER é um software voltado ao meio acadêmico, mais especificamente no auxílio ao aprendizado de disciplinas de modelagem conceitual de banco de dados. O público alvo da ferramenta consiste de alunos de graduação. Portanto, houve uma preocupação com a criação de uma interface gráfica prática, inteligível e intuitiva.

A ferramenta é *open-source* sobre a licença GPL e gratuito³. Isso é importante para melhoria constante da ferramenta, uma vez que os alunos podem contribuir diretamente, seja pela participação ativa no projeto daqueles que têm o perfil desenvolvedor, como também através de críticas, sugestões etc. Como exemplo, um dos alunos contribuiu para a internacionalização da ferramenta, que agora conta com o idioma inglês, além do português. Além disso, TerraER é desenvolvido na linguagem Java e, por isso,

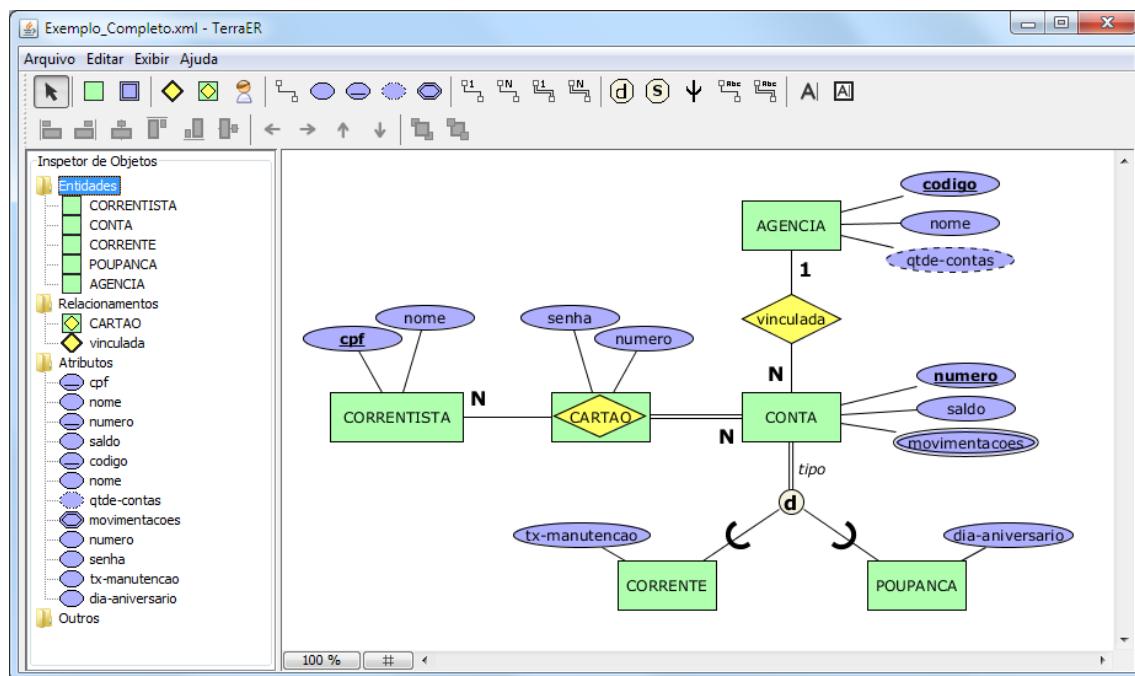


Figura 1. Screenshot do TerraER executado em Windows

é portável a grande parte dos sistemas operacionais, bastando apenas possuir a máquina virtual Java instalada. Convém ainda mencionar que, a cada nova liberação, testes básicos de funcionamento em ambiente Linux, Mac OS e Windows são realizados.

Como ilustrado na Figura 1, a interface da ferramenta é dividida em cinco principais áreas: (i) menu, localizado ao topo; (ii) barra de ferramentas de objetos, logo abaixo do menu; (iii) barra de ferramentas de posicionamento, logo abaixo da barra de ferramentas de objetos; (iv) inspetor de objetos, à esquerda; (v) área de desenho, à direita.

O menu contém as opções básicas da ferramenta. É possível imprimir os modelos, exportá-los como imagem, salvá-los etc. É importante mencionar que os modelos são

³<http://sourceforge.net/projects/terraer>

armazenados em formato XML, que atualmente é considerado o formato universal para compartilhamento de dados. Na barra de ferramentas de objetos existem atalhos para criação de elementos do modelo ER – notação Chen estendida, adotada por Elmasri e Navatthe [3]. A outra barra de ferramentas lida com o posicionamento dos objetos incluídos na área de desenho, permitindo assim uma formatação elegante dos modelos criados.

No inspetor de objetos pode-se ver, selecionar, remover ou editar objetos (entidades, relacionamentos, atributos etc) que estejam na área de desenho. Essa funcionalidade permite um acesso rápido e preciso a qualquer objeto do modelo. Como um outro exemplo de contribuição por parte dos alunos, o desenvolvimento desse inspetor de objetos foi motivado pela sugestão de um aluno que teve dificuldade em localizar objetos específicos.

A área de desenho é a parte principal da ferramenta. Nela estão contidos os elementos dos modelos criados pelo usuário. Essa área possui uma função de *zoom*, que é bastante útil quando se lida com modelos grandes e se deseja ter uma visão geral, ou quando se deseja aumentar o tamanho dos elementos para uma melhor visualização. Existe também uma função de grade que auxilia bastante na tarefa de posicionamento.

3. Experiências

A motivação que levou ao desenvolvimento do TerraER foi pelo simples fato de professores não possuírem uma ferramenta voltada exclusivamente à modelagem de dados conceitual. Assim, logo que desenvolvida a primeira versão da ferramenta, ela foi inicialmente adotada nas disciplinas de banco de dados lecionadas pelos próprios autores deste artigo nas seguintes IES: UNIPAC Bom Despacho, UNIPAC Contagem e FAMINAS-BH.

Como uma primeira experiência de aceitação da ferramenta, no primeiro semestre de 2009, foram planejadas três atividades de modelagem ER para a disciplina de Banco de Dados em cinco turmas distintas. Na primeira atividade, os alunos utilizaram obrigatoriamente o DBDesigner. Na segunda atividade, os alunos utilizaram o TerraER. Na última atividade, os alunos puderam escolher qual ferramenta utilizar e preencheram uma ficha de avaliação das ferramentas. Como resultado, a grande maioria dos alunos optaram pelo TerraER e, sintetizando a avaliação, principalmente porque a ferramenta refletia exatamente o que lhes foi ensinado e que, por isso, sentiam-se mais estimulados.

Como uma segunda experiência, foi apresentada a ferramenta a um professor da Universidade FUMEC e a um outro professor da FAMINAS-BH que concordaram em utilizar a ferramenta. Não foi possível repetir o experimento anterior, pois os professores já tinham o planejamento da disciplina. Contudo, ambos informaram que os alunos gostaram bastante da ferramenta e que os próprios professores a iriam adotar nos próximos semestres, pois, segundo eles, foi a ferramenta que mais se enquadrou no programa didático.

Atualmente, TerraER é também utilizada nas disciplinas de banco de dados do Centro Universitário UNA. A intenção com este artigo é estimular o aprendizado dos alunos e a adoção da ferramenta por professores de outras IES.

4. Trabalhos Relacionados

O DBDesigner é um software comercial para criação de modelos lógicos de bancos de dados. Possui uma ampla gama de funcionalidades como geração do código SQL correspondente, propagação automática de chaves estrangeiras, notações alternativas,

sincronização entre modelo e base de dados etc. Contudo, por ser voltado mais ao modelo lógico, o DBDesigner torna-se pouco ideal para o ensino de modelagem conceitual.

O brModelo⁴ é – assim com o TerraER – um software acadêmico voltado à modelagem de banco de dados. Engloba tanto modelagem conceitual como modelagem lógica. A maior vantagem do TerraER está em sua usabilidade e portabilidade. Em relação à usabilidade, brModelo não possui ferramentas para facilitar o posicionamento e redimensionamento de elementos e não agrupa objetos similares em seu localizador de objetos, dificultando a localização de um elemento específico. Em relação à portabilidade, brModelo só é compatível com o sistema operacional Windows, dificultando sua implantação em universidades que optam por sistemas operacionais gratuitos.

5. Considerações Finais

Mesmo com a popularização da UML, grande parte das instituições de ensino superior aídam utilizam o modelo ER no ensino de modelagem de dados conceitual. Contudo, nota-se uma carência em relação a ferramentas que refletem exatamente o que foi ensinado em sala de aula. Em razão disso, professores vêm adotando ferramentas voltadas para o modelo lógico que, mesmo sendo ferramentas estáveis e populares, não favorecem ao aluno, uma vez que o aluno pratica o que lhe foi ensinado em uma ferramenta voltada a um outro modelo e que não possui fins didáticos. A partir dessa motivação, foi desenvolvida uma ferramenta chamada TerraER voltada ao meio acadêmico, mais especificamente no auxílio ao aprendizado de disciplinas de modelagem conceitual de banco de dados. Além disso, resultados demonstraram uma forte aceitação por parte dos professores e alunos.

Como trabalho futuro pretende-se: (i) estender a ferramenta a demais notações, como a notação “pé de galinha” e IDEFIX, de forma que você possa alternar entre as notações através de uma simples opção de menu; (ii) desenvolver um módulo de verificação de modelo, a fim de relatar ao usuário (possivelmente um aluno) os erros que ele está cometendo; (iii) gerar o script para criação do modelo relacional correspondente.

Agradecimentos: Gostaríamos de agradecer aos professores Virgílio Borges de Oliveira (FAMINAS-BH) e Luiz Eduardo de Mello (Universidade FUMEC) o apoio e a adoção da ferramenta. Gostaríamos de agradecer também aos ex-alunos Rogério Correia e Wallace Alexander (UNIPAC Contagem) o desenvolvimento do sítio da ferramenta TerraER⁵.

Referências

- [1] S. Bagui and R. Earp. *Database Design Using Entity-Relationship Diagrams*. CRC Press LLC, 1964.
- [2] P. P. Chen. The entity-relationship model – towards a unified view of data. *ACM Trans. Database System*, pages 9–36, Março 1976.
- [3] R. Elmasri and S. B. Navathe. *Sistemas de Banco de Dados*. Pearson Education, 4 edition, 2005.
- [4] J. Rumbaugh, I. Jacobson, and G. Booch. *The Unified Modeling Language Reference Manual*. Addison-Wesley, 2 edition, 2005.
- [5] A. Silberschatz, H. F. Korth, and S. Sudarshan. *Sistema de Banco de Dados*. Elsevier, 2006.
- [6] T. Teorey, S. Lightstone, and T. Nadeau. *Projeto e Modelagem de Banco de Dados*. Elsevier, 2007.

