



Fausto Vêras Maranhão Ayres

**QEEF - Uma Máquina de Execução de Consultas
Extensível**

Tese de Doutorado

Tese apresentada ao programa de Pós-graduação em Informática do Departamento de Informática da PUC-Rio como parte dos requisitos para obtenção do título de Doutor em Informática.

Orientador: Rubens Nascimento Melo
Co-Orientador: Fabio André Machado Porto

Rio de Janeiro
Dezembro de 2003



Fausto Vêras Maranhão Ayres

QEEF - Uma Máquina de Execução de Consultas Extensível

Tese apresentada como requisito parcial para obtenção do título de Doutor pelo Programa de Pós-Graduação em Informática da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

Prof. Rubens Nascimento Melo
Orientador, PUC-Rio

Prof. Fabio André Machado Porto
Co-Orientador, IME-RJ

Prof. Sérgio Lifschitz
PUC-Rio

Prof. Carlos José Pereira de Lucena
PUC-Rio

Prof. Carlos Alberto Heuser
UFRGS

Profa. Ana Maria Moura
IME-RJ

Prof. Álvaro César Pereira Barbosa
UFES

Prof. Ney Dumont
Coordenador(a) Setorial do Centro Técnico Científico - PUC-Rio

Rio de Janeiro, 19 de dezembro de 2003

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Fausto Vêras Maranhão Ayres

Graduou-se em Engenharia Civil na Universidade Santa Úrsula do Rio de Janeiro em 1988. Fez mestrado em Informática pela PUC-Rio em 1994.

Ficha Catalográfica

Ayres, Fausto Vêras Maranhão

QEEF – Uma Máquina de Execução de Consultas Extensível / Fausto Vêras Maranhão Ayres; orientador: Rubens Nascimento Melo; co-orientador: Fabio André Machado Porto. – Rio de Janeiro: PUC, Departamento de Informática, 2003.

125 f. : il. ; 30 cm

Tese (Doutorado) – Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática

Inclui referências bibliográficas.

1. Informática – Teses. 2. Banco de Dados. 3. Processamento de consultas. 4. Máquina de execução de consultas. 5. Modelo de execução de consultas. 6. Modelo de dados semi-estruturados. 7. Framework de software. I. Melo, Rubens Nascimento. II. Porto, Fabio André Machado. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. IV. Título.

CDD:004

Para minha esposa Célia e minhas filhas Carolina e Laura
pela paciência, incentivo e confiança.

Agradecimentos

A Deus pelas realizações obtidas e por me dar forças para superar os momentos difíceis;

Ao meu orientador Prof. Rubens Melo pelo apoio e pelas valiosas contribuições no fechamento deste trabalho;

Ao meu co-orientador Prof. Fabio Porto pelas inúmeras horas de discussões que foram fundamentais para a realização deste trabalho;

Aos professores do DI pelo conhecimento recebido;

Aos meus amigos do DI, em especial aos do TecBD, pelos bons momentos de estudo e descontração;

Aos meus familiares, pelo incentivo constante;

Aos funcionários do DI, em especial a Ruth, Débora, “Manu” e Carmen;

Ao CEFET-RJ e a PUC-Rio pelo apoio financeiro, pois sem ele seria impossível realizar este trabalho de tese.

Resumo

Ayres, Fausto Vêras Maranhão. **QEEF – Uma Máquina de Execução de Consultas Extensível**. PUC-RIO, 2003. 125p. Tese de Doutorado - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

O processamento de consultas em Sistemas de Gerência de Banco de Dados tradicionais tem sido largamente estudado na literatura e utilizado comercialmente com enorme sucesso. Isso é devido, em parte, à eficiência das Máquinas de Execução de Consultas (MEC) no suporte ao modelo de execução tradicional. Porém, o surgimento de novos cenários de aplicação, principalmente em consequência do modelo computacional da *web*, motivou a pesquisa de novos modelos de execução, tais como: modelo adaptável e modelo contínuo, além da pesquisa de modelos de dados semi-estruturados, tal como o XML, ambos não suportados pelas MEC tradicionais. O objetivo desta tese consiste no desenvolvimento de uma MEC extensível frente a diferentes modelos de execução e de dados. Adicionalmente, esta proposta trata de maneira ortogonal o modelo de execução e o modelo de dados, o que permite a avaliação de planos de execução de consultas (PEC) com fragmentos em diferentes modelos. Utilizou-se a técnica de *framework* de *software* para a especificação da MEC extensível, produzindo o *framework* QEEF (*Query Execution Engine Framework*). A extensibilidade da solução reflete-se em um meta-modelo, denominado QUEM (QUery Execution Meta-model), capaz de exprimir diferentes modelos em um meta-PEC. O *framework* QEEF pré-processa um meta-PEC e produz um PEC final a ser avaliado pela MEC instanciada. Como parte da validação desta proposta, instanciou-se o QEEF para diferentes modelos de execução e de dados.

