

A Proposal for Customizing Queries on XML documents based on Conditional Preferences

Angélica F. Medeiros, Valéria G. Soares, Eudisley Gomes dos Anjos
 UFPB: Centro de Informática – Universidade Federal da Paraíba (UFPB)
 Programa de Pós-Graduação em Informática – PPGI
 João Pessoa, Brasil
 angelicafelixx@gmail.com, valeria@ci.ufpb.br, eudisley@gmail.com

Abstract— This article presents a proposal to extend the XQuery query language called XQuery-Pref supporting conditional preferences. Aiming to become transparent to the user executing queries written in XQuery-Pref language, we use a system called XQPref, which is responsible for elicitation of dynamic preferences and the processing of these custom queries. We restrict the scope of this paper on Government Open Data that, given the information overload, has driven the demand for sensitive techniques to solve problems associated with querying XML documents. The government open data consist of the publication and dissemination of data and public information on the Web in an open format to facilitate analysis and reuse. However, information overload has attracted the concern of customize the query results according to the needs of each user.

Keywords— *Personalization of Queries; XQuery-Pref; XQPref.*

I. INTRODUÇÃO

O crescimento exponencial de informação disponível em diversos domínios tem chamado atenção dos pesquisadores, pois tem despertado a preocupação em personalizar os resultados de consultas de acordo com o perfil do utilizador [1], abrindo assim, possibilidades de pesquisa em áreas como banco de dados, inteligência artificial e recuperação de informação.

Dentre destes domínios, o XML é uma tecnologia essencial para a gestão do conhecimento e divulgação de dados pela web, no entanto com a sobrecarga de informações o usuário pode ser confrontado com uma quantidade muito grande de resultados, o que tem impulsionado a procura de técnicas sensíveis para resolver problemas associados com a busca em documentos XML [2].

Um exemplo recente de domínio que possui grande quantidade de informação em formato XML são os Dados Abertos, que constituem a publicação e disseminação de dados e informações públicas na Web, respeitando critérios que possibilitam sua reutilização e o desenvolvimento de aplicativos por toda a sociedade.

Como solução para realizar consultas em documentos XML, foram propostas algumas linguagens que devido às diferentes origens, possuem distinções consideráveis em termos de sintaxe e abrangência. Dentre elas, a linguagem XML-QL [3] é baseada numa sintaxe XML textual, já XML

Query Language -XQL [4], que foi proposta pela comunidade de processamento de documentos, explora expressões similares às expressões de caminho em diretórios. Enquanto a XQuery [5] é uma linguagem flexível, originada da mistura de ambos os tipos de linguagem e que dentre outras vantagens, possui o diferencial de possibilitar a criação de funções extensíveis.

No entanto, mesmo a linguagem mais completa para consulta em XML não possui base para personalização de consultas e assim realizar buscas em dados XML tem se tornado maçante devido à sobrecarga da informação, o que tem dificultado ao usuário o acesso a essas informações.

O presente trabalho busca personalizar as consultas em documentos XML através da mineração de preferências. Com isso, pode-se reduzir os esforços do usuário na busca por conteúdos específicos. Para isso, é proposta uma extensão da linguagem XML com suporte a preferências condicionais. Essa extensão utiliza um sistema denominado XQPref que será responsável pela elicitação de preferências dinâmicas e pelo processamento destas consultas personalizadas. Essa escolha permite-nos tornar as consultas escritas na linguagem XQuery-Pref mais transparentes para o usuário.

Este trabalho está organizado como a seguir. A Seção 2 descreve alguns trabalhos relacionados a esta proposta, bem como apresenta o estado da arte que envolve esta pesquisa. As Seções 3 e 4 descrevem a extensão proposta a linguagem XQuery e o sistema XQPref, respectivamente. Para finalizar, na Seção 5 são apresentadas as considerações finais.

II. TRABALHOS RELACIONADOS

A. Personalização de Consultas

A personalização de consultas tem como objetivo fornecer soluções que permitam filtrar o que é mais relevante, ou o que corresponda à expectativa e interesse de cada usuário. Para tanto técnicas de personalização vêm sendo pesquisadas e esta área de estudo vem crescendo e aplicando seus conceitos e em diversas áreas da computação, com o simples objetivo de dinamizar as consultas e produzir respostas personalizadas.

Como principais fundamentos desta área de conhecimento, diversos estudos foram tomados como base, a exemplo dos

trabalhos [6], [7] e tantos outros que desenvolvem muito bem os conceitos relacionados à personalização de consultas e modelagem de preferências. No entanto, a pesquisa apresentada em [8] merece ressalvas, para diferenciar as contribuições.

O trabalho de [8] também possui seu foco na utilização de preferências do usuário ao consultar documentos XML. Esta pesquisa propõe um método de consulta baseado em preferência contextual em documento XML com o objetivo de resolver o problema de respostas vazias ao realizar consultas, para tal ele propõe o relaxamento e pontuação de conteúdo. O modelo proposto é chamado de XCP e tem como objetivo permitir aos usuários expressar seus interesses sobre nós da árvore XML, e então o usuário atribuir pontuações para seus nós interessantes para fornecer as melhores respostas.

Comparando a pesquisa realizada em [8] ao presente trabalho, destaca-se que o autor não utiliza uma linguagem de consulta para XML em sua proposta, de modo que o usuário fica preso ao modelo e não consegue realizar buscas envolvendo artifícios disponíveis nestas linguagens (como agrupamento, consultas aninhadas, dentre outros). Outro ponto a ser destacado é que no presente trabalho o processo de elicitación das preferências ocorre de forma bastante simples para o usuário, pois o mesmo necessita apenas selecionar os elementos que lhe são prediletos, enquanto no modelo proposto em [8] um usuário leigo pode ter dificuldade ao se deparar com uma árvore para selecionar seus nodos prediletos.