⁴<http://sis4.com/brModelo>

⁵<http://www.ricardoterra.com.br/terraer>

EdMaSe - Editor de Marcação semântica de dados na Web baseado em ontologias

Débora Cabral Nazário¹, Tiago Brandes¹

¹Departamento de Ciéncia da Computação – DCC

Universidade do Estado de Santa Catarina (UDESC)

Campus Universitário Prof. Avelino Marcante s/n - Bairro Bom Retiro -

Joinville – SC

Brasil CEP 89223-100

debora@joinville.udesc.br, tiago.brandes@gmail.com

Resumo. Uma das tendências da Web atual é a crescente utilização de conteúdo marcado semanticamente, possibilitando que as informações das páginas sejam facilmente processadas por agentes de software. A criação manual necessita ser feita pelo usuário final, que na maioria das vezes não possui qualquer conhecimento sobre as tecnologias envolvidas. Portanto, são necessárias ferramentas que facilitem o processo de autoria de marcação semântica de dados para estes usuários. Neste trabalho, depois da análise de diversas soluções encontradas, é proposta e desenvolvida a ferramenta de autoria EdMaSe, com o objetivo de facilitar ainda mais este processo.

1. Introdução

Desde a sua criação, a *World Wide Web* (WWW) tem experimentado um crescimento exponencial. A enorme quantidade de conteúdo disponível tornou necessária a existéncia de motores de busca para auxiliar os usuários na tarefa de encontrar os documentos mais relevantes. O principal problema associado aos motores de busca tradicionais, é que suas pesquisas são meramente textuais, o que compromete sua eficácia e abre brechas para ambigüidades [Daconta, Smith e Obrst 2003]. Neste contexto está inserida a Web Semântica, cujo principal objetivo é tornar a informação da Web processável por computadores [Berners-Lee, Hendler e Lassila 2001].

Microformatos são considerados a forma mais simples de se expressar marcação semântica na Web, permitem expressar diversas formas de informação estruturada. Porém, não possui um modelo de dados definido, as regras sintáticas usadas para extrair a informação estruturada do HTML são diferentes para cada microformato [Graf 2007], o que limita o uso desta tecnologia.

Estas limitações são superadas através do uso do modelo de dados RDF (*Resource Description Framework*), que é extensível por natureza e permite a utilização de qualquer vocabulário para criação de marcação semântica. Assim sendo, o W3C (*World Wide Web Consortium*) elaborou uma sintaxe que permite expressar triplas RDF dentro de documentos HTML: o *RDF in HTML Attributes* (RDFa). O objetivo do RDFa é permitir expressar RDF dentro de documentos (X)HTML, de forma que seja possível embutir informações estruturadas em páginas da Web [Adida e Hausenblas 2007].

O objetivo deste trabalho é desenvolver um editor de marcação semântica de dados, de fácil utilização, que funcione em um navegador web. Para isso, é gerado um código em formato padronizado (RDFa) e também são utilizadas ontologias, de acordo com o contexto a ser trabalhado. A idéia é que, uma vez que os produtores de conteúdo possam expressar informações estruturadas e os navegadores estejam preparados para entendê-las, um novo leque de funcionalidades estará disponível na Web.

2. Ferramentas de Autoria

Existem basicamente duas formas de criar conteúdo semântico para a Web: automaticamente e manualmente. A criação automática normalmente envolve algum tipo de *software* conversor, que mapeia as informações de um banco de dados relacional, ou de textos escritos em linguagem natural, para o modelo de dados RDF. A criação manual é mais complexa, porque necessita ser feita pelo usuário final, que na maioria das vezes, não possui qualquer conhecimento sobre RDF e vocabulários / ontologias.

De acordo com [Heese e Luczak-Roesch 2009], uma das principais barreiras para adoção em larga escala das tecnologias semânticas, é a falta de ferramentas que permitam aos usuários comuns da Web marcarem semanticamente o conteúdo que produzem. Foram analisadas funcionalidades e limitações das principais ferramentas para autoria de marcação semântica encontradas na literatura como: *IkeWiki*, *Semantic MediaWiki*, *Loomp*, mais detalhes em [Brandes 2009]. A partir desta análise foi então elaborado o projeto de uma nova ferramenta, chamada EdMaSe, que tem o objetivo de facilitar a autoria deste tipo de informação na Web.

3. Ferramenta Proposta - EdMaSe

Foram identificados diversos requisitos interessantes para uma ferramenta de autoria de marcação semântica. Nenhuma das ferramentas encontradas apresenta todas estas características em conjunto:

- Facilidade de uso: a principal finalidade de uma ferramenta de autoria é facilitar a criação de conteúdo. Portanto, a ferramenta deverá ocultar os detalhes particulares das tecnologias envolvidas.
- Capacidade de carregar ontologias dinamicamente: para obter uma maior flexibilidade e poder ser utilizada em diferentes contextos, a ferramenta deverá ser capaz de analisar uma ontologia descrita nas linguagens RDFS/OWL e gerar uma interface dinâmica com base na hierarquia de classes e propriedades definidas por ela.
- Marcação semântica em formato padronizado: a marcação semântica gerada pela ferramenta deverá ser criada em um formato padronizado (RDFa).
- Independência em relação ao servidor: uma característica presente em todas as ferramentas encontradas é que elas são fortemente acopladas e dependentes da aplicação que roda no servidor. Para possibilitar a criação de marcação semântica em diferentes sistemas, é interessante que a ferramenta seja independente do servidor. A marcação será gerada dentro dos navegadores dos usuários, portanto a ferramenta poderá ser incorporada a diferentes *blogs*, *wikis* e Sistemas Gerenciadores de Conteúdo na forma de *plugins*, sem que estes sistemas possuam qualquer conhecimento sobre RDF e OWL.

A Tabela 1 mostra um comparativo entre as características das ferramentas

analisadas e da ferramenta EdMaSe.

Tabela 1: Comparativo de funcionalidades das ferramentas analisadas

	Carregamento dinâmico de ontologias	Marcação em formato RDFa	Independência em relação ao servidor
IkeWiki	X	X	
SemanticMediaWiki	X	X	
Loomp		X	
EdMaSe	X	X	X

4. Implementação

Na ferramenta EdMaSe todas as interações com o usuário seguem o mesmo padrão: inicialmente é necessário selecionar a porção de texto que se deseja manipular; em seguida o botão de marcação semântica deve ser pressionado para iniciar o processo. A tela de manipulação da marcação permite ao usuário realizar todas as funcionalidades da ferramenta. A Figura 1 mostra um exemplo da sua interface.

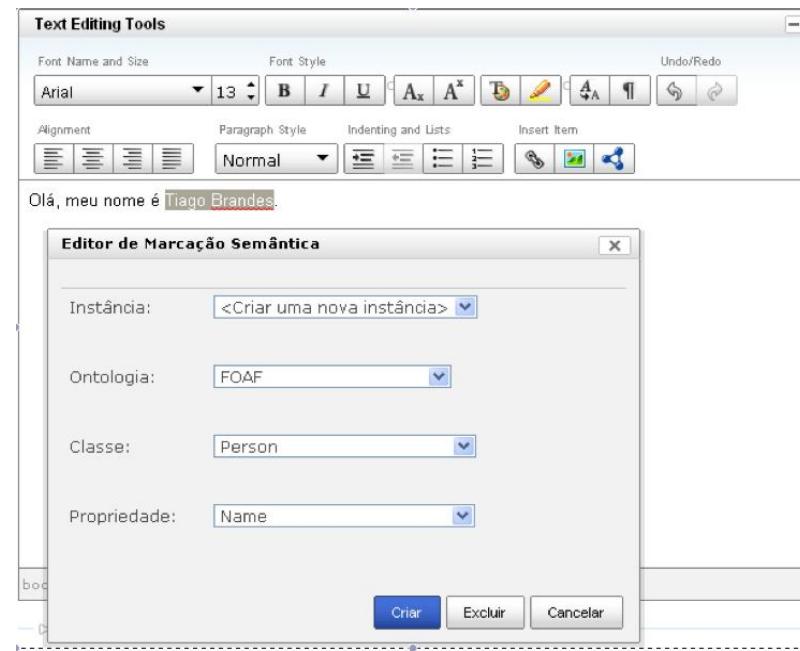


Figura 1. Interface do EdMaSe

Para criar uma nova marcação, o usuário deve escolher os seguintes parâmetros: instância, ontologia, classe e propriedade, e em seguida pressionar o botão Criar. Caso tenha escolhido criar uma nova instância, a ferramenta irá requisitar um nome, responsável por identificá-la dentro da página em questão. A partir deste momento, a nova instância estará disponível para ser utilizada. O processo de alteração é bastante

similar ao processo de criação. Também tem-se a funcionalidade de exclusão de marcação, onde é eliminado o respectivo RDFa.

Para demonstrar a utilização do editor EdMaSe em um cenário real, foi realizada a integração com o Sistema Gerenciador de Conteúdo (SGC) *Drupal*. A facilidade com que esta integração foi realizada demonstra que foi atingido o requisito de independência em relação ao servidor, bastando adicionar as dependências ao documento e enviar um arquivo *Javascript* para que o editor possa ser utilizado.

5. Conclusões e Trabalhos Futuros

A ferramenta desenvolvida possui dois diferenciais em relação às outras soluções existentes. O primeiro é a capacidade de criar e editar (ou excluir) a marcação diretamente no navegador dos usuários, independentemente da aplicação executada no servidor, o que permite a sua integração com diversos Sistemas Gerenciadores de Conteúdo sob a forma de *plugins*. O segundo diferencial é a capacidade de carregar ontologias dinamicamente para definir o tipo de marcação que os usuários podem realizar. Estas estruturas são carregadas em tempo de execução via JavaScript, de forma que a ferramenta não está acoplada a nenhuma ontologia em particular, o que permite a sua utilização em diferentes contextos de acordo com as necessidades dos usuários.

Como sugestões para trabalhos futuros, foram encontradas algumas possibilidades. A capacidade de processar ontologias em outros formatos, uma vez que esta implementação somente é capaz de processar o formato RDF/XML. Embora este seja o formato padrão definido pelo Consórcio W3C, existem diversos outros formatos populares que podem ser explorados no futuro. Também podem ser desenvolvidos *plugins* para outros Sistemas Gerenciadores de Conteúdo populares baseados na Web, como *Wordpress*, *Moveable Type* e *Plone*.

