

**UNIVERSIDADE FEDERAL DA PARAÍBA**  
**CENTRO DE CIÊNCIAS E TECNOLOGIA**  
**DEPARTAMENTO DE SISTEMAS E COMPUTAÇÃO**  
**COORDENAÇÃO DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

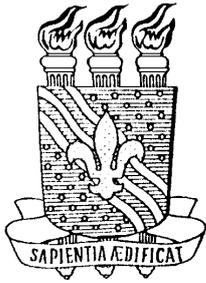
**DISSERTAÇÃO DE MESTRADO**

"BDRV - Um Sistema Integrador de Múltiplos  
Bancos de Dados Relacionais"

Carlos Eduardo Santos Pires

Campina Grande – PB

Agosto de 2000



**UNIVERSIDADE FEDERAL DA PARAÍBA**

**CENTRO DE CIÊNCIAS E TECNOLOGIA - CCT**

**DEPARTAMENTO DE SISTEMAS E COMPUTAÇÃO - DSC**

**COORDENAÇÃO DE PÓS-GRADUAÇÃO EM INFORMÁTICA - COPIN**

**MESTRADO EM INFORMÁTICA**

**DISSERTAÇÃO DE MESTRADO**

**"BDRV - UM SISTEMA INTEGRADOR DE MÚLTIPLOS BANCOS DE  
DADOS RELACIONAIS"**

**Carlos Eduardo Santos Pires**

**(MESTRANDO)**

**Marcus Costa Sampaio**

**(ORIENTADOR)**

**Campina Grande – PB**

**Agosto de 2000**

CARLOS EDUARDO SANTOS PIRES

"BDRV - UM SISTEMA INTEGRADOR DE MÚLTIPLOS  
BANCOS DE DADOS RELACIONAIS"

DISSERTAÇÃO apresentada à COORDENAÇÃO  
DE PÓS-GRADUAÇÃO EM INFORMÁTICA do  
DEPARTAMENTO DE SISTEMAS E COMPUTAÇÃO  
do CENTRO DE CIÊNCIAS E TECNOLOGIA da  
UNIVERSIDADE FEDERAL DA PARAÍBA como  
requisito parcial para obtenção do grau de  
MESTRE EM INFORMÁTICA.

Área de Concentração : CIÊNCIA DA COMPUTAÇÃO

LINHA DE PESQUISA: SISTEMAS DE INFORMAÇÃO E BANCOS DE DADOS

Orientador : PROF. DR. MARCUS COSTA SAMPAIO

CAMPINA GRANDE

Agosto 2000

## Ficha Catalográfica

---

PIRES, Carlos Eduardo Santos

R672G

BDRV - Um Sistema Integrador de Múltiplos Bancos de Dados  
Relacionais

Dissertação (Mestrado) – UFPB/CCT/COPIN, Campina Grande,  
Agosto de 2000.

151 p. Il.

Orientador: Marcus Costa Sampaio

1. Bancos de Dados
2. Multidatabases
3. Mediadores
4. Integração de Esquemas Relacionais
5. Bancos de Dados Virtuais

CDU: 681.3.07B

---

**“BDRV – UM SISTEMA INTEGRADOR DE MÚLTIPLOS BANCOS DE  
DADOS RELACIONAIS”**

**CARLOS EDUARDO SANTOS PIRES**

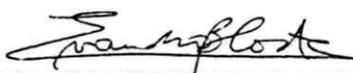
**DISSERTAÇÃO APROVADA EM 30.08.2000**



**PROF. MARCUS COSTA SAMPAIO, Dr.  
Orientador**



**PROF. ULRICH SCHIEL, Dr.  
Examinador**



**PROF. EVANDRO DE BARROS COSTA, D.Sc  
Examinador**

**CAMPINA GRANDE – PB**

# Agradecimentos

A Deus, por todos os prazeres oferecidos, especialmente o de viver.

Ao Prof. Marcus Costa Sampaio pelo apoio e amizade desde meus tempos de graduação.

Às maravilhosas Estelita N. França e Alaíde J. da Silva, pela forma como me acolheram em seu lar durante minha estadia nesta cidade. Vocês são únicas!

Aos meus pais pelo total apoio, principalmente durante estes dois últimos anos em que permaneci em Campina Grande.

A minha querida namorada e futura noiva Karla Leal E. de Amorim

Às fantásticas Ana Lúcia e Vera, pelo carinho e paciência com que atendem aos alunos da Copin.

Aos grandes amigos do LSI pela sadia convivência.

A todos os colegas, professores e funcionários do DSC pela amizade e companheirismo.

À CAPES, pela bolsa de dois anos.

A todos que de alguma forma contribuíram para realização deste trabalho,

OBRIGADO !

# Resumo

PIRES, Carlos Eduardo Santos, "BDRV - Um Sistema Integrador de Múltiplos Bancos de Dados Relacionais", Dissertação de Mestrado, Departamento de Sistemas e Computação, Universidade Federal da Paraíba, Agosto 2000.

A informação é um dos mais importantes patrimônios de uma organização. Este patrimônio é quase sempre armazenado de forma descentralizada, constituindo-se em diversas fontes de dados. Uma fonte de dados opera com dados detalhados e atualizados. Seu objetivo principal é apoiar as operações da organização no dia-a-dia, ou sua gerência no nível operacional.

Ao longo da evolução dos Sistemas de Informação, mudanças significativas aconteceram no que diz respeito às necessidades de informação, especialmente das gerências de nível mais alto de uma organização (gerências tática e estratégica). As informações armazenadas em uma única fonte de dados passaram a ser insuficientes. Surgiu então a necessidade de obtenção de informações armazenadas em mais de uma fonte de dados, ou seja, a disponibilização de uma visão integrada corporativa, formada a partir da união de várias fontes de dados.

Neste trabalho, discutimos a problemática de integração de múltiplas fontes de dados, incluindo os diversos tipos de conflito de informação entre fontes de dados e sua resolução, e apresentamos os requisitos, projeto e implementação de um sistema integrador de múltiplas fontes de dados, exclusivamente bancos de dados relacionais.

# Abstract

PIRES, Carlos Eduardo Santos, "BDRV - Um Sistema Integrador de Múltiplos Bancos de Dados Relacionais", Dissertação de Mestrado, Departamento de Sistemas e Computação, Universidade Federal da Paraíba, Agosto 2000.

The most important asset of an Organization is its information. This asset is nearly always stored in the form of operational systems, or production systems (data sources, by short). Broadly speaking, a data source works with updated and detailed data. Its main objective is to support Organization operations in the everyday life. Unfortunately, these systems do not satisfy requirements of higher information levels, namely tactical and strategic levels.

Along the evolution of Information Systems, significant changes have happened with respect to the information needs for the tactical and strategic levels. Information in this case should be integrated (that is, presented as a corporate data view).

In this work, we discuss the problematic of multiple data sources integration, including several information conflict types and their resolution, and we present the requirements, design and implementation of a system integrating multiple data sources, exclusively relational databases.

# Sumário

<b>Agradecimentos</b>	
<b>Resumo</b>	?
<b>Abstract</b>	?
<b>Lista de Figuras</b>	?
<b>Lista de Tabelas</b>	?
<b>Capítulo 1</b>	
<b>Introdução</b>	1
1.1 Contextualização	1
1.2 Objetivo	5
1.3 Organização	6
<b>Capítulo 2</b>	
<b>Abordagens e Tecnologias para Integração de Informações de Múltiplas Fontes de Dados</b>	7
2.1 Introdução	7
2.2 Sistemas de Bancos de Dados Distribuídos	10
2.2.1 Processamento de Consultas	11
2.2.2 Críticas dos Sistemas de Bancos de Dados Distribuídos	11
2.3 Sistemas de Bancos de Dados Múltiplos	12
2.3.1 Processamento de Consultas	14
2.3.2 Aplicações	14
"UniSQL/M Multidatabase System"	14
"Multibase"	15
2.3.3 Críticas dos Sistemas de Bancos de Dados Múltiplos	16
2.4 "Data Warehousing"	17
2.4.1 "Executive Information Systems"	18
2.4.2 Modelagem Multidimensional	18
2.4.3 "On-line Analytical Processing"	20
2.4.4 Arquitetura	23
2.4.5 Críticas da Tecnologia "Data Warehousing"	27
2.5 Conclusões	28
<b>Capítulo 3</b>	
<b>Requisitos para o Sistema Integrador de Bancos de Dados Relacionais BDRV</b>	29
3.1 Introdução	29
3.2 Requisitos para o Sistema BDRV	30
3.2.1 Arquitetura	30
3.2.2 Integração	34

3.2.3	Segurança	42
3.2.4	Fontes de Dados	43
3.2.5	Interface	44
3.2.6	Configuração	45
3.2.7	Plataformas	45
<b>Capítulo 4</b>		
<b>A Arquitetura BDRV e uma Visão Geral de seu Funcionamento</b>		55
4.1	Objetivo	55
4.2	Funcionamento	58
4.3	Diagramas de Seqüência de Interação Entre os Módulos	65
4.3.1	Exemplo 1	68
4.3.2	Exemplo 2	69
4.3.3	Exemplo 3	69
<b>Capítulo 5</b>		
<b>Projeto e Implementação da Arquitetura BDRV</b>		71
5.1	Arquitetura de Cadastramento de Relatórios	71
5.1.1	Banco de Dados Virtual	74
5.1.2	Software de Definição de Relatórios	74
5.1.2.1	Estrutura de um Relatório	76
5.1.2.2	Descrição de uma Consulta	76
5.1.3	Linguagem Visual de Definição de Relatórios	74
5.1.3.1	Poder de Expressão da LVDC	76
5.1.3.2	Criação de Relatório	76
5.1.3.3	Elaboração de Consultas	76
5.1.3.3.1	Seleção de Relatório	76
5.1.3.3.2	Seleção de Esquema Global	76
5.1.3.3.3	Seleção de Tabela(s) Virtual(is)	76
5.1.3.3.4	Seleção de Atributo(s) Virtual(is)	76
5.1.3.3.5	Definição de Filtro(s)	76
5.1.3.3.6	Definição de Ordenação(ões)	76
5.1.3.3.7	Definição de Agrupamento(s)	76
5.1.3.3.8	Descrição de Consulta	76
5.1.4	Integrador	74
5.1.4.1	Analisador de Consulta	76
5.1.4.2	Coletor de Fontes	76
5.1.4.3	Estimador de Fontes	76
5.1.4.4	Seletor de Fontes	76
5.1.4.5	Transformador de Consulta	76
5.1.5	Metabanco de Dados	74
5.1.5.1	Armazenamento de Esquemas Globais e Locais	76
5.1.5.2	Armazenamento de Mapeamentos Esquemas Globais → Esquemas Locais	76
5.1.5.3	Armazenamento de Informações para Solução de Conflitos de Esquema	76
5.1.5.4	Armazenamento de Relatórios, Consultas e Subconsultas	76
5.2	Arquitetura de Consulta de Relatórios	71
5.2.1	Metabanco de Dados	74
5.2.1.1	Armazenamento de Informações para Solução de Conflitos de Esquema e Dados	76

5.2.2 Combinador de Resultados	74
<b>Capítulo 6</b>	
<b>Conclusões e Trabalhos Futuros</b>	127
7.1 Conclusões	127
7.2 Trabalhos Futuros	130
<b>Anexo A</b>	
<b>Estudo de Caso</b>	108
A.1 Descrição da Aplicação	108
A.3 Criando Consulta	117
A.4 Transformação da Consulta	122
A.5 Execução e Visualização da Consulta	122
<b>Anexo B</b>	
<b>Estudo de Caso</b>	108
A.1 Descrição da Aplicação	108
A.3 Criando Consulta	117
A.4 Transformação da Consulta	122
A.5 Execução e Visualização da Consulta	122
<b>Bibliografia</b>	132

# Lista de Figuras

Figura 1.1 - Os Três Níveis Organizacionais [33]	02
Figura 2.1 – Arquitetura Clássica, em Três Camadas, de um Sistema Baseado na Abordagem de Mediadores	09
Figura 2.2 - Um Sistema de Banco de Dados Distribuído	11
Figura 2.3 - Arquitetura de Integração de Esquemas Conceituais dos Bancos de Dados Locais	13
Figura 2.4 - Arquitetura de Integração de Esquemas de "UniSQL/M"	15
Figura 2.5 – Cubo Multidimensional com Três Dimensões [32]	19
Figura 2.6 - Arquitetura Funcional Básica de um Ambiente de DWing [29]	24
Figura 3.1 - Arquitetura BDRV em Camadas	31
Figura 4.1 (a) – Arquitetura BDRV Genérica	58
Figura 4.1 (b) – Arquitetura BDRV de Cadastramento de Relatórios	58
Figura 4.1 (c) – Arquitetura BDRV de Consulta de Relatórios	58
Figura 4.2 (a) – Esquema "Companhia"	60
Figura 4.2 (b) – Esquema "Empresa"	60
Figura 4.3 (a) - Processo de Transformação de uma Consulta em Subconsulta(s) e Armazenamento de Subconsultas	62
Figura 4.3 (b) - Processo de Consulta de Relatório	64
Figura 5.1 - Visão de um Relatório por um UF	66
Figura 5.2 - Cadastro de Relatórios	69
Figura 5.3 - Seleção de Relatório	69
Figura 5.4 - Seleção de Esquema Global	74
Figura 5.5 - Seleção de Tabela(s) Virtual(is)	78
Figura 5.6 - Seleção de Atributo(s) Virtual(is)	78
Figura 5.7 - Definição de Filtro(s)	80
Figura 5.8 - Definição de Ordenações	81
Figura 5.9 - Definição de Agrupamento(s)	82
Figura 5.10 - Descrição de Consulta	83
Figura 6.1 - Cadastro do "Relatório dos Alunos"	83
Figura 6.2 - Etapa 1 de Geração de Consulta: Seleção do "Relatório dos Alunos"	85

Figura 6.3 - Etapa 2 de Geração de Consulta: Seleção do Esquema Global "COPIN"	86
Figura 6.4 - Etapa 3 de Geração de Consulta: Seleção da Tabela Virtual "Aluno"	89
Figura 6.5 - Etapa 4 de Geração de Consulta: Seleção dos Atributos Virtuais "Aluno.Nm_Aluno", "Aluno.Orientador" e "Aluno.Nm_Email"	95
Figura 6.6 - Etapa 5 de Geração de Consulta: Definição do Filtro "Aluno.Pr_Entrada IGUAL 981"	104
Figura 6.7 - Etapa 6 de Geração de Consulta: Definição da Ordenação "Aluno.Nm_Aluno"	106
Figura 6.8 - Etapa 8 de Geração de Consulta: Descrição da Consulta	???
Figura 6.9 - Seleção de Categoria	???
Figura 6.10 - Relatórios por Título	???
Figura 6.11 - Relatórios iniciados pela letra "R"	???
Figura 6.12 - Consultas pertencentes ao "Relatório dos Alunos"	???
Figura 6.13 - Resultado Final da Consulta	???

# Lista de Tabelas

Tabela 3.1 – Tabela TB1	36
Tabela 3.2 – Tabela TB2	36
Tabela 3.3 – Tabela TB3	36
Tabela 3.4 – Tabela TB4	36
Tabela 3.5 – Tabela TB3	37
Tabela 3.6 – Tabela TB2	37
Tabela 3.7 – Conflitos de Nome de Tabela Detectados nos Esquemas EQ1 e EQ2	38
Tabela 3.8 – Conflitos de Nome de Atributo Detectados nos Esquemas EQ1 e EQ2	38
Tabela 4.1 –	
Tabela 4.2 –	
Tabela 4.3 –	
Tabela 4.4 –	
Tabela 4.5 –	
Tabela 4.6 –	
Tabela 4.7 –	
Tabela 4.8 –	
Tabela 4.9 –	
Tabela 4.10 -	

# Capítulo 1

## *Introdução*

### 1.1 Contextualização

As organizações se caracterizam por uma diferenciação de poder, isto é, uma hierarquização de autoridade [33], defrontando-se com uma multiplicidade de problemas de informação a resolver. Tais problemas podem ser classificados e categorizados segundo os diferentes níveis hierárquicos da organização. Historicamente, os níveis de informação são assim classificados [33] (Figura 1.1):

- **Estratégico ou institucional:** é o nível organizacional mais elevado, responsável pela definição dos principais objetivos e das estratégias da organização. Relaciona-se com o ambiente externo. É constituído por diretores que tomam as decisões fundamentais da organização;
- **Tático ou gerencial:** é o nível intermediário, que cuida do relacionamento e da integração dos níveis estratégico e operacional (ver a seguir). Uma vez tomadas as decisões no nível estratégico, o nível tático é o responsável pela sua transformação em planos e programas para que o nível operacional os execute. É constituído por gerentes e chefes;
- **Operacional ou técnico:** é o nível mais baixo da organização, onde as tarefas são executadas, os programas são desenvolvidos, as técnicas são aplicadas. É constituído por supervisores e executores.

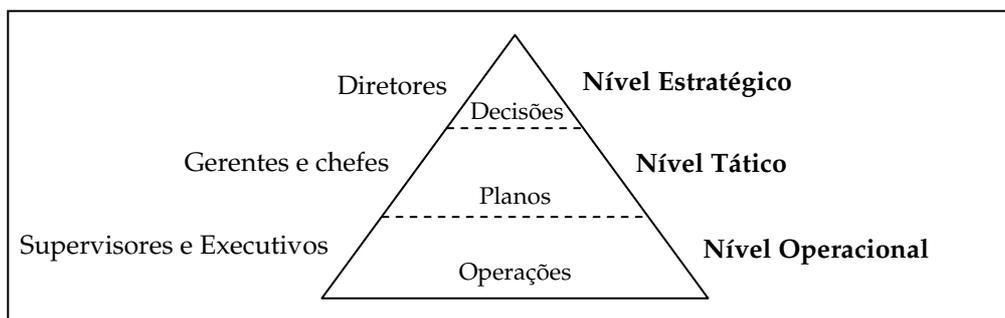


Figura 1.1 - Os Três Níveis Organizacionais [33]

Um fato bastante comum de ser verificado nas mais diferentes organizações é a forma descentralizada de armazenamento das informações, isto é, mantidas em diferentes bancos de dados. É normal encontrarmos cada departamento de uma determinada organização utilizando seu próprio banco de dados, mantendo informações referentes ao seu respectivo departamento. Estes bancos de dados são *alimentados* pelos usuários operacionais de cada departamento, através de chamados programas "front-end", cuja finalidade é atualizar constantemente um banco de dados. Os usuários operacionais movem as engrenagens de uma organização, caracterizando-se por executar as mesmas tarefas inúmeras vezes [18]. Algumas de suas funções são: preencher pedidos, abrir e fechar o caixa, conquistar novos clientes, registrar reclamações, inserir novos dados e corrigir dados antigos. Normalmente, as informações armazenadas em tais bancos de dados são suficientes para atender as necessidades de informação dos usuários locais ao banco de dados em questão.

Dois fatores principais identificados que acarretaram esta descentralização de informações foram [27]:

- A ausência de um administrador de banco de dados único fez com que grupos individuais dentro das organizações criassem e mantivessem bancos de dados de forma separada, no intuito de satisfazer suas próprias necessidades;
- A distribuição de dados em vários bancos de dados é essencial em alguns ambientes para gerenciar grandes volumes de dados.

Ao longo da evolução dos Sistemas de Informação (SIs), mudanças significativas aconteceram no que diz respeito às necessidades de informação por parte

de determinados tipos de usuário. As informações armazenadas em um único banco de dados passaram a ser insuficientes para satisfazer as necessidades de certos usuários. Alguns deles passaram a ter necessidade de consultar informações armazenadas em mais de um banco de dados. Surgiu, então, um novo tipo de usuário, o qual chamaremos de usuário global<sup>1</sup>, com a necessidade de *enxergar* uma visão integrada corporativa, formada a partir da união de vários bancos de dados.

A crescente necessidade por informações compartilhadas intra e interorganizacional levou ao desenvolvimento de inúmeras pesquisas no campo da integração de múltiplos bancos de dados.

No que diz respeito ao problema de integração de múltiplos bancos de dados, academia e indústria seguiram caminhos divergentes durante certo tempo. O problema da integração foi atacado inicialmente pela academia através da **abordagem sob demanda** [4, 13]. Nesta abordagem, também conhecida como abordagem de mediadores, a informação é extraída e integrada das diversas fontes<sup>2</sup> somente quando a consulta é submetida.

A alternativa pesquisada na indústria é a **abordagem antecipada**. Nela a informação é extraída das fontes, integrada, filtrada e armazenada num repositório central, especialmente projetado para as necessidades de análise (nível estratégico). Quando uma consulta é submetida, é executada diretamente sobre este repositório, sem necessidade de acesso a outras bases. Este repositório central é conhecido como "Data Warehouse" [29].

A idéia de integração de múltiplos bancos de dados, baseada na abordagem sob demanda, surgiu no contexto dos Sistemas de Bancos de Dados Distribuídos (SBDDs), onde um único Sistema Gerenciador de Bancos de Dados (SGBD) controla um ou mais bancos de dados, armazenados ao longo de servidores<sup>3</sup> de uma rede de computadores [6, 9]. Um SBDD gerencia transações globais as quais, diferentemente de transações

---

<sup>1</sup> Geralmente, são usuários a nível estratégico, todavia alguns usuários a nível tático podem apresentar as mesmas necessidades de informação.

<sup>2</sup> Consideraremos uma fonte dados um Sistema Gerenciador de Bancos de Dados, um Sistema de Arquivos ou até mesmo um documento "Web".

<sup>3</sup> Os computadores de um SBDD recebem diversos nomes, como "sites" ou nós, dependendo do contexto em que estão inseridos.

locais, acessam vários bancos de dados simultaneamente. Segundo [21], alguns dos principais problemas enfrentados pelos SBDDs são: necessidade de combinar e mover dados armazenados em diferentes computadores, no intuito de responder a uma consulta; necessidade de lidar com transações globais no intuito de preservar a consistência dos dados. Resumindo a crítica aos SBDDs, eles se mostraram incapazes de bem gerenciar transações globais.

Para contornar o problema das transações globais, surgiram os Sistemas de Bancos de Dados Múltiplos<sup>4</sup> (SBDMs). Neste ambiente, não somente os bancos de dados, como também os SGBDs são distribuídos, e possivelmente heterogêneos. Um SBDM proporciona um ambiente em que aplicações de bancos de dados podem manter acesso a dados de diversos bancos preexistentes, localizados em vários ambientes com "software" e "hardware" heterogêneos. Os sistemas de bancos de dados locais podem empregar modelos lógicos diferentes e linguagens de definição e manipulação de dados distintas, podem ainda diferir em relação aos mecanismos para controle de concorrência e gerenciamento de transações. Um SBDM cria a ilusão da integração lógica do banco de dados, sem impor uma integração física. Um SBDM é chamado de homogêneo se os SGBDs *rodando* nos diferentes "sites" são os mesmos, e heterogêneo caso contrário [23].

A maioria das tecnologias atuais de integração de múltiplos bancos de dados está fazendo uso de uma outra tecnologia que cresce em ritmo espantoso - a "World Wide Web", ou "Web", ou ainda WWW [3, 7, 8]. A "Web" é um recurso de informação globalmente distribuído residindo sob a rede mundial de computadores, a "Internet" [19].

Uma importante característica da "Web" é que ela é a utilização de três padrões abertos que permitem:

1. Transferir a informação: para o transporte de informação foi proposto um protocolo de comunicação denominado "Hypertext Transfer Protocol" (HTTP) [25].

---

<sup>4</sup> Do inglês, "Multidatabase".

2. Descrever a formatação da informação: para apresentação de documentos na "Web" é utilizada a linguagem padrão "Hypertext Markup Language" (HTML) [1].
3. Localizar a informação: para identificação e localização de documentos "Web" distribuídos pela "Internet" foi proposta a utilização de um formato para endereçamento de documentos "Web" denominado "Uniform Resource Locator" (URL) [36].

Os problemas de integração de múltiplos bancos de dados que ainda subsistem podem ser sintetizados assim:

- O desempenho dos SBDDs ainda deixa muito a desejar;
- SBDMs e "Web" têm sido tratados até o presente momento de maneira ortogonal, sem nenhuma razão para ser desta forma;
- As abordagens sob demanda e antecipada também têm sido tratadas de forma ortogonal, sem nenhuma razão de ser.

A presente dissertação é sobre integração de múltiplos bancos de dados. Seus objetivos são temas da próxima seção.

## 1.2 Objetivo da Dissertação

O objetivo do nosso trabalho é propor uma solução eficiente para integração de múltiplas fontes de dados, baseada em uma abordagem híbrida (sob demanda e antecipada), proporcionando transparência em relação à localização de fontes de dados através de uma interface "Web". Para o usuário global, tudo deve se passar como se ele estivesse consultando um banco de dados centralizado.

Dada a ubiqüidade dos bancos de dados relacionais nas organizações, as fontes de dados consideradas pelo sistema são exclusivamente relacionais.

## 1.3 Estrutura da Dissertação

O restante do trabalho está estruturado da seguinte forma:

- No Capítulo 2 descrevemos diferentes tecnologias de integração de múltiplas fontes de dados. Apresentamos também algumas aplicações baseadas nas duas abordagens. Uma crítica a cada tecnologia é feita, na tentativa de justificar o desenvolvimento de um novo sistema, ao qual chamaremos de *Sistema Integrador de Múltiplos Bancos de Dados Relacionais BDRV (Banco de Dados Relacional Virtual)* ;
- No Capítulo 3 exibimos os requisitos necessários para este novo sistema;
- O Capítulo 4 oferece uma visão geral do Sistema BDRV;
- O Capítulo 5 apresenta aspectos de projeto e implementação do sistema. Detalhes de cada módulo podem ser encontrados;
- Finalmente, o Capítulo 6 refere-se às conclusões do trabalho realizado e suas perspectivas.

# Capítulo 2

## *Abordagens e Tecnologias para Integração de Informações de Múltiplas Fontes de Dados*

### **2.1 Introdução**

Nos dias de hoje, o acesso integrado a múltiplas fontes de dados heterogêneas está se tornando muito importante, especialmente devido ao crescimento da "Web" e de outros sistemas distribuídos [4].

Duas tendências no que diz respeito à tecnologia de informação estão transformando rapidamente a forma como inúmeras organizações processam seus dados. Em primeiro lugar, a excessiva quantidade de sistemas de computação ligados em rede tem colocado juntos "workstations" e "mainframes", formando coleções de serviços de informação distribuídos e "on-line". O maior exemplo que podemos observar é a "Web". A segunda tendência é que nem todas as unidades de informação disponíveis nestas redes são homogêneas, podendo ser composições de tipos de dados heterogêneos, como documentos texto, imagens e registros de bancos de dados.

Numa tentativa de disponibilizar o poder destes sistemas distribuídos e heterogêneos ao usuário global, os sistemas existentes possibilitam a extração de informações de cada um dos vários tipos de fonte de dados. Entretanto, tais soluções não tornam a procura por informação mais fácil ou simples, visto que é necessário que o usuário aprenda a lidar com a complexidade de uma gama de interfaces, características de sistemas e formatos de dados. Ainda, é necessário que o usuário saiba como receber, manipular e combinar tipos de dados complexos obtidos a partir de inúmeras fontes.

No que diz respeito ao problema de integração de múltiplos bancos de dados, academia e indústria seguiram caminhos divergentes durante certo tempo. O problema da integração foi atacado inicialmente pela academia através da **abordagem sob demanda** (mediadores) [4, 13]. Nesta abordagem, a informação é extraída e integrada das diversas fontes somente quando a consulta é submetida.

As tecnologias que seguem a abordagem sob demanda normalmente incluem um componente bastante importante em suas arquiteturas, chamado de mediador [13]. Segundo [4], mediador é um serviço que fornece uma visão virtual e integrada de fontes de dados heterogêneas. Os usuários globais podem ter acesso a tal visão utilizando uma interface de consulta transparente aos modelos de dados e localização das fontes. Para isto, um mediador utiliza um esquema global, baseado em algum modelo de dados, que pode ser diferente dos modelos de dados das fontes. Para [13], mediador é um módulo ocupando uma camada explícita e ativa entre as aplicações dos usuários e as fontes de dados. O acesso aos mesmos é realizado através de programas aplicativos residindo nas estações dos usuários. Um mediador representa uma camada intermediária, fazendo com que as aplicações dos usuários se tornem independentes em relação às fontes de dados.

Conforme podemos observar na Figura 2.1, a arquitetura clássica, em três camadas, de um sistema baseado na abordagem sob demanda é composta pelas seguintes camadas:

- Usuário: formada pelas aplicações de usuários;
- Mediador: possui o papel de coletar e agrupar informações oriundas de múltiplas fontes de dados. Interage com as fontes através de "wrappers" [24] (também chamados de "proxies"), os quais constituem um mecanismo de abstração, contribuindo para a extensibilidade do sistema. "Wrappers" convertem dados obtidos de fontes de dados em um modelo de dados comum;
- Fontes de dados: composta pelas fontes de dados, possivelmente heterogêneas, sendo integradas.

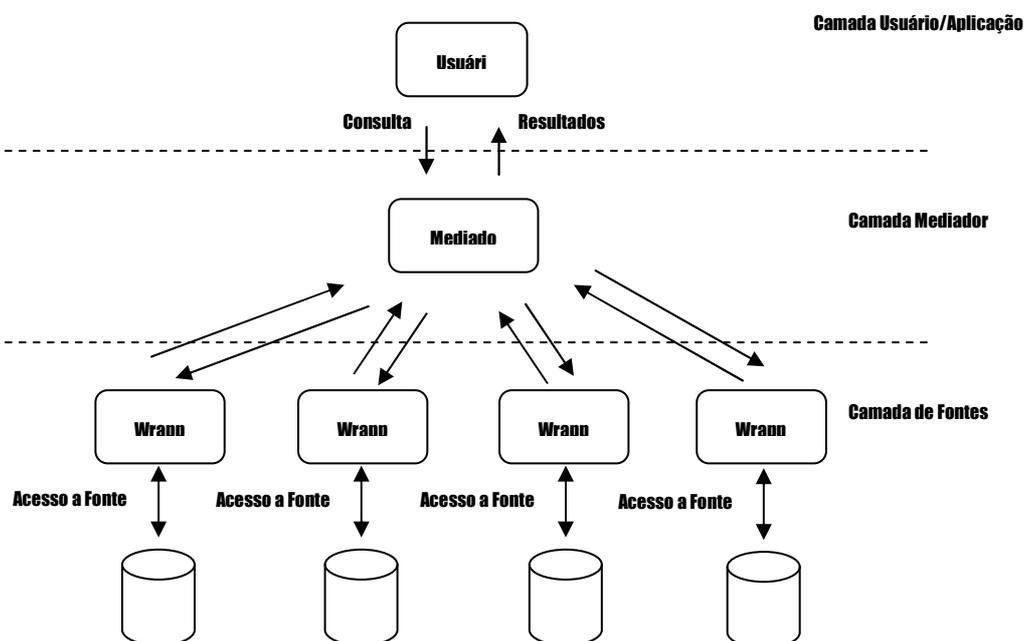


Figura 2.1 – Arquitetura Clássica, em Três Camadas, de um Sistema Baseado na Abordagem Sob Demanda

As três principais tarefas de um mediador são:

- Seleção das Fontes de Dados: determinar o conjunto apropriado de fontes para responder a uma certa consulta;
- Tradução/Processamento da Consulta: determinar os fragmentos de consulta a serem enviados a cada fonte selecionada no passo anterior; gerar as consultas apropriadas para cada fonte; recuperar os resultados de cada fonte;
- Integração dos Resultados Parciais: filtrar e integrar as informações de cada fonte, e retornar o resultado final ao usuário.

Para cada fonte de dados deve existir um "wrapper". Sua função é traduzir uma consulta enviada pelo mediador em uma consulta específica da fonte de dados em que atua. Após receber os resultados da fonte, o "wrapper" traduz os mesmos para o formato do esquema de dados global. O "wrapper" mantém, desta forma, uma interface entre um mediador (esquema global) e uma fonte de dados (esquema local).

A alternativa pesquisada na indústria, por sua vez, é a **abordagem antecipada**. Nela a informação é extraída das fontes, integrada, filtrada e armazenada num repositório central, especialmente projetado para as necessidades de análise (nível estratégico). Quando uma consulta é submetida, é executada diretamente sobre este repositório, sem necessidade de acesso a outras bases. Este repositório central é conhecido como "Data Warehouse" [29].

A seguir, descreveremos sucintamente algumas das tecnologias existentes que fazem uso de ambas as abordagens mencionadas anteriormente. Como tecnologias que usam a abordagem sob demanda, temos os Sistemas de Bancos de Dados Distribuídos e os Sistemas de Bancos de Dados Múltiplos. Como tecnologia que utiliza a abordagem antecipada, temos "Data Warehousing".

## 2.2 Sistemas de Bancos de Dados Distribuídos

Um Sistema de Banco de Dados Distribuído (SBDD) [34] é constituído por um ou mais bancos de dados distribuídos fisicamente, e um Sistema Gerenciador de Bancos de Dados Distribuídos (SGBDD) possibilitando a execução de consultas e atualizações. Um SBDD implica em homogeneidade, isto é, todos os componentes físicos utilizam o mesmo SGBDD. O SGBDD suporta um único modelo de dados e linguagem de consulta, com um esquema sem ambigüidade.

Um usuário local de um SBDD *enxerga* somente uma partição do banco de dados, podendo atualizar apenas os dados armazenados em seu "site". Um usuário global, entretanto *enxerga* o banco de dados em sua plenitude, podendo realizar operações sobre dados contidos em quaisquer "sites".