Palavras-chave

Banco de Dados; Processamento de Consultas; Máquina de Execução de Consultas; Modelo de Execução de Consultas; Modelo de Dados Semi-Estruturados; Framework de Software.

Abstract

Ayres, Fausto Vêras Maranhão. **QEEF – An Extensible Query Execution Engine**. PUC-RIO, 2003. 125p. PhD. Thesis. Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Querying processing in traditional Database Management Systems (DBMS) has been extensively studied in the literature and adopted in industry. Such success is, in part, due to the performance of their Query Execution Engines (QEE) for supporting the traditional query execution model. The advent of new query scenarios, mainly due to the *web* computational model, has motivate the research on new execution models such as: adaptive and continuous, and on semi-structured data models, such as XML, both not natively supported by traditional query engines. This thesis proposes the development of an extensible QEE adapted to the new execution and data models. Achieving this goal, we use a software design approach based on framework technique to produce the Query Execution Engine Framework (QEEF). Moreover, we address the question of the orthogonality between execution and data models, witch allows for executing query execution plans (QEP) with fragments in different models. The extensibility of our solution is specified by in a QEP by an execution meta-model named QUEM (QUery Execution Meta-model) used to express different models in a meta-QEP. During query evaluation, the latter is pre-processed by the QEEF producing a final QEP to be evaluated by the running QEE. The QEEF is instantiated for different execution and data models as part of the validation of this proposal.

Keywords

Database; Query Processing; Query Execution Engine; Query Execution Model; Semi-Structured Data Model; Software Framework.

Sumário

1 Introdução	14
1.1. Motivação	14
1.2. Objetivo da Tese	17
1.3. Visão Geral da Solução Proposta	18
1.4. Contribuições da Tese	24
1.5. Foco da Tese	25
1.6. Contexto da Tese	25
1.7. Organização da Tese	26
1.8. Síntese do Capítulo	27
2 Cenários de Execução de Consultas	28
2.1. Processamento de Consultas em SGBDs	28
2.2. Máquinas de Execução de Consultas (MEC)	44
2.3. Cenários de Execução de Consultas	51
2.4. Síntese do Capítulo	56
3 A Solução Proposta	58
3.1. Suporte a Diferentes Modelos de Execução de Consultas	58
3.2. Suporte a Diferentes Modelos de Dados	77
3.3. Ortogonalidade entre Modelos de Execução e de Dados	78
3.4. Síntese do Capítulo	78
4 A Máquina Extensível	79
4.1. Introdução a <i>Frameworks de Software</i>	79
4.2. O <i>Framework</i> QEEF	82
4.3. Síntese do Capítulo	89
5 Meta-Modelo de Execução de Consultas	90
5.1. Meta-Modelo QUEM	90
5.2. Especificação de meta-PEC em XML	92

5.3. Síntese do Capítulo	95
6 Estudo de Casos	96
6.1. RQEE	96
6.2. AQEE	101
6.3. XQEE	108
6.4. Síntese do Capítulo	115
7 Conclusões e Trabalhos Futuros	116
8 Referências Bibliográficas	120
Anexo A: Diagramas	124