B. Extensão de Linguagens de Consultas

Uma extensão a linguagem de consulta XQuery foi proposta em [9] visando proporcionar recursos para a recuperação de informações temporais e de versão representadas em documentos XML, para tanto são definidas funções temporais e versionadas que são incorporadas à linguagem base.

A importância da pesquisa realizada em [9] na fundamentação do presente trabalho se dá principalmente pelo fato da linguagem escolhida pelo autor ser a XQuery, e se atenua pela extensão ter sido realizada através da implementação de funções extensíveis. Sendo a diferença marcante entre os trabalhos, o foco principal das funções.

Outra pesquisa que fundamentou este artigo teve como resultado a CPref-SQL que foi proposta em [10], onde foi proposta a implementação da linguagem CPref-SQL, onde o modelo de preferência da CPref-SQL foi tornado mais expressivo e desenvolveram algoritmos que implementam os novos operadores de preferência Select-Best e SelectK-Best. Esses operadores são capazes de avaliar consultas top-K com preferências, ou seja, consultas que retornam as K tuplas mais preferidas de acordo com a hierarquia de preferências do usuário. Por fim, implementaram a linguagem no core do processador de consultas do Postgre-SQL.

Muitas semelhanças regem a especificação das extensões CPref-SQL e XQuery-Pref, uma vez que ambas permitem expressar preferências condicionais sob a abordagem qualitativa para possibilitar a modelagem das preferências

condicionais e se baseiam no formalismo de teorias de preferências condicionais especificadas em [6].

Entre as diferenças nestas extensões podem ser ressaltadas o foco e a forma como foram realizadas, pois a CPref-SQL tem como foco a consulta a banco de dados relacionais e foi implementada através da definição de novos operadores na linguagem base. A XQuery-Pref possui como foco a consulta personalizada em documentos XML, mas especificamente em dados abertos governamentais e esta extensão foi realizada através da definição de funções extensíveis, permitidas na própria XQuery. Outro fator de destaque, é que esta pesquisa foi mais além ao propor seu próprio sistema (XQPref) onde a elicitación de preferência é viabilizada.

C. Dados Abertos Governamentais

A inclusão digital aliada à informatização dos procedimentos governamentais e a integração entre os diversos repositórios de dados públicos provocam crescentes demandas da população por mais transparência e participação através de meios tecnológicos. Nessa direção o governo brasileiro tem definido políticas e desenvolvido plataformas tecnológicas na intenção de promover a disseminação das informações públicas.

Através do site <http://dados.gov.br/aplicativos/> é possível ter acesso a essas ferramentas, porém cada uma dessas ferramentas utiliza um conjunto específico de dados abertos governamentais no seu desenvolvimento de modo que, cada ferramenta possibilita o acesso a estes dados especificamente.

A exemplo destas ferramentas, o [11] é um aplicativo que utiliza os dados do Cadastro Nacional de Reclamações Fundamentadas para fornecer diversas informações sobre as empresas em formato ilustrativo e intuitivo, sendo de fácil utilização em smartphones e tablets devido ao layout vertical. Tal aplicativo está disponível também através do endereço: <http://reputacao-sa.org/>.

É importante destacar que estas ferramentas são desenvolvidas a partir de um conjunto específico de dados e dessa forma para que fosse possível analisar todos os dados abertos disponíveis pelo governo seria necessário um aplicativo para cada conjunto de dados, o que notoriamente é pouco viável. Sendo assim, este trabalho percebe a demanda de se criar uma ferramenta universal que possibilite a consulta independente desses dados, de modo que qualquer conjunto de dados possa ser analisado através dela.

Com isso, é justificado o desenvolvimento do XQPref que além de propiciar a elicitación das preferências necessárias para a extensão XQuery-Pref ainda viabiliza a realização de consultas nestes dados abertos governamentais que estejam em formato XML. Uma importante ressalva em relação à limitação pelo formato exigido do sistema XQPref é que os dados que não estejam disponíveis neste formato no site <http://dados.gov.br>, podem ser solicitados através do mesmo.

III. EXTENSÃO DA LINGUAGEM XQUERY

Para melhor apresentar a proposta, esta seção foi dividida em quatro subseções: Na primeira são especificadas as definições preliminares da pesquisa, e na segunda, o problema no tratamento das preferências do escopo deste trabalho é formalizado. A terceira subseção apresenta a especificação da extensão XQuery-Pref e na quarta do sistema XQPref, que proporciona a utilização desta extensão.

A. Definições preliminares

A XQuery-Pref é uma extensão da linguagem XQuery que permite realizar consultas em documentos XML contendo preferências condicionais. A sua grande vantagem é a possibilidade de expressar preferências condicionais de maneira intuitiva através de um conjunto de regras de preferência.

Em relação ao modelo de preferência, como visto, este é um formalismo lógico para especificação e raciocínio com preferências. Neste trabalho, o modelo de preferência segue a linha do formalismo de Wilson [2004], mas também serão uteis conceitos especificados com a CP-net [Boutilier et al, 2004], para melhor compreender como serão tratadas as preferências na personalização de consultas, no sentido de representar os desejos do usuário através de regras de preferências condicionais e teorias de preferências condicionais, que, por sua vez, induzem uma ordem de preferência sobre os elementos de um documento.

As expressões condicionais pré-estabelecidas no XQuery são bastante úteis no contexto da personalização de consultas baseado em preferências condicionais, pois estas expressões baseadas em if, then, else são favoráveis quando a estrutura da informação a ser retornada depende de alguma condição, podendo ainda ser aninhadas e usadas em qualquer lugar onde um valor é esperado.