Em um SBDD, destacamos dois tipos de transação: locais, acessam um único "site", justamente no qual ela se inicia; e globais, busca acesso a diferentes "sites", ou a outro "site" além daquele em que se inicia. Desta forma, cada "site" pode participar da execução de uma transação que acessa um dado em um ou diversos "sites".

A Figura 2.2 apresenta a estrutura geral de um sistema distribuído. Podemos observar uma transação global acessando dados armazenados nos "sites" A e B e uma

transação local acessando dados armazenados no "site" C. Analogamente, poderíamos ter uma transação global acessando dados contidos nos "sites" A, B e C e uma transação local acessando dados apenas do "site" A.

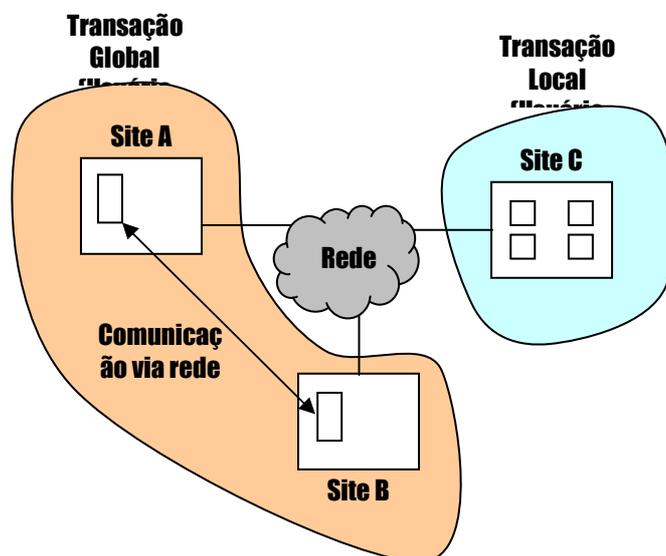


Figura 2.2 - Um Sistema de Banco de Dados Distribuído

## 2.2.1 Processamento de Consultas

Conceitualmente, uma consulta em um SBDD é processada em duas etapas:

1. Tradução: cada subconsulta é *quebrada* em uma subconsulta correspondente ao seu respectivo "site" e, em seguida, enviada para ser executada;
2. Combinação: os resultados oriundos da execução de subconsultas nos diferentes "sites" são combinados. Para tal, pode ser necessário a transferência de dados de um "site" para outro.

## 2.2.2 Críticas dos Sistemas de Bancos de Dados Distribuídos

A seguir, citamos algumas das razões para a utilização de SBDDs:

- Compartilhamento de dados: usuários de um "site" podem ter acesso a dados residentes em outros "sites";
- Administração compartilhada: a responsabilidade pela administração do banco de dados distribuído pode ser compartilhada pelos administradores locais dos "sites";
- Disponibilidade: se um "site" sai do ar, os demais "sites" podem continuar em operação.

A principal desvantagem dos SBDDs é a potencial ineficiência na obtenção de resultados, no que diz respeito ao tempo de resposta de uma consulta. Isto se deve a sua arquitetura puramente distribuída. Para responder a determinadas consultas, é possível que dados armazenados em "sites" distintos necessitem ser movidos e combinados. Como os dados estão sob controles distintos, é necessário coordenar as transações para preservar a consistência dos mesmos.

## 2.3 Sistemas de Bancos de Dados Múltiplos

A proliferação de sistemas de bancos de dados baseados em diferentes modelos de dados impossibilita aos usuários globais um acesso uniforme a dados armazenados em fontes de dados heterogêneas. Nem todos os SGBDs disponíveis apresentam modelos de dados e linguagens de consulta idênticas. Desta forma, usuários acostumados a acessar dados em um determinado sistema apresentam uma certa dificuldade quando necessitam ter acesso a dados contidos em outro sistema. Uma solução para este tipo de problema é desenvolver uma camada de "software" adicional suportando um único modelo de dados e linguagem de consulta no topo de diferentes tipos de sistema já existentes (modelo e esquema globais). Esta camada, juntamente com os sistemas subjacentes, constitui o que se chama Sistema de Bancos de Dados Múltiplos (SBDM), também conhecida como Sistema de Bancos de Dados Heterogêneos, ou ainda, do inglês, "Multidatabase".

Um SBDM disponibiliza um esquema de banco de dados global e único, através do qual os usuários globais podem gerar consultas e realizar atualizações. Um SBDM

armazena apenas o esquema global, enquanto os sistemas de bancos de dados locais mantêm os dados. O esquema global é construído a partir da integração dos esquemas pertencentes aos bancos de dados locais. A Figura 2.3, extraída de [31], ilustra os conceitos de esquemas local e global.

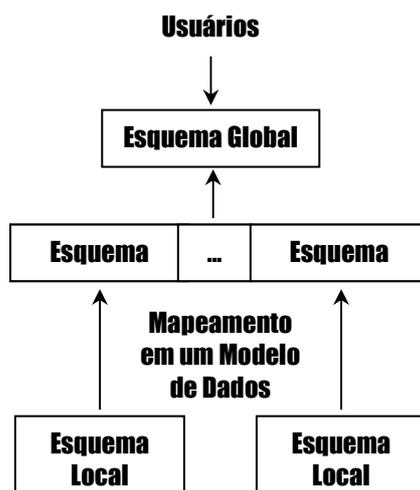


Figura 2.3 - Arquitetura de Integração de Esquemas Conceituais dos Bancos de Dados Locais

A integração dos esquemas não é simplesmente um esforço de translação entre linguagens de definição de dados. Um mesmo atributo pode aparecer em vários SGBDs locais, mas com significados diferentes. Os tipos de dados usados em um sistema podem não ter suporte em outros e a translação entre ambos pode não ser tão simples. Mesmo para tipos de dados idênticos, pode haver problemas originados em função da representação física do dado. Um sistema pode usar ASCII e outro EBCDIC. Os números inteiros podem ser representados pelas formas "big-indian" ou "little-indian". No nível semântico, o valor de um inteiro para medidas pode estar em polegadas em um sistema e em milímetros em outro, criando assim uma situação delicada na qual a igualdade entre inteiros é apenas uma noção aproximada (como acontece sempre no caso de números com ponto flutuante). O mesmo nome pode aparecer em linguagens diferentes em sistemas diferentes. Por exemplo, um sistema com base no Brasil pode referir-se à cidade de Colônia enquanto a um alemão refere-se a "Köln". Também, as representações de pontos flutuantes podem divergir.

### 2.3.1 Processamento de Consultas

Um SBDM traduz as consultas e/ou atualizações em consultas e/ou atualizações a serem enviadas e processadas nos sistemas de bancos de dados locais apropriados, e integra os conjuntos-resposta, gerando o resultado final a ser apresentado ao usuário. Além disto, um SBDM coordena a validação ou cancelamento de transações globais (consultas e atualizações) pelos sistemas de bancos de dados locais no intuito de manter a consistência dos dados armazenados nos bancos de dados locais.

Conceitualmente, uma consulta é processada em três etapas:

1. Decomposição: a consulta global é decomposta em subconsultas destinadas aos bancos de dados que de fato contenham os dados solicitados. Mesmo após a decomposição, as subconsultas continuam na linguagem de consulta global;
2. Tradução: cada subconsulta é traduzida em uma subconsulta correspondente ao seu respectivo sistema de banco de dados e, em seguida, enviada para ser executada;
3. Combinação: os resultados oriundos da execução de cada subconsulta são retornados pelos seus respectivos sistemas de bancos de dados e agrupados posteriormente.

### 2.3.2 Aplicações

#### **"UniSQL/M Multidabase System"**

UniSQL/M [21] é um SBDM que integra fontes de dados heterogêneas (bancos de dados relacionais e orientados a objeto).

Utiliza uma linguagem de definição e manipulação de dados, chamada de SQL/M, que fornece inúmeras facilidades em termos de abstrações (orientação a objeto) como generalização, especialização, agregação, entre outras. SQL/M estende a noção de visão relacional com inúmeras facilidades de abstração proporcionada pela orientação a objeto (generalização, especialização e agregação).

O esquema global é formado por uma visão uniforme dos bancos de dados locais, podendo ser modificado caso haja alterações nos requisitos dos usuários globais ou modificações nos esquemas locais dos bancos de dados. É composto por dois componentes: um esquema de transformação formado por "proxies" e um esquema conceitual formado por classes virtuais (visões), conforme podemos ver na Figura 2.4. Os atributos de um "proxy" mantêm uma correspondência um-para-um com os atributos de uma entidade local (relação ou classe em um banco de dados local). A função principal de um "proxy" é auxiliar na transformação de um comando (consulta ou atualização) global em comandos referentes aos bancos de dados locais

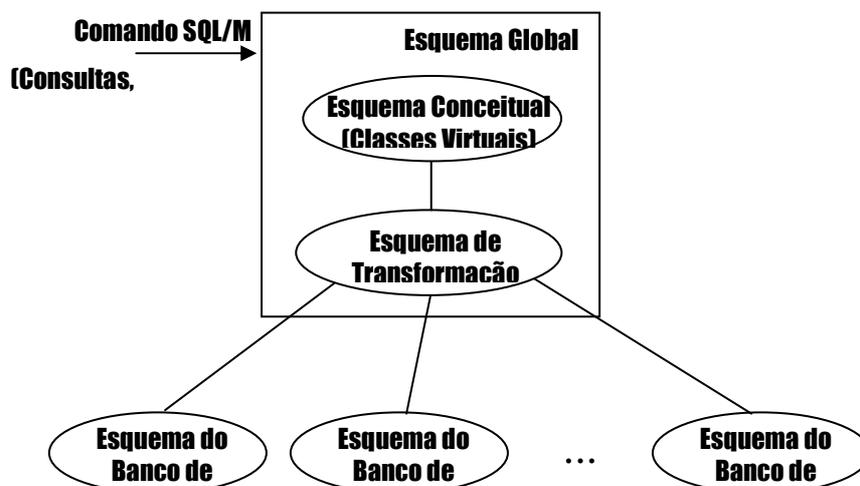


Figura 2.4 - Arquitetura de Integração de Esquemas de "UniSQL/M"

## "Multibase"

"Multibase" [27] é um sistema que fornece uma interface uniforme para recuperação de dados a partir de bancos de dados heterogêneos e distribuídos já existentes, utilizando uma linguagem de consulta única.

Proporciona a execução de consultas que solicitem dados armazenados em diferentes bancos de dados, apresentando esquemas, modelos de dados e linguagens de consulta distintas. Para tanto, não são necessárias quaisquer modificações nos bancos de dados, SGBDs ou programas "front-end".

Fornece ao usuário global a ilusão de um banco de dados único, integrado e centralizado. Para tal, armazena a localização dos bancos de dados locais, acessa os dados em cada SGBD, resolve possíveis conflitos e combina os resultados retornados pelos vários bancos de dados.

Para executar consultas globais, "Multibase" segue uma série de etapas: transformação de consultas expressas na linguagem de consulta global em um conjunto de subconsultas expressas em, possivelmente, diferentes linguagens suportadas pelos SGBDs locais; formulação de um plano eficiente para executar uma seqüência de subconsultas; implementação de um módulo eficiente que possibilite acesso aos dados de uma fonte local; movimentação de dados oriundos de subconsultas através das diferentes fontes; resolução de conflitos entre os bancos de dados; combinação dos dados para compor uma resposta única. "Multibase" não possibilita a atualização dos dados armazenados nos bancos de dados locais.

### **2.3.3 Críticas dos Sistemas de Bancos de Dados Múltiplos**

As principais vantagens da utilização dos SBDMs são:

- propiciam a obtenção de informações em tempo real;
- possibilidade de acesso a um grande número de fontes de dados, exclusivamente bancos de dados, que podem apresentar modelos de dados e linguagens de consultas distintas;
- possibilitar que usuários e aplicações globais possam interagir com uma única linguagem de consulta a banco de dados;

- ao contrário dos bancos de dados distribuídos, cada SGBD local possui autonomia [14, 23], ou seja, controle completo sobre os dados locais e processamentos sobre os mesmos;

Como qualquer tecnologia inspirada na abordagem sob demanda, "Multidatabases" enfrentam a potencial ineficiência na obtenção de resultados, no que diz respeito ao tempo de resposta de uma consulta. Isto se deve a sua arquitetura puramente distribuída. O aumento no tempo de processamento pode ocorrer em decorrência de dois fatores. Cada consulta necessita ser traduzida numa forma que cada fonte de dados a entenda. Os resultados obtidos de cada fonte de dados devem ser traduzidos, interpretados e agrupados, até serem apresentados ao usuário global.

No que diz respeito ao sistema "Multibase" (seção 2.3.2), podemos assinalar outra desvantagem: não possibilita a realização de atualizações nos bancos de dados locais.

## 2.4 "Data Warehousing"

Segundo [18], "Data Warehousing" (DWing) é uma coleção de tecnologias de apoio à decisão, que ajudam *tomadores de decisão* a tomar decisões rápida e corretamente [18]. Para [26], DWing é um processo, e não um produto, para montar e gerenciar dados oriundos de várias fontes com o objetivo de prover uma visão simples, detalhada sobre parte de ou todo o negócio.

Na tecnologia "Data Warehousing" (DWing), inspirada na abordagem antecipada para integração de dados, os dados de cada fonte são extraídos antecipadamente, traduzidos, filtrados e integrados apropriadamente, e armazenados em um "Data Warehouse" (DW). Segundo [29], o DW é um conjunto de dados baseados em assuntos, integrados, não-volátil, e variável em relação ao tempo, de apoio às decisões gerenciais. Quando uma consulta é submetida, a mesma é avaliada diretamente no DW, sem a necessidade, portanto, de ter acesso às várias fontes de dados em tempo real.

Diferentemente da abordagem sob demanda, que utiliza visões virtuais, DWing faz uso de visões materializadas [18, 28]. Visões materializadas são freqüentemente agregações ou sumários pré-computados e pré-armazenados de outros registros de diferentes bancos de dados. Tais visões materializadas têm a vantagem de reduzir o tempo de processamento das consultas que as utilizam.

### 2.4.1 "Executive Information Systems"

A tecnologia de DWing é voltada principalmente para executivos da alta e média gerência, que necessitam muitas vezes de informações não somente integradas, mas também sumariadas, ao contrário dos sistemas de bancos de dados tradicionais, dirigidos especificamente à baixa gerência, ou pessoal ao nível de controle operacional.

Para obter as informações desejadas, os usuários de nível estratégico (global) utilizam os "Executive Information Systems" (EIS)<sup>1</sup>. Um EIS permite localizar problemas e determinar tendências relacionadas aos vários aspectos de um negócio [29]. Os usuários deste nível possuem visões diferentes dos dados, é a visão corporativa, que em nada se assemelha à visão operacional. À alta-gerência interessam dados resumidos, agregados sob diversos aspectos, orientados a assuntos e históricos. O acesso aos dados deve ser rápido, direto, fácil, intuitivo e sob a perspectiva dos negócios.

Comumente, EIS são confundidos com "Decision Support Systems" (DSS)<sup>2</sup> por autores e fabricantes de "softwares", devido talvez ao impacto causado pela sua nomenclatura - Apoio à Decisão -, que torna o "marketing" dos fabricantes mais apelativo.

### 2.4.2 Modelagem Multidimensional

A técnica ideal de modelagem de dados para construção de repositórios para o processamento de EIS é conhecida como Modelagem Multidimensional [32].

---

<sup>1</sup> Em português, Sistemas de Informações Executivas.

<sup>2</sup> Em português, Sistemas de Apoio à Decisão.

Na Modelagem Multidimensional, os dados operacionais são visualizados sob diversos ângulos administrativos, refletindo os modelos mentais da alta-gerência. Esses ângulos administrativos são chamados *dimensões* do negócio (ex.: produto, loja, promoção, região, etc). Referentes a cada dimensão existem  *fatos* comerciais relevantes à gerência (ex.: vendas, compras, etc). Fatos são valores quantitativos referentes ao desempenho comercial de um grupo de dimensões. Assim, por exemplo, podemos visualizar o fato vendas por loja, por produto e por região.

Para melhor compreender o descrito acima, observe a Figura 2.5. Esta estrutura é um cubo tridimensional com as dimensões: produto, região e tempo. Note que cada face do cubo representa uma dimensão. O cubo em si é preenchido com valores quantitativos que representam o número de unidades vendidas de determinado produto, durante um determinado tempo numa determinada loja. Esta estrutura é conhecida como Cubo Multidimensional [32]. No nosso exemplo, representamos apenas três dimensões pela nossa capacidade de visualização espacial limitar-se a esse número. Todavia, é possível estruturas com  $n$  dimensões.

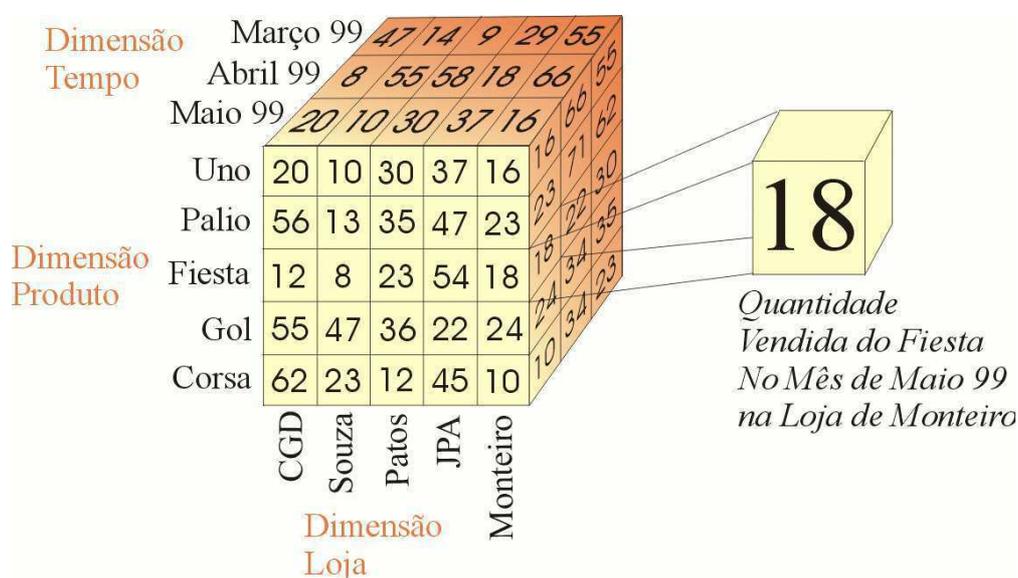


Figura 2.5 - Cubo Multidimensional com Três Dimensões [32]

O tipo de processamento aplicado sobre estruturas dimensionais difere substancialmente daquele utilizado por aplicações do nível operacional<sup>3</sup>. Surge o processamento analítico denominado "On-line Analytical Processing" (OLAP) [32], com

<sup>3</sup> O Processamento Transacional Clássico utilizando por grande parte das aplicações do nível operacional através dos SGBDs Relacionais é chamado de "On-Line Transaction Processing" (OLTP).

a necessária flexibilidade esperada pelos usuários de EIS. O processamento OLAP tornou-se então a alma do EIS.

### 2.4.3 "On-line Analytical Processing"

O termo OLAP vem sendo utilizado para caracterizar uma nova gama de aplicações voltadas para análise gerencial. Essas aplicações executam o processamento OLAP cujo objetivo é prover ao usuário a visualização dos dados sob diferentes ângulos gerenciais e comportar todas as necessidades da atividade de análise. Ferramentas OLAP possuem capacidades analíticas para visualizar os dados sob diferentes ângulos de forma fácil e rápida.

Segundo [32], OLAP é o processo de criar e gerenciar dados corporativos multidimensionais para visualização e análise do usuário que deseja *enxergar* o que realmente os dados dizem. OLAP conduz analistas, gerentes e executivos a descobrirem conhecimentos nos dados através de acesso rápido, interativo e consistente a uma variedade de visões possíveis da informação.

Note-se a referência a termos como *diferentes ângulos*, *dados multidimensionais* e *variedade de visões possíveis*. Todos eles referem-se à Análise Multidimensional, que é o *coração* do OLAP. Esses diferentes ângulos referem-se aos diferentes ramos, áreas, aspectos ou componentes do negócio: as dimensões do negócio (produto, loja, fornecedor, promoção, etc). Referentes às dimensões existem fatos quantificáveis sobre o desempenho dessas em atividades do negócio (fato vendas, fato compras, etc.). Dados esses conceitos que refletem os dois pontos importantes na Análise Multidimensional, trazemo-los para o mundo dos SIs. Os atributos dos dados operacionais podem ser classificados no processamento OLAP, segundo sua natureza, em:

- Atributos de Fatos: atributos quantitativos sobre o desempenho do negócio em determinado(s) ramo(s). Exemplo : sobre o fato vendas, a quantidade vendida, o preço da unidade, a margem de lucro, etc;

- Atributos de Dimensão: atributos qualitativos que caracterizam os ramos do negócio envolvidos na medida de desempenho de determinado fato. Exemplo : sobre a dimensão produto, descrição, embalagem, preço, etc.;

Os conceitos de fatos e dimensões permeiam todo o universo da Análise Multidimensional. Esses dois conceitos são suficientes para prover a visão gerencial dos dados. As dimensões, como representantes de áreas, aspectos, ramos ou componentes do negócio provêm a visão dos dados que realmente importa aos *tomadores de decisão*. Os fatos, por sua vez, expressam a medida de desempenho dos componentes das dimensões. São os fatores quantitativos que permitem a introspecção nos dados para descoberta de tendências e reconhecimento de padrões.

## **Modelos para Repositórios OLAP**

Está claro até este ponto que as necessidades ditas gerenciais diferem muito das necessidades ditas operacionais. Os sistemas operacionais geralmente se baseiam em processamento transacional clássico, onde um grande número de transações é executado num pequeno intervalo de tempo, com alto grau de concorrência. As atividades dos sistemas operacionais tendem a acontecer numa taxa relativamente constante. Os dados são detalhados. As atualizações dos dados são tão frequentes quanto às leituras. Os bancos de dados refletem o estado atual dos dados. As consultas geralmente ocorrem sobre um pequeno conjunto de dados e são conhecidas de antemão [32]. A estrutura dos dados é fixa enquanto que o conteúdo é variável. O foco dos sistemas operacionais está no armazenamento, na confiabilidade, no desempenho e na disponibilidade dos dados. Eles mantêm a empresa funcionando [18].

No outro extremo, os sistemas projetados para análise procuram facilitar consultas e navegação na estrutura dimensional e assim prover informação realmente gerencial. Os dados para aplicações OLAP são resumidos, históricos e baseados em assuntos. As consultas OLAP geralmente "ad-hoc" e acessam grandes conjuntos de dados. Não há atualizações como as que executamos no nível operacional, essas geralmente são feitas em "batch" e por causa disso não há controle de concorrência. Os dados não necessariamente seguem os princípios da normalização [9] já que o foco do

OLAP está na recuperação e visualização e não no armazenamento. Existem duas grandes vertentes para Modelagem Multidimensional: MOLAP e ROLAP, as quais descrevemos nas subseções seguintes.

### **MOLAP (Multidimensional OLAP)**

Baseia-se nos Bancos de Dados Multidimensionais [32]. A estrutura por trás dessa abordagem é o Cubo Multidimensional (ou "Multidimensional Array", em português, Vetor Multidimensional) como o da Figura 2.5.

Em um cubo multidimensional, cada aresta do cubo representa uma dimensão do negócio. Cada linha ou cada coluna de cada aresta representa um elemento de uma dimensão (ex. CGD, Monteiro, etc. para a dimensão Loja; e Uno, Palio e Fiesta para a dimensão Produto). As células do cubo são preenchidas com valores quantitativos que são as medidas ou fatos do negócio.

### **ROLAP (Relational OLAP)**

Para reagir às tentadoras facilidades oferecidas pelos BDs Multidimensionais, a indústria relacional logo propôs sua alternativa. O ROLAP [18, 28] utiliza-se de tabelas relacionais para simular o cubo multidimensional.

O esquema proposto concentra numa tabela as medidas ou fatos do negócio. Esta tabela é conhecida como tabela de fatos (do inglês: "fact table"). Os componentes das dimensões são armazenados em outras tabelas, chamadas tabelas de dimensão (do inglês: "dimension tables"). As tabelas de dimensão compõem com a tabela de fatos um relacionamento de chave estrangeira 1-N (um para N). Uma tabela de fatos está ligada a várias tabelas de dimensão, formando uma estrutura semelhante a uma estrela, fato esse que deu nome ao esquema: Esquema em Estrela (do inglês: "Star Schema").

## **Operações OLAP**

A maioria das consultas OLAP é "ad-hoc" e algumas operações dessas consultas sobre as estruturas dimensionais são típicas e merecem atenção especial. Elas são chamadas operações OLAP e são divididas em dois grandes grupos:

- Análise Prospectiva: consiste em analisar partindo de um conjunto de dados até chegar a conjuntos mais detalhados, menos detalhados ou outros conjuntos de dados [29]. Operações :
  - "Drill-Down": descer no nível de detalhes de uma dimensão;
  - "Roll-Up": contrário de "Drill-Down", ou seja, *caminhar* para a visão e dados mais resumidos;
  - "Drill-Across": significa caminhar a partir de uma dimensão para outra dimensão, combinando-as para mudar o enfoque da análise.
- Análise Seletiva: objetiva selecionar pedaços do conjunto de dados para análise. Operações:
  - "Slice&Dice": Em português, cortar e fatiar. Seleciona pedaços transversais do cubo e em seguida aplica critérios de seleção sobre este pedaço;
  - "Pivoting": Rotação. Muda a orientação dimensional para um relatório.

## 2.4.4 Arquitetura

A literatura fornece uma variedade de arquiteturas para DWing. Contudo, no intuito de melhor entendermos o funcionamento geral de um ambiente de DWing, preferimos utilizar uma arquitetura funcional básica [29].

Na Figura 2.6, podemos observar os principais componentes de um ambiente de DWing inseridos em retângulos sombreados. As setas indicam o fluxo de dados entre os componentes. A seguir, detalharemos os componentes desta arquitetura e, posteriormente, o fluxo de dados.

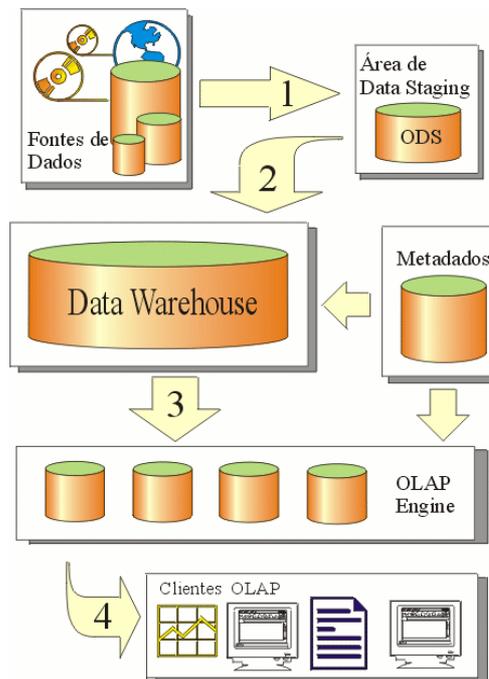


Figura 2.6 – Arquitetura Funcional Básica de um Ambiente de DWing [29]

## Componentes

### a) Fontes de dados

As fontes de dados podem ser divididas em dois grupos:

- Fontes de Dados Internas: são os sistemas de automação do nível operacional. Os dados são provenientes de sistemas "On-line Transaction Processing" (OLTP) ou dos chamados "legacy systems" (em português: sistemas legados ou herdados). Os sistemas legados são os sistemas em "batch" ou arquivados. Esses sistemas não utilizam processamento transacional e nem controle de concorrência;
- Fontes de Dados Externas: são fontes que não fazem parte dos SIs da empresa. Os dados dessas fontes geralmente chegam através de redes públicas de comunicação. A "Internet" é uma grande fonte de dados externa.

### b) Área de "Data Staging"

É uma área de armazenamento intermediária entre as fontes de dados e o DW. Nesta área são realizados o tratamento, a integração e o controle de qualidade sobre os dados para que eles possam ser transferidos para o DW.

O repositório desta área também é conhecido como ODS ("Operational Data Store"), cujo objetivo é integrar as várias fontes de dados operacionais [28, 32].

Não necessariamente existe apenas um repositório para a Área de "Data Staging", porém, os vários que venham a existir devem ser integrados e homogêneos.

c) "Data Warehouse"

É o repositório central, integrado, histórico e orientado a assuntos.

d) Clientes OLAP

Generalizamos o termo clientes OLAP para caracterizar todas as aplicações que acessam os dados armazenados no DW.

e) OLAP "Engine"

Este componente está presente em arquiteturas baseadas em ROLAP ou MOLAP [18]. Nela, dados do DW são extraídos para repositórios menores chamados "Data Marts", que são DWs departamentais, menores em tamanho e orientados aos assuntos específicos de um departamento.

f) "Metadados"

Os "metadados" têm a função de descrever o DW [29].

## **Fluxo de Dados**

Num ambiente de DWing os dados seguem um fluxo determinado. Inicialmente esses dados precisam ser (1) integrados de várias fontes para posteriormente serem estruturados (2), armazenados (3) e distribuídos para análise (4).

Cada seta direcionada na arquitetura da Figura 2.6 é um estágio do fluxo de dados no DWing que agrega quatro atividades:

1 - Extração das várias fontes, armazenamento e integração na Área de "Data Staging".

Uma vez na Área de "Data Staging", os dados passam por diversas etapas de transformação as quais incluem [18, 28]:

- Integração e geração de novas chaves ;
- Adição do tempo;
- Checagem da integridade referencial para tabelas de fato e tabelas de dimensão;
- Desnormalização ou normalização;
- Conversão de tipos de atributos;
- Cálculos, derivações e alocações;
- Construção de agregados;
- Tratamento de valores nulos.

2 – Carga dos dados no "Data Warehouse".

Uma vez feita toda a transformação na Área de "Data Staging" os dados são carregados no DW. As principais atividades do processo de carga são:

- Particionamento de tabelas;
- Reconstrução de Índices;
- Controle de Qualidade;
- Publicação dos Dados.

3 - A partir do "Data Warehouse", os dados são carregados nos "Data Marts".

O uso de "Data Marts" melhora o desempenho do sistema à medida que as consultas analíticas são divididas entre vários DWs departamentais, evitando assim a sobrecarga de consultas sobre o DW corporativo.

As principais atividades envolvidas nesta carga são:

- Separação dos dados por áreas funcionais;
- Transformação dos dados do modelo lógico corporativo para o modelo lógico departamental.

4 – Neste ponto o fluxo de dados é basicamente composto de conjuntos-resposta para os Clientes OLAP. As atividades incluem :

- Processamento e otimização de consultas;
- Geração de relatórios;
- Controle de acesso e segurança.

## 2.4.5 Críticas da Tecnologia "Data Warehousing"

Em resumo, a abordagem antecipada é apropriada para:

- usuários exigindo informações sumariadas e, de uma certa maneira, previsíveis;
- usuários requerendo alto nível de desempenho (dados disponíveis localmente no DW), mas não necessariamente exigindo informações rigorosamente atualizadas;
- compartilhamento entre as aplicações locais às fontes e as aplicações OLAP globais;
- informações não contidas nas fontes (históricos, sumários, etc).

Como desvantagens deste enfoque, podemos apontar:

- complexidade de se criar e manter o DW;

- impossibilidade de acesso direto às fontes, por executivos da alta gerência<sup>4</sup>.

## 2.5 Conclusões

Tendo em vista as desvantagens encontradas nas diferentes abordagens e tecnologias de integração de múltiplas fontes de dados, justifica-se o investimento em um novo sistema integrador de fontes de dados. Um tal sistema é a essência desta dissertação, constituindo o assunto do restante do documento.

---

<sup>4</sup> Isto muitas vezes é necessário!

# Capítulo 3

## *Requisitos para o Sistema Integrador de Bancos de Dados Relacionais BDRV*

### **3.1 Introdução**

Considere um usuário global que deseja utilizar um Sistema de Informação heterogêneo para obter informações a partir de qualquer banco de dados relacional. Neste caso, o usuário não está interessado em saber de que fonte provém as mesmas. As informações procuradas estão descentralizadas, ou seja, armazenadas em diferentes fontes de dados.

O Sistema Integrador de Múltiplos Bancos de Dados Relacionais BDRV (Banco de Dados Relacional Virtual) necessita de mecanismos para ocultar as diferentes fontes de dados, identificando as fontes que de fato contenham os dados solicitados e combinando os resultados a serem apresentados ao usuário global. Para ele, tudo se passa como se estivesse consultando um banco de dados centralizado.

Como forma de demonstrar a validade do Sistema BDRV, desenvolvemos uma aplicação que consiste da integração de dados da Coordenação de Pós-Graduação em Informática (Copin) e do Departamento de Sistemas e Computação (DSC) (ver Anexo A).

## 3.2 Requisitos para o Sistema BDRV

Apresentaremos, a seguir, os requisitos do Sistema BDRV, no que diz respeito aos seguintes tópicos:

- Arquitetura;
- Integração;
- Segurança;
- Fontes de Dados;
- Interface;
- Configuração;
- Plataforma.

### 3.2.1 Arquitetura

A arquitetura do Sistema BDRV é composta por um conjunto de componentes que permite o acesso, pelos UGs, à informações provenientes de múltiplas fontes de dados, exclusivamente bancos de dados relacionais. O acesso é feito de forma transparente em relação à localização das fontes de dados. As informações integradas são unicamente informações estruturadas, do tipo registros de uma tabela.