Referências

- ADIDA, B.; HAUSENBLAS, M. RDFa (2007) Use Cases: Scenarios for Embedding RDF in HTML. [S.I.]. Disponível em: <<http://www.w3.org/TR/xhtml-rdfa-scenarios/>>, Fevereiro.
- BERNERS-LEE T.; HENDLER, J.; LASSILA, O. (2001) The semantic web. Scientific American, Maio.
- DACONTA, M. C.; SMITH, K. T.; OBRST, L. J. (2003) The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management. New York, NY, USA: John Wiley & Sons, Inc. ISBN 0471432571.
- DEAN, M. et al. (2003) OWL Web Ontology Language Reference. [S.I.], Fevereiro. Disponível em: <<http://www.w3.org/TR/owl-ref/>>.
- GRAF, A. (2007) RDFa vs. Microformats: a comparison of inline metadata formats in (X)HTML. [S.I.], Abril 2007.
- HEESE, R.; LUCZAK-ROESCH, M. (2009) Linked data authoring for non-experts. In: Proceedings of the WWW09, Workshop Linked Data on the Web (LDOW2009). [S.I.: s.n.], 2009.

KDD Cleaning – Ferramenta para pré-processamento de dados em Descoberta de Conhecimento em Bases de Dados

Juliano Augusto Carreira¹, Tiago Luis de Andrade¹, Carlos Roberto Valêncio¹

¹Grupo de Banco de Dados – Departamento de Ciência da Computação e Estatística – Universidade Estadual Paulista (UNESP) – São José do Rio Preto – SP - Brasil

julianocarreira.tlacac@gmail.com, valencio@ibilce.unesp.br

Abstract. *Inherently to current large databases, there are some problems such as null values, missing values, duplicated tuples, outliers and others, if untreated, can affect the reliability of the knowledge extracted in the KDD process. Thus, there is a step in this process called “data cleaning” which must be performed before the stage of data mining and is responsible for tasks of correction and adjustment in the data. The tool proposed here focuses on that stage and implements some of the existing techniques for handling these problems in order to ensure greater consistency and reliability of the data that are used as raw for data mining.*

Resumo. *Inerentemente às grandes bases de dados atuais, existem alguns problemas tais como valores nulos, valores ausentes, tuplas duplicadas, valores fora de domínio, entre outros, que, se não tratados, podem prejudicar a confiabilidade do conhecimento extraído no processo de KDD. Para isso, existe uma etapa nesse processo denominada “limpeza de dados”, que deve ser executada antes da etapa de data mining e que é responsável por realizar correções e ajustes nos dados. A ferramenta, aqui apresentada, foca nessa etapa e implementa algumas das técnicas existentes para tratamento desses problemas, visando garantir uma maior consistência e confiabilidade nos dados que serão utilizados como alvo na mineração de dados.*

1. Introdução

Em virtude da evolução das tecnologias de hardware e software, a capacidade de armazenar e processar dados vem aumentando muito e, em consequência disso, é cada vez mais natural deparar-se com grandes volumes de dados nas mais variadas áreas. Nas bases de dados existentes atualmente, seja por um projeto ruim ou por falhas no processo de alimentação, é muito comum encontrar problemas relacionados à inconsistência de dados e ao esquema, como erros de digitação, valores nulos ou fora do domínio, tuplas duplicadas, inexistência de chave primária e etc. Para tratar esses problemas, o processo de *Knowledge Discovery in Databases* (KDD) provê a fase de limpeza de dados dentro da etapa de preparação, cujo objetivo principal é garantir a integridade e consistência destes para etapas futuras, mais precisamente a etapa de *data mining* que é responsável pela mineração de dados cujos resultados, posteriormente, serão transformados em conhecimento útil (Han; Kamber, 2006).

Atualmente já existem algumas ferramentas focadas na limpeza de bases de dados: *Address Doctor*, *Trillium*, *help IT System*, *Winpure*, *DataFlux*, *DQ Global*, *Grit*

Bot e *proMiss* (Piatetsky-Shapiro, 2010). Porém, estas ferramentas são especializadas e realizam somente parte do trabalho, tratando problemas específicos e nem sempre utilizando as melhores técnicas, o que obriga a utilização de várias ferramentas para a realização de uma limpeza completa de uma base de dados. Sendo assim, o intuito da ferramenta proposta neste trabalho é realizar a limpeza de dados atacando a maioria dos problemas que uma base de dados real pode apresentar. Para cada problema em específico, tentou-se e continua-se tentando implementar a maior quantidade de técnicas possíveis para melhor atender o propósito da etapa de limpeza de dados e também acabar com a necessidade de utilização de várias ferramentas no processo de limpeza.

2. Descrição da Ferramenta

A ferramenta *KDD Cleaning* foi implementada utilizando a linguagem de programação Java e trabalha com o auxílio do Sistema Gerenciador de Banco de Dados MySQL. Os módulos implementados para o tratamento dos problemas citados serão apresentados nas subseções a seguir. A figura 1 apresenta a ferramenta proposta neste trabalho. Pode-se notar que cada aba da ferramenta é responsável pelo tratamento de um problema em específico.

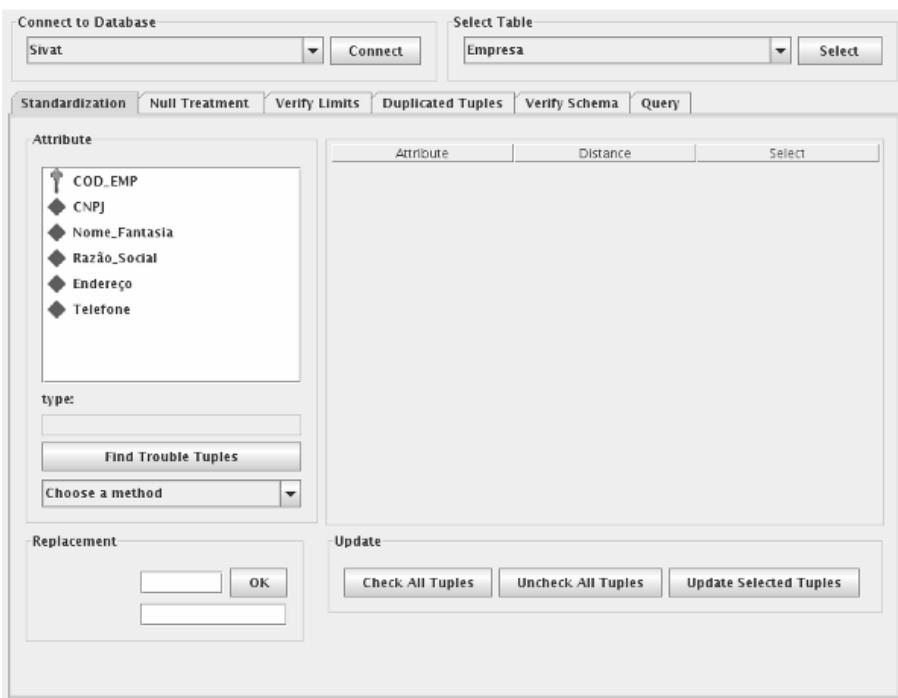


Figura 1. Ferramenta *KDD Cleaning*.

2.1. Padronização

Módulo que implementa técnicas que atuam na padronização de valores da base de dados que por ventura tenham sido digitados incorretamente. As técnicas implementadas para este módulo são:

- Busca por *substring*: identifica registros, em uma base de dados, que possuam como parte de um dos valores de seus atributos uma *string* fornecida pelo usuário. Essa funcionalidade recebe o auxílio de uma tabela *hash* que armazena

valores que se repetem e sua respectiva quantidade de repetições. Sendo assim, pode-se realizar buscas baseadas nos valores que mais se repetem.

- Distância de *Levenshtein*: algoritmo proposto por Levenshtein (1966) o qual possui sua essência na programação dinâmica e exerce a função de calcular a menor distância entre duas *strings* A e B, a qual é mensurada pela menor quantidade de operações de edição de caracteres para transformar uma *string* A em uma *string* B ou vice-versa.
- *Soundex*: algoritmo proposto por Knuth (1973) que encontra similaridade entre palavras por meio dos sons fonéticos que podem ser avaliados pelo conjunto de consoantes dessas palavras. Ou seja, é capaz de dizer se a pronúncia de duas *strings* é parecida, apenas utilizando suas consoantes.

2.2. Tratamento de valores nulos

Módulo que implementa técnicas para o preenchimento de valores nulos de atributos que não foram preenchidos por algum motivo desconhecido. As técnicas implementadas para este módulo são:

- Preenchimento manual: permite ao usuário preencher manualmente os valores ausentes.
- Preenchimento baseado em média: estratégia muito útil para atributos numéricos que calcula uma média dos valores existentes para o atributo em questão e preenche os registros, cujos valores são nulos, com a média encontrada.
- Preenchimento baseado em média por grupo: também realiza o cálculo de média baseando-se em valores existentes, porém o faz para grupos específicos, por exemplo: gênero (Masculino | Feminino), estações do ano (Primavera | Verão | Outono | Inverno), classe social (Baixa | Média | Alta) e etc.
- Constante global: permite ao usuário preencher os valores ausentes utilizando uma única constante.
- Classificador de *Naïve Bayes*: faz parte do conjunto de classificadores considerados *Bayesianos* que são estatísticos e baseiam-se no teorema de *Bayes* para predizerem a probabilidade de algo, como uma tupla de uma tabela que faça parte de uma classe em particular (Han; Kamber, 2006). Geralmente utilizada como estratégia de *data mining*, neste trabalho foi utilizada com o propósito de auxiliar no preenchimento de valores nulos encontrados em bases de dados reais.

2.3. Verificação de limites

Módulo responsável por encontrar valores fora do domínio e oferecer ao usuário a possibilidade de corrigir essas discrepâncias por meio da alteração do valor ou remoção da tupla. A técnica implementada, neste módulo, consiste em uma combinação de consultas SQL regidas por parâmetros de restrição de domínio oferecidos pelo usuário.

2.4. Tuplas Duplicadas

Módulo responsável por encontrar e tratar tuplas duplicadas por meio da remoção. As técnicas implementadas para este módulo são:

- Duplicatas idênticas: identifica tuplas idênticas armazenadas na base de dados por meio de uma combinação de consultas SQL.
- Duplicatas *Fuzzy*: identifica tuplas que são idênticas no mundo real, mas que estão armazenadas de forma diferente na base de dados. Para a implementação desta técnica, foi utilizado o algoritmo AA-SNM proposto em YAN et al. (2007).

2.5. Verificação de esquema

Módulo responsável por oferecer um conjunto de recursos que promova alterações na estrutura da base de dados, visto que muitos problemas encontrados nas bases surgem na fase de projeto e precisam ser corrigidos.