Desta forma, quando o usuário fornecer suas preferências estas serão armazenadas em variáveis, onde uma relação de preferências sobre um conjunto finito de objetos $A=\{a_1, a_2, \dots, a_n\}$ formam uma relação de ordem sobre A , e esta relação de ordem será aplicada à consulta através das expressões condicionais.

A declaração de funções é uma das mais simples maneiras de estender uma linguagem e a XQuery implanta o conceito de funções extensíveis que atuam sobre documentos, seções ou valores atômicos. A grande vantagem na definição de funções na XQuery concentra-se na linguagem em que elas são definidas, pois a própria XQuery, com suas construções, permite que funções sejam escritas, não sendo necessário especificar funções numa linguagem de programação à parte.

Diante do exposto, um bloco de consulta na linguagem XQuery-Pref utiliza o domínio das expressões condicionais já existentes na linguagem XQuery aliada à flexibilidade da linguagem para criar funções customizadas.

De forma transparente ao usuário, cada preferência expressa se transforma em uma regra que é declarada seguindo a sintaxe IF <a> THEN <relacao_pref > e várias

regras são conectadas pela palavra-chave and. Estas regras podem ser criadas com ou sem antecedentes, resguardando a existência de preferências independentes, por exemplo, no caso de um usuário preferir informações sobre determinada região independente do tema tratado no conteúdo.

O parâmetro opcional <relação_pref>, é representado entre colchetes e indica os elementos que são menos importantes ou que não levam em conta a semântica ceteris paribus, dependendo das preferências já expressas.

Para nível de explicação, o termo ceteris paribus no contexto deste trabalho significa que os elementos que não são citados na preferência do usuário não influenciam no resultado, mas para que duas tuplas possam ser comparadas tais elementos precisam ser iguais.

B. Formalização do problema

Inicialmente, é importante destacar que no decorrer do trabalho será utilizado o termo tupla a qual é definida matematicamente como uma sequência finita (também chamada de "lista ordenada") de objetos, cada um deles sendo de um tipo específico. Chama-se atenção para o fato de o termo tupla utilizado no trabalho não se relacionar com o contexto do banco de dados relacional.

Uma regra de preferência condicional (regra-pc) sobre um conjunto de elementos E é uma expressão φ da forma:

$$\varphi: u \rightarrow Q_1(X) \wedge Q_2(X) [W] \quad (1)$$

- $X \in E, W \subseteq E, X \notin W$;
- $Q_i(X)$ para $i=1,2$, é um predicado unário sobre o domínio de X ($\text{dom}(X)$);
- $\{x \in \text{dom}(X) \mid x \models Q_1(X)\} \cap \{x \in \text{dom}(X) \mid Q_2(X)\} = \emptyset$;
- u é chamado de condição de regra-pc φ , onde este é uma condição simples de expressões do tipo: $P_1(A_1) \wedge P_2(A_2) \wedge \dots \wedge P_k(A_k)$, e cada $P_i(A_i)$ é um predicado unário sobre o atributo A_i , para $i=1,2,\dots,k$;
- Seja $E(u)$ o conjunto de atributos que aparecem em u . Então, $(\{X\} \cup W) \cap E(u) = \emptyset$.

Para cada elemento E que está do lado esquerdo de φ , uma tupla u sobre E é dita compatível com uma regra-pc. Tem-se que $u[A]$ satisfaz $P_\varphi(A)$, onde $u[A]$ constitui o valor do atributo A na tupla u e $P_\varphi(A)$ constitui a propriedade P associada ao atributo A na regra-pc φ .

A representação de um conjunto maior de preferências é feita através de uma teoria de preferência condicional (teoria-pc), onde uma teoria-pc sobre E é um conjunto finito de regras-pc sobre E .

Para melhor ilustrar a formalização, tomemos o exemplo a seguir o qual é um trecho de código XML que será utilizado durante todos os exemplos desta seção. A lista de elementos a

seguir, com seus respectivos domínios e suas siglas serão utilizadas nos exemplos abaixo.

- Modalidade educacional (M): educação especial (m_1), educação à distância (m_2) e educação infantil (m_3);
- Região (R): nordeste (r_1), sudeste (r_2), sul (r_3), norte (r_4);
- Dados (D): estatísticos (d_1) e orçamentais (d_2);
- Dados Estatísticos (d_1): evasão (d_{11}), aprovação (d_{12}), reprovação (d_{13});
- Dados orçamentais (d_2): empenho anual (d_{21}), alimentação (d_{22}), material (d_{23});

Exemplo: Tomando o trecho de um documento XML, que contém informações sobre Educação do Brasil, traz dados abertos como orçamento e dados estatísticos sobre Educação nas modalidades de ensino Educação Especial, Educação à Distância e Educação Infantil. Diante disto, para o usuário conseguir acessar o conteúdo de seu interesse, ele precisa enfrentar uma gama muito grande de informações, o que o levou a utilizar o sistema XQPref. Para que realize suas consultas ao documento de forma personalizada, o usuário forneceu as seguintes preferências:

- φ_1 Entre informações de mesma modalidade (M), prefiro as relacionadas com a região (R) nordeste (r_1) a norte (r_4), ou seja, $(R = r_1) > (R = r_4) \llbracket \{D\} \rrbracket$;
- φ_2 Para informações da mesma região (R), prefiro informações sobre educação especial (m_1) à educação à distância (m_2), ou seja, $(M = m_1) > (M = m_2) \llbracket \{D\} \rrbracket$;
- φ_3 Para informações sobre a educação especial (m_1) da mesma região (R), prefiro dados estatísticos (d_1), ou seja, $(M = m_1) \cdot (D = d_1) > (D = d_2)$;

Observando a regra φ_2 , por exemplo, observa-se que dadas duas tuplas x_1 e x_2 se elas tiverem informações sobre a mesma região, e x_1 abordar educação profissional enquanto x_2 a educação à distância, então x_1 é preferida a x_2 , logo $x_1 > x_2$.