A arquitetura segue o modelo de três camadas de [13]. Seus principais componentes são mostrados na Figura 3.1 e descritos a seguir.

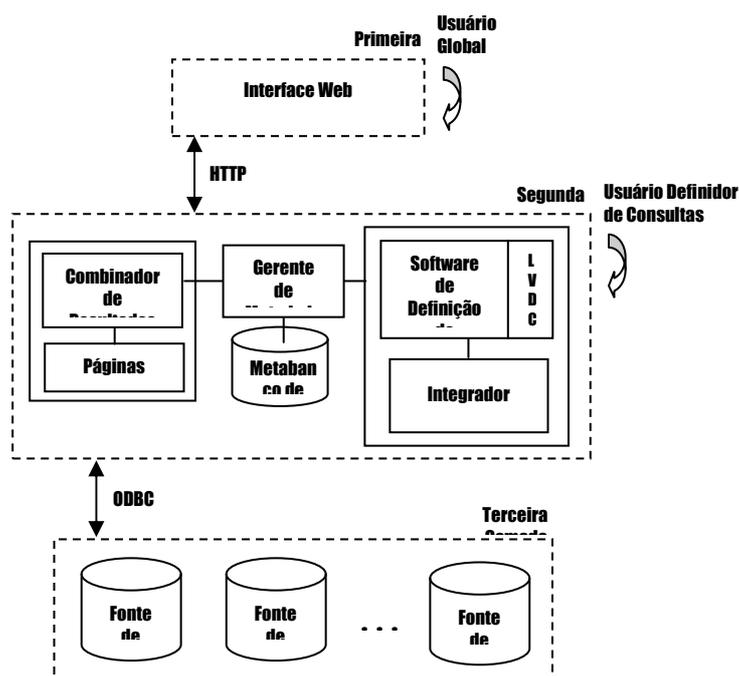


Figura 3.1 – Arquitetura BDRV em Camadas

### Primeira Camada

A Primeira Camada apresenta uma interface "Web" com o UG (ver Seção 3.2.3).

Considere um ambiente acadêmico, onde existem dois bancos de dados a serem integrados. O primeiro deles - *Seminários* - armazena informações sobre os seminários apresentados por alunos de um certo departamento. O segundo - *Controle Acadêmico* - armazena informações pertinentes aos alunos, disciplinas e professores do departamento mencionado anteriormente.

Suponha que um determinado UG deseja obter os seminários apresentados por alunos cuja área de pesquisa seja Banco de Dados, no ano de 2000. Para que isto seja possível, é necessário que exista uma consulta cadastrada anteriormente (ver Segunda Camada), que possa recuperar os dados solicitados. Uma consulta obrigatoriamente deve possuir um nome que a identifique. Suponhamos, então, que o nome da consulta seja *Seminários Apresentados pelos Alunos de Banco de Dados, em 2000*.

As consultas cadastradas são apresentadas ao UG através de uma interface "Web". Assim, para obter as informações desejadas, o UG necessita apenas selecionar a

consulta *Seminários Apresentados pelos Alunos de Banco de Dados, em 2000*, através de um simples clique de "mouse".

## Segunda Camada

Na Arquitetura BDRV, são utilizados dois níveis de esquema: **esquemas globais**, que são esquemas relacionais virtuais construídos a partir da integração de vários esquemas relacionais locais, servindo de base para elaboração de consultas; e **esquemas locais**, que representam o esquema de cada banco de dados sendo integrado.

Uma **consulta** é basicamente uma coleção de tabelas e atributos selecionados a partir de um esquema global<sup>1</sup>. Consultas são criadas pelos Usuários Definidores de Consultas (UDCs) (ver Seção 3.2.3).

A **Interface de Definição de Consultas (IDC)** tem a função de possibilitar a elaboração de consultas pelo UDC. Internamente, a IDC utiliza uma **Linguagem Visual de Definição de Consultas (LVDC)**, cuja função é descrever um conjunto de regras e opções que podem ser aplicadas na formulação de consultas.

Para elaborar uma consulta o UDC deve primeiramente decidir a que esquema global deseja formular sua consulta. Por conseguinte, é preciso escolher as tabelas virtuais que farão parte da consulta - tabelas *Aluno*, *Área* e *Seminários*. Posteriormente, é necessário selecionar os atributos virtuais a serem apresentados no conjunto-resposta final. Como restrição imposta pela LVDC, são apresentados apenas os atributos virtuais pertencentes às tabelas virtuais selecionadas. O UDC deve escolher os atributos virtuais *Título (Seminário)* e *Nome (Aluno)*.

O passo seguinte é a definição de filtros. A LVDC permite que sejam definidos filtros apenas sobre atributos virtuais de tabelas selecionadas. O UDC deve definir os seguintes filtros: *Nome da Área (Área) = Banco de Dados* e *Ano (Seminário) = 2000*. A LVDC permite ainda a definição de ordenações apenas sobre atributos virtuais de tabelas selecionadas e de agrupamentos somente sobre atributos virtuais selecionados.

---

<sup>1</sup> Os elementos de uma consulta a um esquema global são chamados de tabelas e atributos virtuais.

Para finalizar, o UDC deve fornecer algumas informações referentes à consulta em questão, entre as quais podemos destacar: título (*Seminários Apresentados pelos Alunos de Banco de Dados, em 2000*).

Uma vez elaborada, a consulta é enviada ao **Integrador**, que tem a função de determinar e gerar os fragmentos de consulta (subconsultas) referentes à mesma. Em resumo, o Integrador seleciona os bancos de dados que de fato contenham os dados solicitados gerando uma subconsulta para cada banco de dados escolhido.

Para que o UG ao qual nos referimos na Primeira Camada possa obter as informações almeçadas, se faz necessário a existência de uma consulta previamente cadastrada que possa recuperar as informações solicitadas.

As informações pertinentes às consultas, assim como as subconsultas referentes a cada consulta, ficam armazenadas no **Metabanco de Dados**. São ainda funções do Metabanco de Dados: armazenar a descrição de fontes de dados locais; guardar a descrição de esquemas globais; manter mapeamentos esquemas globais → esquemas locais; armazenar informações para resolução de conflitos (ver Seção 3.2.2).

O processo de transformação de uma consulta em subconsultas envolve uma série de etapas, conforme observaremos em detalhes nos Capítulos 4 e 5. Estas etapas são: análise de consulta, coleta de fonte(s), estimação de fonte(s), seleção de fonte(s) e transformação de consulta. Algumas delas necessitam realizar consultas ao Metabanco de Dados. É função do **Gerente de Metadados** coordenar o processo de realização de consultas ao Metabanco de Dados.

Retornando ao nosso exemplo, a consulta elaborada anteriormente deve ser transformada, pelo Integrador, em subconsultas para ambos os bancos de dados sendo integrados. A primeira subconsulta, destinada ao banco de dados *Seminários*, recupera o título dos seminários e o nome dos alunos que apresentaram seminários em 2000. A segunda subconsulta, destinada ao banco de dados *Controle Acadêmico*, recupera o nome dos alunos cuja área de pesquisa seja Banco de Dados. Ambas as subconsultas ficam armazenadas no Metabanco de Dados.

Quando uma consulta é acionada pelo UG, (ver Primeira Camada), ou seja, quando o UG, através de uma interface "Web" (protocolo HTTP), clica no "link" rotulado pelo nome da consulta, as subconsultas referentes à mesma são recuperadas no Metabanco de Dados e enviadas para serem executadas nos bancos de dados locais (ver Terceira Camada) através do protocolo ODBC.

Após a execução das subconsultas e conseqüente recuperação de cada conjunto-resposta, é necessário ainda detectar e resolver possíveis conflitos existentes entre os conjuntos-resposta recuperados, e agrupar cada um deles. Esta é a tarefa do **Combinador de Resultados**.

De acordo com o nosso exemplo, o Combinador de Resultados deve agrupar os conjuntos-resposta referentes aos bancos de dados *Seminários*, composto pelos atributos *título do seminário* e *nome do aluno*, e *Controle Acadêmico*, formado unicamente pelo atributo *nome de aluno*. O conjunto-resposta final referente ao nosso exemplo é composto pelos atributos *título do seminário* e *nome do aluno*, sendo apresentado ao UG via interface "Web".

### Terceira Camada

A Terceira Camada é composta unicamente pelas fontes de dados, exclusivamente bancos de dados relacionais. A função de cada banco de dados é executar as subconsultas enviadas e retornar os conjuntos-resposta.

## 3.2.2 Integração

A integração de bancos de dados relacionais é uma atividade extremamente complexa, em virtude de eventuais conflitos. Nosso propósito com relação a este ponto é trabalhar somente com formas singelas de integração. Sendo assim, propomos a utilização de soluções já bem conhecidas para resolver alguns dos principais tipos de conflito.

Em [22], podemos encontrar uma classificação dos principais tipos de conflito que podem surgir quando diferentes bancos de dados, todos eles baseados no modelo

relacional, necessitam ser integrados. Utilizaremos esta classificação para indicar os tipos de conflito que o Sistema BDRV pretende detectar e solucionar, além da maneira como os mesmos serão resolvidos.

De acordo com [22], como um banco de dados é definido pelo seu esquema e os dados nele armazenados, os conflitos podem ser classificados em duas categorias:

- **Conflitos de Esquema** - resultam do uso de diferentes definições de esquema em bancos de dados distintos. Existem duas causas básicas para os conflitos de esquema. A primeira é a utilização de estruturas diferentes (tabelas e atributos) para representar uma mesma informação. A segunda é o uso de especificações diferentes para uma mesma estrutura; isto inclui nomes, tipos de dados e restrições diferentes representando tabelas e/ou atributos semanticamente equivalentes;
- **Conflitos de Dados** - resultam da presença de inconsistência nos dados armazenados. Existem dois tipos de conflitos de dados: (1) conflitos de dados errados que violam restrições de integridade especificadas implícita ou explicitamente em dados e (2) conflitos baseados em diferentes representações para um mesmo dado.

A seguir, apresentaremos uma classificação de conflitos mais refinada, também proposta em [22], assim como as soluções a serem adotadas no Sistema BDRV.

Os exemplos utilizados para demonstrar cada tipo de conflito de esquema são baseados nos esquemas relacionais genéricos descritos a seguir. Para nos referirmos a uma determinada tabela de um certo esquema, utilizaremos a notação "[Esquema] . [Tabela]". Analogamente, utilizaremos a notação "[Esquema] . [Tabela] . [Atributo]" para nos referirmos ao atributo de uma certa tabela pertencente a um determinado esquema.

**Esquema Relacional EQ1**

Tabela				
Nome		Semântica		
TB1		ST1		
Atributos				
Nome	Tipo	Chave Estrangeira	Referência	Semântica
AT1	TP1			SA1
AT2	TP2			SA2
AT3	TP1	✓	TB2.AT1	SA3
AT4	TP1	✓	TB3.AT1	SA4

Tabela 3.1 – Tabela TB1

Tabela				
Nome		Semântica		
TB2		ST2		
Atributos				
Nome	Tipo	Chave Estrangeira	Referência	Semântica
AT5	TP1			SA5
AT6	TP2			SA6

Tabela 3.2 – Tabela TB2

Tabela				
Nome		Semântica		
TB3		ST3		
Atributos				
Nome	Tipo	Chave Estrangeira	Referência	Semântica
AT7	TP1			SA7
AT8	TP2			SA8
AT9	TP2			SA9

Tabela 3.3 – Tabela TB3

**Esquema Relacional EQ2**

Tabela				
Nome		Semântica		
TB4		ST1		
Atributos				
Nome	Tipo	Chave Estrangeira	Referência	Semântica
AT10	TP1			SA1
AT11	TP3			SA2
AT12	TP1	✓	TB2.AT1	SA4

Tabela 3.4 – Tabela TB4

Tabela				
Nome			Semântica	
TB3			ST3	
Atributos				
Nome	Tipo	Chave Estrangeira	Referência	Semântica
AT7	TP1			SA7
AT8	TP3			SA8
AT9	TP1	✓	TB3.AT1	SA9

Tabela 3.5 – Tabela TB3

Tabela				
Nome			Semântica	
TB2			ST4	
Atributos				
Nome	Tipo	Chave Estrangeira	Referência	Semântica
AT5	TP1			SA10
AT6	TP3			SA11

Tabela 3.6 – Tabela TB2

## Classificação de Conflitos

### A. Conflitos de Esquema

Os principais tipos de Conflito de Esquema são os seguintes:

#### A.1 Conflitos de Nome de Tabela

São dois os tipos de Conflito de Nome de Tabela:

- **Sinonímia entre tabelas:** consideramos duas tabelas sinônimas se possuírem a mesma semântica, apresentarem nomes diferentes e pertencerem a esquemas distintos. As tabelas EQ1.TB1 e EQ2.TB4 são consideradas sinônimas visto que possuem a mesma semântica (S1), apresentam nomes diferentes (TB1 ≠ TB4) e pertencem a esquemas distintos (EQ1 ≠ EQ2);
- **Homonímia entre tabelas:** consideraremos duas tabelas homônimas se possuírem nomes iguais, apresentarem semânticas diferentes e pertencerem a esquemas distintos. As tabelas EQ1.TB2 e EQ2.TB2 são consideradas

homônimas visto que possuem nomes iguais ( $TB2 = TB2$ ), apresentam semânticas diferentes ( $S2 \neq S4$ ) e pertencem a esquemas distintos ( $EQ1 \neq EQ2$ ).

**Solução adotada:** Para o problema de sinonímia entre tabelas, propomos mapear as tabelas sinônimas em uma única tabela virtual a ser inserida em um esquema global. Por exemplo, as tabelas  $EQ1.TB1$  e  $EQ2.TB4$  devem ser mapeadas na tabela virtual  $EG.TBV1$ . Para o problema de homonímia entre tabelas, propomos mapear cada tabela homônima em uma tabela virtual distinta a ser inserida em um esquema global. Portanto, as tabelas  $EQ1.TB2$  e  $EQ2.TB2$  devem ser mapeadas nas tabelas virtuais  $EG.TBV2$  e  $EG.TBV3$ , respectivamente.

A Tabela 3.7 exibe todos os Conflitos de Nome de Tabela relativos ao nosso exemplo genérico, bem como as tabelas virtuais em que as tabelas de cada esquema são mapeadas.

	Esquema EQ1	Esquema EQ2	Esquema Global EG
Sinônimas	TB1	TB4	TBV1
Homônimas	TB2	TB2	TBV2 e TBV3

Tabela 3.7 – Conflitos de Nome de Tabela Detectados nos Esquemas EQ1 e EQ2

## A.2 Conflitos de Nome de Atributo

São dois os tipos de Conflito de Nome de Atributo:

- **Sinonímia entre atributos:** consideramos dois atributos sinônimos se pertencerem a tabelas sinônimas, possuírem a mesma semântica e apresentarem nomes distintos. Os atributos  $EQ1.TB1.AT1$  e  $EQ2.TB4.AT10$  são considerados sinônimos visto que pertencem a tabelas sinônimas ( $TB1$  e  $TB4$ ), possuem a mesma semântica ( $S1 = S1$ ) e apresentam nomes distintos ( $AT1 \neq AT10$ );
- **Homonímia entre atributos:** consideramos dois atributos homônimos se pertencerem a tabelas homônimas, possuírem semânticas distintas e apresentarem nomes iguais. Os atributos  $EQ1.TB2.AT5$  e  $EQ2.TB2.AT5$  são considerados homônimos visto que pertencem a tabelas homônimas ( $TB2$  e

TB2), possuem semânticas distintas ( $S2 \neq S4$ ) e apresentam nomes iguais ( $AT5 = AT5$ ).

**Solução adotada:** Adotamos soluções semelhantes às propostas para Conflitos de Nome de Tabela. Para o problema de sinonímia entre atributos, propomos mapear os atributos sinônimos em um único atributo virtual. Os atributos sinônimos EQ1.TB1.AT1 e EQ2.TB4.AT10, por exemplo, devem ser mapeados no atributo virtual EG.TBV1.ATV1. Para o problema de homonímia entre atributos, propomos mapear cada atributo homônimo em atributos virtuais distintos. Os atributos homônimos EQ1.TB2.AT5 e EQ2.TB2.AT5, por exemplo, devem ser mapeados nos atributos virtuais EG.TBV2.ATV4 e EG.TBV2.ATV5, respectivamente.

A Tabela 3.8 exibe todos os Conflitos de Nome de Atributo relativos ao nosso exemplo genérico, bem como os atributos virtuais em que os atributos de cada esquema são mapeados.

	Esquema EQ1	Esquema EQ2	Esquema Global EG
Sinônimos	TB1.AT1	TB4.AT10	TBV1.ATV1
	TB1.AT2	TB4.AT11	TBV1.ATV2
	TB1.AT4	TB4.AT12	TBV1.ATV3
Homônimos	TB2.AT5	TB2.AT5	TBV2.ATV4 e TBV3.ATV5
	TB2.AT6	TB2.AT6	TBV4.ATV6 e TBV5.ATV7
	TB1.A13	TB4.A13	TBV6.ATV8 e TBV7.ATV9

Tabela 3.8 – Conflitos de Nome de Atributo Detectados nos Esquemas EQ1 e EQ2

### A.3 Conflitos de Tipo de Dado

Conflitos de Tipo de Dado ocorrem quando atributos sinônimos possuem tipos de dado diferentes.

**Solução adotada:** Os atributos sinônimos EQ1.TB1.AT2 e EQ2.TB4.AT11 são do tipo TP2 e TP3, respectivamente. Sabemos que os mesmos são mapeados no atributo virtual TBV1.ATV2. A escolha do tipo de dados do atributo virtual TBV1.ATV2 é feita observando o tipo de dados dos atributos sinônimos EQ1.TB1.AT2 e EQ2.TB4.AT11. A escolha do tipo de dados de um atributo virtual criado a partir de dois ou mais atributos sinônimos é sempre feita com base no tipo de dado mais amplo. Por exemplo, na hipótese do tipo de dado TP2 ser um subconjunto do tipo TP3, então TP3 deve ser

escolhido como o tipo de dados do atributo virtual TBV1.ATV2. Assim, quando os atributos sinônimos EQ1.TB1.AT2 e EQ2.TB4.AT11 são solicitados, é necessária uma conversão ( $T2 \rightarrow T3$ ) nos dados referentes ao atributo EQ1.TB1.AT2.

## B. Conflitos de Dados

Os tipos de conflito de dados são os seguintes:

### B.1 Representações distintas para um mesmo dado (mesma representação para dados distintos)

Existem três diferentes aspectos para a representação de um dado através de bancos de dados: expressões, unidades e precisão.

#### B.1.1 Expressões Diferentes

Este tipo de conflito ocorre quando bancos de dados distintos utilizam um mesmo tipo de dado ou diferentes tipos de dados. Os exemplos seguintes mostram várias expressões para um mesmo dado:

- Palavras distintas para um mesmo dado: Paraíba, PB, Pb;
- Diferentes "strings" para um mesmo dado:

s/n Aprígio Veloso - Bodocongó - Campina Grande - PB

Rua Aprígio Veloso, s/n, Bodocongó, Campina Grande, CG

- Códigos diferentes para um mesmo dado:

★★★★★, A, Excelente, 1, 10

★★★★, B, Bom, 2, 8

★★★, C, Razoável, 3, 6

★★, D, Ruim, 4, 4

★, E, Péssimo, 5, 2

O terceiro exemplo ilustra o conflito que ocorre quando diferentes tipos de valores são utilizados por diferentes bancos de dados para um mesmo dado; por exemplo, enquanto alguns bancos de dados usam "strings", outros usam inteiros.

**Solução adotada:** Propomos manter uma lista das expressões que necessitam ser mapeadas. Por exemplo, caso um dado esteja armazenado em um banco de dados com um valor X e em outro com um valor Y, então quando solicitado, este dado deve ser mapeado ( $X \rightarrow Y$  ou vice-versa).

### B.1.2 Unidades Diferentes

Estes conflitos ocorrem quando diferentes bancos de dados utilizam diferentes unidades para dados numéricos. Unidades distintas oferecem diferentes significados para valores numéricos. Por exemplo, suponha que uma tabela de um banco de dados expresse o período de empréstimo em termos de dias (de 1 a 28 dias) e que outra tabela, de outro banco de dados, expresse o mesmo atributo em semanas (de 1 a 4 semanas). Assim, mesmo que os bancos de dados possuam um mesmo valor, digamos 2 (dois), como valor do atributo, o significado dos números é diferente em cada banco de dados.

**Solução adotada:** Propomos diferenciar os atributos que utilizam diferentes unidades para dados numéricos considerando-os atributos homônimos.

### B.1.3 Precisões Diferentes

Conflitos de precisão ocorrem quando dois bancos de dados utilizam valores de domínios de diferentes cardinalidades para um mesmo dado. Por exemplo, suponha que o tipo de dado de um atributo peso-do-navio em um banco de dados é definido como {pesado, médio, leve, ultra-leve}, enquanto o mesmo atributo em um outro banco de dados é definido como inteiro, com tonelada como unidade. Assim, o tamanho do domínio para o valor do peso do navio no primeiro banco de dados é quatro; enquanto no outro banco de dados pode ser um milhão (isto é, qualquer inteiro entre 1 e 1.000.000).

**Solução adotada:** Propomos a mesma solução utilizada para resolver o Conflito de Expressões Diferentes.

### 3.2.3 Segurança

Como forma de introduzir um nível de segurança no sistema, utilizamos mecanismos de controle de acesso. Desta forma, permissões de acesso devem ser atribuídas aos usuários.

Os usuários do sistema são classificados em três tipos:

- **Usuário Administrador de Sistema (UAS):** com permissão irrestrita para manipulação dos objetos configuráveis do sistema. Exemplo: administrador de bancos de dados;
- **Usuário Definidor de Consultas (UDC):** é o usuário que, a partir da IDC, insere, exclui e/ou modifica consultas. Exemplo: projetista de banco de dados;
- **Usuário Global (UG):** é o usuário que obtém as informações oriundas da execução de consultas previamente cadastrados pelo UDC. Exemplo: qualquer usuário em potencial que saiba navegar na "Web" e que tenha permissões atribuídas (ver a seguir).

O acesso ao Sistema BDRV é controlado da seguinte forma:

- **Atribuição de Permissões:** permissões devem estar associadas aos UDCs e UGs. As informações de cada usuário a serem armazenadas são: nome, "login" e permissões de acesso;
- **Identificação de Usuários:** para ter acesso ao sistema, o usuário deverá fornecer um "login" e uma senha, recebendo, então, permissões apropriadas. Para ter acesso à IDC deve ser checada a permissão do usuário.

### 3.2.4 Fontes de Dados

Apresentaremos a partir de agora alguns requisitos que dizem respeito às fontes de dados a serem integradas pela Arquitetura BDRV:

- **Autonomia**

O Sistema BDRV não oferece a possibilidade de atualização - inserção, remoção e modificação - de dados contidos nos bancos de dados locais, e sim apenas de consulta aos mesmos. As fontes de dados continuam sendo atualizadas pelos seus respectivos programas "front-end". Diferentemente dos SBDDs, manteremos a autonomia de cada SGBD local.

- **Descrição**

A descrição de cada banco de dados não necessita ser alterada em função de sua inclusão no Sistema BDRV.

- **Atualização de Informações**

A execução de consultas *sob demanda* nas fontes de dados propicia a obtenção de informações atualizadas.

- **Bancos de Dados**

O Sistema BDRV possibilita a integração de um grande número de fontes de dados. Estas podem ser qualquer banco de dados relacional, desde que sua descrição seja devidamente armazenada no Metabanco de Dados. Uma fonte de dados local pode ser também um "Data Warehouse" (Servidor ROLAP).

O fato do Sistema BDRV integrar unicamente bancos de dados relacionais é justificado pela grande quantidade de fontes deste tipo inserida nas principais organizações. Bancos de dados baseados no modelo relacional se tornaram praticamente uma unanimidade no que diz respeito ao armazenamento de dados nas empresas.

- **Linguagem de Manipulação de Dados (DML)**

As subconsultas geradas pelo Integrador para as fontes de dados sendo integradas baseiam-se no SQL padrão.

- **Resultados**

Não existe necessidade do UG combinar os conjuntos-resposta oriundos das várias fontes de dados. Isto é feito automaticamente pelo componente da Arquitetura BDRV chamado Combinador de Resultados.

### **3.2.5 Interface**

O Sistema BDRV apresenta três tipos de interface, cada uma destinada a um tipo de usuário diferente - UAS, UDC e UG.

#### **Interface de Criação e Atualização do Metabanco de Dados**

É uma interface gráfica onde o UAS pode inserir novas fontes a serem integradas, remover fontes sendo integradas, criar esquemas globais e definir mapeamentos esquemas globais → esquemas locais. Através dela, é possível também inserir novos usuários, remover usuários existentes e modificar atributos pertencentes a usuários já cadastrados.

#### **Interface de Definição de Consultas**

É uma interface gráfica apresentada através da IDC, que utiliza uma LVDC para elaboração de consultas com base em esquemas globais. Através desta interface, o UDC pode construir suas consultas com simples cliques de "mouse". Para isto, deve selecionar tabelas e atributos virtuais, definir filtros, ordenações e/ou agrupamentos.

#### **Interface de Execução de Consultas**

É uma interface "Web" de seleção de consultas previamente cadastradas, e de apresentação de resultados integrados referentes à execução de uma ou mais subconsultas.

### 3.2.6 Configuração

Os seguintes objetos podem ser configurados por usuários com permissão administrativa:

- **Usuários:** é possível adicionar ou remover usuários, além de editar atributos de usuários já cadastrados ("login", senha, permissões de acesso);
- **Consultas:** deve ser possível inserir, remover e modificar consultas. Para a elaboração de uma consulta é também necessário fornecer os seguintes atributos: título, descrição e criador;
- **Fontes de Dados:** deve ser possível adicionar ou remover fontes de dados locais. Para inserir ou remover uma fonte de dados, é necessário atualizar os esquemas globais que referenciam a fonte em questão, e os possíveis mapeamentos esquema global → esquema local. Não nos preocuparemos com possíveis atualizações no Metabanco de Dados, oriundas de modificações no esquema das fontes de dados locais;
- **Esquemas Globais:** deve ser possível atualizar a estrutura dos esquemas globais, em virtude da adição ou remoção de bancos de dados locais;
- **Mapeamentos Esquemas Globais → Esquemas Locais:** deve ser possível a adição, remoção e alteração de mapeamentos esquemas globais → esquemas locais, em virtude da adição ou remoção de bancos de dados.

### 3.2.7 Plataformas

Para cada camada da arquitetura são necessários os seguintes requisitos:

#### Primeira Camada

O "browser Web" a ser utilizado é o "Internet Explorer" versão 3.0 ou superior.

#### Segunda Camada

- Integrador

O Integrador foi desenvolvido através de uma ferramenta de programação gráfica, mais especificamente Delphi 4.0. Escolhemos a mesma ferramenta para desenvolver a IDC.

- Metabanco de Dados

O SGBD escolhido para implementação do Metabanco de Dados é o SQL Server, versão 6.5.

- Servidor "Web"

Utilizaremos a tecnologia "Active Server Pages" (ASP) [37]. Assim, é possível a utilização de dois tipos de servidores "Web": "Internet Information Server" (IIS 4.0) e "Personal Web Server" (PWS). Isto se deve ao fato de que "scripts" ASP contidos em páginas "Web" são executados no servidor "Web" ("server-side"). Assim, o mesmo deve suportar a execução de tais "scripts".

- Sistema Operacional

O sistema BDRV deve funcionar em ambiente Windows 95, 98 ou NT.

### **Terceira Camada**

As fontes de dados locais são exclusivamente banco de dados relacionais ("Oracle", "SQL Server", "Interbase", "Access", entre outros).

# Capítulo 4

## *BDRV: Um Sistema Integrador de Bancos de Dados Relacionais - Visão Geral de Seu Funcionamento*

### 4.1 Introdução

A integração de bancos de dados relacionais é uma atividade extremamente complexa. O objetivo deste capítulo é mostrar, através de exemplos, os principais problemas enfrentados nas etapas necessárias à integração: construção de esquemas globais, elaboração e execução de consultas.

### 4.2 Objetivos do Sistema BDRV

O objetivo do Sistema Integrador de Múltiplos Bancos de Dados Relacionais, ou simplesmente Sistema BDRV, é integrar múltiplas fontes de dados, exclusivamente bancos de dados relacionais. Para tal, se faz necessária a utilização de técnicas que proporcionem:

- Transparência em relação à localização de fontes de dados locais - o Sistema BDRV proporciona meios de ocultar a real localização das fontes de dados locais. Os Usuários Globais (UGs) enxergam uma única fonte de dados para elaboração de consultas e obtenção de resultados;
- Modos de identificação de fontes que de fato contenham dados relevantes - uma vez elaboradas as consultas, o Sistema BDRV deve dispor de meios para

identificação das fontes de dados locais que de fato contenham as informações solicitadas, no intuito de transformar e redirecionar as consultas recebidas de forma apropriada;

- Formas de combinar resultados oriundos de fontes de dados sendo integradas, eliminando possíveis conflitos - no Sistema BDRV, a função principal de cada fonte de dados local é receber consultas e enviar resultados parciais. Antes de serem apresentados ao UG, possíveis conflitos entre os resultados parciais necessitam ser detectados e solucionados e, uma vez consistentes, os resultados parciais devem ser combinados.

No Sistema BDRV, utilizamos o conceito de **Banco de Dados Virtual (BDV)** [15], um banco de dados composto apenas do esquema, sem nenhum dado. Propomos ainda um mapeamento deste esquema para esquemas locais relacionais materializados (fontes de dados). Em resumo, enquanto bancos de dados materializados são formados por pares <esquema, dados>, BDVs são formados por pares <esquema, mapeamentos>. Para aplicações em que um único banco de dados não é suficiente, um BDV deve ser criado. Ele vai integrar os esquemas (ou parte deles) dos bancos de dados fontes relevantes em um único esquema global, que vai atender às aplicações globais.

Desta forma, o Sistema BDRV utiliza dois níveis de esquema: **esquemas globais**, ou simplesmente BDVs, que são esquemas relacionais, construídos a partir da integração de vários esquemas relacionais locais, servindo de base para elaboração de consultas; e os **esquemas locais**.

O Sistema BDRV oferece ainda a possibilidade de elaboração de consultas através de uma interface gráfica (ver detalhes na Seção 4.2.2), onde o Usuário Definidor de Consultas (UDC) realiza seleções de tabelas virtuais e atributos virtuais<sup>1</sup>, definição de filtros, ordenações e agrupamentos, com simples cliques de "mouse".

Podemos observar a arquitetura do Sistema BDRV na Figura 4.1(a). A mesma ainda pode ser detalhada como **Módulo de Definição de Consultas (MDC)** e **Módulo**

de **Execução de Consultas (MEC)**, com as funções de gerar e executar consultas, respectivamente. No caso do MEC, as consultas são executadas através de uma interface "Web". O MDC, por sua vez, pode ainda ser detalhado na Figura 4.1(b).

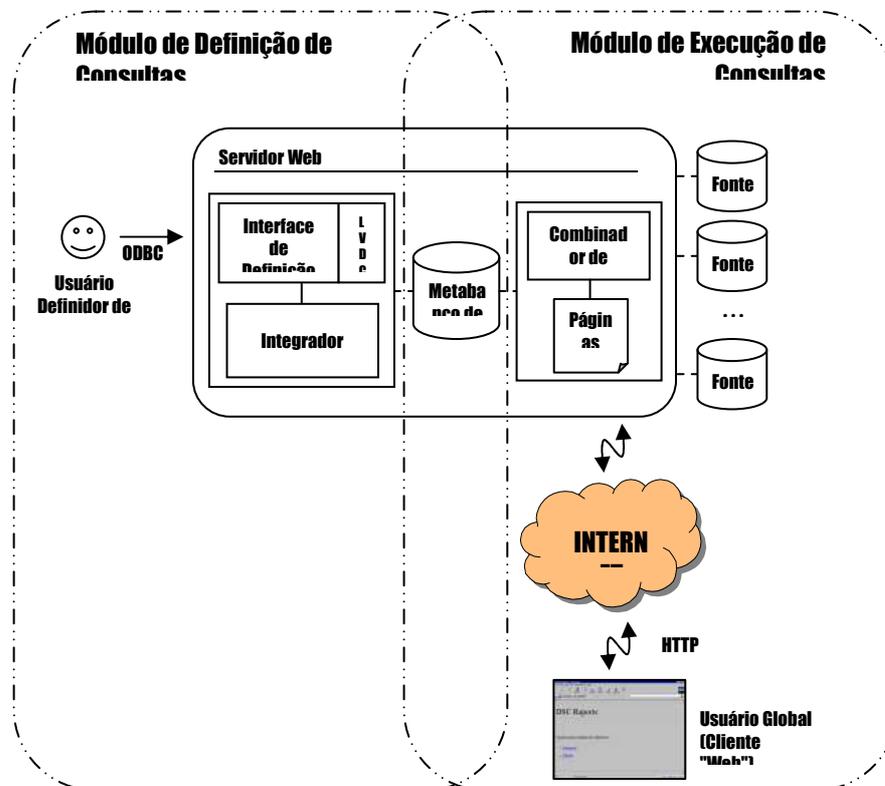


Figura 4.1(a) - Arquitetura do Sistema BDRV

A **Interface de Definição de Consultas (IDC)** é utilizada para elaboração de consultas a um esquema global, pelo UDC. Internamente, a IDC utiliza uma **Linguagem Visual de Definição de Consultas (LVDC)**, cuja função é descrever um conjunto de regras e opções que podem ser aplicadas na formulação de consultas.