2.6. Consulta

Módulo responsável por realizar e apresentar resultados de consultas na base de dados. Este módulo foi desenvolvido para proporcionar maior comodidade ao usuário que utiliza a ferramenta, evitando que o mesmo necessite utilizar alguma outra ferramenta para verificação dos resultados de suas alterações.

3. Aplicação da Ferramenta e Conclusões

A ferramenta já foi aplicada com êxito em uma base de dados real que armazena aproximadamente trinta mil registros sobre acidentes de trabalho que ocorrem na região de São José do Rio Preto-SP. O principal problema identificado nessa base de dados foi a tabela “empresa”, pelo fato de que não foi possível conseguir uma tabela que contivesse todas as empresas localizadas na região. Diante das circunstâncias, a solução adotada foi a criação da mesma pelos próprios usuários a medida que as fichas fossem cadastradas no sistema. Sendo assim, a criação de duplicatas *fuzzy* já era esperada e precisava ser tratada. A ferramenta mostrou-se eficaz ao reduzir aproximadamente sete mil registros duplicados da tabela de empresas. Outras funcionalidades da ferramenta também foram aplicadas com êxito na base de dados, porém com uma quantidade menos expressiva de correções.

5. Referências Bibliográficas

- Levenshtein, Vladimir I. (1966) “Binary Codes Capable of Correcting Deletions, Insertions and Reversals”, In: Soviet Physics Doklady, p. 707-710, 1966.
- Knuth, Ervin D. (1973) “The Art of Computer Programming vol. 3”, p. 391-392, 1973.
- Han, J., Kamber, M. (2006) “Data Mining: Concepts and Techniques 2. ed.”, p. 310-317, 2006.
- Yan, S. et al. (2007) “Adaptive Sorted Neighborhood Methods for Efficient Record Linkage”, In: JCDL'2007, p. 17-22, 2007.
- Piatetsky-Shapiro, G. (2010) “Kdnuggets”. Disponível em: <www.kdnuggets.com>. Acesso em: 07 jan 2010.

Processo de Descoberta de conhecimento em Base de Dados aplicado a Bolsa de Valores

Débora Cabral Nazário¹, Rogério Giacomini de Almeida¹

¹Departamento de Ciência da Computação – DCC
Universidade do Estado de Santa Catarina (UDESC)
Campus Universitário Prof. Avelino Marcante s/n - Bairro Bom Retiro -
Joinville – SC
Brasil CEP 89223-100

debora@joinville.udesc.br, rogeriogiacomini@gmail.com

Resumo. *Este trabalho realiza análises sobre ações da BOVESPA aplicando técnicas de classificação de Data Mining. Dessa forma, foi possível extrair conhecimento útil e válido para investidores. Foi desenvolvida uma aplicação que resultou na classificação das ações de uma bolsa de valores, utilizando um algoritmo conhecido como Nearest Neighbors, onde foi possível obter bons resultados em comparação a um método utilizado cotidianamente por analistas de ações, que é o método de Médias Móveis.*

1. Introdução

Em um ambiente complexo, como o de uma bolsa de valores, é útil a existência de sistemas que facilitem a tomada de decisão de um investidor. O objetivo destes sistemas é observar oscilações do mercado em períodos determinados, auxiliando assim, usuários a optarem e realizarem melhores investimentos baseados nas informações obtidas destes sistemas. Então, de acordo com informações passadas pelo usuário, o sistema conseguirá buscar nos dados, já armazenados, informações relevantes para determinada aplicação.

[Minardi 2001], afirma que na década de 50 já existiam aplicações econômicas que analisavam em séries temporais o valor de ações, tentando prever o progresso da economia. Acrescenta-se que uma boa análise do comportamento de ações envolve uma quantidade de dados elevada, por exemplo, as cotações do ano de 1998 (um período economicamente estável para o país) da Bolsa de Valores de São Paulo (BOVESPA), somam um total de 104.972 registros. Essa grande massa de dados dificulta a tarefa realizada pelo investidor que precisa analisar e entender esses grandes volumes de dados. Para isso os sistemas computacionais serão utilizados para automatizar esta tarefa.

O objetivo deste trabalho concentra-se em viabilizar ao investidor informações relevantes para tomar suas respectivas decisões de acordo com os resultados obtidos com a utilização do sistema desenvolvido.

2. Técnica e Algoritmo Escolhidos

O foco do trabalho é então, auxiliar investidores a tomar decisões sobre investimento em ações. Para solucionar este problema, a aplicação do processo de Descoberta de Conhecimento em Bases de Dados (DCBD) é utilizada na busca de padrões sobre o

comportamento histórico do valor das ações. Esses padrões são definidos em uma etapa do processo conhecida como etapa de mineração [Fayaad 1996].

Para realizar a mineração de dados, a técnica de classificação foi aplicada por atender o objetivo geral deste trabalho, pois identificando se o valor da ação subiu, caiu ou manteve, ou seja, atribuindo uma classe para cada dia da cotação de uma ação, é possível estudar uma tendência e, dessa forma, auxiliar o usuário do sistema a tomar a decisão de como investir [Witten 2005].

Conforme o teorema de *No-Free-Lunch Theorem* (NFL), os algoritmos de classificação possuem a mesma importância em qualquer problema de classificação. Logo, a cada nova aplicação todos devem ser testados, identificando os de melhor desempenho [Soares 2007]. Foram realizados testes com 20 algoritmos de classificação. Foram selecionados algoritmos que apresentaram melhores resultados, baseando-se no desempenho avaliado. Este é avaliado por informações estatísticas obtidas com a aplicação de cada uma das técnicas testadas.

De acordo com os resultados que foram obtidos, os algoritmos que se baseiam no método do vizinho mais próximo, atingiram o melhor resultado, já que uma instância é comparada ao(s) seu(s) vizinho(s) e dessa forma tem sua classificação de forma mais coerentemente atribuída. Quando se trata da bolsa de valores, essa característica é muito importante, pois os valores apresentam um comportamento sazonal mesmo que esse período seja variável. O algoritmo *Nearest Neighbors* foi o que apresentou os melhores resultados nos testes que foram realizados [Almeida 2009].

3. Ferramenta para o Estudo de Caso

A arquitetura do sistema divide-se em *frontend* e *backend*, correspondendo as rotinas de *interface* com o usuário e as rotinas de acesso ao módulo do sistema que implementa a técnica de *Data Mining* - DM, respectivamente.

O *frontend* corresponde a uma interface de acesso para os usuários do sistema. Enquanto que o *backend* à interface que disponibiliza funcionalidades das técnicas de DM. É no *backend* que se faz acesso ao banco de dados MySQL, de onde coleta os dados que irá analisar, além de retornar ao os resultados das análises.

Optou-se pelo uso da linguagem Java no desenvolvimento de uma aplicação que faz o acesso direto a API do *software* de mineração de dados *RapidMiner*. Para desenvolvimento do trabalho foram utilizadas tecnologias como PHP, MySQL, Apache, Java e também foram pesquisadas tecnologias que empregassem técnicas de DM. Duas ferramentas foram encontradas, WEKA e *RapidMiner*. A ferramenta WEKA já é bastante difundida entre o meio acadêmico na área de mineração de dados. A outra, *RapidMiner*, é mais recente e inclui funcionalidades da ferramenta WEKA. Logo, optou-se pela utilização do *RapidMiner*.

4. Resultados

Foram desenvolvidas três formas de apresentação dos resultados obtidos, dois gráficos e um relatório resumido. A utilização dos dois gráficos foi realizada para comparar o método utilizado usualmente com a solução proposta por este trabalho. Dessa forma os dois resultados podem ser analisados juntos, para se atingir maiores informações do

processo, conforme Figura 1. As curvas, verde claro e marrom são os valores da ação. A curva verde escuro representa a média móvel dos valores da ação analisada. A curva azul representa as classes, identificando as instâncias treinadas. Quando a linha está no topo significa que a ação SUBIU, no meio MANTEVE e abaixo CAIU (gerado pelo sistema).

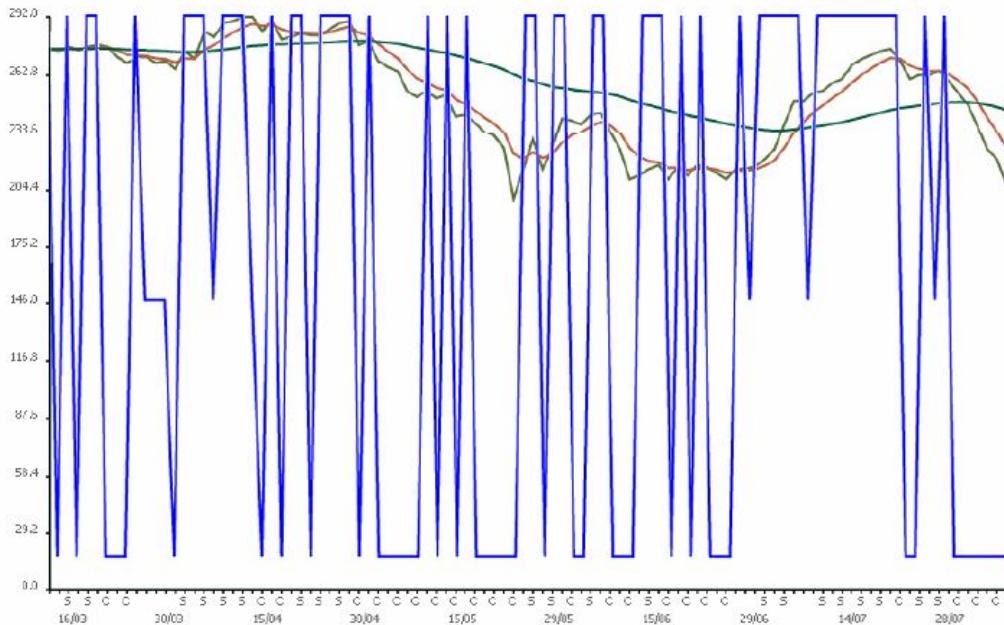


Figura 1. Gráfico Comparativo

Os resultados apresentados na Figura 1 foram obtidos analisando as ações da Petrobrás preferenciais (PETR4), no período (tanto para geração do modelo, quanto para aplicação) de março a agosto de 1998. Enquanto o gráfico dos valores (curvas verde e marrom) indicava apenas sete variações, o gráfico de classes (curva azul) indica um número bastante expressivo. Observando os pontos de cruzamento das curvas do gráfico de valores, nota-se que por volta do dia 30 de abril este gráfico inicia um período de queda que só vai terminar no final do mês de junho, talvez início do mês de julho. Os períodos que determinam a alta / queda através deste gráfico são muito vagos, por se tratar de uma análise continua de dados, são médias.