Lista de figuras

Figura 1 - Processamento de Consultas em SGBDs	14
Figura 2 - SGBD Distribuído	30
Figura 3 - Arquitetura Mediador-Adaptador	38
Figura 4 - Processamento de Consultas no Mediador-Adaptador	40
Figura 5 - Exemplo de operadores algébricos e de controle	45
Figura 6 - Exemplo de um PEC	49
Figura 7 - Topologia de PEC: linear à esquerda, ramificada e linear à direita	49
Figura 8 - PEC com paralelismo no SGBD Oracle	52
Figura 9 - Arquitetura do sistema <i>Le Select</i>	55
Figura 10 - O servidor do <i>Le Select</i>	56
Figura 11 - Módulos de Sincronismo <i>Wait</i> (a), <i>Nowait</i> (b) e <i>Waitall</i> (c)	62
Figura 12 - Combinando Módulos de Sincronismo	63
Figura 13 - Módulo de Paralelismo <i>Inter-operator</i>	64
Figura 14 - Módulo de Paralelismo <i>Intra-operator</i>	65
Figura 15 - Módulo de Distribuição sobre PEC linear (a) e PEC <i>bushy</i> (b)	65
Figura 16 - Módulos de Fluxo de Controle <i>Demand-driven</i> (a) e <i>Data-driven</i> (b)	66
Figura 17 - Operadores de controle <i>passive</i> (a) e <i>active</i> (b)	66
Figura 18 - O problema de <i>back pressure</i>	67
Figura 19 - Módulos de Fluxo de Dados <i>Fixed</i> (a) e <i>Adaptive</i> (b)	67
Figura 20 - Módulo de Tempo de Resposta <i>First-tuple</i> (a) e <i>Last-tuple</i> (b)	69
Figura 21 - Modelo Tradicional: Exemplo 1	70
Figura 22 - Execução do Exemplo 1	71
Figura 23 - Modelo Tradicional: Exemplo 2	71
Figura 24 - Execução do Exemplo 2	71
Figura 25 - Modelo Tradicional: Exemplo 3	72
Figura 26 - Execução do Exemplo 3	72
Figura 27 - Modelo Contínuo: Exemplo 4	72
Figura 28 - Execução do Exemplo 4	73
Figura 29 - Modelo Adaptável: Exemplo 5	73

Figura 30 - Execução do Exemplo 5	74
Figura 31 - Modelo Adaptável: Exemplo 6	74
Figura 32 - Execução do Exemplo 6	74
Figura 33 - Modelo baseado em Streams: Exemplo 7	75
Figura 34 - Execução do Exemplo 7	75
Figura 35 - Novo Modelo: Exemplo 8	76
Figura 36 - Execução do Exemplo 8	76
Figura 37 - O Framework QEEF	84
Figura 38 - Relacionamentos entre Módulos, Controles e Estruturas	85
Figura 39 - Execução de Consultas no QEEF	91
Figura 40 - Os dois níveis de execução do QEEF	92
Figura 41 - Elemento metaplano (raiz)	94
Figura 42 - Detalhamento do elemento MODULO	94
Figura 43 - Detalhamento do modulo Adaptive e do operador Algebrico	94
Figura 44 - Modelo de execução do PEC	97
Figura 45 - Ambiente de execução do RQEE	98
Figura 46 - Dados e resultados da consulta SQL	99
Figura 47 - Conteúdo da Metabase	99
Figura 48 - Instanciação do QEEF para o RQEE	100
Figura 49 - Modelo de execução da consulta de integração	103
Figura 50 - Execução do PEC adaptável	103
Figura 51 - Especificação do meta-PEC	105
Figura 52 - A metabase da consulta de integração	106
Figura 53 - A interface do usuário da aplicação	106
Figura 54 - Instanciação do QEEF para o AQEE	107
Figura 55 - Amarrações das variáveis de ligação	113
Figura 56 - Tuplas XML produzidas a partir das amarrações	113
Figura 57 - As classes instanciadas do QEEF para o XQEE	114

Lista de quadros

Quadro 1 - Exemplo de Dados XML	41
Quadro 2 - A DTD para o documento de livros	41
Quadro 3 - Implementação da interface iterador para o NestedLoop	48
Quadro 4 - Implementação dos métodos getNext, putnext e hasNext	86
Quadro 5 - Exemplo de uso de uma MEC	87
Quadro 6 - Instanciação do QEEF	89
Quadro 7 - DTD do meta-modelo QUEM (quem.dtd)	93
Quadro 8 - Uma consulta SQL sobre dados demográficos	96
Quadro 9 - Especificação do meta-PEC da consulta 1	98
Quadro 10 - A classe RQEE	101
Quadro 11 - Uma consulta SQL sobre preços de produtos	101
Quadro 12 - A classe AQEE	108
Quadro 13 - Documento bib.xml	109
Quadro 14 - Consulta XQuery sobre dados bibliográficos	109
Quadro 15 - Resultado da consulta XQuery	110
Quadro 16 - PEC da consulta XQuery	111
Quadro 17 - A classe XQEE	115

Lista de tabelas

Tabela 1 - Características de Execução de Consultas e seus Estados	19
Tabela 2 - Conceitos de execução de consultas.	20
Tabela 3 - Níveis de composição dos conceitos de execução de consultas	58
Tabela 4 - Classificação dos Modelos de Execução de Consultas	70
Tabela 5 - MECs produzidas pelo QEEF	96