Uma vez elaboradas, as consultas são enviadas ao **Integrador**, que tem a função de determinar e gerar os fragmentos de consulta (subconsultas) destinados aos bancos de dados locais. Em resumo, o Integrador seleciona os bancos de dados que de fato contenham os dados solicitados em uma consulta, gerando uma subconsulta para cada banco de dados escolhido.

<sup>1</sup> Chamaremos de tabela virtual e atributo virtual qualquer tabela e atributo, respectivamente, pertencente a um esquema global (BDV). De forma análoga, chamaremos pura e simplesmente de tabela e atributo qualquer tabela e atributo, respectivamente, pertencente a um esquema local.

O processo de transformação de uma consulta em subconsultas envolve uma série de etapas, conforme veremos em detalhes no Capítulo 5. Desta forma, o Integrador é composto ainda por outros componentes, conforme podemos observar na Figura 4.1(b). São eles:

- **Analisador de Consulta** - recebe conjuntos de tabelas virtuais, atributos virtuais, filtros, ordenações e agrupamentos, estruturando-os, para que possam vir a ser utilizados pelos outros componentes;
- **Coletor de Fontes** - identifica os bancos de dados que de fato contenham os dados solicitados em uma consulta;
- **Estimador de Fontes** - identifica os bancos de dados que melhor podem responder a uma consulta;
- **Seletor de Fontes** - decide para que bancos de dados devem ser geradas subconsultas;
- **Transformador de Consulta** - gera as subconsultas destinadas aos bancos de dados seleccionados pelo Seletor de Fontes.

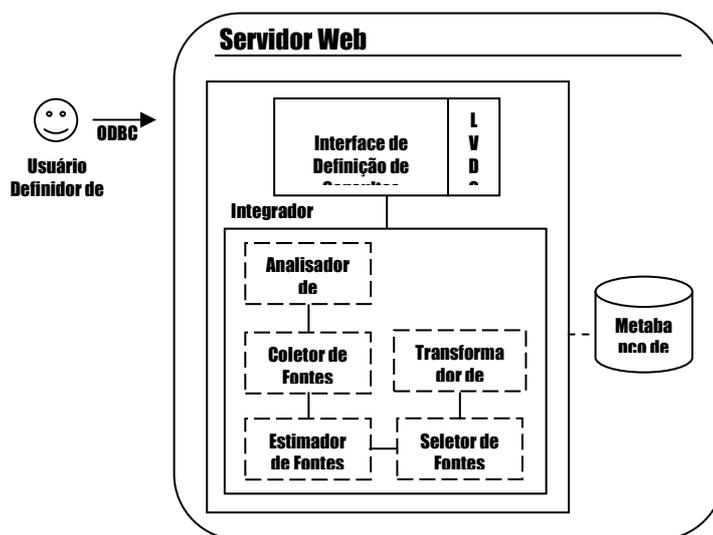


Figura 4.1(b) – Detalhamento do Módulo de Definição de Consultas

As consultas e suas respectivas subconsultas geradas pelo Integrador não são imediatamente redirecionadas aos bancos de dados locais, ficando armazenadas no **Metabanco de Dados** para posterior execução. São ainda funções do Metabanco de

Dados: armazenar a descrição dos bancos de dados locais; guardar a descrição de BDVs; manter mapeamentos BDVs → esquemas locais; armazenar informações para resolução de conflitos (ver a seguir).

No MEC, o UG pode executar as consultas armazenadas no Metabanco de Dados, através de uma interface "Web". Quando isto acontece, as subconsultas referentes à consulta escolhida são recuperadas no Metabanco de Dados e enviadas para serem executadas nos bancos de dados locais.

A função do **Combinador de Resultados** é receber os resultados retornados por cada banco de dados, detectar e remover possíveis conflitos existentes entre os resultados, integra-los e apresenta-los ao UG.

## 4.2 "Modus Operandi" do Sistema BDRV

Na seqüência, apresentaremos o "modus operandi" do Sistema BDRV, no que diz respeito aos Módulos de Definição e Execução de Consultas. Uma descrição mais precisa de cada um dos módulos será apresentada no Capítulo 5.

Inicialmente, ilustraremos o processo manual de construção de esquemas globais, integrando dois esquemas locais em um esquema global.

Em seguida, mostraremos como elaborar consultas ao esquema global construído.

Finalizando, descreveremos o processo de execução das consultas elaboradas anteriormente.

### 4.2.1 Construção de Um Esquema Global

Consideremos dois esquemas relacionais triviais. O primeiro deles, *Companhia*, é composto pelas tabelas: *Empregado*, *Departamento* e *Endereço* (Tabelas 4.2 a 4.4). O segundo, *Empresa*, é formado pelas tabelas: *Funcionario*, *Departamento*, *Projeto*, *FuncionarioTrabalhaProjeto*, *DepartamentoControlaProjeto* e *Endereço* (Tabelas 4.5 a 4.10).

## Esquema Relacional *Companhia*

Empregado				
Semântica		Armazena informações sobre os empregados da companhia		
Tabelas Referenciadas		Endereco e Departamento		
Nome	Tipo	Chave Estrangeira	Referência	Semântica
Cd_Empregado	Inteiro			Código do Empregado
Nm_Empregado	Char			Nome do Empregado
Nm_Est_Civil	Char			Estado Civil
Vl_Salario	Moeda			Salário Semanal
Nm_Cargo	Char			Cargo
Dt_Admissao	Data			Data de Admissão
Cd_Endereco	Inteiro	✓	Endereco.Cd_Endereco	Código do Endereço
Cd_Departamento	Inteiro	✓	Departamento.Cd_Departamento	Código do Departamento

Tabela 4.2 - Tabela *Empregado*

Departamento				
Semântica		Armazena informações sobre os departamentos da companhia		
Tabelas Referenciadas				
Nome	Tipo	Chave Estrangeira	Referência	Semântica
Cd_Departamento	Inteiro			Código do Departamento
Nm_Departamento	Char			Nome do Departamento
Nu_Telefone	Inteiro			Número do Telefone

Tabela 4.3 - Tabela *Departamento*

Endereco				
Semântica		Armazena o endereço dos empregados		
Tabelas Referenciadas				
Nome	Tipo	Chave Estrangeira	Referência	Semântica
Cd_Endereco	Inteiro			Código do Endereço
Nm_Rua	Char			Nome da Rua
Vl_Numero	Inteiro			Número da Residência
Nm_Bairro	Char			Nome do Bairro

Tabela 4.4 - Tabela *Endereco*

## Esquema Relacional *Empresa*

Funcionário				
Semântica		Armazena informações sobre os funcionários da empresa		
Tabelas Referenciadas		Departamento		
Nome	Tipo	Chave Estrangeira	Referência	Semântica
Cd_Funcionario	Inteiro			Código do Funcionário
Nm_Funcionario	Varchar			Nome do Funcionário
Nm_Est_Civil	Varchar			Estado Civil
Vl_Salario	Inteiro			Salário
Dt_Admissao	Data			Data de Admissão
Cd_Departamento	Inteiro	✓	Departamento.Cd_Departamento	Código do Departamento

Tabela 4.5 - Tabela *Funcionario*

Departamento				
Semântica		Armazena informações sobre os departamentos da empresa		
Tabelas Referenciadas		Endereco		
Nome	Tipo	Chave Estrangeira	Referência	Semântica
Cd_Departamento	Inteiro			Código do Departamento
Nm_Departamento	Varchar			Nome do Departamento
Nu_Do_Telefone	Varchar			Número do Telefone
Cd_Endereco	Inteiro	✓	Endereco.Cd_Endereco	Código do Endereço

Tabela 4.6 - Tabela *Departamento*

Projeto				
Semântica		Armazena informações sobre os projetos da empresa		
Tabelas Referenciadas				
Nome	Tipo	Chave Estrangeira	Referência	Semântica
Cd_Projeto	Inteiro			Código do Projeto
Nm_Projeto	Varchar			Nome do Projeto

Tabela 4.7 - Tabela *Projeto*

FuncionarioTrabalhaProjeto				
Semântica		Armazena informações sobre os funcionários da empresa que trabalham em projetos		
Tabelas Referenciadas		Funcionario e Projeto		
Nome	Tipo	Chave Estrangeira	Referência	Semântica
Cd_Funcionario	Inteiro	✓	Funcionario.Cd_Funcionario	Código do Funcionário
Cd_Projeto	Inteiro	✓	Projeto.Cd_Projeto	Código do Projeto

Tabela 4.8 - Tabela *FuncionarioTrabalhaProjeto*

DepartamentoControlaProjeto				
Semântica		Armazena informações sobre os departamentos que gerenciam projetos		
Tabelas Referenciadas		Departamento e Projeto		
Nome	Tipo	Chave Estrangeira	Referência	Semântica
Cd_Departamento	Inteiro	✓	Departamento.Cd_Departamento	Código do Departamento
Cd_Projeto	Inteiro	✓	Projeto.Cd_Projeto	Código do Projeto

Tabela 4.9 - Tabela *DepartamentoControlaProjeto*

Endereco				
Semântica		Armazena informações sobre os endereços dos departamentos		
Tabelas Referenciadas				
Nome	Tipo	Chave Estrangeira	Referência	Semântica
Cd_Endereco	Inteiro			Código do Endereço
Nm_Rua	Varchar			Nome da Rua
Vl_Numero	Varchar			Número
Nm_Bairro	Varchar			Nome do Bairro

Tabela 4.10 - Tabela *Endereco*

Como forma de integrarmos os esquemas locais *Companhia* e *Empresa*, utilizamos uma solução inspirada em [5]. Os passos para integração podem ser vistos a seguir:

**Passo 1:** Obtenção dos esquemas locais a serem integrados.

Inicialmente, deve ser obtido o nome de cada esquema local a ser integrado. Conforme podemos observar na Tabela 4.11, os esquemas são *Companhia* e *Empresa*.

Esquema
Companhia
Empresa

Tabela 4.11 - Esquemas Locais

**Passo 2:** Obtenção das tabelas de cada esquema local a ser integrado.

O segundo passo é obter as tabelas pertencentes a cada esquema local listado no passo anterior. Observe a Tabela 4.12.

Tabela	Esquema
Empregado	Companhia
Departamento	Companhia
Endereco	Companhia
Funcionario	Empresa
Departamento	Empresa
Projeto	Empresa
FuncionarioTrabalhaProjeto	Empresa
DepartamentoControlaProjeto	Empresa
Endereco	Empresa

Tabela 4.12 - Tabelas dos Esquemas Locais

**Passo 3:** Obtenção das tabelas sinônimas e homônimas.

Como exemplo de tabelas sinônimas temos *Empregado* e *Funcionario*, pertencentes aos esquemas *Companhia* e *Empresa*, respectivamente. Já como tabelas homônimas temos as tabelas *Endereco*, presentes em ambos os esquemas. No esquema *Companhia*, a tabela *Endereco* refere-se ao endereço dos empregados, enquanto que no esquema *Empresa*, refere-se ao endereço dos departamentos.

Nº	Tabela	Esquema	Tabela(s) Sinônima(s)	Tabela(s) Homônima(s)
1	Empregado	Companhia	{4}	-
2	Departamento	Companhia	-	-
3	Endereço	Companhia	-	{9}
4	Funcionário	Empresa	{1}	-
5	Departamento	Empresa	-	-
6	Projeto	Empresa	-	-
7	FuncionarioTrabalhaProjeto	Empresa	-	-
8	DepartamentoControlaProjeto	Empresa	-	-
9	Endereco	Empresa	-	{3}

Tabela 4.13 - Obtendo Tabelas Sinônimas e Homônimas

**Passo 4:** Resolução de Conflitos de Nome de Tabela.

Conforme podemos observar na Tabela 4.14, as tabelas sinônimas *Empregado* e *Funcionario*, pertencentes aos esquemas *Companhia* e *Empresa*, respectivamente, são mapeadas na tabela virtual *Empregado*.

A tabela *Endereco*, presente em ambos os esquemas, é mapeada em duas tabelas virtuais distintas, *EnderecoEmpregado* e *EnderecoDepartamento* (Tabela 4.14).

Tabela Virtual	{Esquema, Tabela}
Empregado	{{Companhia, Empregado}, {Empresa, Funcionario}}
EnderecoEmpregado	{{Companhia, Endereco}}
EnderecoDepartamento	{{Empresa, Endereco}}

Tabela 4.14 - Mapeamentos Tabelas Virtuais → Tabelas de Esquemas Locais

**Passo 5:** Mapeamento de tabelas sem conflito de nome.

As tabelas que não apresentam qualquer tipo de conflito de nome são mapeadas diretamente em tabelas virtuais. Por exemplo, a tabela *Projeto*, presente no esquema *Empresa* é mapeada diretamente na tabela virtual *Projeto*.

Tabela Virtual	{Esquema, Tabela}
Departamento	{{Companhia, Departamento}, {Empresa, Departamento}}
Projeto	{{Empresa, Projeto}}
EmpregadoTrabalhaProjeto	{{Empresa, FuncionarioTrabalhaProjeto}}
DepartamentoControlaProjeto	{{Empresa, DepartamentoControlaProjeto}}

Tabela 4.15 - Mapeamentos Tabelas Virtuais → Tabelas de Esquemas Locais

**Passo 6:** Obtenção dos atributos pertencentes às tabelas dos esquemas locais.

A Tabela 4.16 sintetiza todos os atributos das tabelas pertencentes aos esquemas *Companhia* e *Empresa*.

Atributo	Tabela	Esquema	Tipo	CE	Referência
Cd_Empregado	Empregado	Companhia	Inteiro		
Nm_Empregado	Empregado	Companhia	Char		
Nm_Est_Civil	Empregado	Companhia	Char		
Vl_Salario	Empregado	Companhia	Moeda		
Nm_Cargo	Empregado	Companhia	Char		
Dt_Admissao	Empregado	Companhia	Data		
Cd_Endereco	Empregado	Companhia	Inteiro	✓	Endereco.Cd_Endereco
Cd_Departamento	Empregado	Companhia	Inteiro	✓	Departamento.Cd_Departamento
Cd_Funcionario	Funcionario	Empresa	Inteiro		
Nm_Funcionario	Funcionario	Empresa	Varchar		
Nm_Est_Civil	Funcionario	Empresa	Varchar		
Vl_Salario	Funcionario	Empresa	Inteiro		
Dt_Admissao	Funcionario	Empresa	Data		
Cd_Departamento	Funcionario	Empresa	Inteiro	✓	Departamento.Cd_Departamento
Cd_Departamento	Departamento	Companhia	Inteiro		
Nm_Departamento	Departamento	Companhia	Char		
Nu_Telefone	Departamento	Companhia	Inteiro		
Cd_Departamento	Departamento	Empresa	Inteiro		
Nm_Departamento	Departamento	Empresa	Varchar		
Nu_Do_Telefone	Departamento	Empresa	Varchar		
Cd_Endereco	Departamento	Empresa	Inteiro	✓	Endereco.Cd_Endereco
Cd_Endereco	Endereco	Companhia	Inteiro		
Nm_Rua	Endereco	Companhia	Char		
Vl_Numero	Endereco	Companhia	Inteiro		
Nm_Bairro	Endereco	Companhia	Char		
Cd_Endereco	Endereco	Empresa	Inteiro		
Nm_Rua	Endereco	Empresa	Char		
Vl_Numero	Endereco	Empresa	Varchar		
Nm_Bairro	Endereco	Empresa	Char		
Cd_Projeto	Projeto	Empresa	Inteiro		
Nm_Projeto	Projeto	Empresa	Varchar		
Cd_Funcionario	FuncionarioTrabalhaProjeto	Empresa	Inteiro	✓	Funcionario.Cd_Funcionario
Cd_Projeto	FuncionarioTrabalhaProjeto	Empresa	Inteiro	✓	Projeto.Cd_Projeto
Cd_Departamento	DepartamentoControlaProjeto	Empresa	Inteiro	✓	Departamento.Cd_Departamento
Cd_Projeto	DepartamentoControlaProjeto	Empresa	Inteiro	✓	Projeto.Cd_Projeto

Tabela 4.16 - Atributos das Tabelas dos Esquemas Locais

**Passo 7:** Obtenção dos atributos sinônimos e homônimos.

Como exemplo de atributos sinônimos temos *Nm\_Empregado* e *Nm\_Funcionario*, pertencentes aos esquemas *Companhia* e *Empresa*, respectivamente. Já como atributos homônimos temos o atributo *Nm\_Rua*, presente na tabela *Endereco* contida em ambos os esquemas. No esquema *Companhia*, refere-se à rua onde residem os empregados, enquanto no esquema *Empresa* significa a rua onde estão localizados os departamentos.

Outro exemplo de atributos homônimos é o atributo *Vl\_Salario*, presente nas tabelas *Empregado* e *Funcionário* dos esquemas *Companhia* e *Empresa*, respectivamente. No esquema *Companhia*, *Vl\_Salario* refere-se ao salário semanal de um empregado, enquanto que no esquema *Empresa* descreve o salário diário de um funcionário.

Nº	Atributo	Tabela	Esquema	Atributo(s) Sinônimo(s)	Atributo(s) Homônimo(s)
1	Cd_Empregado	Empregado	Companhia	{9}	-
2	Nm_Empregado	Empregado	Companhia	{10}	-
3	Nm_Est_Civil	Empregado	Companhia	{11}	-
4	Vl_Salario	Empregado	Companhia	-	{12}
5	Nm_Cargo	Empregado	Companhia	-	-
6	Dt_Admissao	Empregado	Companhia	{13}	-
7	Cd_Endereco	Empregado	Companhia	-	-
8	Cd_Departamento	Empregado	Companhia	{14}	-
9	Cd_Funcionario	Funcionario	Empresa	{1}	-
10	Nm_Funcionario	Funcionario	Empresa	{2}	-
11	Nm_Est_Civil	Funcionario	Empresa	{3}	-
12	Vl_Salario	Funcionario	Empresa	-	{4}
13	Dt_Admissao	Funcionario	Empresa	{6}	-
14	Cd_Departamento	Funcionario	Empresa	{8}	-
15	Cd_Departamento	Departamento	Companhia	-	-
16	Nm_Departamento	Departamento	Companhia	-	-
17	Nu_Telefone	Departamento	Companhia	-	-
18	Cd_Departamento	Departamento	Empresa	-	-
19	Nm_Departamento	Departamento	Empresa	-	-
20	Nu_Do_Telefone	Departamento	Empresa	-	-
21	Cd_Endereco	Departamento	Empresa	-	-
22	Cd_Endereco	Endereco	Companhia	-	{26}
23	Nm_Rua	Endereco	Companhia	-	{27}
24	Vl_Numero	Endereco	Companhia	-	{28}
25	Nm_Bairro	Endereco	Companhia	-	{29}
26	Cd_Endereco	Endereco	Empresa	-	{22}
27	Nm_Rua	Endereco	Empresa	-	{23}
28	Vl_Numero	Endereco	Empresa	-	{24}
29	Nm_Bairro	Endereco	Empresa	-	{25}
30	Cd_Projeto	Projeto	Empresa	-	-
31	Nm_Projeto	Projeto	Empresa	-	-
32	Cd_Funcionario	FuncionarioTrabalhaProjeto	Empresa	-	-
33	Cd_Projeto	FuncionarioTrabalhaProjeto	Empresa	-	-
34	Cd_Departamento	DepartamentoControlaProjeto	Empresa	-	-
35	Cd_Projeto	DepartamentoControlaProjeto	Empresa	-	-

Tabela 4.17 - Atributos Sinônimos e Homônimos

**Passo 8:** Resolução de conflitos de nome de atributo e tipo de dados.

O objetivo desta etapa é resolver os conflitos de nome e tipo de dados dos atributos pertencentes aos esquemas locais.

### Conflitos de Nome

Os atributos sinônimos *Nm\_Empregado* e *Nm\_Funcionario*, pertencentes aos esquemas *Companhia* e *Empresa*, respectivamente, são mapeados no atributo virtual *Nm\_Empregado* (observe a Tabela 4.18).

Esquema Companhia		Esquema Empresa		Esquema Global	
Atributo	Tabela	Atributo	Tabela	Atributo Virtual	Tabela Virtual
Cd_Empregado	Empregado	Cd_Funcionario	Funcionario	Cd_Empregado	Empregado
Nm_Empregado	Empregado	Nm_Funcionario	Funcionario	Nm_Empregado	Empregado
Nm_Est_Civil	Empregado	Nm_Est_Civil	Funcionario	Nm_Est_Civil	Empregado
Dt_Admissao	Empregado	Dt_Admissao	Funcionario	Dt_Admissao	Empregado
Cd_Departamento	Empregado	Cd_Departamento	Funcionario	Cd_Departamento	Empregado
Cd_Departamento	Departamento	Cd_Departamento	Departamento	Cd_Departamento	Departamento
Nm_Departamento	Departamento	Nm_Departamento	Departamento	Nm_Departamento	Departamento
Nu_Telefone	Departamento	Nu_Do_Telefone	Departamento	Nu_Telefone	Departamento

Tabela 4.18 - Mapeamento Nome de Atributos Sinônimos → Nome de Atributos Virtuais

Conforme podemos observar na Tabela 4.19, o atributo homônimo *Nm\_Rua*, presente em ambos os esquemas, é mapeado nos atributos virtuais *Nm\_Rua\_Empregado* e *Nm\_Rua\_Departamento*, a serem inseridos nas tabelas virtuais *EnderecoEmpregado* e *EnderecoDepartamento*, respectivamente. Outro exemplo ocorre com o atributo *Vl\_Salario*, presente em ambos os esquemas. No esquema *Companhia*, o atributo deve ser mapeado no atributo virtual *Vl\_Salario\_Semanal*, enquanto que no esquema *Empresa* deve ser mapeado no atributo virtual *Vl\_Salario\_Diario*.

Esquema	Atributo	Tabela	Atributo Virtual	Tabela Virtual
Companhia	Vl_Salario	Empregado	Vl_Salario_Semanal	Empregado
Empresa	Vl_Salario	Funcionario	Vl_Salario_Diario	Empregado
Companhia	Cd_Endereco	Endereco	Cd_Endereco_Empregado	EnderecoEmpregado
Companhia	Nm_Rua	Endereco	Nm_Rua_Empregado	EnderecoEmpregado
Companhia	Vl_Numero	Endereco	Vl_Numero_Empregado	EnderecoEmpregado
Companhia	Nm_Bairro	Endereco	Nm_Bairro_Empregado	EnderecoEmpregado
Empresa	Cd_Endereco	Endereco	Cd_Endereco_Departamento	EnderecoDepartamento
Empresa	Nm_Rua	Endereco	Nm_Rua_Departamento	EnderecoDepartamento
Empresa	Vl_Numero	Endereco	Vl_Numero_Departamento	EnderecoDepartamento
Empresa	Nm_Bairro	Endereco	Nm_Bairro_Departamento	EnderecoDepartamento

Tabela 4.19 - Mapeamento Nome de Atributos Homônimos → Nome de Atributos Virtuais

### Conflitos de Tipo de Dados

O tipo de dado dos atributos sinônimos *Dt\_Admissao* (*Companhia*) *Dt\_Admissao* (*Empresa*), é Data. Portanto, no esquema global o tipo do atributo virtual que irá representá-los (neste caso, *Dt\_Admissao*), também será Data (observe a Tabela 4.20).

Esquema Companhia			Esquema Empresa			Esquema Global		
Atributo	Tabela	Tipo	Atributo	Tabela	Tipo	Atributo Virtual	Tabela Virtual	Tipo do Atributo Virtual
Cd_Empregado	Empregado	Inteiro	Cd_Funcionario	Funcionario	Inteiro	Cd_Empregado	Empregado	Inteiro
Nm_Empregado	Empregado	Char	Nm_Funcionario	Funcionario	Varchar	Nm_Empregado	Empregado	Varchar
Nm_Est_Civil	Empregado	Char	Nm_Est_Civil	Funcionario	Varchar	Nm_Est_Civil	Empregado	Varchar
Dt_Admissao	Empregado	Data	Dt_Admissao	Funcionario	Data	Dt_Admissao	Empregado	Data
Cd_Departamento	Empregado	Inteiro	Cd_Departamento	Funcionario	Inteiro	Cd_Departamento	Empregado	Inteiro
Cd_Departamento	Departamento	Inteiro	Cd_Departamento	Departamento	Inteiro	Cd_Departamento	Departamento	Inteiro
Nm_Departamento	Departamento	Char	Nm_Departamento	Departamento	Varchar	Nm_Departamento	Departamento	Varchar
Nu_Telefone	Departamento	Inteiro	Nu_Do_Telefone	Departamento	Varchar	Nu_Telefone	Departamento	Varchar

Tabela 4.20 - Mapeamento Tipo de Atributos Sinônimos → Tipo de Atributos Virtuais

Diferentemente, os atributos sinônimos *Nm\_Empregado* e *Nm\_Funcionario*, pertencentes aos esquemas *Companhia* e *Empresa*, respectivamente, são do tipo Char e Varchar (ver Tabela 4.20). O tipo de dado do atributo virtual que representa ambos no esquema global (*Nm\_Empregado*) é Varchar.

**Passo 9:** Mapeamento de atributos sem conflito de nome e de dado.

Conforme podemos observar na Tabela 4.21, o tipo de dado dos atributos virtuais que representam atributos sem conflitos de tipo mantém inalterado.

Esquema	Atributo	Tabela	Tipo	Atributo Virtual	Tabela Virtual	Tipo do Atributo Virtual
Companhia	VI_Salario	Empregado	Moeda	VI_Salario_Semanal	Empregado	Moeda
Empresa	VI_Salário	Funcionario	Moeda	VI_Salario_Diario	Empregado	Moeda
Companhia	Cd_Endereco	Endereco	Inteiro	Cd_Endereco_Empregado	EnderecoEmpregado	Inteiro
Companhia	Nm_Rua	Endereco	Char	Nm_Rua_Empregado	EnderecoEmpregado	Char
Companhia	VI_Numero	Endereco	Inteiro	VI_Numero_Empregado	EnderecoEmpregado	Inteiro
Companhia	Nm_Bairro	Endereco	Char	Nm_Bairro_Empregado	EnderecoEmpregado	Char
Empresa	Cd_Endereco	Endereco	Inteiro	Cd_Endereco_Departamento	EnderecoDepartamento	Inteiro
Empresa	Nm_Rua	Endereco	Varchar	Nm_Rua_Departamento	EnderecoDepartamento	Varchar
Empresa	VI_Numero	Endereco	Varchar	VI_Numero_Departamento	EnderecoDepartamento	Varchar
Empresa	Nm_Bairro	Endereco	Varchar	Nm_Bairro_Departamento	EnderecoDepartamento	Varchar
Companhia	Nm_Cargo	Empregado	Char	Nm_Cargo	Empregado	Char
Companhia	Cd_Endereco	Empregado	Inteiro	Cd_Endereco	Empregado	Inteiro
Empresa	Cd_Endereco	Departamento	Inteiro	Cd_Endereco	Departamento	Inteiro
Empresa	Cd_Projeto	Projeto	Inteiro	Cd_Projeto	Projeto	Inteiro
Empresa	Nm_Projeto	Projeto	Varchar	Nm_Projeto	Projeto	Varchar
Empresa	Cd_Funcionario	FuncionarioTrabalhaProjeto	Inteiro	Cd_Funcionario	FuncionarioTrabalhaProjeto	Inteiro
Empresa	Cd_Projeto	FuncionarioTrabalhaProjeto	Inteiro	Cd_Projeto	FuncionarioTrabalhaProjeto	Inteiro
Empresa	Cd_Departamento	DepartamentoControlaProjeto	Inteiro	Cd_Departamento	DepartamentoControlaProjeto	Inteiro
Empresa	Cd_Projeto	DepartamentoControlaProjeto	Inteiro	Cd_Projeto	DepartamentoControlaProjeto	Inteiro

Tabela 4.21 - Mapeamento de Atributos Sem Conflitos de Nome e de Dado

**Passo 10:** Criação do esquema global.

Após serem solucionados os conflitos, é chegada a hora de *montar* o esquema global, o qual chamaremos de *Organização*.

Utilizando as informações geradas nos passos 4 e 5, dispomos das tabelas virtuais a serem inseridas em nosso esquema global. As informações levantadas nos passos 8 e 9, por sua vez, nos fornecem o nome e o tipo de dado referente a cada atributo virtual.

### Gerando as Tabelas Virtuais

Tomando como base a Tabela 4.14, gerada no passo 4, observamos que a primeira tabela virtual a ser formada é *Empregado*. A partir das Tabelas 4.18 a 4.20, produzidas no passo 8, e 4.21, gerada no passo 9, obtemos nome e tipo de dado dos atributos virtuais pertencentes à tabela virtual *Empregado*.

Empregado	
Nome	Tipo
Cd_Empregado	Inteiro
Nm_Empregado	Varchar
Nm_Est_Civil	Varchar
Dt_Admissao	Data
Cd_Departamento	Inteiro
VI_Salario_Semanal	Moeda
VI_Salario_Diario	Moeda
Cd_Endereco	Inteiro

Tabela 4.22 - Tipo de Dado dos Atributos Virtuais da Tabela Virtual *Empregado*

As demais tabelas virtuais pertencentes ao nosso esquema global são:

Departamento	
Nome	Tipo
Cd_Departamento	Inteiro
Nm_Departamento	Varchar
Nu_Telefone	Varchar
Cd_Endereco	Inteiro

Tabela 4.23 - Tabela Virtual *Departamento*

Projeto	
Nome	Tipo
Cd_Projeto	Inteiro
Nm_Projeto	Varchar

Tabela 4.24 - Tabela Virtual *Projeto*

EmpregadoTrabalhaProjeto	
Nome	Tipo
Cd_Empregado	Inteiro
Cd_Projeto	Inteiro

Tabela 4.25 - Tabela Virtual *EmpregadoTrabalhaProjeto*

DepartamentoControlaProjeto	
Nome	Tipo
Cd_Departamento	Inteiro
Cd_Projeto	Inteiro

Tabela 4.26 - Tabela Virtual *DepartamentoControlaProjeto*

Endereço do Departamento	
Nome	Tipo
Cd_Endereco_Departamento	Inteiro
Nm_Rua_Departamento	Varchar
VI_Numero_Departamento	Varchar
Nm_Bairro_Departamento	Varchar

Tabela 4.27 - Tabela Virtual *EnderecoDepartamento*

Endereço do Empregado	
Nome	Tipo
Cd_Endereco_Empregado	Inteiro
Nm_Rua_Empregado	Varchar
VI_Numero_Empregado	Inteiro
Nm_Bairro_Empregado	Varchar

Tabela 4.28 - Tabela Virtual *EnderecoEmpregado*

## 4.2.2 Definição de Consulta

Nesta seção, mostraremos, através de uma série de exemplos, as etapas necessárias para que um Usuário Definidor de Consultas (UDC) possa elaborar consultas ao BDV. Em seguida, exibiremos o processamento envolvido na transformação de uma consulta em subconsultas a serem executadas nos bancos de

dados locais. Para facilitar, faremos uso dos diagramas de seqüência (Figuras 4.2(a) e 4.2(b)). Durante cada um dos exemplos faremos referências às várias fases dos diagramas.

### Diagramas de Seqüência de Interação Entre Componentes

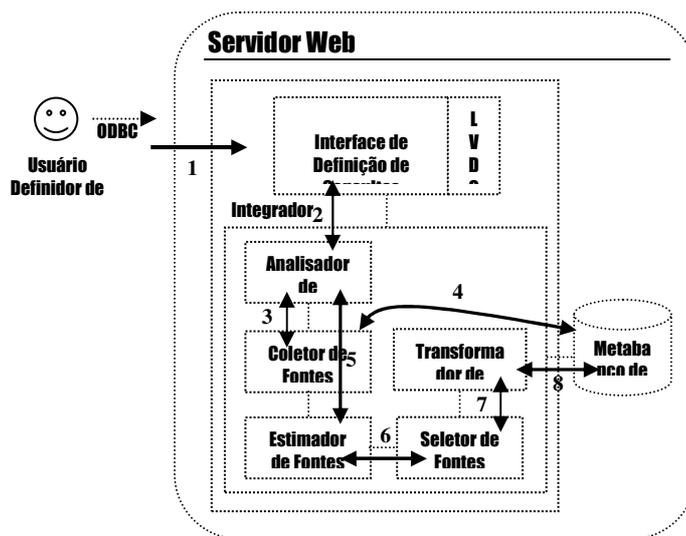


Figura 4.2(a) - Processo de Transformação de uma Consulta em Subconsulta(s) e Armazenamento de Subconsulta(s)

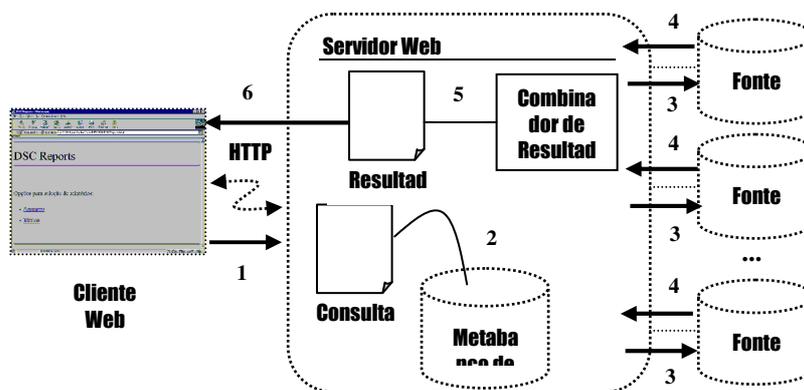


Figura 4.2(b) - Processo de Execução de Consulta

Apresentaremos, a partir de agora, uma seqüência de três exemplos em ordem crescente de complexidade. O conjunto de exemplos *cobre* praticamente os principais tipos de consulta que podem ser elaboradas. Detalharemos bastante o primeiro exemplo. As etapas de elaboração de consultas serão exibidas através das telas da Interface de Definição de Consultas (IDC).