Uma ordem de preferência sobre $X(A)$ é uma relação binária Pref sobre $X(R)$, isto é, um subconjunto $\text{Pref} \subseteq X(R) \times X(R)$ desde que seja irreflexiva e transitiva. Logo, uma ordem de preferência é uma relação binária irreflexiva e transitiva sobre duas tuplas X_i e X_j , que permite estabelecer se X_i é preferida a X_j , se X_j é preferida a X_i ou se X_i e X_j são incomparáveis. No contexto da presente proposta, uma teoria-pc Γ sobre uma relação A induz uma ordem de preferência sobre as tuplas de A.

As regras de preferências obtidas seguem o formalismo das Cp-nets [Boutilier et al., 2004], e, em conformidade com estas regras de preferências é possível inferir uma ordem sobre um conjunto de tuplas que pertence ao domínio do problema. De fato, a semântica de uma regra-pc está relacionada com a ordem de preferência induzida pela teoria-pc sobre as tuplas da relação. De modo que:

Seja $A(A_1, A_2, \dots, A_n)$ um documento XML e X um conjunto de tuplas sobre A, sabendo que X_i e X_j são tuplas no conjunto X, um elemento de $X_i \times X_j \times \{1,0\}$, com $X_i \neq X_j$ é denominado uma amostra de preferência. Ou seja, $\llbracket (a, b), 1 \rrbracket$ com $a, b \in X$, significa que a tupla a é preferida à tupla b e assim, $a < b$, enquanto $\llbracket (a, b), 0 \rrbracket$ significa que $b > a$.

Uma ordem de preferência sobre $X(A)$ é uma relação binária Pref sobre $X(R)$, isto é, um subconjunto $\text{Pref} \subseteq X(R) \times X(R)$ desde que seja irreflexiva e transitiva. Logo, uma ordem de preferência é uma relação binária irreflexiva e transitiva sobre duas tuplas X_i e X_j , que permite estabelecer se X_i é preferida a X_j , se X_j é preferida a X_i ou se X_i e X_j são incomparáveis. No contexto da presente proposta, uma teoria-pc Γ sobre uma relação A induz uma ordem de preferência sobre as tuplas de A.

Em um primeiro momento, este trabalho propõe para a consulta XQuery-Pref a obtenção de uma ordem de preferência forte sobre um conjunto de tuplas, inspirada no formalismo de Wilson [2004]. A ordem forte é baseada na semântica *ceteris paribus*, mais conservativa e intuitiva. Com os estudos sobre a ordem forte espera-se que ela seja suficiente para resultados satisfatórios no contexto da busca em documentos XML, tendo em vista a representação estruturada dos dados. Onde:

Seja φ uma regra-pc sobre os elementos de A, $E(A)$. Denotamos \succ_{φ} o conjunto de pares de tuplas $(x_1, x_2) \in X(A)$ que satisfazem as seguintes condições:

- Para cada elemento $A \in E(A)$ que aparece do lado esquerdo de φ , temos $x_1[A] = x_2[A]$ e $x_1[A] \models P_{\varphi}[A]$ (predicado P associado ao atributo A em φ).
- Para cada atributo $B \in E(A)$ que não aparece em φ , temos $x_1[B] = x_2[B]$;
- Para cada atributo X que aparece do lado direito de φ , $t_1[X] \models Q_1(X)$ e $t_2[X] \models Q_2[X]$.

Seja Γ uma teoria-pc sobre $E(A)$. Denotamos por \succ_{Γ} o fecho transitivo de $\bigcup_{\varphi \in \Gamma} \succ_{\varphi}$. Γ induz uma ordem de preferência forte sobre $X(A)$. Logo, para todo par de tuplas $(x_1, x_2) \in \succ_{\Gamma}$, dizemos que x_1 é preferido a x_2 de acordo com uma de preferência forte e denotamos $x_1 \succ_{\Gamma} x_2$.

Tomando outro exemplo, ainda utilizando o trecho do documento XML citado acima, onde cada preferência expressa sobre os elementos do documento compõe uma tupla ordenada, de modo que, seria representado da seguinte forma: $A(R, M, D)$ e para $\text{dom}(R) = \{r_1, r_2, r_3, r_4\}$, $\text{dom}(M) = \{m_1, m_2, m_3\}$ e $\text{dom}(D) = \{d_1, d_2\}$. Seja a teoria-pc $\Gamma = \{\varphi_1, \varphi_2\}$ sobre $E(A)$, onde:

- $\varphi_1 \llbracket (R = r_1) \rightarrow (M = m_2) > (M = m_1) \llbracket \{D\} \rrbracket$;
- $\varphi_2 \llbracket (M = m_2) > (M = m_3) \rrbracket$;

De acordo com a ordem de preferência forte induzida por Γ , temos que:

- $\succ_{\varphi_1} = \{((r_1, m_2, d_1), (r_1, m_1, d_1)), ((r_1, m_2, d_2), (r_1, m_1, d_2)), ((r_1, m_2, d_2), (r_1, m_1, d_1)), ((r_1, m_2, d_1), (r_1, m_1, d_2))\}$,
 - $\succ_{\varphi_2} = \{((r_1, m_2, d_1), (r_1, m_3, d_1)), ((r_1, m_2, d_2), (r_1, m_3, d_2))\}$ e,
- $$\succ_{\Gamma} = \{((r_1, m_1, d_1), (r_1, m_2, d_1)), ((r_1, m_1, d_1), (r_1, m_2, d_2)), ((r_1, m_1, d_2), (r_1, m_2, d_2)), ((r_1, m_1, d_2), (r_1, m_3, d_2)), ((r_1, m_2, d_1), (r_1, m_3, d_1)), ((r_1, m_2, d_2), (r_1, m_3, d_2))\} \cup \succ_{\varphi_1} \cup \succ_{\varphi_2}$$

Desta forma, a tupla $x_1 = (r_1, m_1, d_1)$ é preferida a $x_2 = (r_1, m_2, d_1)$, assim sendo, $x_1 \succ_{\Gamma} x_2$.