O gráfico das classes apresenta resultados diferentes, é mais preciso em relação ao valor da ação que está sendo analisada. Em um pequeno período de alta, este gráfico aponta este período, mesmo que seja pequeno, como por exemplo, a última semana do mês de maio apresentou um pequeno período de alta e o gráfico de classes correspondeu.

O motivo do gráfico de classes apresentar melhores resultados é que o algoritmo utilizado avalia instância por instância e certo relacionamento entre essas, neste caso, de acordo com o número de vizinhos que são analisados no algoritmo *Nearest Neighbors* e também aos outros atributos que constituem a instância completa que neste caso, contou apenas com o volume total de negociação.

Finalmente, uma alternativa para o usuário interpretar os resultados obtidos, é através de um relatório com algumas informações ao final da aplicação do classificador.

Neste relatório, informações como o dia da semana favorável para compra e venda, foram muito bem recebidos por usuários experientes e atuantes desse ramo. O número de ocorrências das classes em separado por dias da semana e meses também teve boa aceitação, pois auxilia o investidor a identificar a situação atual a que se encontra a empresa que está sendo analisada. Mais detalhes em [Almeida 2009].

5. Conclusões e Trabalhos Futuros

Os resultados obtidos pela aplicação do algoritmo *Nearest Neighbors*, de classificação em DM a dados da bolsa de valores foram bons, visto que, um usuário utilizando o sistema desenvolvido, tem nele uma ferramenta interessante para o auxílio da análise de períodos de cotações. Com a ajuda do sistema desenvolvido, um usuário, sendo este especialista ou leigo, pode realizar tal análise, sendo guiado pela classificação automática executada pelo algoritmo de DM.

O custo computacional em relação ao método de Médias Móveis é superior, pois existem recursos que exigem maior processamento, porém é possível apresentar ao usuário do sistema algo mais do que apenas gráficos com várias curvas. Com o sistema foi possível demonstrar ao usuário, gráficos e relatórios com uma interpretação, apresentando uma suposta afirmação de tendência, identificando ao usuário oportunidades de investimento. Foi interessante notar a reação de analistas e acionistas que viram o sistema, identificando resultados inovadores, como a indicação dos dias da semana para compra e venda de determinada ação.

[Stahnke 2008] desenvolveu um trabalho semelhante, porém comparando apenas classificadores. Unir uma especialização em classificadores aos resultados obtidos com este trabalho (Métodos estatísticos e outras técnicas de DM) pode ser interessante para desenvolver modelos de dados mais “inteligentes”, ou seja, com capacidade de interferir nos resultados utilizando diferentes técnicas de mineração.

Referências

- Minardi, A. M. A. F. (2001) Preços Passados Prevendo Desempenho de Ações Brasileiras. Artigo apresentado à Bovespa. São Paulo.
- SOARES., J. A. (2007) Pré-processamento em Mineração de Dados: Um estudo Comparativo em Complementação. Universidade Federal do Rio de Janeiro.
- Almeida, R. G. (2009) Aplicação do Processo de Descoberta de Conhecimento em Bases de Dados a uma Bolsa de Valores. Trabalho de Conclusão de Curso. Universidade do Estado de Santa Catarina. Joinville.
- Fayyad, U; Piatetski-Shapiro, G.; Smyth, P. The KDD Process For Extracting Useful Knowledge From Volumes of Data. In: Communications of the ACM, Nov.1996.
- WITTEN, I. H.. Data Mining: practical machine learning tolls and techniques. 2 ed. Elsevier, , 2005.
- STAHNKE, Fernando R. Uso de data mining no mercado financeiro. Novo Hamburgo, RS: 2008. 121 p. Monografia – Instituto de Ciências Exatas e Tecnológicas, 2008.

Uma Aplicação de Banco de Dados Espacial para Filtragem de Dados para Agricultura de Precisão

**Edson Murakami¹, Fabiana S. Santana², Antonio M. Saraiva³
Bruno U. Grisi³, Marcos Nogueira³, Albert M. Kuniyoshi³**

¹Departamento de Ciência da Computação – Universidade do Estado de Santa Catarina
Caixa Postal 631 – CEP 89.223-100 – Joinville – SC – Brasil

²Centro de Matemática, Computação e Cognição – Universidade Federal do ABC
Santo André – SP – Brasil

³Departamento de Engenharia de Computação e Sistemas Digitais – USP
São Paulo – SP – Brasil

murakami@joinville.udesc.br, fabiana.santana@gmail.com,
saraiva@usp.br, {grisi.bruno, nomarcos, albert.kuniyoshi}@gmail.com

Abstract. This paper presents a precision agriculture web application using spatial database as basis for handling georeferenced information of agricultural yield data. Open technologies and standards were used in order to facilitate the map generation and integration of specialized software artifacts.

Resumo. Este artigo apresenta uma aplicação web para agricultura de precisão que utiliza banco de dados espacial como base para manipulação de informações de produtividade agrícola georeferenciadas. Tecnologias e padrões abertos foram utilizados com o objetivo de facilitar a geração de mapas e integração de artefatos de software especializados.

1. Introdução

O Agronegócio brasileiro [Saraiva 2003], em particular a Agricultura de Precisão (AP), demanda cada vez mais de suporte da Tecnologia da Informação (TI) em várias das suas atividades. A coleta, o armazenamento, o processamento e a análise das numerosas variáveis e da grande quantidade de dados envolvidos em seus processos requerem sistemas de informação.

A característica principal da maioria dos sistemas de informação desenvolvidos para AP é o seu caráter monolítico e proprietário. Isso dificulta ou mesmo impede a integração com outros sistemas, sejam eles fonte ou destino dos dados gerados como resultado do processo de análise e decisão [Saraiva 2003] [Murakami 2007].

Portanto, o desenvolvimento desses tipos de software deve ser realizado com base em paradigmas que favoreçam a interoperabilidade e o reuso e permita sua evolução, para que as dificuldades sejam minimizadas e se utilize de modo mais efetivo o potencial da TI no agronegócio.

A tecnologia de serviços web permite o desenvolvimento e a integração de sistemas de informações como uma alternativa para resolver os problemas de interoperabilidade e reuso de software. Ela tem sido fortemente adotada e apoiada pelas principais empresas de computação no mundo, razão pela qual, embora seja relativamente recente no mercado, teve forte avanço [Saraiva 2003].

Neste artigo é apresentada uma aplicação web que utiliza as tecnologias de serviços web, banco de dados espacial e um algoritmo de filtragem de dados de produtividade. Essa aplicação utiliza recursos de um Sistema de Informações Geográficas (SIG) para geração de mapas, que auxiliam os usuários na quantificação, entendimento e gerenciamento da variabilidade espaço-temporal para tomada de decisões na AP.

2. Material e Métodos

Um Banco de Dados Espacial, em termos gerais, é um banco de dados comum que adiciona a funcionalidade de manipular informações espaciais vetoriais, isto é, as primitivas espaciais: Pontos, Linhas e Polígonos [Obe e Hsu 2009]. Os bancos de dados espaciais fornecem funções específicas para consulta e manipulação de dados usando linguagens como a *Structured Query Language* (SQL). Embora eles não necessitem ser de natureza relacional, a maioria dos mais conhecidos é [Obe e Hsu 2009].

Nesse trabalho foi utilizado o banco de dados espacial PostGIS [Obe e Hsu 2009], uma extensão do sistema gerenciador de banco de dados objeto-relacional PostgreSQL, que fornece capacidades espaciais. Os serviços web geradores de mapas são fornecidos pelo MapServer [MapServer 2009] e as funcionalidades da aplicação web cliente pelo OpenLayers [OpenLayers 2009]. A motivação pela escolha se deve a flexibilidade dessas soluções na geração de mapas e por serem implementações de padrões mantidos pelo *Open Geospatial Consortium* [OGC 2009]. Outras soluções que utilizam essas tecnologias de padrões abertos podem ser vistas em [Kulawiak et al. 2008]. O PostGIS foi escolhido porque é uma solução madura e bastante utilizada no mercado. Na Figura 1 é mostrada a arquitetura em três camadas com as respectivas tecnologias utilizadas neste trabalho.



Figura 1. Arquitetura em camadas.

Na camada Cliente está o OpenLayers que fornece as funcionalidades de visualização e manipulação dos mapas. Na camada Servidor está o MapServer, que implementa os serviços padrões da OGC para manipulação de mapas (WMS, WFS e WCS) [OGC 2009] e os serviços especializados, como o algoritmo de filtragem. Esses serviços manipulam os dados no banco de dados espacial PostGIS na camada Dados.

O sistema de projeção utilizado para desenhar os mapas e armazenar os dados foi o EPSG:4326, por ser o mais comum e também mais usado em *Global Positioning Systems* (GPS). Um sistema de projeção é uma representação, ou modelo, da superfície terrestre. Ele especifica uma maneira de representar a Terra, através de uma aproximação matemática do planeta (elipsóides, também conhecidas como esferóides) aliado a um tipo específico de representação de coordenadas [Chapman et al. 2005].

A linguagem de manipulação dos dados espaciais é a *Simple Features for SQL* (SFSQL) [Obe e Hsu 2009]. Basicamente, é um conjunto de funções que adiciona capacidades espaciais à linguagem SQL. Por exemplo, traçando um paralelo entre a SQL e a SFSQL, pode-se afirmar que, enquanto a SQL é capaz de responder a

perguntas como “Quais foram as vendas totais naquele mês?”, com a SFSQL é possível responder perguntas como “Quantas ocorrências de espécimes temos num raio de 50 km?”. A SFSQL é capaz de responder questões relativas ao espaço. A seguir é apresentado um exemplo do uso dessa linguagem:

```
SELECT ST_AsText(the_point_geospatial_elements) from geospatial_elements;
```

Nesta consulta, *the_point_geospatial_elements* trata da coluna da tabela *geospatial_elements*, que tem capacidade para guardar informações geométricas. A função *ST_AsText* é da SFSQL e é capaz de converter o objeto armazenado no formato binário *Well Known Binary* (WKB), para o formato texto *Well Known Text* (WKT).

3. O Algoritmo e a Aplicação de Filtragem de Dados de Produtividade

O algoritmo de filtragem de dados de produtividade é uma implementação da metodologia para identificação, caracterização e remoção de erros em mapas de produtividade gerados por sensores de produtividade instalados em máquinas colhedeiras equipadas com receptores de GPS e sensores de produtividade, umidade, velocidade, etc. Os dados coletados (latitude, longitude, umidade, produtividade, entre outros) são armazenados em arquivo texto para posterior tratamento. Mais detalhes sobre a metodologia podem ser obtidos em [Molin e Menegatti 2002].