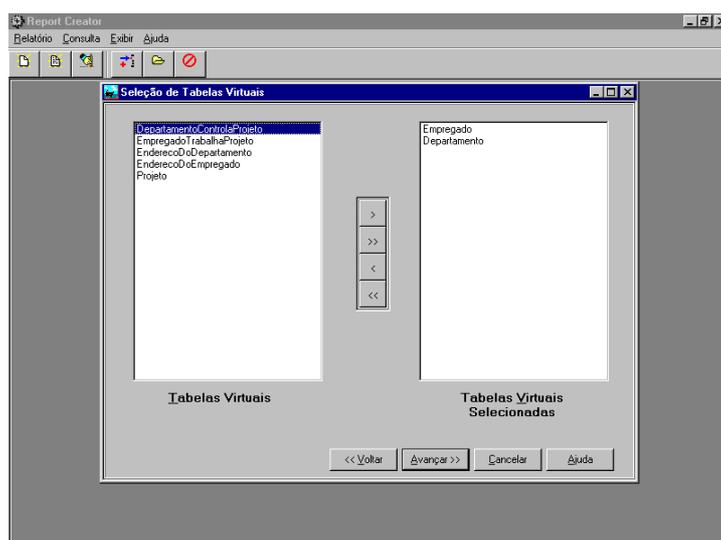


Figura 4.4 - Seleção das Tabelas Virtuais *Empregado* e *Departamento*

Conforme podemos observar na Figura 4.5, os atributos virtuais a serem selecionados são *Empregado.Nm\_Empregado*, *Empregado.Dt\_Admissao* e *Departamento.Nm\_Departamento*.

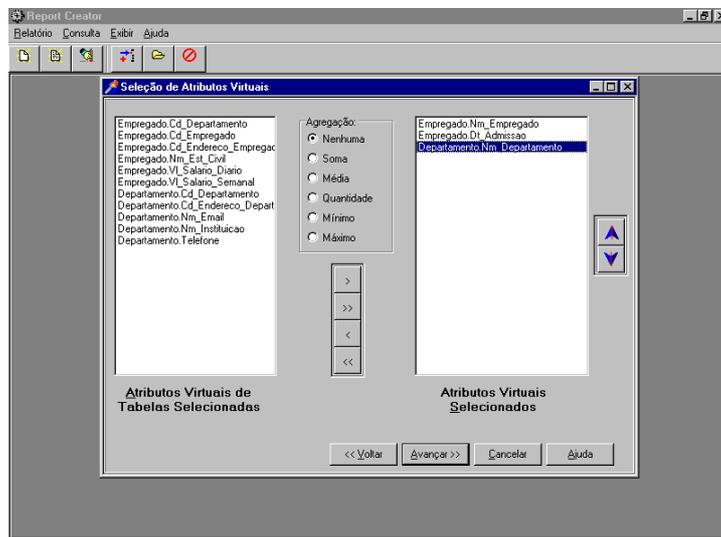


Figura 4.5 - Escolha dos Atributos Virtuais *Empregado.Nm\_Empregado*, *Empregado.Dt\_Admissao* e *Departamento.Nm\_Departamento*

A próxima etapa é a de definição de filtros nos atributos virtuais pertencentes às tabelas virtuais selecionadas anteriormente. Neste exemplo, todavia, esta etapa não será utilizada.

Na Figura 4.6, podemos observar a definição de ordenações nos atributos virtuais *Departamento.Nm\_Departamento*, *Empregado.Nm\_Empregado* e *Empregado.Dt\_Admissao*, nesta ordem, no intuito de que nosso conjunto-resposta final seja apresentado de forma ordenada.

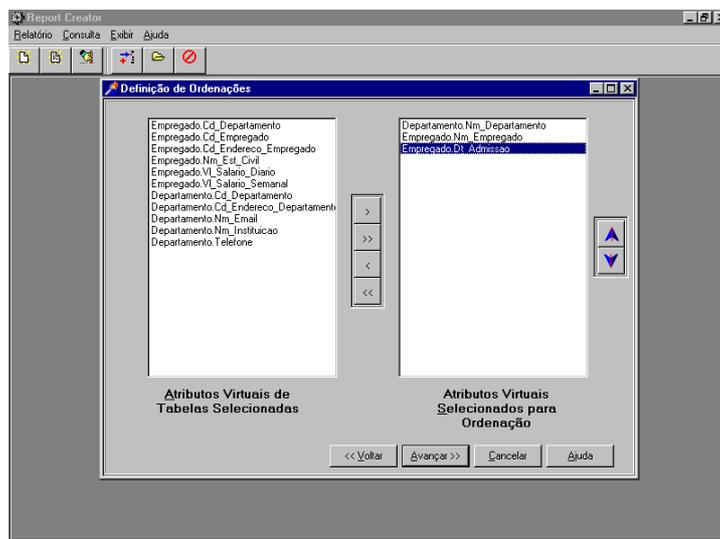


Figura 4.6 - Definição de Ordenações nos Atributos Virtuais *Departamento.Nm\_Departamento*, *Empregado.Nm\_Empregado* e *Empregado.Dt\_Admissao*

A próxima etapa é a de definição de agrupamentos sobre os atributos virtuais selecionados. Neste exemplo, contudo, esta etapa não será utilizada.

Finalizando, é necessário definir algumas informações sobre a consulta, como: título, descrição e nome do usuário.

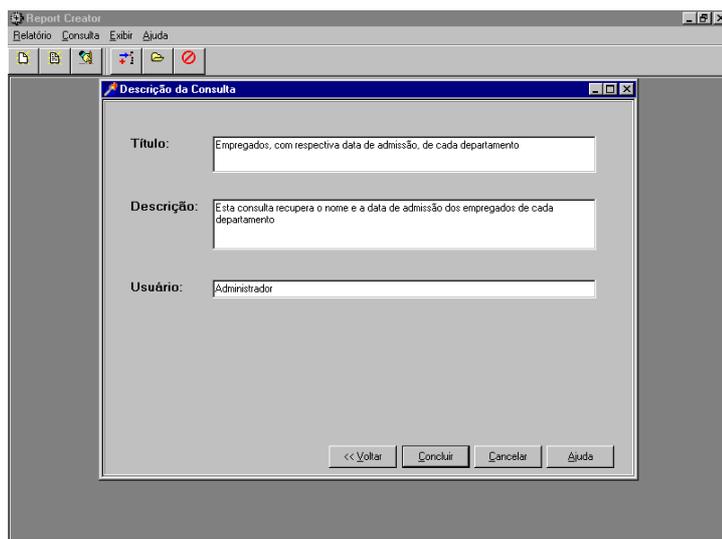


Figura 4.7 - Descrição da Consulta

O leitor mais atento pode estar perguntando se não é realizar uma equijunção entre os atributos *Departamento.Cd\_Departamento* e *Empregado.Cd\_Departamento*. A definição de equijunções é realizada internamente pela IDC, com base nas informações armazenadas no Metabanco de Dados.

Após a elaboração da consulta, a IDC repassa as seguintes informações ao Integrador, mas precisamente ao módulo Analisador de Consultas (Figura 4.2(a) (2)):

- Tabelas Virtuais = {Empregado, Departamento}
- Atributos Virtuais = {{Empregado, Nm\_Empregado, Ø}, {Empregado, Dt\_Admissao, Ø}, {Departamento, Nm\_Departamento, Ø}}
- Junções = {{Empregado, Cd\_Departamento, =, Departamento, Cd\_Departamento}}
- Filtros = Ø
- Ordenações = {{Departamento, Nm\_Departamento}, {Empregado, Nm\_Empregado}, {Empregado, Dt\_Admissao}}
- Agrupamentos = Ø

O componente **Analisador de Consultas** recebe conjuntos de Tabelas Virtuais, Atributos Virtuais, Junções, Filtros, Ordenações e Agrupamentos, e estrutura toda a informação recebida para que possa ser utilizada pelos componentes seguintes. O resultado deste trabalho é a geração das Tabelas 4.29 a 4.31.

A Tabela 4.29 contém dados referentes aos atributos virtuais incluídos em uma consulta. Nela, os campos Tabela Virtual, Atributo Virtual, Selecionado, Função, Junção, OrderBy e GroupBy indicam, respectivamente, nome da tabela virtual, nome do atributo virtual, se o atributo virtual será exibido no conjunto-resposta final, função de agregação (Quantidade, Soma, Valor Máximo, Valor Mínimo e Média) definida no atributo virtual, se existe alguma restrição definida no atributo virtual, se alguma ordenação está definida no atributo virtual e se algum agrupamento está definido no atributo virtual. O atributo virtual *Nm\_Empregado*, por exemplo, pertence à tabela virtual *Empregado*, será exibido no conjunto-resposta final e será utilizado para ordenar o mesmo conjunto.

Tabela Virtual	Atributo Virtual	Selecionado	Função	Filtro	Junção	OrderBy	GroupBy
Empregado	Nm_Empregado	✓				✓	
Empregado	Dt_Admissao	✓				✓	
Departamento	Nm_Departamento	✓				✓	
Departamento	Cd_Departamento				✓		
Empregado	Cd_Departamento				✓		

Tabela 4.29 - Atributos Virtuais

A Tabela 4.30 contém informações sobre restrições impostas aos atributos virtuais de uma consulta. Nela, os campos Atributo Virtual, Operador e Constante indicam, respectivamente, nome do atributo virtual, operador de comparação (igual, diferentes, maior, menor, maior do que, menor do que) utilizado na definição de uma restrição e valor literal, como número ou caractere. Neste exemplo, como nenhum filtro foi definido, a Tabela 4.33 não se encontra preenchida.

Tabela Virtual	Atributo Virtual	Operador	Constante

Tabela 4.30 - Filtros

A Tabela 4.31 contém informações sobre junções definidas entre atributos virtuais. Nela, os campos Tabela Virtual, Atributo Virtual, Operador, Tabela Virtual de

Junção e Atributo Virtual de Junção significam, respectivamente, nome da primeira tabela virtual de uma junção, nome do primeiro atributo virtual de uma junção, operador de comparação utilizado na definição de uma junção, nome da segunda tabela virtual de uma junção e nome do segundo atributo virtual de uma junção. De acordo com a Tabela 4.31, uma equijunção está definida entre os atributos virtuais *Cd\_Departamento*, pertencente à tabela virtual *Departamento*, e *Cd\_Departamento*, pertencente à tabela virtual *Empregado*.

Tabela Virtual	Atributo Virtual	Operador	Tabela Virtual de Junção	Atributo Virtual de Junção
Departamento	Cd_Departamento	=	Empregado	Cd_Departamento
Empregado	Cd_Departamento	=	Cd_Departamento	Departamento

Tabela 4.31 - Junções

O **Coletor de Fontes** recebe esses resultados do Analisador de Consultas (Figura 4.2(a) (3)) e seleciona as possíveis fontes que podem *atender* a cada par tabela e atributo virtual contido na Tabela 4.29, isto é, a(s) fonte(s) que contenha(m) tabelas e atributos, respectivamente, correspondentes ao par em questão, utilizando o Metabanco de Dados (Figura 4.2(a) (4)). O resultado gerado pelo componente é a Tabela 4.32. Nela, os campos Tabela Virtual, Atributo Virtual e Fonte significam, respectivamente, nome da tabela virtual, nome do atributo virtual e fonte que *atende* ao par tabela e atributo virtual. Conforme podemos notar, o par tabela virtual *Empregado* e atributo virtual *Nm\_Empregado*, por exemplo, é *atendido* tanto pelo banco de dados *Companhia* como pelo banco *Empresa*.

Tabela Virtual	Atributo Virtual	Fonte
Empregado	Nm_Empregado	Companhia
		Empresa
Empregado	Dt_Admissao	Companhia
		Empresa
Departamento	Nm_Departamento	Companhia
		Empresa
Departamento	Cd_Departamento	Companhia
		Empresa
Empregado	Cd_Departamento	Companhia
		Empresa

Tabela 4.32 - Fontes

O **Estimador de Fontes** recebe como entrada a Tabela 4.32 (Figura 4.2(a) (5)) e realiza um levantamento estatístico de cada fonte de dados que possui tabelas e atributos correspondentes aos selecionados no esquema global, através da divisão do número de atributos virtuais que uma determinada fonte possa *atender* pelo número de atributos virtuais selecionados em uma consulta. Este módulo também verifica se o conjunto de atributos *atendidos* por uma das fontes forma um subconjunto dos atributos *atendidos* pelas outras fontes.

Pela Tabela 4.32 é possível perceber que os bancos de dados *Companhia* e *Empresa* *atendem* a cinco de um total de cinco atributos virtuais solicitados, ou seja, 100% de atributos *atendidos*. Obviamente, o conjunto de atributos virtuais *atendidos* pelo banco de dados *Companhia* é o mesmo do conjunto de atributos virtuais *atendidos* pelo banco de dados *Empresa*.

O resultado produzido pelo Estimador de Fontes (Figura 4.2(a) (6)) pode ser visto na Tabela 4.33(a). Nela, os campos Fonte, Porcentagem e Subconjunto indicam, respectivamente, nome da fonte de dados, porcentagem de atributos virtuais *atendidos* por cada fonte e indicação se o conjunto de atributos virtuais *atendido* por uma fonte é um subconjunto dos atributos virtuais *atendidos* por outra fonte de dados. Por exemplo, o banco de dados *Companhia* *atende* a 100% dos atributos virtuais incluídos na consulta e o conjunto de atributos *atendido* pelo banco é um subconjunto dos atributos *atendidos* pelo banco de dados *Empresa*.

Fonte	Porcentagem	Subconjunto
Companhia	100,0%	✓
Empresa	100,0%	✓

Tabela 4.33(a) - Estatística

O módulo **Seletor de Fontes**, baseado nos resultados calculados pelo Estimador de Fontes, decide para que(quais) fonte(s) deve(m) ser geradas subconsultas (Figura 4.2(a) (6)). A escolha das fontes é baseada nas seguintes heurísticas:

- Totalidade: se uma fonte *atende* a todos os atributos virtuais incluídos em uma consulta, então a mesma deve ser escolhida e todas as outras ignoradas. Contudo, de acordo com as estatísticas levantadas pelo Estimador de Fontes

(ver Tabela 4.36 (a)), ambas as fontes *atendem* a todos os atributos virtuais. Portanto, as fontes *Companhia* e *Empresa* devem ser selecionadas;

- **Subconjunto:** se o conjunto de atributos *atendidos* por uma determinada fonte for um subconjunto dos atributos *atendidos* por outra fonte, então a primeira deve ser descartada. Todavia, o conjunto de atributos *atendidos* pelo banco *Companhia* é, na verdade, um subconjunto dos atributos virtuais *atendidos* pelo banco *Companhia*, e vice-versa. Novamente, selecionaremos ambas as fontes.

O resultado produzido pelo *Seletor de Fontes* (Figura 4.2(a) (7)) pode ser visualizado na Tabela 4.33(b). O campo *Selecionada* indica se uma fonte foi ou não escolhida para responder a uma consulta, isto é, se deve ou não ser gerada uma subconsulta para uma determinada fonte. Podemos ver que os banco de dados *Companhia* e *Empresa* foram selecionados para responder à consulta formulada anteriormente.

Fonte	Porcentagem	Subconjunto	Selecionada
Companhia	100,0%	✓	✓
Empresa	100,0%	✓	✓

Tabela 4.33(b) - Estatística

O **Transformador de Consultas**, de acordo com a(s) fonte(s) selecionada(s) pelo *Seletor de Fontes*, gera subconsultas apropriadas para cada fonte escolhida (Figura 4.2(a) (7)). Para tal, utiliza as informações armazenadas no *Metabanco de Dados* (Figura 4.2(a) (8)). O *Metabanco de Dados* também tem a função de armazenar as subconsultas geradas pelo *Integrador*. As subconsultas serão executadas posteriormente pelo *UG* no *MEC* (ver a seguir).

Continuando nosso exemplo, deve ser gerada uma subconsulta para cada banco de dados selecionado pelo *Seletor de Fontes*, *Companhia* e *Empresa*. A subconsulta referente ao banco de dados *Companhia* é:

```
SELECT Empregado.Nm_Empregado, Empregado.Dt_Admissao,  
Departamento.Nm_Departamento  
FROM Empregado, Departamento  
WHERE Departamento.Cd_Departamento = Empregado.Cd_Departamento  
ORDER BY Departamento.Nm_Departamento, Empregado.Nm_Empregado,  
Empregado.Dt_Admissao
```

A subconsulta gerada para o banco de dados *Empresa* é:

```
SELECT Funcionario.Nm_Funcionario, Funcionario.Dt_Admissao,  
Departamento.Nm_Departamento  
FROM Funcionario, Departamento  
WHERE Departamento.Cd_Departamento = Funcionario.Cd_Departamento  
ORDER BY Departamento.Nm_Departamento, Funcionario.Nm_Funcionario,  
Funcionario.Dt_Admissao
```

## Exemplo 2

Suponha agora que queremos elaborar a consulta: *Qual o total de empregados trabalhando em cada departamento que foram admitidos depois de 20/02/1997?*

Após selecionarmos o esquema global *Organização* (Figura 4.2(a) (1)), devemos escolher as tabelas virtuais que irão participar de nossa consulta: *Empregado* e *Departamento* (ver Figura 4.8).

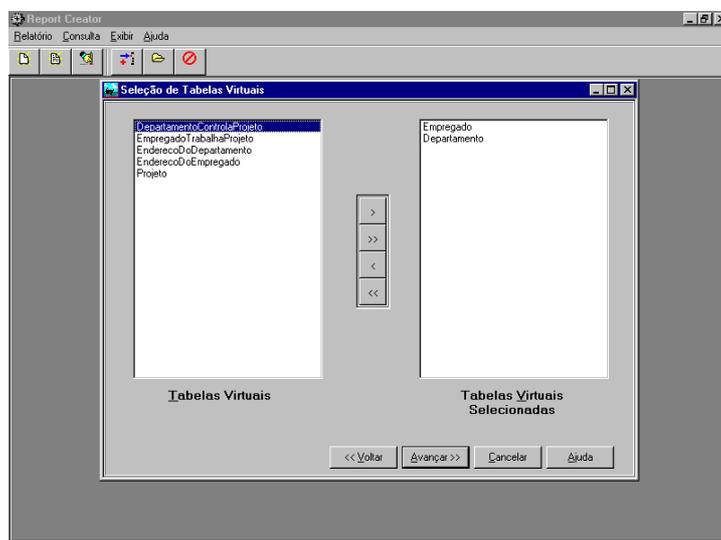


Figura 4.8 - Seleção das Tabelas Virtuais *Empregado* e *Departamento*

Conforme podemos observar na Figura 4.9, os atributos virtuais a serem selecionados são *Departamento.Nm\_Departamento* e *Empregado.Cd\_Empregado*, sendo que devemos associar a função de agregação *Quantidade* a este último.

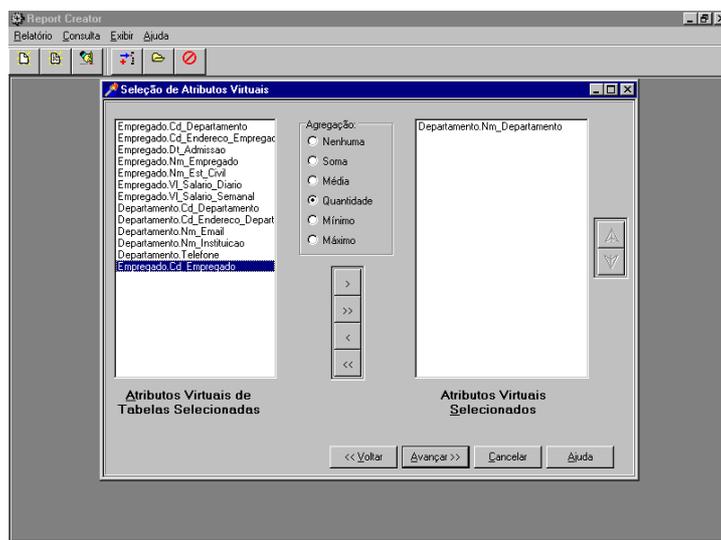


Figura 4.9 - Escolha dos Atributos Virtuais *Departamento.Nm\_Departamento* e *Quantidade(Empregado.Cd\_Empregado)*

A próxima etapa é a de definição de filtros em atributos virtuais pertencentes às tabelas virtuais *Empregado* e *Departamento*. Nossa consulta está interessada apenas nos empregados admitidos após 20/02/1997. Portanto, se faz necessário a definição de um filtro no atributo virtual *Empregado.Dt\_Admissao* (ver Figura 4.10).

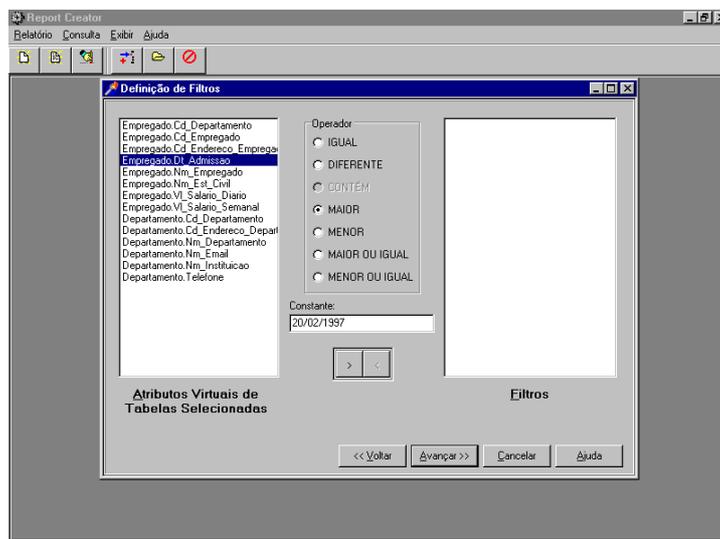


Figura 4.10 - Definição do Filtro *Empregados Admitidos Após 20/02/1997*

A próxima etapa é de a definição de ordenações. Ordenaremos nosso conjunto-resposta final pelo nome do departamento, como se pode observar na Figura 4.11.

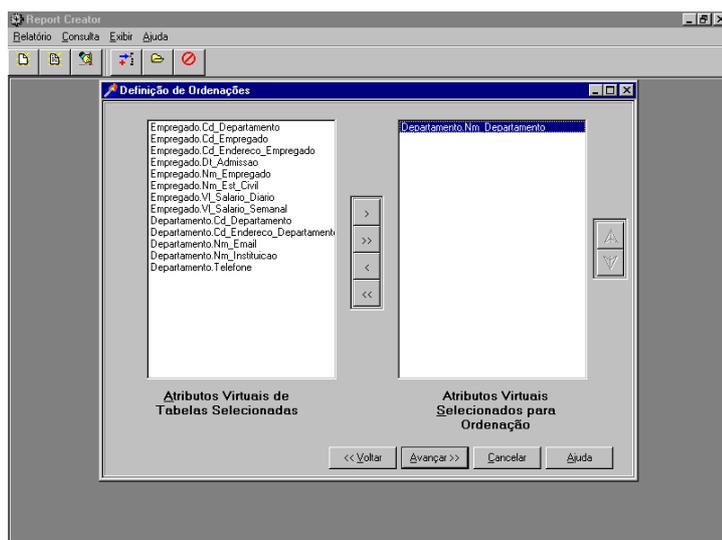


Figura 4.11 - Ordenando o Conjunto-Resposta Final pelo Nome do Departamento

Como queremos saber a quantidade de empregados por departamento, é necessário definir um agrupamento no atributo virtual *Departamento.Nm\_Departamento* (ver Figura 4.12).

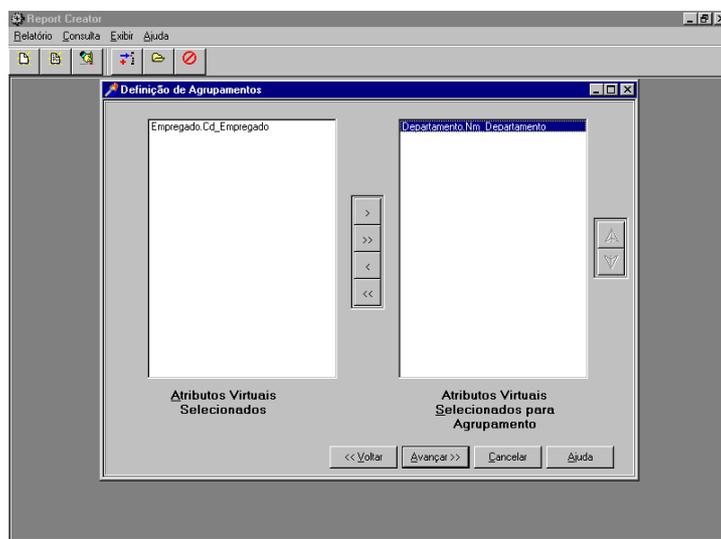


Figura 4.12 - Agrupando o Conjunto-Resposta Final por Departamento

Para finalizar, é necessário definir algumas informações sobre a consulta, como: título, descrição e nome do usuário (Figura 4.13).

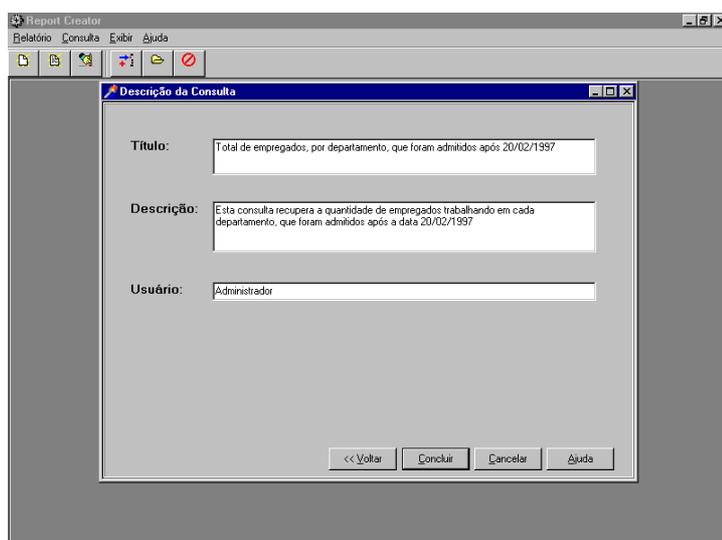


Figura 4.13 - Descrição da Consulta

A equijunção entre os atributos *Departamento.Cd\_Departamento* e *Empregado.Cd\_Departamento* é realizada internamente pela IDC, com base nas informações armazenadas no Metabanco de Dados.

Após a elaboração da consulta, a IDC repassa as seguintes informações ao Integrador, mas precisamente ao módulo Analisador de Consultas (Figura 4.2(a) (2)):

- Tabelas Virtuais = {Empregado, Departamento}
- Atributos Virtuais = {{Departamento, Nm\_Departamento, Ø}, {Empregado, Cd\_Empregado, Quantidade}}
- Junções = {{Departamento, Cd\_Departamento, =, Empregado, Cd\_Departamento}}
- Filtros = {{Empregado, Dt\_Admissao, >, 20/02/1997}}
- Ordenações = {{Departamento, Nm\_Departamento}}
- Agrupamentos = {{Departamento, Nm\_Departamento}}

O componente **Analisador de Consultas** recebe os conjuntos de Tabelas Virtuais, Atributos Virtuais, Junções, Filtros, Ordenações e Agrupamentos, e estrutura toda a informação recebida para que possa ser utilizada pelos componentes seguintes.

O resultado deste trabalho é a geração das Tabelas 4.34 a 4.36.

Tabela Virtual	Atributo Virtual	Selecionado	Função	Filtro	Junção	OrderBy	GroupBy
Departamento	Nm_Departamento	✓				✓	✓
Empregado	Cd_Empregado	✓	Quantidade		✓		
Empregado	Dt_Admissao			✓			
Departamento	Cd_Departamento				✓		
Empregado	Cd_Departamento				✓		

Tabela 4.34 - Atributos Virtuais

Tabela Virtual	Atributo Virtual	Operador	Constante
Empregado	Dt_Admissao	>	20/02/1997

Tabela 4.35 - Filtros

Tabela Virtual	Atributo Virtual	Operador	Tabela Virtual de Junção	Atributo Virtual de Junção
Departamento	Cd_Departamento	=	Empregado	Cd_Departamento
Empregado	Cd_Departamento	=	Cd_Departamento	Departamento

Tabela 4.36 - Junções

O **Coletor de Fontes** recebe esses resultados do Analisador de Consultas (Figura 4.2(a) (3)) e seleciona as possíveis fontes que podem *atender* a cada par tabela e atributo virtual contido na Tabela 4.34, utilizando o Metabanco de Dados (Figura 4.2(a) (4)). O resultado gerado pelo módulo é a Tabela 4.37.

Tabela Virtual	Atributo Virtual	Fonte
Departamento	Nm_Departamento	Companhia
		Empresa
Empregado	Cd_Empregado	Companhia
		Empresa
Empregado	Dt_Admissao	Companhia
		Empresa
Departamento	Cd_Departamento	Companhia
		Empresa
Empregado	Cd_Departamento	Companhia
		Empresa

Tabela 4.37 - Fontes

O **Estimador de Fontes** recebe como entrada a Tabela 4.37 (Figura 4.2(a) (5)) e produz como resultado a Tabela 4.38(a) (Figura 4.2(a) (6)).

Podemos perceber que os bancos de dados *Companhia* e *Empresa* *atendem* 100% dos atributos virtuais solicitados. Obviamente, o conjunto de atributos virtuais *atendidos* pelo banco de dados *Companhia* é o mesmo do conjunto de atributos virtuais *atendidos* pelo banco de dados *Empresa*.

Fonte	Porcentagem	Subconjunto
Companhia	100,0%	✓
Empresa	100,0%	✓

Tabela 4.38(a) - Estatística

De forma semelhante ao exemplo 1, o componente **Seleto de Fontes**, baseado nos resultados produzidos pelo Estimador de Fontes, decide que devem ser geradas

subconsultas para ambas as fontes (Figura 4.2(a) (6)). O resultado produzido pelo Seletor de Fontes (Figura 4.2(a) (7)) pode ser visualizado na Tabela 4.38(b).

Fonte	Porcentagem	Subconjunto	Selecionada
Companhia	100,0%	✓	✓
Empresa	100,0%	✓	✓

Tabela 4.38(b) - Estatística

O **Transformador de Consultas**, de acordo com a(s) fonte(s) selecionada(s) pelo Seletor de Fontes, gera subconsultas apropriadas para os bancos de dados *Companhia* e *Empresa* (Figura 4.2(a) (7)).

A subconsulta referente ao banco de dados *Companhia* é:

```
SELECT Departamento.Nm_Departamento, COUNT(Empregado.Cd_Empregado)
FROM Empregado, Departamento
WHERE Departamento.Cd_Departamento = Empregado.Cd_Departamento AND
Empregado.Dt_Admissao > 20/02/1997
GROUP BY Departamento.Nm_Departamento
ORDER BY Departamento.Nm_Departamento
```

A subconsulta gerada para o banco de dados *Empresa* é:

```
SELECT Departamento.Nm_Departamento, COUNT(Funcionario.Cd_Funcionario)
FROM Funcionario, Departamento
WHERE Departamento.Cd_Departamento = Funcionario.Cd_Departamento AND
Funcionario.Dt_Admissao > 20/02/1997
GROUP BY Departamento.Nm_Departamento
ORDER BY Departamento.Nm_Departamento
```

### Exemplo 3

Suponha agora que queremos elaborar a consulta: *Quais os empregados, com respectivos cargos, trabalhando em projetos controlados pelo departamento de Informática?*

Após selecionarmos o esquema global *Organização* (Figura 4.2(a) (1)), devemos escolher as tabelas virtuais que irão participar de nossa consulta: *Empregado*, *Departamento*, *Projeto*, *EmpregadoTrabalhaProjeto* e *DepartamentoControlaProjeto* (ver Figura 4.14).

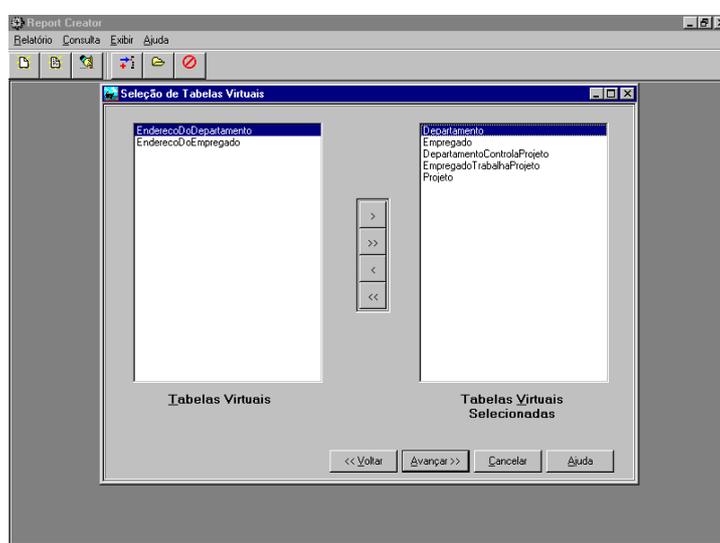


Figura 4.14 - Seleção das Tabelas Virtuais *Empregado*, *Departamento*, *Projeto*, *EmpregadoTrabalhaProjeto* e *DepartamentoControlaProjeto*

Os atributos virtuais a serem selecionados (Figura 4.15) são *Empregado.Nm\_Empregado*, *Empregado.Cargo* e *Projeto.Nm\_Projeto*.