Com a definição da Ordem de Preferência Forte dois pontos merecem destaque:

1. Cada regra φ individualmente induz uma ordem sobre as tuplas. Nessa ordem, as preferências locais sobre os valores dos atributos das tuplas não são absolutas, ou seja, só é possível inferir que uma tupla é preferida a outra em relação ao atributo consequente de φ , dependendo de como os outros atributos estão instanciados;
2. Para testar se duas tuplas $x = (x_1, x_2, \dots, x_n)$ e $x' = (x'_1, x'_2, \dots, x'_n)$ são comparáveis de acordo com a teoria-pc $\Gamma = \{\varphi_1, \dots, \varphi_n\}$, isto é, $x \succ_{\Gamma} x'$, deve-se verificar se existe um subconjunto de $\Gamma' = \{\delta_1, \dots, \delta_m\}$ para $\Gamma' \subseteq \Gamma$, e tuplas y_1, \dots, y_{m-1} tal que $x \succ_{\delta_1} y_1, y_1 \succ_{\delta_2} y_2, \dots, y_{m-1} \succ_{\delta_m} x'$.

É importante frisar que para cada regra-pc φ , a relação \succ_{φ} mantém a semântica *ceteris paribus* em relação aos atributos que não pertencem à regra φ . Ou seja, tuplas com valores diferentes para cada elemento $Z \in E(A)$, que não estão em φ , são incomparáveis de acordo com \succ_{φ} . Em resumo, a ordem forte de preferência do XQuery-Pref não induz uma preferência absoluta sobre os atributos (daí preferências condicionais) e é inferida a partir do fecho transitivo da união das ordens de preferência de cada regra-pc. Mais detalhes de como a extensão será especificada serão abordados na próxima seção.

C. XQuery-Pref

A linguagem XQuery atua sobre a estrutura lógica e abstrata dos documentos, de modo que o modelo de dados representa os documentos como árvores onde os nós podem corresponder a documentos, elementos, atributos, textos, espaços de nomes, instruções de processamento ou comentários, e cada nó tem um identificador único.

A declaração de funções é uma das mais simples maneiras de estender uma linguagem e a XQuery implanta o conceito de

funções extensíveis que atuam sobre documentos, seções ou valores atômicos. A vantagem na definição de funções na XQuery concentra-se no fato que as funções são implementadas na própria linguagem, não sendo necessário especificar funções numa linguagem de programação à parte.

Diante do exposto, um bloco de consulta na linguagem XQuery-Pref utiliza o domínio das expressões condicionais já existentes na linguagem XQuery aliada a flexibilidade da linguagem para criar funções customizadas que possibilitem a personalização das consultas.

Para melhor compreender o funcionamento da função criada, destaca-se que a partir das preferências expressas por um usuário o sistema XQPref, que será abordado mais profundamente na próxima seção, realizará o tratamento das preferências de modo que terá como entrada e saída:

- Entrada: Um conjunto de amostras de preferências consistentes sobre um conjunto X de tuplas, onde: $A \subseteq X \times X \times \{0,1\}$;
- Saída: Uma ordenação que seja compatível com as amostras de preferências A , com boa acurácia, onde: $O: X \times X \rightarrow \{0,1\}$;

Para cada preferência expressa pelo usuário será atribuída uma variável denominada *relevance* que armazenará a relevância associada a uma expressão, e esta variável fica associada a um valor de tipo *xs:float* no intervalo $[0,1]$, onde um valor maior indica uma maior relevância. É importante ressaltar aqui, que o valor desta relevância será calculado de acordo com a teoria-pc explicado na seção anterior.

A função proposta, *fnpreference*, pode ser usada sempre que uma comparação é possível, e esta inclui um caminho para especificar os nós onde a função será aplicada, e a expressão com a palavra-chave que está sendo procurada no conteúdo dos nós. Onde:

```
declare function fnpreference($i as string) as return String
{Let $i relevance $r in doc ("...") /elemento1/elemento2
 [tema fnpreferences "tema"]
 Order by $r
 Return $i };
```

Por exemplo, em uma consulta para retornar informações relacionadas à evasão sobre a educação da região nordeste ordenado de acordo com as preferências do usuário:

```
For $i in doc ("...") /regiões/nordeste
 Where $i/tema fnpreferences "evasão"
 Return $i;
```

Levando em consideração que o usuário já expressou suas preferências e nelas prevaleceu que para a região nordeste prefere informações sobre o índice de evasão na educação especial a demais modalidades, então esta consulta terá como resultado os índices de evasão desta modalidade no topo dos resultados, tendo em vista que a função ordenará o resultado de acordo com esta preferência.

Por fim, destaca-se o fato que para uma execução transparente de consultas escritas na linguagem XQuery utilizando os recursos da extensão XQuery-Pref requer um processo automático de tradução de consultas. Esta tarefa além de não ser trivial, pois é necessário que o tradutor conheça as estruturas do esquema de representação utilizado para armazenar os documentos XML, pode também não ser eficiente principalmente em consultas recursivas.