Este algoritmo automatiza as etapas de identificação, caracterização e remoção de erros dos mapas de produtividade. Originalmente, essas etapas eram executadas com auxílio de planilha de cálculos. O algoritmo foi implementado como um serviço web e faz parte da aplicação web denominada *Filtering* [Murakami 2007]. Essa aplicação faz o upload do arquivo texto com dados de produtividade brutos, armazena os dados no banco de dados e aplica o algoritmo de filtragem. Sobre os dados filtrados são feitas as consultas utilizando a SFSQL, cujos resultados são usados para a geração de mapas, que por sua vez são usados para tomada de decisões na agricultura de precisão.

4. Análise dos Resultados

A utilização de tecnologias “open source” permitiu a integração de componentes que facilitaram a construção de aplicações web para AP. Tecnologias de SIG foram utilizadas, pois são fundamentais para AP, uma vez que a AP consiste na manipulação de dados georeferenciados para manejo localizado de áreas agrícolas (talhão). As tecnologias utilizadas permitem visualizar mapas, comparar e combinar variáveis que influenciam na produtividade. Essas tecnologias utilizam padrões reconhecidos que facilitam a integração e reutilização de serviços prontos, o que permite evoluir e criar novas aplicações e consequentemente acompanhar a rápida evolução da AP.

Uma das tecnologias fundamentais, que facilitou o desenvolvimento da aplicação, devido às funcionalidades pré-existentes, foi o banco de dados espacial PostGIS. Porém, alguns cuidados devem ser tomados, por exemplo, o PostGIS armazena as informações espaciais na forma (longitude, latitude) e não (latitude, longitude) como normalmente se usa, o que é uma fonte de erro bastante comum. Além disso, quando um objeto geométrico é salvo no banco, ele é salvo no formato binário e não em texto.

Na Figura 2, são apresentados dois mapas, à esquerda o mapa gerado com dados brutos e à direita com dados filtrados. No mapa da direita é possível perceber que os dados fora dos limites do talhão foram retirados.

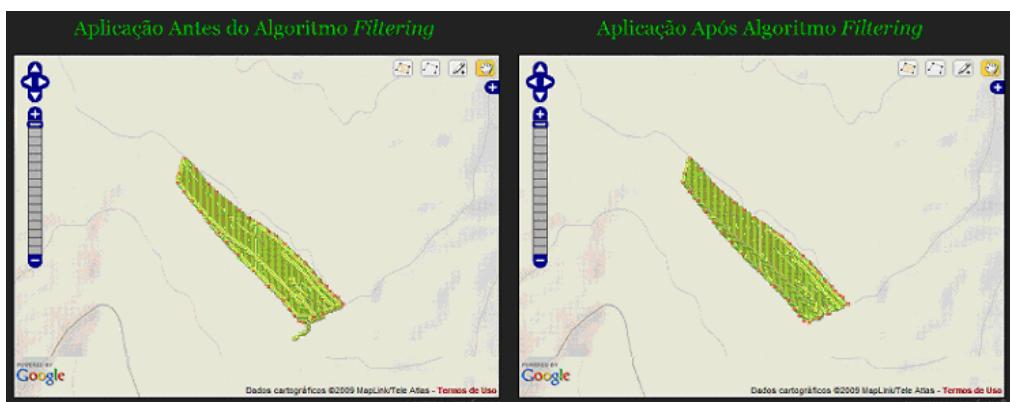


Figura 2. Mapas de produtividade antes e depois da filtragem. A geração é dinâmica e usa o mapa terrestre do Google Maps.

5. Conclusão

Atualmente diversas tecnologias para construção de SIGs para web estão disponíveis, várias delas “open source”. O que pode contribuir consideravelmente para a mudança do paradigma dos sistemas para AP, que são predominantemente monolíticos e proprietários [Murakami 2007]. Este trabalho demonstrou pragmaticamente a viabilidade de se construir sistemas de informações web para AP utilizando apenas soluções “open source”, e que o uso dos recursos de banco de dados espaciais para esses tipos de sistemas facilitam o seu desenvolvimento, pois simplificam a manipulação de dados espaciais, que são fundamentais para sistemas para AP.

6. Agradecimentos

Os autores são muito agradecidos ao CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico – Brasil, pelo suporte ao projeto “Infra-estrutura para sistemas orientados a serviços para agronegócio e ambiente”, sob concessão 476252/2008 3.

7. Referências

- Saraiva, A.M. (2003). Tecnologia da Informação na agricultura de precisão e biodiversidade: estudos e proposta de utilização de Web services para desenvolvimento e integração de sistemas. 2003. 185p. Tese (Livre-Docência) – Escola Politécnica, Universidade de São Paulo. São Paulo.
- Murakami, E., Saraiva A.M.; Junior L.C.M.R., Cugnasca, C.E., Hirakawa A.R. and Correa, P.L.P. (2007). An infrastructure for the development of distributed service-oriented information systems for precision agriculture. Computers and Electronics in Agriculture, v.58, pages 37 - 48.
- Obe, R.O. and Hsu, L.S. (2009). PostGIS in Action. Manning Publications.
- MapServer. (2009). MapServer open source web mapping. <http://mapserver.org/>. April.
- OpenLayers. (2009). OpenLayers free maps for the web. www.openlayers.org/. May.
- Kulawiak, M., Luba, M., Chybicki, A. Web-based GIS technologies dedicated for presenting semi-dynamic geospatial data. In: International Conference on Information Technology, 2008. Gdansk. Proceedings. Poland.
- OGC. (2009). Open Geospatial Consortium. <http://www.opengeospatial.org/>. April.
- Chapman, A.D., Muñoz, M.E.S. and Koch, I. (2005) Environmental information: placing biodiversity phenomena in an ecological and environmental context. Biodiversity Informatics 2. pages 24-41.
- Molin, J.P. and Menegatti, L.A.A. (2002). Methodology for identification, characterization and removal errors on yield maps. In: 2002 ASAE Annual International Meeting/CIGR XVth World Congress, 2002. Chicago. Proceedings. Illinois.

Uma ferramenta para gerar bancos de dados geográficos a partir de diagramas OMT-G

Klaus Werner Schaly, Angelo Augusto Frozza

Universidade do Planalto Catarinense (UNIPLAC)
Santa Catarina – SC – Brasil

klausschaly@globo.com; frozza@uniplac.net

Resumo. Este artigo apresenta uma ferramenta para gerar bancos de dados geográficos a partir de um diagrama OMT-G criado no software Star UML. Inicialmente, é apresentado um modelo genérico de mapeamento de elementos do modelo OMT-G para elementos correspondentes em um banco de dados objeto-relacional para, em seguida, apresentar a ferramenta proposta. Ao final, as considerações finais retratam o estado atual do projeto e desenvolvimentos futuros.

1. Introdução

Em computação, os objetos do mundo real são bastante complexos para serem representados por completo utilizando unicamente a tecnologia dos Sistemas Gerenciadores de Bancos de Dados (SGBD) atuais. Essa complexidade torna-se maior quando se trata de Banco de Dados Geográficos (BDG), os quais armazenam informações referentes à localização espacial, aos dados descritivos e às formas geométricas dos tipos geográficos [Casanova *et al.* 2005].

A modelagem de dados tem como objetivo, abstrair os objetos reais para objetos mais simples, mantendo as características mais importantes que possam ser representadas por um SGBD [Queiroz e Ferreira 2006]. No caso de BDG, este trabalho destaca o modelo OMT-G (*Object Modeling Technique for Geographic Applications*) [Borges 1997] [Casanova *et al.* 2005], o qual vem se consolidando como padrão para modelo de dados geográficos, a ponto de ser adotado como modelo padrão por instituições como a CONCAR - Comissão Nacional de Cartografia e a ANA - Agência Nacional de Águas [Frozza 2007]. O modelo OMT-G é um modelo de dados utilizado para o projeto conceitual de bancos de dados geográficos, baseado em alguns componentes da UML (*Unified Modeling Language*) e acrescido de componentes geográficos que tornam possível uma melhor transcrição de um modelo geográfico mental para um modelo de representação. Além de definir elementos geográficos, o modelo OMT-G permite a descrição de atributos alfanuméricos e métodos [Borges 1997].

Porém, atualmente, uma dificuldade para utilizar o OMT-G é a falta de ferramentas que automatizam o desenho de diagramas de BDG para este modelo. Como ferramentas conhecidas, há um *stencil* para o MS Visio 2000, o qual apresenta diversos problemas, e um software comercial para o qual foi desenvolvida uma extensão com os componentes do OMT-G [Davis Junior e Laender 2000]. Assim, disponibilizar ferramentas para modelagem de BDG com base no modelo OMT-G é uma necessidade imediata. Nesse sentido, uma interface gráfica para a geração de diagramas OMT-G é proposta em [Pinheiro 2009]. A partir desses diagramas, este artigo apresenta uma ferramenta que permite criar os *scripts* SQL para implementar o banco de dados no SGBD PostgreSQL [Schaly 2009].

O presente artigo está dividido em mais três seções. Na segunda seção é descrito um modelo genérico para mapeamento de um diagrama OMT-G para um banco de dados objeto-relacional. Na terceira seção é apresentada a ferramenta proposta. Na última seção são apresentadas as considerações finais.

2. Mapeando um diagrama OMT-G para um banco de dados objeto-relacional

A Tabela 1 resume o mapeamento entre os conceitos do modelo OMT-G e os conceitos compatíveis em um banco de dados objeto-relacional [Borges 1997].

Tabela 1. Relação entre os modelos OMT-G e relacional [adaptado de Borges 1997]

Modelo OMT-G	Modelo Objeto-Relacional
Classe Georreferenciada	Relação <i>Entidade</i> com representação geométrica associada; se for do tipo geo-campo, incluir restrições de integridade referentes à representação adotada.
Classe Convencional	Relação <i>Entidade</i> .
Associação simples com cardinalidade 1:1 ou 1:N	Par <i>chave estrangeira-chave primária</i> .
Associação simples com cardinalidade N:M	Relação <i>Relacionamento</i> e dois pares <i>chave estrangeira-chave primária</i> .
Relacionamento espacial topológico	Restrição de integridade relativa ao tipo de relacionamento espacial.
Relacionamento em rede arco-nó	Dois pares <i>chave estrangeira-chave primária</i> entre a relação <i>arco</i> e a relação <i>nó</i> (nó anterior e nó posterior); restrição de integridade espacial adequada.
Relacionamento em rede arco-arco	Dois pares <i>chave estrangeira-chave primária</i> em auto-relacionamento sobre a relação <i>arco</i> ; restrição de integridade espacial adequada.
Agregação	Par <i>chave estrangeira-chave primária</i> entre a classe <i>parte</i> e a classe <i>todo</i> .
Agregação espacial	Restrição de integridade relativa a agregação espacial.
Generalização/especialização	Restrições de integridade entre subclasses e superclasse.
Atributo simples	Atributo simples (coluna).
Atributo composto	Conjunto de atributos simples.
Métodos ou operações	<i>Triggers</i> ou programas associados.