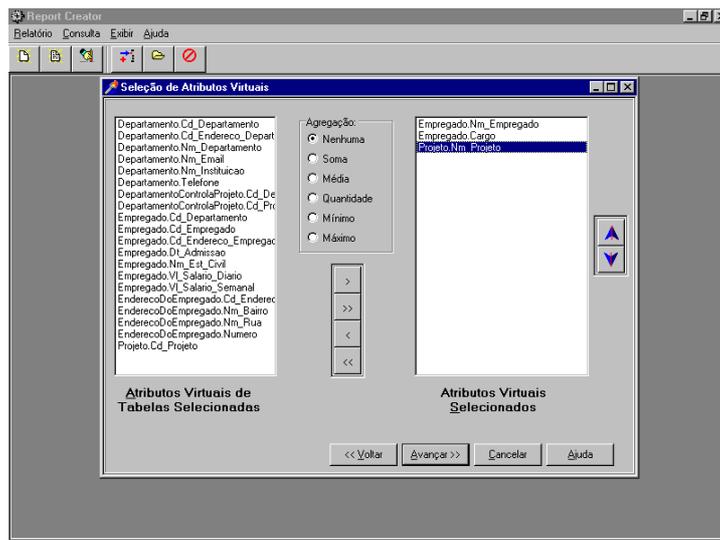


Figura 4.15 - Escolha dos Atributos Virtuais *Empregado.Nm\_Empregado*, *Empregado.Cargo* e *Projeto.Nm\_Projeto*

Como estamos interessados apenas nos empregados do departamento de Informática, devemos estabelecer um filtro no atributo virtual *Departamento.Nm\_Departamento* (ver Figura 4.16).

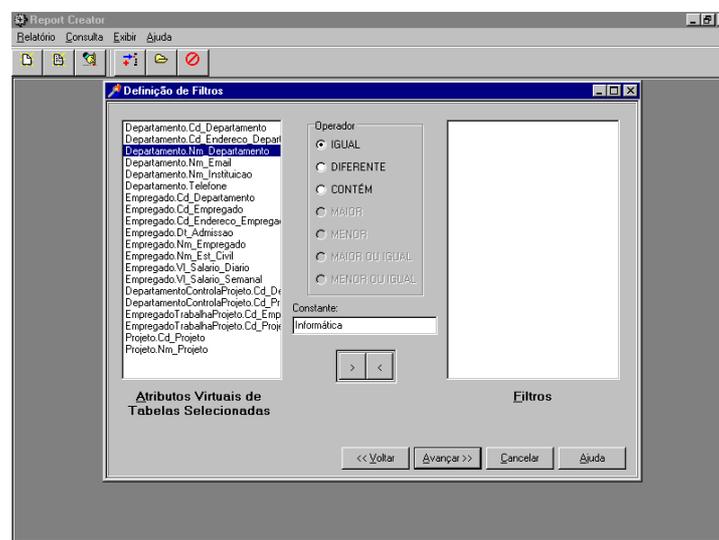


Figura 4.16 - Definição do Filtro *Departamento de Informática*

Conforme podemos observar na Figura 4.17, ordenaremos nosso conjunto-resposta pelos atributos virtuais *Projeto.Nm\_Projeto* e *Empregado.Nm\_Empregado*, nesta ordem.

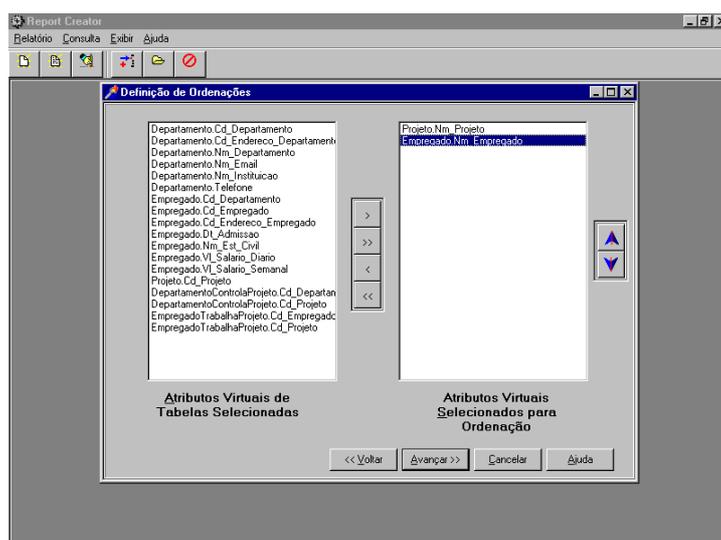


Figura 4.17 - Ordenando o Conjunto-Resposta Final pelos Atributos Virtuais *Projeto.Nm\_Projeto* e *Empregado.Nm\_Empregado*

Em razão da inexistência de agrupar o conjunto-resposta final, não utilizaremos a etapa de definição de agrupamentos neste exemplo.

Para finalizar, é necessário definir algumas informações sobre a consulta, como: título, descrição e nome do usuário (Figura 4.18).

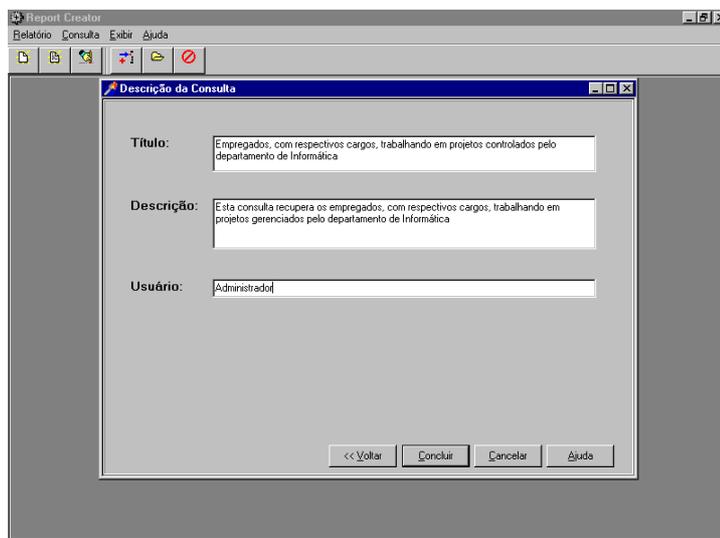


Figura 4.18 - Descrição da Consulta

As equijunções entre os atributos *Departamento.Cd\_Departamento* e *Empregado.Cd\_Departamento*, *Empregado.Cd\_Empregado* e

*EmpregadoTrabalhaProjeto.Cd\_Empregado*, *Projeto.Cd\_Projeto* e  
*EmpregadoTrabalhaProjeto.Cd\_Projeto*, *Projeto.Cd\_Projeto* e  
*DepartamentoControlaProjeto.Cd\_Projeto*, *DepartamentoControlaProjeto.Cd\_Departamento* e  
*Departamento.Cd\_Departamento* são realizadas internamente pela IDC, com base nas informações armazenadas no Metabanco de Dados.

Após a elaboração da consulta, a IDC repassa as seguintes informações ao Integrador, mas precisamente ao módulo Analisador de Consultas (Figura 4.2(a) (2)):

- Tabelas Virtuais = {Empregado, Departamento, Projeto, EmpregadoTrabalhaProjeto, DepartamentoControlaProjeto}
- Atributos Virtuais = {{Empregado, Nm\_Empregado, Ø}, {Empregado, Cargo, Ø}, {Projeto, Nm\_Projeto, Ø}}
- Junções = {{Departamento, Cd\_Departamento, =, Empregado, Cd\_Departamento}, {Empregado, Cd\_Empregado, =, EmpregadoTrabalhaProjeto, Cd\_Empregado}, {Projeto, Cd\_Projeto, =, EmpregadoTrabalhaProjeto, Cd\_Projeto}, {Projeto, Cd\_Projeto, =, DepartamentoControlaProjeto, Cd\_Projeto}, {DepartamentoControlaProjeto, Cd\_Departamento, =, Departamento, Cd\_Departamento}}
- Filtros = {{Departamento, Nm\_Departamento, =, Informática}}
- Ordenações = {{Projeto, Nm\_Projeto}, {Empregado, Nm\_Empregado}}
- Agrupamentos = Ø

O componente **Analisador de Consultas** recebe os conjuntos de Tabelas Virtuais, Atributos Virtuais, Junções, Filtros, Ordenações e Agrupamentos, e estrutura toda a informação recebida para que possa ser utilizada pelos componentes seguintes. O resultado deste trabalho é a geração das Tabelas 4.39 a 4.41.

Tabela Virtual	Atributo Virtual	Selecionado	Função	Filtro	Junção	OrderBy	GroupBy
Empregado	Nm_Empregado	✓				✓	
Empregado	Cargo	✓					
Projeto	Nm_Projeto	✓				✓	

Departamento	Nm_Departamento			✓			
Empregado	Cd_Departamento				✓		
Departamento	Cd_Departamento				✓		
Empregado	Cd_Empregado				✓		
EmpregadoTrabalhaProjeto	Cd_Empregado				✓		
Projeto	Cd_Projeto				✓		
EmpregadoTrabalhaProjeto	Cd_Projeto				✓		
DepartamentoControlaProjeto	Cd_Projeto				✓		
DepartamentoControlaProjeto	Cd_Departamento						

Tabela 4.39 - Atributos Virtuais

Tabela Virtual	Atributo Virtual	Operador	Constante
Departamento	Nm_Departamento	=	Informática

Tabela 4.40 - Filtros

Tabela Virtual	Atributo Virtual	Operador	Tabela Virtual de Junção	Atributo Virtual de Junção
Departamento	Cd_Departamento	=	Empregado	Cd_Departamento
Empregado	Cd_Departamento	=	Cd_Departamento	Departamento
Empregado	Cd_Departamento	=	EmpregadoTrabalhaProjeto	Cd_Empregado
EmpregadoTrabalhaProjeto	Cd_Empregado	=	Empregado	Cd_Departamento
Projeto	Cd_Projeto	=	EmpregadoTrabalhaProjeto	Cd_Projeto
EmpregadoTrabalhaProjeto	Cd_Projeto	=	Projeto	Cd_Projeto
Projeto	Cd_Projeto	=	DepartamentoControlaProjeto	Cd_Projeto
DepartamentoControlaProjeto	Cd_Projeto	=	Projeto	Cd_Projeto
DepartamentoControlaProjeto	Cd_Departamento	=	Departamento	Cd_Departamento
Departamento	Cd_Departamento	=	DepartamentoControlaProjeto	Cd_Departamento

Tabela 4.41 - Junções

O **Coletor de Fontes** recebe esses resultados do Analisador de Consultas (Figura 4.2(a) (3)) e seleciona as possíveis fontes que podem *atender* a cada par tabela e atributo virtual contido na Tabela 4.39, utilizando o Metabanco de Dados (Figura 4.2(a) (4)). O resultado gerado pelo módulo é a Tabela 4.42.

Tabela Virtual	Atributo Virtual	Fonte
Empregado	Nm_Empregado	Companhia
		Empresa
Empregado	Cargo	Companhia
Projeto	Nm_Projeto	Empresa
Departamento	Nm_Departamento	Companhia
		Empresa
Empregado	Cd_Departamento	Companhia
		Empresa
Departamento	Cd_Departamento	Companhia
		Empresa
Empregado	Cd_Empregado	Companhia
		Empresa
EmpregadoTrabalhaProjeto	Cd_Empregado	Empresa
Projeto	Cd_Projeto	Empresa
EmpregadoTrabalhaProjeto	Cd_Projeto	Empresa
DepartamentoControlaProjeto	Cd_Projeto	Empresa
DepartamentoControlaProjeto	Cd_Departamento	Empresa

Tabela 4.42 - Fontes

O **Estimador de Fontes** recebe como entrada a Tabela 4.42 (Figura 4.2(a) (5)), produzindo como resultado a Tabela 4.43(a) (Figura 4.2(a) (6)).

Fonte	Porcentagem	Subconjunto
Companhia	50%	
Empresa	92%	

Tabela 4.43(a) - Estatística

Concluimos pela Tabela 4.43(a) que o banco de dados *Companhia atende* a 50% dos atributos virtuais solicitados, enquanto o banco *Empresa atende* a aproximadamente 92% dos atributos. Percebemos também que o conjunto de atributos virtuais *atendidos* pelo banco de dados *Companhia* não é um subconjunto do conjunto de atributos virtuais *atendidos* pelo banco *Empresa* e vice-versa.

De forma semelhante aos exemplos 1 e 2, o componente **Seletor de Fontes**, baseado nos resultados produzidos pelo Estimador de Fontes, decide que devem ser geradas subconsultas para ambas as fontes (Figura 4.2(a) (6)), visto que nenhuma das duas fontes *atende* a 100% dos atributos virtuais solicitados e o conjunto de atributos *atendidos* por uma fonte não é subconjunto do da outra fonte. O resultado produzido pelo Seletor de Fontes (Figura 4.2(a) (7)) pode ser visualizado na Tabela 4.43(b).

Fonte	Porcentagem	Subconjunto	Selecionada
Companhia	50%		✓
Empresa	92%		✓

Tabela 4.43(b) - Estatística

O **Transformador de Consultas**, de acordo com a(s) fonte(s) selecionada(s) pelo Seletor de Fontes, gera subconsultas apropriadas para os bancos de dados *Companhia* e *Empresa* (Figura 4.2(a) (7)).

A subconsulta referente ao banco de dados *Companhia* é:

```
SELECT Empregado.Nm_Empregado, Empregado.Nm_Cargo
FROM Empregado, Departamento
WHERE Empregado.Cd_Departamento = Departamento.Cd_Departamento AND
Departamento.Nm_Departamento = 'Informática'
ORDER BY Empregado.Nm_Empregado
```

A subconsulta gerada para o banco de dados *Empresa* é:

```
SELECT Funcionario.Nm_Funcionario, Projeto.Nm_Projeto
FROM Funcionario, Departamento, Projeto, DepartamentoControlaProjeto,
FuncionarioTrabalhaProjeto
WHERE Funcionario.Cd_Departamento = Departamento.Cd_Departamento AND
Funcionario.Cd_Funcionario = FuncionarioTrabalhaProjeto.Cd_Funcionario AND
Projeto.Cd_Projeto = FuncionarioTrabalhaProjeto.Cd_Projeto AND Projeto.Cd_Projeto =
DepartamentoControlaProjeto.Cd_Projeto AND
DepartamentoControlaProjeto.Cd_Departamento = Departamento.Cd_Departamento AND
Departamento.Nm_Departamento = 'Informática'
ORDER BY Projeto.Nm_Projeto, Funcionario.Nm_Funcionario
```

## 4.2.3 Execução de Consulta

Vamos agora acompanhar o funcionamento do Módulo de Execução de Consultas, levando em consideração cada um dos exemplos da seção anterior.

### Exemplo 1

As subconsultas, produzidas pelo Integrador, ficam armazenadas no Metabanco de Dados, podendo ser visualizadas pelos UGs através de uma interface "Web".

Conforme podemos observar na Figura 4.19, para executar a consulta do exemplo 1, devemos selecionar a consulta intitulada de *Empregados, com respectiva data de admissão, de cada departamento* (Figura 4.2(b) (1)). Ao escolhermos a consulta, as subconsultas correspondentes à mesma, armazenadas no Metabanco de Dados, são recuperadas (Figura 4.2(b) (2)) e enviadas para serem executadas em suas respectivas fontes de dados (Figura 4.2(b) (3)), *Companhia* e *Empresa*, que retornam seus respectivos conjuntos-resposta (Figura 4.2(b) (4)).

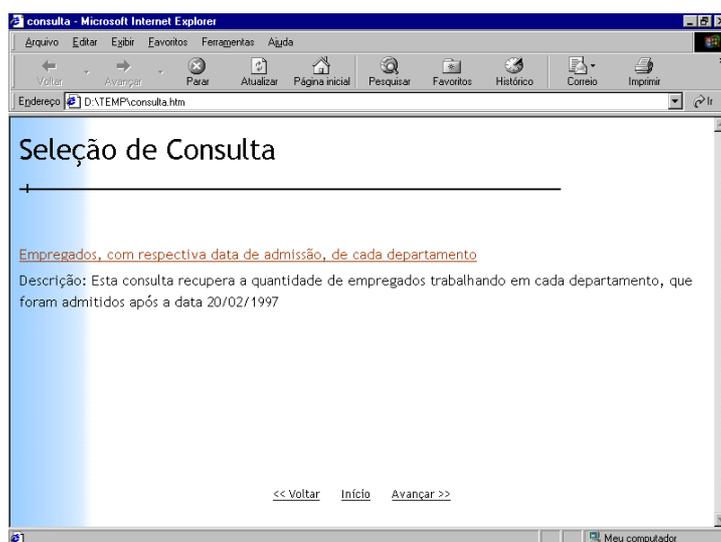


Figura 4.19 - Seleção da Consulta *Empregados, com respectiva data de admissão, de cada departamento*

O resultado referente ao banco de dados *Companhia* é o que podemos observar na Tabela 4.44.

Nm_Empregado	Dt_Admissao	Nm_Departamento
José S Campos	15/02/1999	Administração
Sérgio B Castro	14/02/1999	Administração
Marcelo M Correia	30/01/1994	Financeiro
Sandro F Carvalho	15/01/1996	Financeiro
Francisco M de Souza	15/09/1998	Informática
Jorge C Silva	16/11/1999	Informática
Manuela F de Lima	18/01/1997	Marketing
Paula A Andrade	22/02/1994	Marketing
Antônio L Farias	22/08/1997	Pessoal
Joana B de Souza	19/01/1998	Pessoal

Tabela 4.44 – Resultado Retornado pelo Banco de Dados *Companhia*

Na Tabela 4.45, podemos ver o resultado produzido pelo banco de dados *Empresa*.

Nm_Funcionario	Dt_Admissao	Nm_Departamento
Marcelo Morais Correia	30/01/1994	Financeiro
Sandro Ferreira Carvalho	15/01/1996	Financeiro
Francisco Maturana de Souza	15/09/1998	Informática
Jorge Castro Silva	16/11/1999	Informática
Manuela Farias de Lima	18/01/1997	Marketing
Paula Almeida Andrade	22/02/1994	Marketing

Tabela 4.45 – Resultado Retornado pelo Banco de Dados *Empresa*

O **Combinador de Resultados** tem a função de integrar os conjuntos-resposta resultantes da execução de subconsultas nas fontes de dados locais, no intuito de serem apresentados ao UG, eliminando qualquer tipo de conflito.

Neste caso, foram encontrados três tipos de conflito. São eles:

- Conflito de tipo de dado: os atributos *Nm\_Empregado* e *Nm\_Departamento*, pertencentes ao conjunto-resposta retornado pelo banco de dados *Companhia*, são do tipo Char, enquanto os atributos *Nm\_Funcionario* e *Nm\_Departamento*, pertencente ao conjunto-resposta produzido pelo banco *Empresa*, são do tipo Varchar. É necessária a utilização de uma função de conversão nos dois primeiros atributos (*Nm\_Empregado* e *Nm\_Departamento*). Ambos devem ser convertidos de Char para Varchar;
- Conflito de expressões diferentes: o empregado *Jorge Castro Silva* é identificado como *Jorge C Silva* no banco de dados *Companhia* e como *Jorge Castro Silva* no banco *Empresa*. Desta forma, deve ser realizado um mapeamento de valores com base nas informações contidas no Metabanco de Dados, isto é, neste caso, o dado *Jorge C Silva* deve ser mapeado em *Jorge Castro Silva*;
- Conflito de sinonímia de atributos: os atributos *Nm\_Empregado* e *Nm\_Funcionario*, pertencentes aos bancos de dados *Companhia* e *Empresa*, respectivamente, são sinônimos (ver Tabela 4.17). O atributo virtual que representa ambos no esquema global é *Nm\_Empregado*. Portanto, o atributo *Nm\_Funcionario* deve ser mapeado para *Nm\_Empregado*.

Após a detecção e eliminação de conflitos pelo Combinador de Resultados é necessário ainda combinar os conjuntos-resposta. Somente devem ser apresentados os atributos selecionados na consulta. Os atributos devem ser ordenados e/ou agrupados de acordo como indica a consulta. Assim, o resultado final apresentado ao UG é o que se pode observar na Tabela 4.46.

Nm_Empregado	Dt_Admissao	Nm_Departamento
José S Campos	15/02/1999	Administração
Sérgio B Castro	14/02/1999	Administração
Marcelo Morais Correia	30/01/1994	Financeiro
Sandro Ferreira Carvalho	15/01/1996	Financeiro
Francisco Maturana de Souza	15/09/1998	Informática
Jorge Castro Silva	16/11/1999	Informática
Manuela Farias de Lima	18/01/1997	Marketing
Paula Almeida Andrade	22/02/1994	Marketing
Antônio L Farias	22/08/1997	Pessoal
Joana B de Souza	19/01/1998	Pessoal

Tabela 4.46 - Resultado Final da Consulta Após a Eliminação de Conflitos

## Exemplo 2

Conforme podemos observar na Figura 4.20, para executar a consulta do exemplo 2, devemos selecionar a consulta intitulada de *Total de empregados, por departamento, que foram admitidos após 20/02/1997* (Figura 4.2(b) (1)). Ao escolhermos a consulta, as subconsultas correspondentes à mesma, armazenadas no Metabanco de Dados, são recuperadas (Figura 4.2(b) (2)) e enviadas para serem executadas em suas respectivas fonte de dados (Figura 4.2(b) (3)), *Companhia* e *Empresa*, que retornam seus conjuntos-resposta (Figura 4.2(b) (4)).

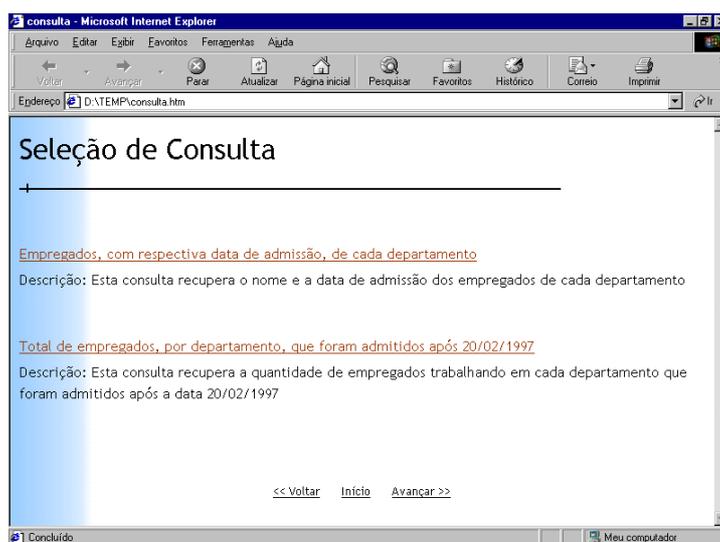


Figura 4.20 - Seleção da Consulta *Total de Empregados, por departamento, que foram admitidos após 20/02/1997*

O resultado referente ao banco de dados *Companhia* é o que podemos observar na Tabela 4.47.

Nm_Departamento	Total
Administração	2
Financeiro	0
Informática	2
Marketing	0
Pessoal	2

Tabela 4.47 – Resultado Retornado pelo Banco de Dados *Companhia*

Na Tabela 4.48, podemos ver o resultado produzido pelo banco de dados *Empresa*.

Nm_Departamento	Total
Financeiro	0
Informática	2
Marketing	0

Tabela 4.48 – Resultado Retornado pelo Banco de Dados *Empresa*

Neste exemplo, o componente **Combinador de Resultados** detecta e resolve um único tipo de conflito antes de combinar os resultados:

- Conflito de tipo de dado: o atributo *Nm\_Departamento*, pertencente ao conjunto-resposta retornado pelo banco de dados *Companhia*, é do tipo Char, enquanto o atributo *Nm\_Departamento*, pertencente ao conjunto-resposta produzido pelo banco *Empresa*, é do tipo Varchar. É necessária a utilização de uma função de conversão no primeiro atributo, no intuito de convertê-lo de Char para Varchar;

O resultado final pode se observado na Tabela 4.49:

Nm_Departamento	Total
Administração	2
Financeiro	0
Informática	2
Marketing	0
Pessoal	2

Tabela 4.49 - Resultado Final da Consulta Após a Eliminação de Conflitos

### Exemplo 3

Conforme podemos observar na Figura 4.21, para executar a consulta do exemplo 3, devemos selecionar a consulta intitulada de *Empregados, com respectivos cargos, trabalhando em projetos controlados pelo departamento de informática* (Figura 4.2(b) (1)). Ao escolhermos a consulta, as subconsultas correspondentes à mesma, armazenadas no Metabanco de Dados, são recuperadas (Figura 4.2(b) (2)) e enviadas para serem executadas em suas respectivas fonte de dados (Figura 4.2(b) (3)), *Companhia* e *Empresa*, que retornam seus conjuntos-resposta (Figura 4.2(b) (4)).

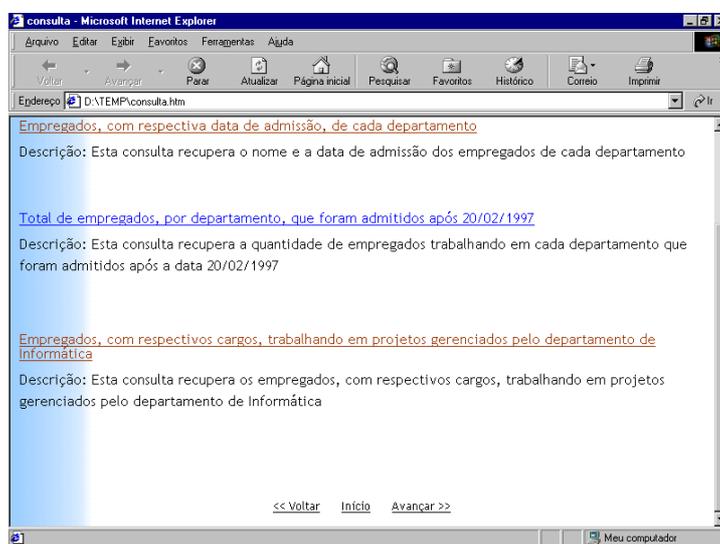


Figura 4.21 - Seleção da Consulta *Empregados, com respectivos cargos, trabalhando em projetos gerenciados pelo departamento de Informática*

O resultado referente ao banco de dados *Companhia* é o que podemos observar na Tabela 4.50.

Nm_Empregado	Cargo
Francisco M de Souza	DBA
Jorge C Silva	Desenvolvedor

Tabela 4.50 – Resultado Retornado pelo Banco de Dados *Companhia*

Na Tabela 4.51, podemos ver o resultado produzido pelo banco de dados *Empresa*.

Nm_Funcionario	Nm_Projeto
Francisco Maturana de Souza	Alpha 1
Jorge Castro Silva	Alpha 1

Tabela 4.51 – Resultado Retornado pelo Banco de Dados *Empresa*

Neste exemplo, o componente **Combinador de Resultados** detecta e resolve três diferentes tipos de conflito antes de combinar os resultados:

- Conflito de tipo de dado: os atributos sinônimos (ver Tabela 4.17) *Nm\_Empregado*, pertencente ao banco de dados *Companhia*, e *Nm\_Funcionario*, pertencente ao banco *Empresa*, são do tipo Char e Varchar, respectivamente. É necessária então a utilização de uma função de conversão no primeiro atributo. O mesmo deve ser convertido de Char para Varchar;
- Conflito de expressões diferentes: o empregado *Francisco Maturana de Souza* é identificado como *Francisco M de Souza* no banco de dados *Companhia* e como *Francisco Maturana de Souza* no banco *Empresa*. Desta forma deve ser realizado um mapeamento de valores com base nas informações contidas no Metabanco de Dados, isto é, neste caso, o dado *Francisco M de Souza* deve ser mapeado em *Francisco Maturana de Souza*;
- Conflito de sinonímia de atributos: os atributos *Nm\_Empregado* e *Nm\_Funcionario*, pertencentes aos bancos de dados *Companhia* e *Empresa*, respectivamente, são sinônimos (ver Tabela 4.17). O atributo virtual que representa ambos no esquema global é *Nm\_Empregado*. Portanto, o atributo *Nm\_Funcionario* deve ser mapeado para *Nm\_Empregado*.

É necessária ainda a combinação dos conjuntos-resposta pelo Combinador de Resultados. O conjunto-resposta final pode se observado na Tabela 4.52.

Nm_Empregado	Cargo	Nm_Projeto
Francisco Maturana de Souza	DBA	Alpha 1
Jorge Castro Silva	Desenvolvedor	Alpha 1

Tabela 4.52 - Resultado Final da Consulta Após a Eliminação de Conflitos

## 4.3 Conclusões

Neste capítulo, vimos que a construção de um esquema global não é uma tarefa trivial, em virtude da presença de conflitos entre os vários esquemas locais. Vimos também como uma consulta a um esquema global, na Arquitetura BDRV, é transformada em subconsultas a serem executadas em suas respectivas fontes de dados locais. Após serem executadas, as subconsultas ainda continham conflitos que também necessitavam de serem resolvidos. No Capítulo 5, trataremos do projeto e implementação do Sistema BDRV, incluindo as estruturas de dados genéricas e os algoritmos que recuperam dados das estruturas.

No Anexo A, o leitor pode encontrar um estudo de caso, onde três bancos de dados, localizados no DSC, são integrados. É possível também observar alguns exemplos de elaboração e execução de consultas.

# Capítulo 5

## *Projeto e Implementação do Sistema BDRV*

### 5.1 Introdução

Conforme pudemos observar nos capítulos anteriores, o Sistema BDRV é composto pelos Módulos de Definição de Consultas (MDC) e de Execução de Consultas (MEC).

No MDC, as consultas aos Bancos de Dados Virtuais (BDVs) são elaboradas pelo Usuário Definidor de Consultas (UDC) através da Interface de Definição de Consultas (IDC). Uma vez definidas, as consultas são repassadas ao Integrador que, por sua vez, gera subconsultas a serem executadas posteriormente nos bancos de dados locais. Neste módulo, o Metabanco de Dados desempenha uma tripla função: guardar a descrição dos BDVs esquemas locais, utilizadas pela IDC; manter mapeamentos BDVs → esquemas locais, utilizados pelo Integrador; e, por fim, armazenar as consultas e respectivas subconsultas geradas pelo Integrador.

No MEC, as consultas geradas no MDC são disponibilizadas para serem executadas pelo Usuário Global (UG). Uma vez selecionada, as subconsultas correspondentes são recuperadas no Metabanco de Dados e enviadas para serem executadas diretamente em suas respectivas fontes de dados. O Combinador de Resultados então recebe os vários conjuntos-resposta, detecta/resolve possíveis conflitos entre os mesmos, e integra-os, apresentando o resultado final através de uma interface "Web". Neste módulo, o Metabanco de Dados desempenha as seguintes funções: manter as consultas e suas respectivas subconsultas; e armazenar informações para resolução de conflitos.

No Capítulo 4, apresentamos uma idéia do funcionamento geral do Sistema BDRV, isto é, como os componentes de cada módulo estão interligados. Em resumo, mostramos uma visão do Sistema BDRV ao nível de usuário.

O objetivo deste capítulo é detalhar individualmente os componentes de cada módulo do Sistema BDRV, sua(s) função(ões), entrada(s) e saída(s), e algoritmo(s). Analisaremos o funcionamento de cada componente de forma seqüencial, utilizando o terceiro exemplo do capítulo anterior.

## 5.2 Metabanco de Dados

O Metabanco de Dados, um banco de dados relacional, é um componente que participa de ambos os módulos do Sistema BDRV.

No MDC, o Metabanco de Dados desempenha as seguintes funções:

- Guardar a descrição de BDVs e esquemas locais, para uso pela IDC;
- Manter mapeamentos BDVs → esquemas locais, a serem utilizados pelo Integrador;
- Armazenar as consultas, geradas pela IDC, e as subconsultas, estas geradas pelo Integrador.

No MEC a função do Metabanco de Dados é armazenar informações a serem utilizadas na solução de possíveis tipos de conflitos, pelo Combinador de Resultados.

Na Figura 5.1, apresentamos o esquema geral do Metabanco de Dados. As setas indicam a existência de referência de integridade entre atributos pertencentes às tabelas interligadas. A origem da seta significa o atributo que referencia, enquanto a parte final indica o atributo que é referenciado. O símbolo *jogo-da-velha* (#) indica que o atributo é chave primária da tabela. Já o símbolo *asterisco* significa que o atributo é chave estrangeira.