Assim, a optou-se pela implementação de um sistema denominado XQPref, que servirá como camada de tradução, que utiliza informações de metadados do documento XML e proporciona a conversão da consulta realizada pelo usuário por uma sentença de consulta XQuery, e realiza a busca efetiva no documento de forma transparente ao usuário.

IV. XQPREF

O sistema XQPref tem como objetivo suprir a necessidade de um mecanismo de consulta personalizada baseada em preferências condicionais em base de dados XML. Para tanto, o sistema permite a execução de consultas XQuery em que funções de preferências estão embutidas na linguagem e o uso destas funções possibilitam a otimização da recuperação de informações nessas bases de dados, reduzindo ainda o esforço do usuário para encontrar conteúdo relevante, de acordo com suas necessidades.

O XQPref proporciona uma interface de usuário para a biblioteca de execução XQuery. As funções propostas para realizar as consultas respeitando as preferências expressas pelo usuário são disponibilizadas através de um namespace na linguagem XQuery e permitem atender a propósitos de busca variados, simplificando o processo de consulta para o usuário. As funcionalidades incluídas na interface de usuário podem ser vistas na Figura 1, a qual apresenta o diagrama de casos de uso da ferramenta.

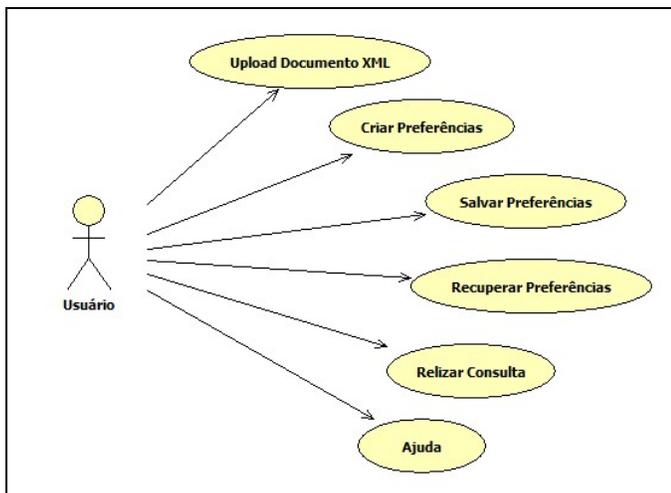


Fig. 1. Diagrama de casos de uso com as funcionalidades do sistema XQPref

A primeira funcionalidade do sistema é o upload do documento XML, esta função visa facilitar o processo de consulta para o usuário, para possibilitar que o sistema realize uma varredura sobre as tags, de modo a mapeá-las de forma

dinâmica para opções que serão oferecidas para o usuário selecionar suas preferências.

Em seguida, a funcionalidade Realizar Consulta permite que usuário busque as informações de seu interesse através da seleção de opções relacionadas ao documento fornecido ao sistema. É importante destacar que nesta funcionalidade não é preciso nenhum esforço do usuário para que o sistema retorne suas preferências, uma vez que a extensão com as funções de relevância já estão acopladas a linguagem. Por fim, a última funcionalidade se refere à ajuda disponível para que o usuário possa tirar alguma dúvida que surja na utilização do sistema.

A Figura 2, é apresentada de forma gráfica a arquitetura do sistema XQPref, onde estão ilustrados os módulos necessários para a implementação do sistema e suas relações.

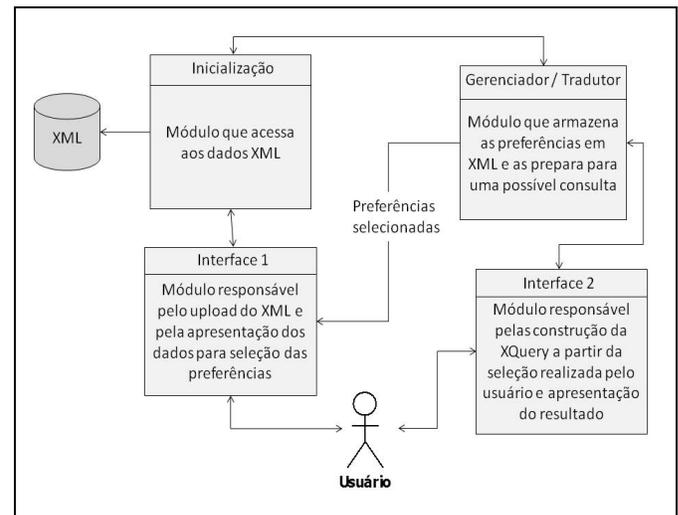


Fig. 2. Arquitetura do sistema XQPref

Inicialmente, percebeu-se a necessidade de criar um repositório onde serão armazenados os documentos XML e que deve ser acessado pelo módulo da inicialização.

O módulo Gerenciador / Tradutor, é a parte do sistema responsável por calcular a relevância das preferências através da teoria-*pc*, com base nos nodos selecionados pelo usuário. E os módulos Interface 1 e 2 se responsabilizam pela interação entre o usuário, e os demais módulos de modo que todo o processo de consulta seja o mais transparente e intuitivo possível para o usuário.

Este estudo de caso tem como proposta realizar consultas personalizadas em dados abertos governamentais utilizando a extensão proposta XQuery-Pref, com o objetivo de verificar o índice de acerto dos resultados recebidos pelos usuários. Para tanto, o mecanismo de consulta utilizado foi o sistema XQPref, onde através dele serão fornecidas as preferências necessárias para que a função Preference realize a ordenação das preferências.

Inicialmente, foi realizada uma demonstração do sistema XQPref utilizando dados XML fictícios ou sintéticos (criados pelo próprio autor), para que viabilizasse uma primeira

impressão do sistema e, assim, fosse possível detectar falhas simples. A Figura 3 abaixo mostra o processo de consulta.