O processo de mapeamento de diagramas OMT-G para um banco de dados objeto-relacional divide-se em etapas [Casanova *et al.* 2005]:

- converter as *classes convencionais* e *georreferenciadas* para tabelas do BD. Cada atributo de uma classe se torna uma coluna na tabela correspondente. Os atributos geográficos definidos no diagrama OMT-G devem ser de um tipo de dados compatível com o SGBD. Cada uma das tabelas deve possuir um atributo chave, que deve ser único para cada registro da tabela. Se nenhum dos atributos da classe possuir as características necessárias para ser um atributo chave, um novo atributo deve ser criado;
- mapear as *associações* e *agregações simples*. Nas relações que possuem cardinalidade 1:1, uma das duas classes deve possuir um atributo que seja uma referência à chave primária da segunda classe. Em relações que possuem cardinalidade 1:N, a classe N deve possuir uma chave estrangeira para o atributo chave da classe 1. Nos relacionamentos de cardinalidade N:N, uma classe intermediária precisa ser criada, contendo referências às chaves primárias de ambas as tabelas N. O mapeamento ideal de relacionamentos espaciais não causa alterações diretamente nas tabelas construídas até este passo, mas requer a implementação de controles dinâmicos (*triggers*) ou estáticos (verificações *offline* de consistência);
- fazer o mapeamento das *generalizações* e *especializações*. Existem três opções para realizar este mapeamento:
 - Primeira opção: cria-se uma tabela para a superclasse e cada subclasse é representada em uma tabela distinta, que possui como chave primária e estrangeira a mesma chave primária definida na tabela da superclasse. O atributo geográfico fica com a subclasse. Esta opção é recomendada para subclasses que possuam atributos próprios;

- Segunda opção: são criadas apenas as tabelas para as subclasses, que possuem todos os seus atributos, mais os atributos da superclasse;
- Terceira opção: apenas uma tabela é criada contendo todos os atributos da superclasse e de todas as subclasses. Para diferenciar os atributos de uma classe das atributos de outra classe são inseridos dois novos atributos, um representando o tipo de classe e outro identificando a qual classe o atributo pertence.

3 Conversão do diagrama OMT-G em *scripts* para o banco de dados PostgreSQL

Arquitetura da ferramenta é apresentada na Figura 1. A ferramenta foi dividida em dois módulos: o módulo interface da aplicação, que faz a leitura e processamento do diagrama OMT-G, e um módulo que faz o mapeamento dos elementos do diagrama para elementos equivalentes no SGBD.

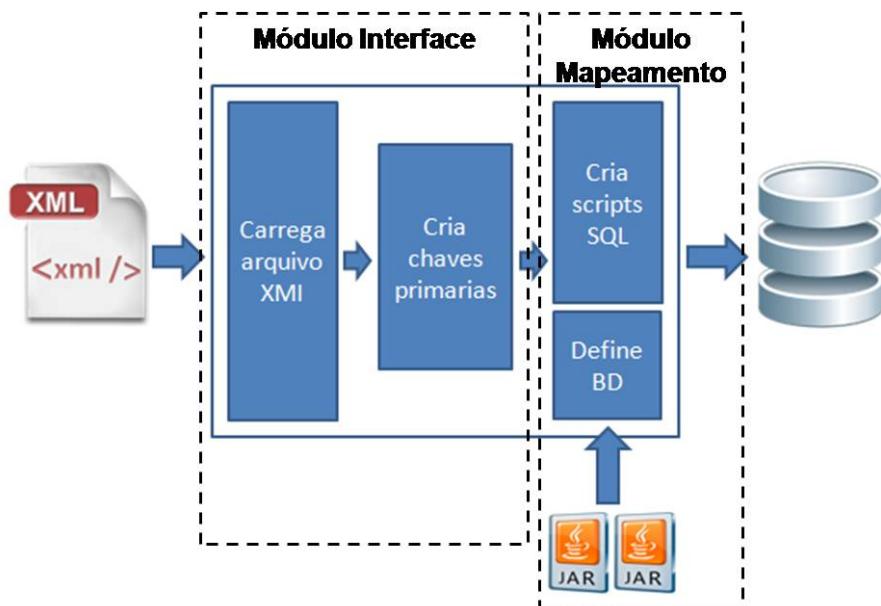


Figura 1. Arquitetura da aplicação

O primeiro módulo corresponde à interface principal da aplicação (Figura 2), a qual gerencia toda a operação. Na parte superior existe um *menu* contendo as opções para: carregar um arquivo XMI; selecionar o módulo referente ao SGBD utilizado na criação dos *scripts* SQL; salvar os *scripts* gerados. Ao lado esquerdo existe uma árvore que representa os objetos contidos no arquivo XMI. No centro da aplicação, os *scripts* SQL gerados são apresentados ao usuário. O botão *Converter* é responsável por disparar a criação dos *scripts*. A operação de mapeamento é dividida em três partes:

- a) Fazer a leitura do arquivo XMI (gerado pelo *plug-in* do StarUML [Pinheiro 2009]) e a identificação dos elementos (objetos) que compõem o diagrama OMT-G. Esses elementos são inseridos em uma estrutura de dados em forma de árvore para que possam ser manipulados pela aplicação;
- b) Configurar as chaves primárias para que as associações possam ser criadas corretamente;
- c) Converter a árvore que contém os dados do arquivo XMI em *scripts* SQL, utilizando o módulo específico para o banco de dados previamente selecionado.

O segundo módulo corresponde a um arquivo *jar* (*Java Archive*) contendo as regras para a criação de *scripts* para um banco de dados específico (etapa “c” acima). Dessa forma, é possível adicionar suporte a novos SGBDs criando-se apenas o arquivo *jar* referente ao

segundo módulo. Para este trabalho, foi desenvolvido apenas a biblioteca específica para ser utilizada com o SGBD PostgreSQL/PostGIS.

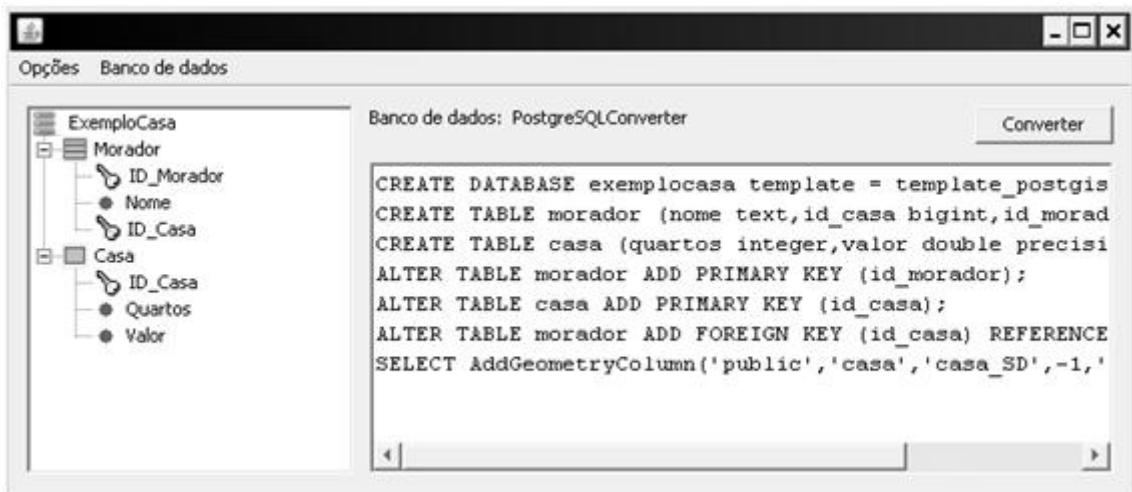


Figura 2. Interface principal da aplicação.

5. Considerações finais

O artigo apresentou uma ferramenta que permite o mapeamento de um diagrama OMT-G para um banco de dados objeto-relacional, gerando os *scripts* SQL para criação do respectivo banco de dados. Esta ferramenta estende as funcionalidades de um *plug-in* [Pinheiro 2009] criado para o *StarUML* que permite a criação de diagramas no modelo OMT-G. Em seu estado atual, a ferramenta obteve sucesso na criação dos *scripts* SQL para implementar BDG com poucos elementos (até 15 classes). Como trabalho futuro, está prevista a realização de experiências com diagramas mais elaborados, contendo construções com classes e relacionamentos relativamente complexos. Posteriormente, está previsto a expansão da ferramenta para uso com outros SGBD. Para tanto, deverão ser gerados os respectivos pacotes *jar* para cada SGBD a ser suportado.

Referencias bibliográficas

- Borges, K. A. V. "Modelagem de Dados Geográficos: Uma Extensão do Modelo OMT para Aplicações Geográficas." 1997. 128 f. Dissertação (Mestrado Tecnologias da Informação) – Fundação João Pinheiro, Belo Horizonte.
- Casanova, M. et al. "Banco de Dados Geográficos." Curitiba: MundoGEO, 2005. 490 p.
- Davis Junior, C. A.; Laender, A. H. F. "Extensões ao modelo OMT-G para produção de esquemas dinâmicos e de apresentação". II Workshop Brasileiro de GeoInformática - GeoInfo, 2000. Anais... São Paulo: SBC, 2000. p. 29-36.
- Frozza, A. A. "Um método para determinar a equivalência semântica entre esquemas GML." 2007. 139 f. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Santa Catarina – UFSC, Florianópolis.
- Pinheiro, H. H. C. "Modelagem de dados geográficos – Adaptação de ferramenta CASE baseada em software livre para suportar o modelo OMT-G." 2009. 21 f.
- Queiroz, G. R.; Ferreira, K. R. "Tutorial sobre Bancos de Dados Geográficos." GeoBrasil, 2006. 104 p.
- Schaly, C. W. "Ferramenta para criação de bancos de dados geográficos a partir de diagramas OMT-G". 79 f. 2009. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) - Universidade do Planalto Catarinense - UNIPLAC, Lages.