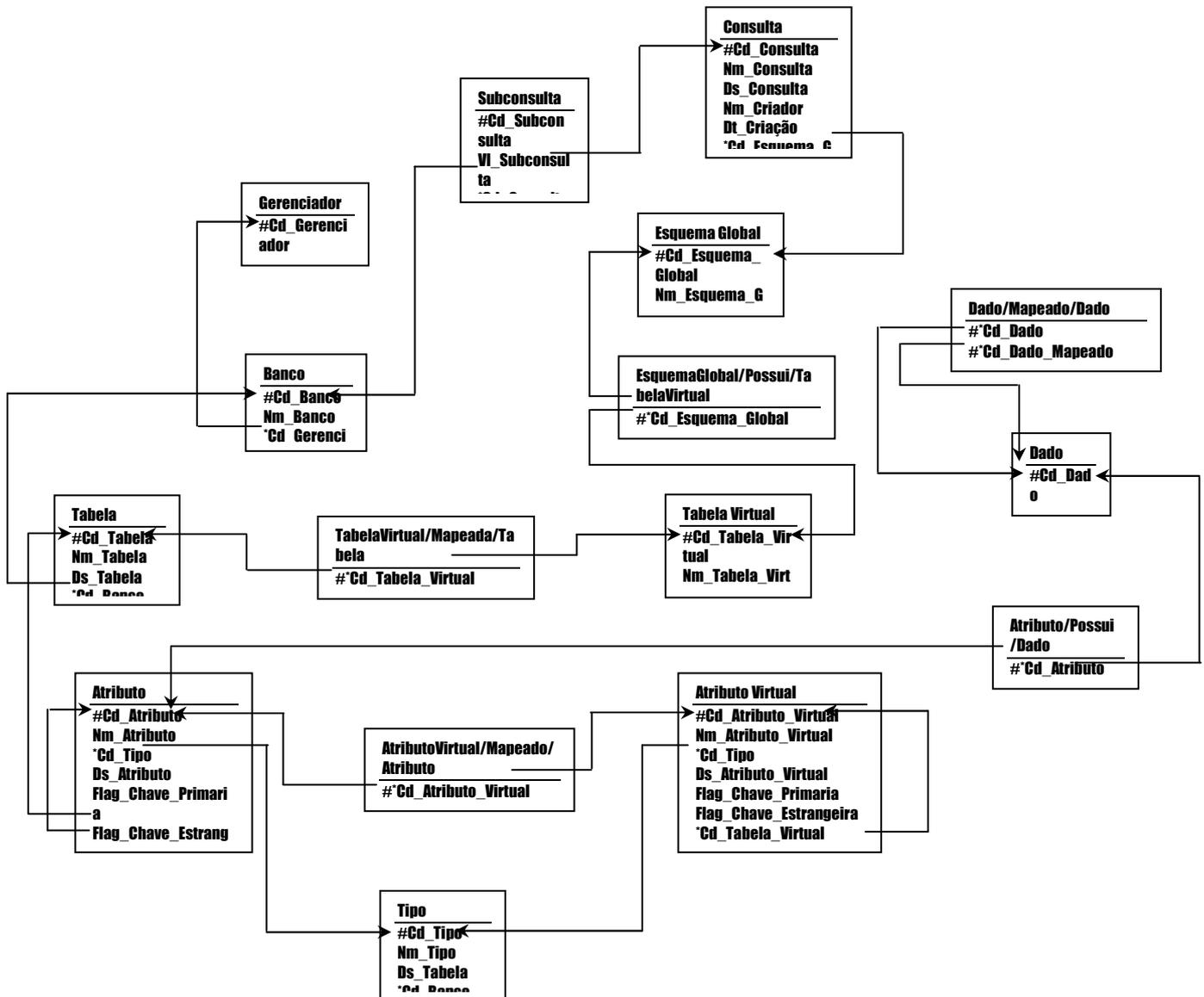


Figura 5.1 - Esquema Geral do Metabanco de Dados

Na seqüência, apresentamos o dicionário de dados do Metabanco de Dados.

Para cada tabela do Metabanco de Dados, mostramos uma breve descrição e as tabelas referenciadas, ou seja, as tabelas que possuem atributos considerados chave estrangeira de atributos da tabela em questão.

No que diz respeito aos atributos descrevemos as propriedades nome do atributo (Nome), tipo de dado (Tipo), chave primária (CP), chave estrangeira (CE), nome da tabela e do atributo ao qual o atributo em questão faz referência (Referência) e significado do atributo (Semântica).

## Dicionário de Dados

Tabela: Atributo					
Nome Simbólico: Atributo					
Descrição		Armazena informações sobre os atributos de bancos de dados locais			
Tabelas Referenciadas		Tipo, Tabela e Atributo			
Nome	Tipo	CP	CE	Referência	Semântica
#Cd_Atributo	Int	✓			Código do atributo
Nm_Atributo	Varchar				Nome do atributo
Cd_Tipo	Int		✓	Tipo.Cd_Tipo	Código do tipo
Ds_Atributo	Varchar				Descrição do atributo
Flag_Chave_Primary	Bit				Indica se o atributo é chave primária
Flag_Chave_Estrangeira	Bit				Indica se o atributo é chave estrangeira
*Cd_Tabela	Int		✓	Tabela.Cd_Tabela	Código da tabela que contém o atributo
*Cd_Atributo_Referencia	Int		✓	Atributo.Cd_Atributo	Código que identifica o atributo que referencia

Tabela 5.1 - Tabela *Atributo*

Tabela: Atributo_Dado					
Nome Simbólico: Atributo / Possui / Dado					
Descrição		Armazena os atributos que apresentam conflitos de dados do tipo: expressões distintas e precisões diferentes			
Tabelas Referenciadas		Atributo e Dado			
Nome	Tipo	CP	CE	Referência	Semântica
#*Cd_Atributo	Int	✓	✓	Atributo.Cd_Atributo	Código do atributo
#*Cd_Dado	Int	✓	✓	Dado.Cd_Dado	Código do dado

Tabela 5.2 - Tabela *Atributo / Possui / Dado*

Tabela: Atributo Virtual					
Nome Simbólico: Atributo_Virtual					
Descrição		Armazena os atributos pertencentes à tabelas virtuais			
Tabelas Referenciadas		Tipo, Tabela_Virtual e Atributo_Virtual			
Nome	Tipo	CP	CE	Referência	Semântica
#Cd_Atributo_Virtual	Int	✓			Código do atributo virtual
Nm_Atributo_Virtual	Varchar				Nome do atributo virtual
Cd_Tipo	Int		✓	Tipo.Cd_Tipo	Código do tipo de dados
Ds_Atributo_Virtual	Varchar				Descrição do atributo virtual
Flag_Chave_Primary	Bit				Indica se é chave primária
Flag_Chave_Estrangeira	Bit				Indica se é chave estrangeira
*Cd_Tabela_Virtual	Número		✓	TabelaVirtual.Cd_Tabela_Virtual	Código da tabela virtual
*Cd_Atributo_Virtual_Referencia	Número		✓	AtributoVirtual.Cd_Atributo_Virtual	Código que identifica cada atributo virtual que referencia

Tabela 5.3 - Tabela *Atributo Virtual*

Tabela: Atributo Virtual / Mapeado / Atributo					
Nome Simbólico: Atrib_Virt_Mapeado_Atributo					
Descrição		Armazena mapeamentos entre atributos virtuais e atributos			
Tabelas Referenciadas		Atributo_Virtual e Atributo			
Nome	Tipo	CP	CE	Referência	Semântica
#*Cd_Atributo_Virtual	Int	✓	✓	AtributoVirtual.Cd_Atributo_Virtual	Código do atributo virtual
#*Cd_Atributo	Int	✓	✓	Atributo.Cd_Atributo	Código do atributo

Tabela 5.4 - Tabela *Atributo Virtual / Mapeado / Atributo*

Tabela: Banco					
Nome Simbólico: Banco de Dados					
Descrição		Armazena o nome dos bancos de dados locais			
Tabelas Referenciadas		Gerenciador			
Nome	Tipo	CP	CE	Referência	Semântica
#Cd_Banco	Int	✓			Código que identifica cada banco de dados
Nm_Banco	Varchar				Nome de banco de dados
*Cd_Gerenciador	Int		✓	Gerenciador.Cd_Gerenciador	Código que identifica cada sistema gerenciador de banco de dados

Tabela 5.5 - Tabela *Banco*

Tabela: Consulta					
Nome Simbólico: Consulta					
Descrição		Armazena informações relativas às consultas			
Tabelas Referenciadas		Esquema_Global			
Nome	Tipo	CP	CE	Referência	Semântica
#Cd_Consulta	Int	✓			Código que identifica cada consulta
Nm_Consulta	Varchar				Nome da consulta
Vl_Consulta	Text				Comando SQL
Ds_Consulta	Varchar				Descrição da consulta
Nm_Criador	Varchar				Criador da consulta
Dt_Criacao	Smalldatetime				Data de criação
*Cd_Esquema_Global	Int		✓	EsquemaGlobal.Cd_Esquema_Global	Código do Esquema Global

Tabela 5.6 - Tabela *Consulta*

Tabela: Dado					
Nome Simbólico: Dado					
Descrição		Armazena os valores a serem mapeados			
Tabelas Referenciadas					
Nome	Tipo	CP	CE	Referência	Semântica
#Cd_Dado	Int	✓			Código que identifica unicamente um valor de dados
Nm_Dado	Varchar				Valor do dado

Tabela 5.7 - Tabela *Dado*

Tabela: Dado / Mapeado / Dado					
Nome Simbólico: Dado_Mapeado_Dado					
Descrição		Armazena os mapeamentos entre dados			
Tabelas Referenciadas		Dado			
Nome	Tipo	CP	CE	Referência	Semântica
#*Cd_Dado	Int	✓	✓	Dado.Cd_Dado	Código que identifica unicamente um dado a ser mapeado
#*Cd_Dado_Mapeado	Int	✓	✓	Dado.Cd_Dado	Código que identifica unicamente um dado que mapeia

Tabela 5.8 - Tabela *Dado\_Mapeado\_Dado*

Tabela: Esquema Global						
Nome Simbólico: EsquemaGlobal						
Descrição		Armazena informações sobre os esquemas globais				
Tabelas Referenciadas						
Nome	Tipo	CP	CE	Referência	Semântica	
#Cd_Esquema_Global	Int	✓			Código do esquema global	
Nm_Esquema_Global	Varchar				Nome do esquema global	
Ds_Esquema_Global	Varchar				Descrição do esquema global	

Tabela 5.9 - Tabela *Esquema Global*

Tabela: Esquema Global / Possui / Tabela Virtual						
Nome Simbólico: Esq_Glob_Possui_Tab_Virt						
Descrição		Armazena as tabelas virtuais pertencentes a cada esquema global				
Tabelas Referenciadas						
Nome	Tipo	CP	CE	Referência	Semântica	
#*Cd_Esquema_Global	Int	✓	✓	EsquemaGlobal.Cd_Esquema_Global	Código do esquema global	
#*Cd_Tabela_Virtual	Int	✓	✓	TabelaVirtual.Cd_Tabela_Virtual	Código da tabela virtual	

Tabela 5.10 - Tabela *Esquema Global / Possui / Tabela Virtual*

Tabela: Gerenciador						
Nome Simbólico: Gerenciador						
Descrição		Armazena os tipos de SGBDRs que podem ser integrados				
Tabelas Referenciadas						
Nome	Tipo	CP	CE	Referência	Semântica	
#Cd_Gerenciador	Int	✓			Código que identifica cada SGBDR	
Nm_Gerenciador	Varchar				Nome do sistema gerenciador de banco de dados	

Tabela 5.11 - Tabela *Gerenciador*

Tabela: Subconsulta						
Nome Simbólico: Subconsulta						
Descrição		Armazena as subconsultas geradas correspondentes a cada consulta				
Tabelas Referenciadas						
Nome	Tipo	CP	CE	Referência	Semântica	
#Cd_Subconsulta	Int	✓			Código que identifica unicamente uma subconsulta	
Vl_Subconsulta	Text				Cláusula SQL	
*Cd_Consulta	Int		✓	Consulta.Cd_Consulta	Código que identifica cada consulta	
*Cd_Banco	Int		✓	Banco.Cd_Banco	Código que identifica unicamente um banco de dados	

Tabela 5.12 - Tabela *Subconsulta*

Tabela: Tabela						
Nome Simbólico: Tabela						
Descrição		Armazena as tabelas dos bancos de dados locais				
Tabelas Referenciadas						
Nome	Tipo	CP	CE	Referência	Semântica	
#Cd_Tabela	Número	✓			Código que identifica cada tabela	
Nm_Tabela	Varchar				Nome da tabela	
*Cd_Banco	Número		✓	Banco.Cd_Banco	Código que identifica cada banco de dados	

Tabela 5.13 - Tabela *Tabela*

Tabela: Tabela Virtual					
Nome Simbólico: Tabela_Virtual					
Descrição			Armazena as tabelas virtuais		
Tabelas Referenciadas					
Nome	Tipo	CP	CE	Referência	Semântica
#Cd_Tabela_Virtual	Int	✓			Código da tabela virtual
Nm_Tabela_Virtual	Varchar				Nome da tabela virtual
Ds_Tabela_Virtual	Varchar				Descrição da tabela virtual

Tabela 5.14 - Tabela *Tabela Virtual*

Tabela: Tabela Virtual / Mapeada / Tabela					
Nome Simbólico: Tab_Virt_Mapeada_Tabela					
Descrição			Armazena os mapeamentos entre tabelas virtuais e tabelas		
Tabelas Referenciadas			Tabela_Virtual e Tabela		
Nome	Tipo	CP	CE	Referência	Semântica
#*Cd_Tabela_Virtual	Int	✓	✓	TabelaVirtual.Cd_Tabela_Virtual	Código da tabela virtual
#*Cd_Tabela	Int	✓	✓	Tabela.Cd_Tabela	Código da tabela

Tabela 5.15 - Tabela *Tabela Virtual / Mapeada / Tabela*

Tabela: Tipo de Dado					
Nome Simbólico: Tipo					
Descrição			Armazena os diversos tipos de dados dos atributos virtuais ou não		
Tabelas Referenciadas					
Nome	Tipo	CP	CE	Referência	Semântica
#Cd_Tipo	Int	✓			Código que identifica unicamente um tipo de dado
Nm_Tipo	Varchar				Tipo de Dado

Tabela 5.16 - Tabela *Tipo de Dado*

### 5.3 Gerente de Metadados

Sua função é manter uma interface entre o Metabanco de Dados e os componentes do Sistema BDRV. Qualquer solicitação de informação ao Metabanco de Dados por parte de algum componente é controlada pelo Gerente de Metadados (ver Figura 5.2). Em resumo, seu papel é receber consultas, envia-las ao Metabanco de Dados, receber os resultados e retorna-los aos componentes.

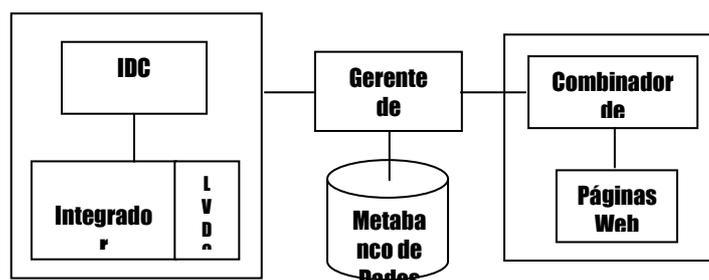


Figura 5.2 - Gerente de Metadados

Na Tabela 5.17 apresentamos as solicitações de informação enviadas ao Gerente de Metadados por alguns componentes do Sistema BDRV.

Componente	Solicitação
IDC	- Descrição dos BDVs
Integrador	- Mapeamentos BDVs → Esquemas locais - Armazenamento de consultas e subconsultas
Combinador de Resultados	- Armazenamento de consultas e subconsultas - Armazenamento de informações a serem utilizadas na solução de alguns tipos de conflitos

Tabela 5.17 - Solicitações de Componentes do Sistema BDRV ao Gerente de Metadados

As informações armazenadas no Metabanco de Dados são obtidas através de comandos SQL. Deixaremos para mostrar tais comandos quando estivermos falando de cada componente individualmente.

## 5.4 Interface de Definição de Consultas

A Interface de Definição de Consultas (IDC) é um componente do Sistema BDRV cuja função é possibilitar a elaboração de consultas por um UDC, com base em um BDV, através de uma Linguagem Visual de Definição de Consultas – LVDC (ver Seção 5.5.2).

Uma consulta é formada por uma coleção de tabelas virtuais, atributos virtuais, filtros, ordenações e agrupamentos, além de alguns elementos que descrevem a mesma, como: título, descrição, data de criação e nome do usuário.

NA IDC, as consultas podem ser construídas através de simples cliques de "mouse". O processo de elaboração de consultas é realizado em etapas seqüenciais. A

qualquer momento, durante o processo de construção de uma consulta, o UDC pode abandonar a elaboração da mesma, cancelando-a. É possível também voltar através das etapas.

A descrição de cada consulta é armazenada no Metabanco de Dados. Os atributos de uma consulta podem ser visualizados na Tabela 5.6.

Na seqüência, apresentaremos as várias etapas para construção de consultas.

### 5.4.1 Etapas para Elaboração de Consultas

Para elaboração de uma consulta o UDC deve, necessariamente, seguir uma seqüência de etapas. São elas:

#### Seleção de Esquema Global

São mostrados todos os esquemas globais para que o UDC possa selecionar um deles e elaborar uma consulta baseada nas tabelas e atributos virtuais pertencentes ao esquema global selecionado.

O nome de cada BDV cadastrado encontra-se armazenado no Metabanco de Dados, mais precisamente na tabela *Esquema\_Global* (Tabela 5.9). Para obtê-los, a IDC remete a seguinte consulta ao Gerente de Metadados para posterior envio ao Metabanco de Dados:

```
SELECT Esquema_Global.Nm_Esquema_Global  
FROM Esquema_Global
```

Tabela 5.18 - Obtendo Esquemas Globais

O resultado é então retornado pelo Metabanco de Dados ao Gerente de Metadados que, por sua vez, o repassa para a IDC que o apresenta ao UDC.

Considerando de agora em diante o terceiro exemplo do Capítulo 4, o resultado retornado é o mesmo apresentado na Figura 4.3. O UDC seleciona então o esquema global *Organização*.

Nm_Esquema_Global
Copin
Organização

Tabela 5.19 - Esquemas Globais Cadastrados no Metabanco de Dados

### Seleção de Tabelas Virtuais

São listadas todas as tabelas virtuais pertencentes ao esquema global selecionado anteriormente, para que o UDC possa selecionar a(s) que pretende incluir em uma consulta. Só é possível passar para a etapa seguinte caso tenha sido selecionada pelo menos uma tabela virtual.

A descrição de cada tabela virtual, independente do BDV a qual pertence, está armazenada na tabela *Tabela\_Virtual* do Metabanco de Dados (ver Tabela 5.14). Na tabela *Esq\_Glob\_Possui\_Tab\_Virt* do Metabanco de Dados (ver Tabela 5.10) é possível obtermos as tabelas virtuais referentes a cada esquema global.

Para obter o nome das tabelas virtuais pertencentes ao esquema global *Organização*, a IDC envia a seguinte consulta ao Gerente de Metadados:

```
SELECT DISTINCT Tabela_Virtual.Nm_Tabela_Virtual
FROM Tabela_Virtual, Esquema_Global, Esq_Glob_Possui_Tab_Virt
WHERE Tabela_Virtual.Cd_Tabela_Virtual = Esq_Glob_Possui_Tab_Virt.Cd_Tabela_Virtual
AND Esquema_Global.Nm_Esquema_Global = 'Organização' AND
Esquema_Global.Cd_Esquema_Global = Esq_Glob_Possui_Tab_Virt.Cd_Esquema_Global
ORDER BY Nm_Tabela_Virtual
```

Tabela 5.20 - Obtendo as Tabelas Virtuais do Esquema Global *Organização*

O resultado retornado é o mesmo apresentado na Figura 4.4 (ver Tabela 5.21).

Nm_Tabela_Global
Departamento
DepartamentoControlaProjeto
Empregado
EmpregadoTrabalhaProjeto
EnderecoDoDepartamento
EnderecoDoEmpregado
Projeto

Tabela 5.21 - Esquemas Globais Cadastrados

De acordo com o exemplo, o UDC seleciona as tabelas virtuais: *Empregado*, *Departamento*, *DepartamentoControlaProjeto*, *EmpregadoTrabalhaProjeto* e *Projeto*.

### Seleção de Atributos Virtuais

São mostrados apenas os atributos virtuais pertencentes às tabelas virtuais selecionadas no passo anterior, para que o UDC possa selecionar os que pretende incluir em uma consulta, ou seja, os atributos que deseja que sejam apresentados no resultado final. Só é possível passar para a etapa seguinte caso tenha sido selecionado pelo menos um atributo virtual.

Neste passo, o UDC pode aplicar as seguintes funções de agregação aos atributos virtuais selecionados:

- **Soma:** realiza a adição de valores contidos no conjunto-resposta referente ao atributo virtual;
- **Média:** retorna a média dos valores contidos no conjunto resposta, referente ao atributo virtual;
- **Quantidade:** retorna o número total de registros contidos no conjunto-resposta;
- **Máximo:** retorna o maior valor contido no conjunto-resposta, referente ao atributo virtual;

**Mínimo:** retorna o menor valor contido no conjunto-resposta, referente ao atributo virtual.

Os atributos virtuais de um BDV estão armazenados na tabela *Atributo\_Virtual* do Metabanco de Dados (ver Tabela 5.3). Para cada tabela virtual selecionada no passo anterior, uma consulta é enviada ao Gerente de Metadados, pela IDC. Por exemplo, para obter os atributos virtuais da tabela *Empregado*, a seguinte consulta é executada:

```

SELECT DISTINCT Tabela_Virtual.Nm_Tabela_Virtual,
Atributo_Virtual.Nm_Atributo_Virtual
FROM Tabela_Virtual, Atributo_Virtual, Esquema_Global, Esq_Glob_Possui_Tab_Virt
WHERE Tabela_Virtual.Cd_Tabela_Virtual = Atributo_Virtual.Cd_Tabela_Virtual AND
Tabela_Virtual.Nm_Tabela_Virtual = 'Empregado' AND
Tabela_Virtual.Cd_Tabela_Virtual = Esq_Glob_Possui_Tab_Virt.Cd_Tabela_Virtual AND
Esquema_Global.Nm_Esquema_Global = 'Organização' AND
Esq_Glob_Possui_Tab_Virt.Cd_Esquema_Global =
Esquema_Global.Cd_Esquema_Global
ORDER BY Atributo_Virtual.Nm_Atributo_Virtual

```

Tabela 5.22 - Obtendo os Atributos Virtuais da Tabela Virtual *Empregado*

O resultado retornado pelo Metabanco de Dados pode ser visto na Tabela 5.23:

Nm_Tabela_Virtual	Nm_Atributo_Virtual
Empregado	Cargo
Empregado	Cd_Departamento
Empregado	Cd_Empregado
Empregado	Cd_Endereço_Empregado
Empregado	Dt_Admissao
Empregado	Nm_Empregado
Empregado	Nm_Est_Civil
Empregado	VI_Salario_Diario
Empregado	VI_Salario_Semanal

Tabela 5.23 - Atributos Virtuais da Tabela Virtual *Empregado*

Este processo se repete para o restante das tabelas virtuais selecionadas (*Departamento*, *DepartamentoControlaProjeto*, *EmpregadoTrabalhaProjeto* e *Projeto*). Podemos ver o resultado final na Tabela 5.24. Compare estes resultados com o da Figura 4.15 e perceba que são os mesmos. A única diferença é que para identificar a tabela virtual de cada atributo virtual na IDC, cada um deles é precedido pelo nome da tabela virtual a qual pertence seguido do caractere ponto (".").

Nm_Tabela_Virtual	Nm_Atributo_Virtual
Empregado	Cargo
Empregado	Cd_Departamento
Empregado	Cd_Empregado
Empregado	Cd_Endereco_Empregado
Empregado	Dt_Admissao
Empregado	Nm_Empregado
Empregado	Nm_Est_Civil
Empregado	VI_Salario_Diario
Empregado	VI_Salario_Semanal
Departamento	Cd_Departamento
Departamento	Nm_Departamento
Departamento	Nu_Telefone
Departamento	Cd_Endereco_Departamento
Projeto	Cd_Projeto
Projeto	Nm_Projeto
EmpregadoTrabalhaDepartamento	Cd_Empregado
EmpregadoTrabalhaDepartamento	Cd_Departamento
DepartamentoControlaProjeto	Cd_Departamento
DepartamentoControlaProjeto	Cd_Projeto

Tabela 5.24 - Atributos Virtuais das Tabelas Virtuais Selecionadas

### Definição de Filtros

A definição de filtros tem o objetivo de estabelecer restrições aos atributos virtuais. Nesta etapa, a IDC exibe apenas os atributos virtuais pertencentes às tabelas virtuais selecionadas anteriormente, para que o UDC possa impor filtros. Portanto, as consultas enviadas Metabanco de Dados são as mesmas da etapa anterior. Esta não é uma etapa obrigatória, ou seja, o UDC pode passar para etapa seguinte sem definir nenhum filtro.

Um filtro é composto por um atributo virtual, um operador e uma constante. Os seguintes operadores podem ser utilizados: maior (>), menor (<), maior ou igual (>=), menor ou igual (<=), igual (=) e diferente (<>). A constante é um valor literal, como número ou caractere.

Como o Metabanco de Dados armazena o tipo de dado de cada atributo virtual, no momento em que um UDC seleciona um atributo virtual para definir algum filtro,

apenas os operadores que fazem sentido serem aplicados ao mesmo são disponibilizados.

No exemplo que estamos seguindo, estamos interessados apenas nos empregados do departamento de informática. Como o atributo *Nm\_Departamento* da tabela *Departamento* é do tipo *Varchar*, então todos os operadores disponíveis podem ser associados a este tipo.

Por outro lado, se quiséssemos aplicar um filtro a um atributo do tipo lógico (booleano), não faria sentido disponibilizar os operadores: maior (>), menor (<), maior ou igual (>=) e menor ou igual (<=), e sim apenas igual (=) ou diferente (<>).

Para obter o tipo de dados do atributo virtual *Nm\_Departamento* é necessário o envio da seguinte consulta ao Metabanco de Dados:

```
SELECT DISTINCT C.Nm_Tipo
FROM Tabela_Virtual A, Atributo_Virtual B, Tipo C, Esquema_Global D,
Esq_Glob_Possui_Tab_Virt E
WHERE A.Cd_Tabela_Virtual = B.Cd_Tabela_Virtual AND
A.Nm_Tabela_Virtual = 'Departamento' AND
B.Nm_Atributo_Virtual = 'Nm_Departamento' AND
B.Cd_Tipo = C.Cd_Tipo AND D.Nm_Esquema_Global = 'Organização' AND
E.Cd_Esquema_Global = D.Cd_Esquema_Global AND E.Cd_Tabela_Virtual =
A.Cd_Tabela_Virtual
```

Tabela 5.25 - Obtendo o Tipo do Atributo Virtual *Departamento.Nm\_Departamento*

O Metabanco de Dados, por sua vez, produz o resultado visto na Tabela 5.26.

Nm_Tipo
Varchar

Tabela 5.26 - Tipo de Dados do Atributo Virtual *Departamento.Nm\_Departamento*

### **Definição de Ordenações**

Nesta etapa são apresentados apenas os atributos virtuais pertencentes às tabelas virtuais selecionadas anteriormente, para que o UDC possa indicar a ordem com que deseja que os resultados lhes sejam apresentados.

A ordem de apresentação depende do tipo de dado do atributo virtual. Caso seja Char, por exemplo, o conjunto resposta será ordenado em ordem alfabética. No caso de atributos do tipo Número, o resultado é exibido em ordem crescente. A IDC não possibilita a ordenação descendente.

As consultas enviadas Metabanco de Dados são as mesmas da etapa anterior. Conseqüentemente, os resultados obtidos também (ver Tabela 5.24). Novamente, esta não é uma etapa obrigatória, ou seja, o UDC pode passar para etapa seguinte sem definir nenhuma ordenação.

No nosso exemplo, o resultado da consulta será ordenado pelos atributos virtuais *Projeto.Nm\_Projeto* e *Empregado.Nm\_Empregado*.

### **Definição de Agrupamentos**

A definição de agrupamento em um atributo virtual significa que o conjunto-resposta final será agrupado de acordo com o atributo virtual. Só é possível definir agrupamentos nos atributos virtuais selecionados na etapa Definição de Atributos Virtuais. Esta também não é uma etapa obrigatória, ou seja, o UDC pode passar para o passo seguinte sem definir nenhum agrupamento.

No nosso exemplo, o resultado da consulta não será agrupado por nenhum atributo virtual.

### Descrição da Consulta

Nesta etapa, o UDC deve fornecer, necessariamente, um título para a consulta que acabou de definir, uma breve descrição da consulta, além de informar o nome do usuário elaborador da consulta. A data de criação é a da elaboração da consulta.

No exemplo, são fornecidas as seguintes informações sobre a consulta em questão:

Título	Empregados, respectivos cargos, trabalhando em projetos controlados pelo departamento de Informática
Descrição	Esta consulta recupera os empregados, com respectivos cargos, trabalhando em projetos gerenciados pelo departamento de Informática
Usuário	Administrador

Tabela 5.27 - Descrição da Consulta

Uma vez finalizada a consulta, a IDC realiza interna e automaticamente equijunções entre as tabelas virtuais selecionadas que mantêm algum tipo de referência ao nível de atributo virtual (chave estrangeira). É possível obter as referências entre tabelas virtuais através das tabelas do Metabanco de Dados *Atributo Virtual* (ver Tabela 5.3) e *Tabela Virtual* (ver Tabela 5.14). A tabela *Atributo Virtual* mantém um auto-relacionamento através dos atributos *Atributo\_Virtual.Cd\_Atributo\_Virtual* e *Atributo\_Virtual.Cd\_Atributo\_Virtual\_Referencia*. Com ele é possível identificar os atributos virtuais que fazem referência entre si. Através da realização de uma equijunção entre as tabelas *Atributo Virtual* e *Tabela Virtual* podemos identificar as tabelas virtuais que mantêm referência.

No exemplo, para obter as referências entre as tabelas virtuais selecionadas e seus respectivos atributos virtuais, a IDC realiza a seguinte consulta ao Metabanco de Dados:

```

SELECT DISTINCT TV1.Nm_Tabela_Virtual, AV1.Nm_Atributo_Virtual,
TV2.Nm_Tabela_Virtual, AV2.Nm_Atributo_Virtual
FROM Tabela_Virtual TV1, Tabela_Virtual TV2, Atributo_Virtual AV1, Atributo_Virtual AV2,
Esquema_Global EG, Esq_Glob_Possui_Tab_Virt EGPTV1, Esq_Glob_Possui_Tab_Virt EGPTV2
WHERE AV1.Cd_Tabela_Virtual = TV1.Cd_Tabela_Virtual AND AV2.Cd_Tabela_Virtual =
TV2.Cd_Tabela_Virtual AND AV1.Cd_Atributo_Virtual =
AV2.Cd_Atributo_Virtual_Referencia AND EGPTV1.Cd_Tabela_Virtual =
TV1.Cd_Tabela_Virtual AND EGPTV2.Cd_Tabela_Virtual = TV2.Cd_Tabela_Virtual AND
EG.Cd_Esquema_Global = EGPTV1.Cd_Esquema_Global AND EG.Cd_Esquema_Global =
EGPTV2.Cd_Esquema_Global AND EG.Nm_Esquema_Global = 'Organização'
ORDER BY TV1.Nm_Tabela_Virtual, AV1.Nm_Atributo_Virtual, TV2.Nm_Tabela_Virtual,
AV2.Nm_Atributo_Virtual

```

Tabela 5.28 - Referências Entre Tabelas Virtuais Selecionadas e Respective Atributos Virtuais

Os resultados retornados pelo Metabanco de Dados podem ser vistos na Tabela 5.29. Nela, podemos perceber que a tabela virtual *Empregado* referencia a tabela virtual *Departamento* através do atributo virtual *Cd\_Departamento*, presente em ambas.