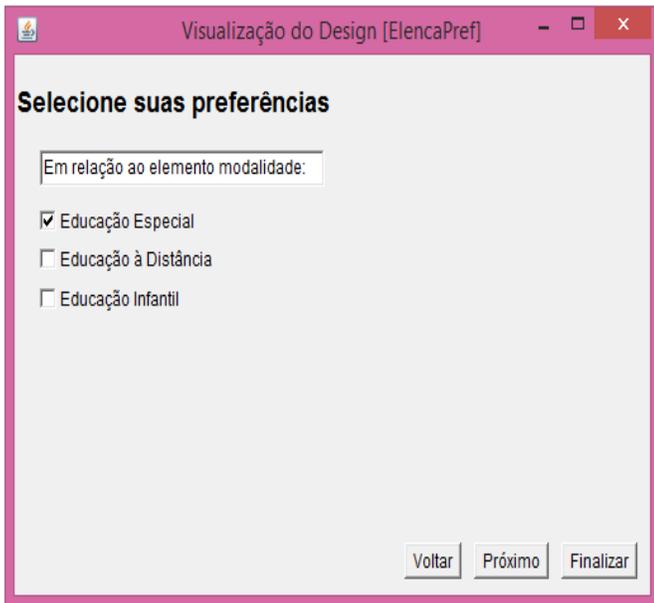


Fig. 3. Tela do XQPref para fornecer preferências

Neste primeiro exemplo o usuário selecionou como sua preferência apenas a modalidade educacao_especial e na hora de fazer a consulta também selecionou educacao_especial e desta forma obteve como resultado todas as informações contidas no documento dentro da tag <educacao_especial>, como ilustra a Figura 4.

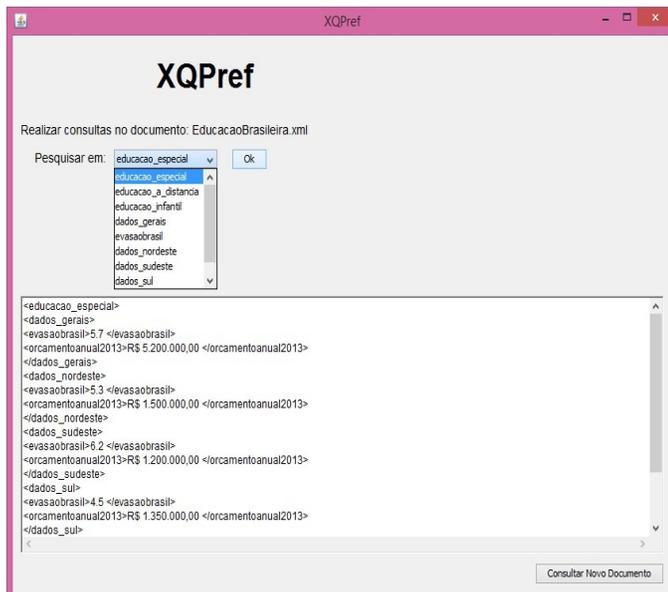


Fig. 4. Tela do XQPref para realizar a consulta e visualizar resultado

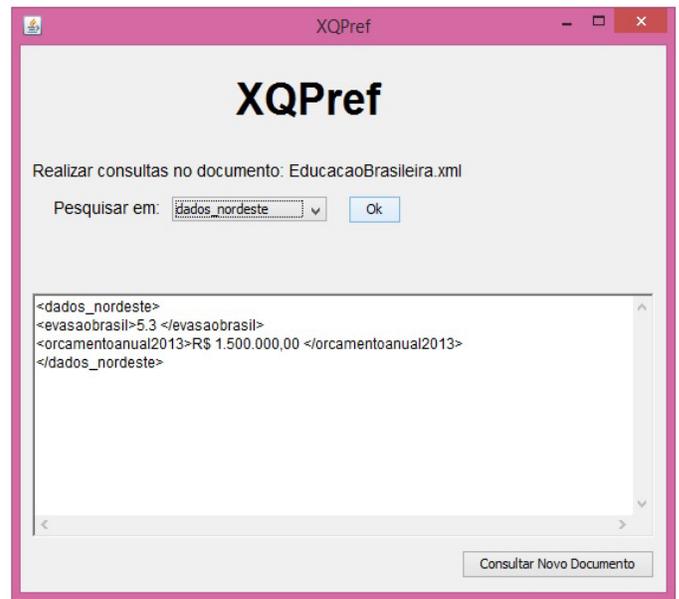


Fig. 5. Tela do XQPref para realizar a consulta e visualizar resultado

Na Figura 5, o mesmo usuário que selecionou como preferência informações sobre educacao_especial, vai seleciona a tag <dados_nordeste> para consulta, com isso recebe os resultados referentes à educacao_especial que pertencem ao nordeste.

A realização deste primeiro estudo de caso com documentos menores facilitou a adequação do sistema para os testes com dados abertos reais. Tendo em vista que, o foco inicial é avaliar se os resultados das consultas estão respeitando as preferências fornecidas pelos usuários.

Desse modo, Com o estudo de caso foi possível verificar que os resultados retornados estão respeitando as preferências expressas pelo usuário. Possibilitando que mesmo usuários que não conheçam os comandos da linguagem XQuery, realizem consultadas de modo otimizado.

Destaca-se que a ferramenta ainda necessita de maior avaliação antes de ser disponibilizada, tendo em vista que o seu foco são para consultas em grandes quantidades de informações. Tal avaliação está em andamento, através da realização de testes automatizados, ficando assegurada como trabalho futuro.