StereoMap: Um *Mashup* para Busca de Eventos Musicais

Josiane M. Diniz Duszczak¹, Lucas B. Zambon¹, Oliver M. Batista¹,
Leandro Ferreira¹, Issam Ibrahim¹, Carmem Satie Hara¹

¹Departamento de Informática – Universidade Federal do Paraná (UFPR) – Curitiba – PR – Brazil

{josiane, carmem}@inf.ufpr.br, benbom@gmail.com,
olivermbatista@gmail.com, lrferr@gmail.com, issamzao@gmail.com

Abstract. *The paper presents StereoMap, a web application for searching musical events by locality. It was developed as a mashup, i.e., by integrating data and services provided by external sources. StereoMap has four data/service providers: Last.fm, Google Maps, Twitter, and Wikipedia. Mashup applications have become a technological trend to develop integrated services and to provide more flexibility for the end user.*

Resumo. *Este artigo apresenta uma aplicação web que permite a busca de eventos musicais por localidade, chamada StereoMap. Ela foi desenvolvida utilizando o conceito de mashup, ou seja, através da integração de serviços e dados provenientes de diversas fontes. O StereoMap utiliza como provedores de conteúdo os seguintes serviços: Last.fm, Google Maps, Twitter e Wikipedia. Os mashups estão se tornando uma tendência tecnológica, facilitando o desenvolvimento de serviços integrados e proporcionando maior flexibilidade ao usuário final*

1. Introdução

Devido a grande disponibilidade de dados na Internet, há uma crescente necessidade de integrá-los de forma que seu acesso e entendimento sejam facilitados. Porém, integrar dados na Web não é uma tarefa fácil, uma vez que o processo envolve tanto o aspecto sintático e tecnológico, como a análise semântica das informações disponíveis. Um *mashup* é uma aplicação ou página web que combina dados de várias fontes para criar um novo serviço integrado. *Mashups* foram concebidos para facilitar o desenvolvimento de páginas web e proporcionar facilidades para combinar dados das mais variadas fontes.

Um exemplo de *mashup* é a página web *Chicago Crime*¹, que reúne dados extraídos da base de dados do Departamento Policial de Chicago e mapas fornecidos pelo Google Maps². Dentre os exemplos de *mashups* com dados musicais podem ser citados o *TuneGlue*³ e *MusicPlace*⁴. A maior parte das páginas web sobre música destinam-se a agrupar dados sobre os artistas e sua produção. Neste artigo é apresentado o StereoMap, um *mashup* para busca de eventos musicais por localidade. Dentre as principais funcionalidades do sistema destacam-se: integração com a rede social Twitter⁵, visualização do local do evento através do Google Maps e pesquisa de informações do

1 <http://chicagocrime.org>

2 <http://maps.google.com>

3 <http://audiomap.tuneglue.net>

4 <http://www.wdot.com.br/musicplace>

5 <http://twitter.com>

cantor ou banda no Wikipedia⁶. Desta forma, a ferramenta StereoMap, disponível em <http://www.sopalmeira.com/stereomap>, alia o conceito de *mashups* às redes sociais, promovendo uma maior interatividade entre grupos de interesse musical. A arquitetura, interface e implementação da ferramenta são apresentadas na seção 2, sendo as conclusões e trabalhos futuros discutidos na seção 3.

2. StereoMap

O StereoMap é um *mashup* que tem como objetivo principal proporcionar ao usuário uma ferramenta de pesquisa de eventos musicais de uma determinada cidade. Os eventos encontrados são apresentados como *marcações* em um mapa. Como objetivo secundário, o aplicativo disponibiliza informações adicionais sobre o evento musical, tais como nome e data do evento, além de informações sobre o artista. Por fim, como terceiro objetivo o StereoMap permite que o usuário comunique-se com seus conhecidos utilizando a ferramenta *Twitter*. Através desta interação é possível que todos os usuários que estão em contato com o autor da mensagem recebam as informações e também façam comentários sobre o evento.

2.1. Arquitetura do Sistema

Um *mashup* é composto pelos seguintes elementos [Merril 2006],[Wang et al. 2008]: provedores de conteúdo, página do *mashup* e aplicação cliente. Os *provedores de conteúdo* são responsáveis por fornecer dados, que são em geral disponibilizados através de APIs (Interface de Programação de Aplicativos), ou seja, possibilitam a utilização de suas funcionalidades sem envolver-se em detalhes da sua implementação. Em aplicações *web*, uma API pode ser definida através do formato de mensagens de requisição e resposta. O StereoMap possui atualmente quatro provedores de conteúdo, como ilustrado na Figura 1: Last.fm⁷, Google Maps, Twitter e Wikipedia. O segundo componente do *mashup*, o sítio do *mashup*, corresponde ao local no qual a aplicação é executada, que pode ser no servidor ou no cliente. No StereoMap, grande parte da lógica do sistema é executada no cliente, caracterizando-o como um *mashup* baseado no cliente. O componente *aplicação cliente* refere-se à aplicação através da qual o *mashup* é visualizado. Para *mashups* disponíveis na Internet uma aplicação cliente pode ser o navegador *web*.

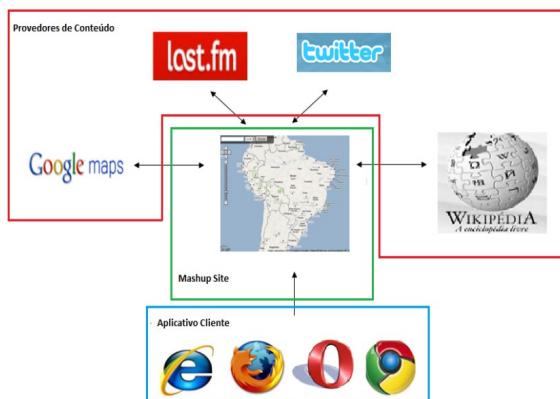


Figura 1. Arquitetura do Stereomap

6 <http://pt.wikipedia.org>

7 <http://www.last.fm>

2.2. Interface

A Figura 2 apresenta a interface do aplicativo. A tela de apresentação consiste no mapa obtido do Google Maps. No canto superior esquerdo há uma caixa de pesquisa, no qual é possível entrar com a cidade de interesse. Os pontos no mapa são as *marcações* que foram adicionados ao site com o auxílio das informações oriundas do Last.fm. Com a expansão da marcação, o StereoMap possibilita que o usuário acesse o Wikipedia e o Twitter.



Figura 2. Interface do Stereomap

2.3. Implementação

O StereoMap é uma aplicação *web* e apresenta uma arquitetura cliente-servidor tradicional. Para minimizar o tempo de espera do usuário, o sistema foi implementado utilizando a tecnologia AJAX (Asynchronous Javascript And XML) [Garrett 2005], com apoio da biblioteca Javascript Jquery [Bibeault et al. 2010].

O StereoMap foi desenvolvido de forma que grande parte do código fosse executado pelo cliente da aplicação. Por este motivo, um dos requisitos para a escolha das linguagens utilizadas é que elas fossem passíveis de interpretação pelos navegadores. Assim, as linguagens utilizadas foram: Javascript [Flanagan 2006] e PHP (Hypertext Preprocessor) [Welling 2004], para o desenvolvimento da parte interativa do sistema e apresentação dos dados; e JSONP (*Java Script Object Notation with Padding*) [Ippolito 2005] para a representação de objetos em JavaScript e transmissão de dados.

A obtenção dos dados dos provedores de conteúdo é realizada através das suas APIs. Dentre as APIs utilizadas, aquela disponibilizada pelo *Twitter* apresenta uma peculiaridade. Para utilizá-la é necessário utilizar um protocolo de comunicação de segurança, que permite o acesso somente de usuários cadastrados no sistema. Esse protocolo é chamado de *o-auth*⁸. A implementação do StereoMap tem aproximadamente 150 linhas de código. O tamanho reduzido do sistema mostra o potencial do conceito de *mashups* no desenvolvimento de aplicações *web*, reutilizando e integrando serviços disponíveis na criação de uma aplicação específica.

8 <http://oauth.net>

3. Conclusão

Este artigo apresenta a aplicação StereoMap, desenvolvida utilizando o conceito de *mashups*. Ele baseia-se na integração de serviços disponíveis, gerando economia de trabalho e facilitando o desenvolvimento de novos serviços. O StereoMap traz como benefício ao usuário final uma interface amigável que oferece um serviço completo. O sistema integra dados de eventos musicais e apresenta sua localização em um mapa, além de disponibilizar informações sobre o artista e permitir que o usuário conecte-se à rede social Twitter para comunicar a existência do show.

Como trabalho futuro, pretende-se utilizar o StereoMap como estudo de caso para determinar a adequação de ferramentas como Yahoo Pipes⁹ e Damia [Simmen 2008] no desenvolvimento de *mashups*. Estas ferramentas têm como objetivo simplificar a criação *mashups* utilizando recursos de arrastar e soltar, possibilitando que usuários finais desenvolvam suas próprias aplicações. Funcionalidades futuras da ferramenta incluem: um mecanismo de busca por banda e gênero musical, integração de outros provedores de conteúdo, tais como MySpace¹⁰, Amazon¹¹, e bases ontológicas como DBpedia[Auer et al, 2007] e DBTune¹².

Referências

- [Auer et al, 2007] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R. and Ives, Z. (2007). DBpedia: A Nucleus for a Web of Open Data, 6th International Semantic Web Conference, Lecture Notes in Computer Science 4825.
- [Bibeault et al. 2010] Bibeault, B. e Katz, Y. (2010). JQuery in action, Second Edition, Manning Publications.
- [Flanagan 2006] Flanagan, D. (2006). JavaScript: The Definitive Guide, Fifth Edition, Ed, O'Reilly Media, ISBN 0596101996.
- [Garrett 2005] Garrett, J. J. (2005). Ajax: A new approach to web applications. Disponível em <http://adaptivepath.com/ideas/essays/archives/000385.php>.
- [Ippolito 2005] Ippolito, B. (2005). Remote JSON – JSONP. Disponível em <http://bob.pythonmac.org/archives/2005/12/05/remote-json-jsonp/>.
- [Merril 2006] Merril, D. (2006). Mashups : The new breed of Web app. Disponível em <http://www.ibm.com/developerworks/xml/library/x-mashups.html>.
- [Simmen 2008] Simmen, D. E., Altinel M., Markl V., Padmanabhan, S. e Singh, A. (2008). Damia: data mashups for intranet applications, SIGMOD Conference, páginas 1171-1182.
- [Wang et al. 2008] Wang, X., Chen, Y. e Sha, J. (2008). The development model of Web applications based on mashups, IEEE International Conference on Service Operations and Logistics, and Informatics, Volume 1, Páginas 1059 – 1062.
- [Welling 2004] Welling, Luke e Thomson, Laura (2004), PHP and MySQL Web Development, (3rd Edition), ISBN: 0672326728.u

9 <http://pipes.yahoo.com/pipes>

10 <http://br.myspace.com>

11 <http://www.amazon.com>

12 <http://dbtune.org>