V. CONCLUSÃO

Este trabalho apresentou a XQuery-Pref, uma extensão da linguagem de consulta XQuery, desenvolvida para viabilizar a personalização de consultas, com suportes a preferências condicionais em documentos XML, necessidade identificada a partir da sobrecarga de informações em domínios que utilizam este formato. Para isto, a XQuery-Pref usa um modelo de preferência baseado no conjunto de regras condicionais proposto por Wilson [2004] e do formalismo CP-net [Boutilier et al. 2004], para induzir uma ordem de preferência sobre os elementos de um documento XML.

Para tanto, foi realizado um estudo em torno da personalização de consultas, bem como das técnicas para representação e modelagem das preferências. Através da literatura estudada percebe-se que existem poucas ferramentas e técnicas que suprem essa necessidade de realizar consulta em dados abertos governamentais, e as que existem são para consultas específicas a determinado conjunto de dados. Através da literatura também foi possível concluir que a personalização de consultas através das preferências condicionais é uma solução eficiente e que pode ser usada para suprir tal necessidade.

Outra contribuição apresentada como resultado da pesquisa é o sistema XQPref, que é responsável pela eliciação de preferências dinâmicas e pelo processamento destas consultas personalizadas. Tal proposta tem como objetivo suprir a necessidade de um mecanismo de consulta personalizada com suporte a preferências condicionais em base de dados XML, tornando transparente para o usuário a execução de consultas escritas na linguagem XQuery-Pref.

A importância do XQPref para o presente trabalho destaca-se também pela necessidade de se realizar consultas testes utilizando a extensão proposta através da linguagem XQuery-Pref, para verificar a consistência e acurácia da pesquisa através da avaliação experimental.

Através da avaliação realizada com o estudo de caso, também foi possível registrar que os resultados retornados estão respeitando as preferências expressas pelo usuário e, de fato, possibilita que mesmo usuários que não conheçam os comandos da linguagem XQuery, realizem as consultas personalizadas de modo otimizado e transparente.

Em resumo, a principal contribuição deste trabalho é, portanto, reduzir o esforço do usuário na busca por conteúdo relevante em bases de dados XML, para que qualquer interessado possa, ao processar livremente os dados governamentais, criar conteúdo a partir da reutilização dos dados.

Esta pesquisa é um trabalho em andamento, e a sequência deste trabalho tem como meta aprimorar as funções implementadas na linguagem XQuery-Pref, analisando a consistência e acurácia dos resultados na realização de consultas em dados abertos governamentais, para que qualquer interessado possa, ao processar livremente os dados governamentais, criar conteúdo a partir da reutilização dos

dados. Para tanto, serão realizados testes com dados reais disponíveis no site <http://dados.gov.br/>.

AGRADECIMENTOS

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (Capes) pela bolsa de mestrado concedida para realização de um trabalho maior no qual, este artigo está inserido.

REFERÊNCIAS

- [1] S. A. Amo, T. M. Ferneda, R. G. Cattelan, V. V. S. Dias, H. M. N. Ferreira, "Contextual Preference Repositories for Personalized Query Answering". In: XXVII Simpósio Brasileiro de Banco de Dados, 2012, São Paulo. Anais do XXVII Simpósio Brasileiro de Banco de Dados. São Paulo, 2012. p. 17-24
- [2] F. G. Santos, R. Pinheiro, V. Braganholo, "Processamento de Consultas XML usando Máquinas de Inferência". In: Simpósio Brasileiro de Banco de Dados (SBBD), 2012, São Paulo, SP. Simpósio Brasileiro de Banco de Dados (SBBD). Porto Alegre, RS: SBC, 2012. p. 129-136
- [3] A. Deutsch,; M. Fernandez, D. Florescu, A. Levy, D. Suciú, "XML-QL: A Query Language for XML. Submission to the World Wide Web Consortium. 1998". Disponível em: <<http://w3.org/TR/1998/NOTE-xml-ql-19980819>>. Acesso em Fev.2014.
- [4] J. Robie, "XQL (XML Query Language): W3C Recommendation. 1999". Disponível em: <<http://www.ibiblio.org/xql/xql-proposal.html>>. Acesso em Fev.2014.
- [5] D. Chamberlin, J. Clark, D. Florescu, J. Robie, J. Siméon, M. Stefanescu, "XQuery 1.0: An Query Language for XML: W3C Working Draft". 2001. Disponível em: <<http://www.w3.org/TR/xquery>>. Acesso em: Mar.2014
- [6] N. Wilson, "Extending CP-Nets with Stronger Conditional Preference Statements". In 19th National Conference on Artificial Intelligence (AAAI), 2004, pp.735-741, San Jose, California, USA.
- [7] C. Boutilier, R. I. Brafman, C. Domshlak, H. H. Hoos, D. Poole, "CP-nets: A Tool for Representing and Reasoning with Conditional Ceteris Paribus Preference Statements". Journal of Artificial Intelligence Research 21, 2004.
- [8] W. Yan, L. Yan, and Z. M. Ma. "Automated Ranking of Relaxing Query Results Based on XML Structure and Content Preferences." International Journal of Systems and Service-Oriented Engineering (IJSSOE) 2.1 (2011): 21-39.
- [9] C. H. P. Gomes, "Extensão de uma Linguagem de Consulta para Documentos XML com Características de Tempo e de Versão". 2002. Dissertação - Universidade Federal do Rio Grande do Sul, UFRGS, Brasil. 2002.
- [10] AMO, S. A. ; Fabiola S.F. Pereira . Evaluation of Conditional Preference Queries.. Journal of Information and Data Management - JIDM, v. 1, p. 521-536, 2010.
- [11] Reputação S.A. 2014. Disponível em: <http://reputacao-sa.org/>. Acesso em Jun. 2014.