Tabela Referencia	Atributo Referenciado	Tabela Referenciada	Atributo Referencia
Departamento	Cd_Departamento	DepartamentoControlaProjeto	Cd_Departamento
Departamento	Cd_Departamento	Empregado	Cd_Departamento
Empregado	Cd_Empregado	EmpregadoTrabalhaProjeto	Cd_Empregado
EnderecoDoDepartamento	Cd_Endereco_Departamento	Departamento	Cd_Endereco_Departamento
EnderecoDoEmpregado	Cd_Endereco_Empregado	Empregado	Cd_Endereco_Empregado
Projeto	Cd_Projeto	DepartamentoControlaProjeto	Cd_Projeto
Projeto	Cd_Projeto	EmpregadoTrabalhaProjeto	Cd_Projeto

Tabela 5.29 - Referências Entre Tabelas Virtuais do Esquema Global *Organização*

Continuando com o nosso exemplo, a consulta final ao BDV *Organização* gerada pela IDC é:

```

SELECT Empregado.Nm_Empregado, Empregado.Nm_Cargo, Projeto.Nm_Projeto
FROM Empregado, Departamento, Projeto, DepartamentoControlaProjeto,
EmpregadoTrabalhaProjeto
WHERE Empregado.Cd_Departamento = Departamento.Cd_Departamento AND
Departamento.Nm_Departamento = 'Informática' AND Empregado.Cd_Empregado =
EmpregadoTrabalhaProjeto.Cd_Empregado AND Projeto.Cd_Projeto =
EmpregadoTrabalhaProjeto.Cd_Projeto AND Projeto.Cd_Projeto =
DepartamentoControlaProjeto.Cd_Projeto AND
DepartamentoControlaProjeto.Cd_Departamento = Departamento.Cd_Departamento
ORDER BY Projeto.Nm_Projeto, Empregado.Nm_Empregado

```

Tabela 5.30 - Consulta ao BDV *Organização*

Os elementos de uma consulta (tabelas virtuais, atributos virtuais, filtros, ordenações e agrupamentos) são repassados ao Integrador para serem processados e transformados (mapeados) em elementos correspondentes aos bancos de dados locais. As informações enviadas ao Integrador, pela IDC, seguem o seguinte formato:

Tabelas Virtuais	{ Tabela Virtual }
Atributos Virtuais	{{ Tabela Virtual, Atributo Virtual, Função de Agregação }}
Filtros	{{ Tabela Virtual, Atributo Virtual, Operador, Constante }}
Junções	Junções = {{ Tabela Virtual 1, Atributo Virtual 1, Operador, Tabela Virtual 2, Atributo Virtual 2 }}
Ordenações	{{ Tabela Virtual, Atributo Virtual }}
Agrupamentos	{{ Tabela Virtual, Atributo Virtual }}
Descrição	{ Título da Consulta, Descrição da Consulta, Nome do Usuário }

Tabela 5.31 - Formato das Informações Enviadas pela IDC ao Integrador

Em nosso exemplo a Tabela 5.31 é preenchida da seguinte forma:

Tabelas Virtuais	{Empregado, Departamento, Projeto, EmpregadoTrabalhaProjeto, DepartamentoControlaProjeto}
Atributos Virtuais	{{Empregado, Nm_Empregado, Ø}, {Empregado, Cargo, Ø}, {Projeto, Nm_Projeto, Ø}}
Filtros	{{Departamento, Nm_Departamento, =, Informática}}
Junções	{{Departamento, Cd_Departamento, =, Empregado, Cd_Departamento}, {Empregado, Cd_Empregado, =, EmpregadoTrabalhaProjeto, Cd_Empregado}, {Projeto, Cd_Projeto, =, EmpregadoTrabalhaProjeto, Cd_Projeto}, {Projeto, Cd_Projeto, =, DepartamentoControlaProjeto, Cd_Projeto}, {DepartamentoControlaProjeto, Cd_Departamento, =, Departamento, Cd_Departamento}}
Ordenações	{{Projeto, Nm_Projeto}, {Empregado, Nm_Empregado}}
Agrupamentos	Ø
Descrição	{Empregados, respectivos cargos, trabalhando em projetos controlado pelo departamento de Informática, Esta consulta recupera os empregados, com respectivos cargos, trabalhando em projetos gerenciados pelo departamento de Informática, Administrador}

Tabela 5.32 - Informações Enviadas ao Integrador pela IDC

## 5.4.2 Linguagem Visual de Definição de Consultas

A IDC é, na verdade, um gerador de código-fonte SQL. Com a IDC é possível produzir consultas SQL a esquemas globais, através de uma Linguagem Visual de Definição de Consultas (LVDC), cuja função é descrever um conjunto de regras e opções que podem ser aplicadas na formulação de consultas.

A LVDC possibilita a geração de consultas obedecendo a seguinte sintaxe:

```
SELECT lista_select
FROM tabela_virtual1 [[, tabela_virtual2] [..., tabela_virtual16]]
[WHERE cláusula]
[GROUP BY cláusula]
[ORDER BY cláusula]
```

### *lista\_select*

Especifica um ou mais atributos virtuais a serem selecionados. A ordem dos atributos contidos na lista significa a ordem com que o conjunto-resposta será apresentado. Se a lista contém mais de um atributo, os mesmos devem ser separados por vírgulas (.). O nome do atributo virtual é ser precedido pelo nome da tabela virtual mais o caractere ponto (.). É possível associar no máximo uma função de agregação a cada atributo virtual. São elas: SUM, MAX, MIN, COUNT e AVG.

## FROM

Indica as tabelas virtuais que podem ser utilizadas nas sentenças SELECT, WHERE, GROUP BY e ORDER BY. A cláusula FROM permite um máximo de 16 tabelas virtuais. Se a lista contém mais de uma tabela virtual, os mesmos devem ser separados por vírgulas (,).

## WHERE cláusula =

*WHERE condições\_de\_pesquisa*

Especifica as condições de restrição a serem utilizadas para a obtenção do conjunto-resposta. Não há limite no número de condições de pesquisa.

*condições\_de\_pesquisa*

Define as condições para recuperação de dados. Condições de pesquisa seguem imediatamente a palavra-chave WHERE. Se mais de uma condição de pesquisa for utilizada em uma mesma sentença WHERE, as mesmas devem ser conectadas com o operador lógico AND. Junções são especificadas nas condições de pesquisa. Só são permitidas equijunções entre atributos virtuais.

Sintaxe:

**WHERE** *nome\_da\_tabela\_virtual.nome\_do\_atributo\_virtual operador\_de\_comparação constante [AND]*

**WHERE** *nome\_da\_tabela\_virtual.nome\_do\_atributo\_virtual = nome\_da\_tabela\_virtual.nome\_do\_atributo\_virtual [AND]*

**WHERE** *nome\_da\_tabela\_virtual.nome\_do\_atributo\_virtual LIKE constante [AND]*

onde,

*nome\_da\_tabela\_virtual*

É o nome da tabela virtual a qual o atributo virtual pertence.

*nome\_do\_atributo\_virtual*

É o nome de um atributo virtual utilizado como comparação. Como forma de evitar alguma ambigüidade, o nome do atributo deve ser precedido pelo nome da tabela virtual mais o caractere ponto (.).

**AND**

Operador lógico cuja função é fazer a junção entre duas condições e retornar resultado quando ambas são verdade.

*operador\_de\_comparação*

É um símbolo utilizado entre duas expressões. A lista completa dos tipos de operadores de comparação pode ser vista na Tabela 5.33.

Símbolo	Semântica
=	Igual a
>	Maior que
<	Menor que
>=	Maior ou Igual a
<=	Menor ou Igual a
<>	Diferente de

Tabela 5.33 - Operadores de Comparação

*constante*

Um valor literal, como número ou caractere. Caracteres são cercados por apóstrofos simples (').

**GROUP BY** cláusula

Especifica os grupos nos quais o conjunto-resposta é dividido. O(s) atributo(s) virtual(is) utilizado(s) para agrupamento deve(m) necessariamente se encontrar na cláusula SELECT. Se a lista contém mais de um atributo virtual, os mesmos devem ser separados por vírgulas (,).

**ORDER BY** cláusula

Ordena o conjunto-resposta por atributos virtuais. É possível ordenar por no máximo 16 atributos virtuais. A cláusula ORDER BY pode incluir itens que não se encontram na cláusula SELECT. Se a lista contém mais de um atributo virtual, os mesmos devem ser separados por vírgulas (,).

## 5.5 Integrador

O papel do Integrador é receber consultas formuladas com base em BDVs e transformá-las em um conjunto de subconsultas referentes aos bancos de dados sendo integrados. Posteriormente, no Módulo de Execução de Consultas (MEC), estas subconsultas são submetidas para serem executadas em seus respectivos bancos de dados. O conjunto-resposta de cada banco de dados é agrupado para produzir a resposta à consulta original.

### Estruturas de Dados

Durante o processo de transformação de uma consulta em subconsultas, o Integrador utiliza basicamente cinco estruturas de dados. São elas:

- Consulta - armazena informações ao nível de atributo virtual, como: nome do atributo virtual, tabela virtual a que pertence, função de agregação, indicação de filtro associado, existência de junção associada, entre outras;
- Filtros - guarda informações a respeito de possíveis filtros definidos em atributos virtuais;
- Junções - mantém informações sobre possíveis equijunções entre atributos virtuais;
- Fontes - guarda informações a respeito de cada fonte de dados que possa atender a um determinado atributo virtual;
- Estatística - armazena informações relativas a cada fonte de dados integrada.

A seguir, apresentamos uma descrição mais detalhada dos campos de cada uma das estruturas de dados mencionadas anteriormente.

Consulta		
Campo	Descrição	Tipo
Índice do Atributo Virtual	Identificador utilizado para distinguir unicamente cada atributo virtual	Número
Nome do Atributo Virtual	Nome do atributo virtual selecionado	String
Nome da Tabela Virtual	Nome da tabela virtual a qual o atributo pertence	String
Selecionado	Indica se o atributo virtual foi ou não selecionado	Lógico
Função de Agregação	Armazena uma possível função de agregação associada ao atributo virtual	String
Filtro	Indica se existe alguma restrição associada ao atributo virtual	Lógico
Junção	Indica se existe alguma junção associada ao atributo virtual	Lógico
OrderBy	Indica se foi definida alguma ordenação no atributo virtual	Lógico
GroupBy	Indica se foi definido algum agrupamento no atributo virtual	Lógico
Filtros	Restrições impostas ao atributo virtual	Filtros
Total de Filtros	Quantidade de filtros estabelecidos no atributo virtual	Número
Junções	Junções estabelecidas com o atributo virtual	Junções
Total de Junções	Quantidade de junções estabelecidas no atributo virtual	Número
Fontes	Fontes de dados que possuem tabelas e atributos que correspondem à tabela e atributo virtuais em questão, respectivamente	Fontes
Total de Fontes	Quantidade de fontes que podem atender ao atributo virtual	Número

Tabela 5.34 - Estrutura de dados *Consulta*

Filtros		
Campo	Descrição	Tipo
Operador	Indica o operador a ser utilizado no filtro	String
Constante	Indica o valor a ser utilizado como comparação no filtro	Varchar

Tabela 5.35 - Estrutura de dados *Filtros*

Junções		
Campo	Descrição	Tipo
Operador	Indica o operador a ser utilizado na junção. Somente são permitidas equijunções. Sendo assim, o operador sempre será o de igualdade (=)	String
Tabela Virtual de Junção	Nome da tabela virtual que deve fazer junção com a tabela virtual que contém o atributo virtual	String
Atributo Virtual de Junção	Nome do atributo virtual que faz junção	String

Tabela 5.36 - Estrutura de dados *Junções*

Fontes		
Campo	Descrição	Tipo
Fonte	Nome da fonte de dados que pode atender ao atributo virtual em questão	String
Tabela	Nome da tabela correspondente à tabela virtual que contém o atributo virtual	String
Atributo	Nome do campo correspondente ao atributo virtual	String

Tabela 5.37 - Estrutura de dados *Fontes*

Estatística		
Campo	Descrição	Tipo
Fonte	Nome da fonte de dados	Número
Porcentagem	Porcentagem de atributos virtuais envolvidos em uma consulta que pode ser atendida pela fonte em questão	Número
Subconjunto	Indica se o conjunto de atributos virtuais atendidos por uma fonte é subconjunto do de outra fonte	boolean
Selecionada	Indica, de fato, se a fonte deve ser selecionada para responder a uma consulta	boolean

Tabela 5.38 - Estrutura de dados *Estatística*

Os componentes do Integrador, a serem descritos a seguir, são:

- Analisador de Consulta;
- Coletor de Fontes;
- Estimador de Fontes;
- Seletor de Fontes;

- Transformador de Consultas.

## 5.5.1 Analisador de Consulta

### Objetivo:

Receber conjuntos<sup>1</sup> de tabelas virtuais, atributos virtuais, filtros, ordenações e agrupamentos, e estrutura-los, para que possam vir a ser utilizados por outros componentes do Integrador.

### Entrada:

- Lista de Tabelas Virtuais - composta por "strings" do tipo *Nome\_da\_Tabela\_Virtual*. Exemplos: *Empregado, Aluno, Funcionário, Tese*, etc;
- Lista de Atributos Virtuais - composta por "strings" do tipo *Nome\_da\_Tabela\_Virtual.Nome\_do\_Atributo\_Virtual*. Exemplos: *Empregado.Nm\_Empregado, Aluno.Matrícula, Tese.Título*, etc;
- Lista de Junções - formada por "strings" do tipo *Nome\_da\_Tabela\_Virtual1.Nome\_do\_Atributo1* *Operador* *Nome\_da\_Tabela\_Virtual2.Nome\_do\_Atributo2*. Exemplo: *Empregado.Cd\_Departamento = Departamento.Cd\_Departamento*;
- Lista de Filtros - formada por "strings" do tipo *Nome\_da\_Tabela\_Virtual.Nome\_do\_Atributo* *Operador* *Constante*. Exemplos: *Empregado.Nome = 'Sheyla Yonara', Aluno.Matrícula = 93110492*, etc;
- Lista de Ordenações - formada por "strings" do tipo *Nome\_da\_Tabela\_Virtual.Nome\_do\_Atributo*. Exemplos: *Empregado.Nome, Aluno.Bairro, Tese.Título*, etc;

---

<sup>1</sup> O formato de cada conjunto pode ser visualizado na Tabela 5.31. Para o nosso exemplo, utilizaremos os conjuntos da Tabela 5.32.

- Lista de Agrupamentos: formada por "strings" do tipo *Nome\_da\_Tabela\_Virtual.Nome\_do\_Atributo*. Exemplos: *Empregado.Área*, *Aluno.Período*, etc.

### Saída:

- Preenchimento dos seguintes campos, referentes à estrutura de dados *Consulta*: índice do atributo virtual, nome do atributo virtual, nome da tabela virtual, selecionado, granularidade, filtro, junção, orderby, groupby, total de filtros e total de junções;
- Preenchimento dos campos, referentes à estrutura de dados *Filtros*: operador e constante;
- Preenchimento dos campos, referentes à estrutura de dados *Junções*: operador, tabela virtual de junção e atributo virtual de junção.

### Algoritmos:

#### Algoritmo de Coleta de Atributos e Tabelas Virtuais

**Para cada** Atributo da Lista de Atributos Virtuais **Faça**

    Crie um registro na Estrutura de Dados Consulta

    Consulta.Nome do Atributo Virtual = nome do atributo

    Consulta.Nome da Tabela Virtual = nome da tabela virtual

    Consulta.Selecionado = ✓

**Se** existir função de agregação aplicada ao atributo **Então**

        Consulta.Função de Agregação = função de agregação

**Fim Se**

**Fim Para**

Seguindo o nosso exemplo, o resultado produzido pelo algoritmo anterior é:

Consulta				
Índice do Atributo Virtual	Nome da Tabela Virtual	Nome do Atributo Virtual	Selecionado	Função de Agregação
1	Empregado	Nm_Empregado	✓	
2	Empregado	Cargo	✓	
3	Projeto	Nm_Projeto	✓	

Tabela 5.39 - Resultado Produzido pelo Algoritmo de coleta de atributos e tabelas virtuais

### Algoritmo de Coleta de Filtros

**Para** cada Atributo da Lista de Filtros **Faça**

**Se** atributo já existe na Estrutura de Dados Consulta **Então**

    Crie um registro na Estrutura de Dados Filtros

    Consulta.Filtros.Operador = operador

    Consulta.Filtros.Constante = constante

    Consulta.Total de Filtros = Consulta.Total de Filtros + 1

**Fim Se**

**Senão**

    Crie um registro na Estrutura de Dados Consulta

    Consulta.Nome do Atributo = nome do atributo

    Consulta.Nome da Tabela Virtual = nome da tabela virtual

    Crie um registro na Estrutura de Dados Filtros

    Consulta.Filtros.Operador = operador

    Consulta.Filtros.Constante = constante

    Consulta.Total de Filtros = Consulta.Total de Filtros + 1

**Fim Senão**

**Fim Para**

O resultado produzido pelo algoritmo com relação ao nosso pode ser visto nas tabelas a seguir:

Consulta						
Índice do Atributo Virtual	Nome da Tabela Virtual	Nome do Atributo Virtual	Selecionado	Função de Agregação	Filtro	Total de Filtros
1	Empregado	Nm_Empregado	✓			
2	Empregado	Cargo	✓			
3	Projeto	Nm_Projeto	✓			
4	Departamento	Nm_Departamento			✓	1

Tabela 5.40 - Consulta: Resultado Produzido Pelo Algoritmo de Coleta de Filtros

Filtro		
Índice do Atributo Virtual	Operador	Constante
4	=	Informática

Tabela 5.41 - Filtros: Resultado Produzido Pelo Algoritmo de Coleta de Filtros

**Algoritmo de Coleta de Junções**

Conjunto A (Tabela Virtual, Atributo Virtual, Tabela Virtual de Junção, Atributo Virtual de Junção)

**Para** cada Tabela Virtual da Lista de Tabelas Virtuais **Faça**

Procure no Metabanco de Dados as tabelas e atributos virtuais que fazem junção com a tabela virtual inserindo no conjunto A (consulta da Tabela 5.27)

**Para** cada Atributo Virtual do conjunto A **Faça**

**Se** Atributo Virtual já existe na Estrutura de Dados Consulta **Então**

Crie um registro na Estrutura de Dados Junções

Consulta.Junções.Operador = operador '='

Consulta.Junções.Tabela Virtual de Junção = Tabela Virtual de Junção

Consulta.Junções.Atributo Virtual de Junção = Atributo de Junção

Consulta.Total de Junções = Consulta.Total de Junções + 1

**Fim Se**

**Senão**

Crie um registro na Estrutura de Dados Consulta

Consulta.Nome do Atributo = nome do atributo

Consulta.Nome da Tabela Virtual = nome da tabela virtual

Crie um registro na Estrutura de Dados Junções

Consulta.Junções.Operador = operador '='

Consulta.Junções.Tabela Virtual de Junção = Tabela Virtual de Junção

Consulta.Junções.Atributo Virtual de Junção = Atributo de Junção

Consulta.Total de Junções = Consulta.Total de Junções + 1

**Fim Senão**

**Se** Atributo Virtual de Junção já existe na Estrutura de Dados Consulta **Então**

Crie um registro na Estrutura de Dados Junções

Consulta.Junções.Operador = operador '='

Consulta.Junções.Tabela Virtual de Junção = Tabela Virtual

Consulta.Junções.Atributo Virtual de Junção = Atributo Virtual

Consulta.Total de Junções = Consulta.Total de Junções + 1

**Fim Se**

**Senão**

Consulta.Nome do Atributo = nome do atributo de junção

Consulta.Nome da Tabela Virtual = nome da tabela virtual de junção

Crie um registro na Estrutura de Dados Junções

<p>Consulta.Junções.Operador = operador '='</p> <p>Consulta.Junções.Tabela Virtual de Junção = Tabela Virtual</p> <p>Consulta.Junções.Atributo Virtual de Junção = Atributo Virtual</p> <p>Consulta.Total de Junções = Consulta.Total de Junções + 1</p> <p><b>Fim Senão</b></p> <p><b>Fim Para</b></p> <p><b>Fim Para</b></p>
--

Para o nosso exemplo, o resultado gerado pelo Algoritmo de Coleta de Junções é o seguinte:

Consulta								
Índice do Atributo Virtual	Nome da Tabela Virtual	Nome do Atributo Virtual	Selecionado	Função de Agregação	Filtro	Total de Filtros	Junção	Total de Junções
1	Empregado	Nm_Empregado	✓					
2	Empregado	Cargo	✓					
3	Projeto	Nm_Projeto	✓					
4	Departamento	Nm_Departamento			✓	1		
5	Departamento	Cd_Departamento					✓	2
6	Empregado	Cd_Departamento					✓	1
7	Empregado	Cd_Empregado					✓	1
8	EmpregadoTrabalhaProjeto	Cd_Empregado					✓	1
9	Projeto	Cd_Projeto					✓	2
10	EmpregadoTrabalhaProjeto	Cd_Projeto					✓	1
11	DepartamentoControlaProjeto	Cd_Projeto					✓	1
12	DepartamentoControlaProjeto	Cd_Departamento					✓	1

Tabela 5.42 - Consulta: Resultado Produzido pelo Algoritmo de Coleta de Junções

Junções			
Índice do Atributo Virtual	Operador	Tabela Virtual de Junção	Atributo Virtual de Junção
5	=	Empregado	Cd_Departamento
6	=	Departamento	Cd_Departamento
7	=	EmpregadoTrabalhaProjeto	Cd_Empregado
8	=	Empregado	Cd_Empregado
9	=	EmpregadoTrabalhaProjeto	Cd_Projeto
10	=	Projeto	Cd_Projeto
9	=	DepartamentoControlaProjeto	Cd_Projeto
11	=	Projeto	Cd_Projeto
12	=	Departamento	Cd_Departamento
5	=	DepartamentoControlaProjeto	Cd_Departamento

Tabela 5.43 - Junções: Resultado Produzido pelo Algoritmo de Coleta de Junções

### Algoritmo de Coleta de Ordenações

**Para** cada Atributo da Lista de Ordenações **Faça**

**Se** atributo já existe na Estrutura de Dados Consulta **Então**

Consulta.OrderBy = ✓

**Fim Se**

**Senão**

Crie um registro na Estrutura de Dados Consulta

Consulta.Nome do Atributo = nome do atributo

Consulta.Nome da Tabela Virtual = nome da tabela virtual

Consulta.OrderBy = ✓

**Fim Senão**

**Fim Para**

Após a execução do Algoritmo de Coleta de Ordenações, o resultado produzido para o exemplo é:

Consulta									
Índice do Atributo Virtual	Nome da Tabela Virtual	Nome do Atributo Virtual	Selecionado	Função de Agregação	Filtro	Total de Filtros	Junção	OrderBy	Total de Junções
1	Empregado	Nm_Empregado	✓					✓	
2	Empregado	Cargo	✓						
3	Projeto	Nm_Projeto	✓					✓	
4	Departamento	Nm_Departamento			✓	1			
5	Departamento	Cd_Departamento					✓		2
6	Empregado	Cd_Departamento					✓		1
7	Empregado	Cd_Empregado					✓		1
8	EmpregadoTrabalhaProjeto	Cd_Empregado					✓		1
9	Projeto	Cd_Projeto					✓		2
10	EmpregadoTrabalhaProjeto	Cd_Projeto					✓		1
11	DepartamentoControlaProjeto	Cd_Projeto					✓		1
12	DepartamentoControlaProjeto	Cd_Departamento					✓		1

Tabela 5.44 - Consulta: Resultado Produzido pelo Algoritmo de Coleta de Junções

**Algoritmo de Coleta de Agrupamentos****Para** cada Atributo da Lista de Agrupamentos **Faça****Se** atributo já existe na Estrutura de Dados Consulta **Então**

Consulta.GroupBy = ✓

**Fim Se****Senão**

Crie um registro na Estrutura de Dados Consulta

Consulta.Nome do Atributo = nome do atributo

Consulta.Nome da Tabela Virtual = nome da tabela virtual

Consulta.GroupBy = ✓

**Fim Senão****Fim Para**

Como não foi definido nenhum agrupamento em nosso exemplo, o Algoritmo de Coleta de Agrupamentos não será utilizado. Contudo, seu funcionamento é semelhante aos do Algoritmo de Coleta de Ordenação.

## 5.5.2 Coletor de Fontes

### Objetivo:

Informar a(s) fonte(s) que pode(m) atender a cada item da estrutura de dados *Consulta*. Tabelas e atributos virtuais que participam de junções, somente podem ser mapeadas em tabelas e campos, respectivamente, pertencentes a uma mesma fonte de dados. Por exemplo, dada a equijunção *Professor.Código = Disciplina.Código do Professor*, onde *Professor.Código* pode ser mapeado para tabelas e atributos pertencentes às fontes de dados hipotéticas 1 e 2, e *Disciplina.Código do Professor* para uma tabela e campo pertencente à fonte 1, então, pelo que mencionamos acima, o Coletor de Fontes escolhe a fonte 1 para atender a ambos.

### Entrada:

- Estrutura de dados *Consulta*.

### Saída:

- Preenchimento do campo total de fontes, pertencente à estrutura de dados *Consulta*;
- Preenchimento dos campos fonte, tabela e atributo, referentes à estrutura de dados *Fontes*.

**Algoritmo:****Algoritmo de Coleta de Fontes de Dados**

**Para** cada Atributo Virtual da Estrutura de Dados Consulta **Faça**

Procure no Metabanco de Dados a(s) Fonte(s) que pode(m) atender ao Atributo Virtual (Consulta.Tabela Virtual, Consulta.Atributo Virtual) (Tabela 5.45)

Insira o nome das Fontes, Tabelas e Atributos que podem atender ao atributo virtual no Conjunto A

**Se** Consulta.Junção = ✓ **Então**

Procure no Metabanco de Dados a(s) Fonte(s) que pode(m) atender ao Atributo Virtual de Junção (Consulta.Tabela Virtual de Junção, Consulta.Atributo Virtual de Junção) (Tabela 5.47)

Insira o nome das Fontes, Tabelas e Atributos que podem atender ao atributo virtual de junção no Conjunto B

Faça a interseção dos conjuntos A e B pelo nome das fontes, inserindo o resultado no conjunto C

**Para** cada Fonte do Conjunto C **Faça**

Crie um registro na Estrutura de Dados Fontes

Consulta.Fontes.Fonte = nome da fonte

Consulta.Fontes.Tabela = nome da tabela

Consulta.Fontes.Atributo = nome do campo

Consulta.Total de Fontes = Consulta.Total de Fontes + 1

**Fim Para**

**Fim Se**

**Senão**

**Para** cada Fonte do Conjunto A que atenda ao atributo **Faça**

Crie um registro na Estrutura de Dados Fontes

Consulta.Fontes.Fonte = nome da fonte

Consulta.Fontes.Tabela = nome da tabela

Consulta.Fontes.Atributo = nome do campo

Consulta.Total de Fontes = Consulta.Total de Fontes + 1

**Fim Para**

**Fim Senão**

**Fim Para**

Para o nosso exemplo, precisamos encontrar as fontes de dados que *atendem* ao atributo *Departamento.Cd\_Departamento*. Para tal, a seguinte consulta ao Metabanco de Dados é criada pelo Seletor de Fontes:

```

SELECT DISTINCT B.Nm_Banco, T.Nm_Tabela, A.Nm_Atributo
FROM Banco B, Tabela T, Atributo A, Tabela_Virtual TV, Atributo_Virtual AV,
Tab_Virt_Mapeada_Tabela TVMT, Atrib_Virt_Mapeado_Atributo AVMA, Esquema_Global
EG, Esq_Glob_Possui_Tab_Virt EGPTV
WHERE T.Cd_Banco = B.Cd_Banco AND A.Cd_Tabela = T.Cd_Tabela AND
TV.Nm_Tabela_Virtual = 'Departamento' AND AV.Nm_Atributo_Virtual = 'Cd_Departamento'
AND TVMT.Cd_Tabela = T.Cd_Tabela AND TVMT.Cd_Tabela_Virtual =
TV.Cd_Tabela_Virtual AND A.Cd_Atributo = AVMA.Cd_Atributo AND
AV.Cd_Atributo_Virtual = AVMA.Cd_Atributo_Virtual AND EG.Nm_Esquema_Global =
'Organização' AND EG.Cd_Esquema_Global = EGPTV.Cd_Esquema_Global AND
EGPTV.Cd_Tabela_Virtual = TV.Cd_Tabela_Virtual
ORDER BY B.Nm_Banco

```

Tabela 5.45 - Recupera as Fontes de Dados que Atendem ao Atributo Virtual  
*Departamento.Cd\_Departamento*

Pelo resultado retornado pelo Metabanco de Dados (ver Tabela 5.46), podemos concluir que o atributo *Departamento.Cd\_Departamento* é *atendido* pelas fontes *Companhia* e *Empresa*.

Banco de Dados	Tabela	Atributo
Companhia	Departamento	Cd_Departamento
Empresa	Departamento	Cd_Departamento

Tabela 5.46 - Resultado da Consulta Recupera as Fontes de Dados que Atendem ao Atributo  
Virtual *Departamento.Cd\_Departamento*

De forma semelhante, é necessário encontrar as fontes de dados que *atendem* ao atributo *DepartamentoControlaProjeto.Cd\_Departamento*. Portanto, a seguinte consulta ao Metabanco de Dados é elaborada pelo Seletor de Fontes:

```

SELECT DISTINCT B.Nm_Banco, T.Nm_Tabela, A.Nm_Atributo
FROM Banco B, Tabela T, Atributo A, Tabela_Virtual TV, Atributo_Virtual AV,
Tab_Virt_Mapeada_Tabela TVMT, Atrib_Virt_Mapeado_Atributo AVMA, Esquema_Global
EG, Esq_Glob_Possui_Tab_Virt EGPTV
WHERE T.Cd_Banco = B.Cd_Banco AND A.Cd_Tabela = T.Cd_Tabela AND
TV.Nm_Tabela_Virtual = 'DepartamentoControlaProjeto' AND AV.Nm_Atributo_Virtual =
'Cd_Departamento' AND TVMT.Cd_Tabela = T.Cd_Tabela AND
TVMT.Cd_Tabela_Virtual = TV.Cd_Tabela_Virtual AND A.Cd_Atributo = AVMA.Cd_Atributo
AND AV.Cd_Atributo_Virtual = AVMA.Cd_Atributo_Virtual AND EG.Nm_Esquema_Global
= 'Organização' AND EG.Cd_Esquema_Global = EGPTV.Cd_Esquema_Global AND
EGPTV.Cd_Tabela_Virtual = TV.Cd_Tabela_Virtual
ORDER BY B.Nm_Banco

```

Tabela 5.47 - Recupera as Fontes de Dados que Atendem ao Atributo Virtual

*DepartamentoControlaProjeto.Cd\_Departamento*

Pelo que podemos observar na Tabela 5.48, que mostra o resultado retornado pelo Metabanco de Dados para a consulta da Tabela 5.47, o atributo virtual *DepartamentoControlaProjeto.Cd\_Departamento* somente é atendido pela fonte de dados *Empresa*.

Banco de Dados	Tabela	Atributo
Empresa	DepartamentoControlaProjeto	Cd_Departamento

Tabela 5.48 - Resultado da Consulta Recupera as Fontes de Dados que Atendem ao Atributo

*Virtual DepartamentoControlaProjeto.Cd\_Departamento*

Conforme dissemos no início da seção, as tabelas e atributos virtuais participantes de junções, somente podem ser mapeadas em tabelas e campos, respectivamente, pertencentes a uma mesma fonte de dados. Portanto, os atributos virtuais *Departamento.Cd\_Departamento* e *DepartamentoControlaProjeto.Cd\_Departamento* devem necessariamente ser atendidos por fontes de dados idênticas, visto que participam de uma junção. Em resumo, só podem ser atendidos pela fonte de dados *Empresa*. O processo é o mesmo para todos os outros atributos virtuais inseridos na consulta.

Para o nosso exemplo o resultado final produzido pelo Algoritmo de Coleta de Fontes de Dados é:

Consulta										
Índice do Atributo Virtual	Nome da Tabela Virtual	Nome do Atributo Virtual	Selecionado	Função de Agregação	Filtro	Junção	OrderBy	Total de Filtros	Total de Junções	Total de Fontes
1	Empregado	Nm_Empregado	✓				✓			2
2	Empregado	Cargo	✓							1
3	Projeto	Nm_Projeto	✓				✓			1
4	Departamento	Nm_Departamento			✓			1		2
5	Departamento	Cd_Departamento				✓			2	2
6	Empregado	Cd_Departamento				✓			1	2
7	Empregado	Cd_Empregado				✓			1	2
8	EmpregadoTrabalhaProjeto	Cd_Empregado				✓			1	1
9	Projeto	Cd_Projeto				✓			2	1
10	EmpregadoTrabalhaProjeto	Cd_Projeto				✓			1	1
11	DepartamentoControlaProjeto	Cd_Projeto				✓			1	1
12	DepartamentoControlaProjeto	Cd_Departamento				✓			1	1

Tabela 5.49 - Consulta: Resultado Produzido pelo Algoritmo de Coleta de Fontes de Dados

Fontes			
Índice do Atributo Virtual	Fonte de Dados	Tabela	Atributo
1	Companhia	Empregado	Nm_Empregado
1	Empresa	Funcionario	Nm_Funcionario
2	Companhia	Empregado	Cargo
3	Empresa	Projeto	Nm_Projeto
4	Companhia	Departamento	Nm_Departamento
4	Empresa	Departamento	Nm_Departamento
5	Companhia	Departamento	Cd_Departamento
5	Empresa	Departamento	Cd_Departamento
6	Companhia	Empregado	Cd_Departamento
6	Empresa	Funcionario	Cd_Departamento
7	Companhia	Empregado	Cd_Empregado
7	Empresa	Funcionario	Cd_Funcionario
8	Empresa	FuncionarioTrabalhaProjeto	Cd_Funcionario
9	Empresa	Projeto	Cd_Projeto
10	Empresa	FuncionarioTrabalhaProjeto	Cd_Projeto
11	Empresa	DepartamentoControlaProjeto	Cd_Projeto
12	Empresa	DepartamentoControlaProjeto	Cd_Departamento

Tabela 5.50 - Fontes: Resultado Produzido pelo Algoritmo de Coleta de Fontes de Dados

### 5.5.3 Estimador de Fontes

#### Objetivos:

- Realizar um levantamento estatístico de cada fonte de dados, através da divisão do número de atributos virtuais que uma determinada fonte possa atender pelo número de atributos virtuais selecionados em uma consulta;
- Verificar se os atributos virtuais que uma determinada fonte pode atender formam um subconjunto dos atributos virtuais que outra fonte pode atender. Neste caso, a primeira fonte é ignorada.

#### Entrada:

- Estrutura de dados *Fontes*.

#### Saída:

- Preenchimento dos campos fonte, porcentagem e subconjunto, referentes à estrutura de dados *Estatística*.

**Algoritmos:****Algoritmo de Preenchimento do Campo Porcentagem**

**Para** cada fonte da Estrutura de Dados Estatística **Faça**

Atributos Atendidos Pela Fonte = 0

**Para** cada atributo da Estrutura de Dados Consulta **Faça**

**Para** cada fonte em Consulta.Fontes **Faça**

**Se** Estatística.Fonte = Consulta.Fontes.Fonte **Então**

Incrementa(Atributos Atendidos Pela Fonte)

Estatística.Porcentagem := (Atributos Atendidos Pela Fonte/  
Total de Atributos de Consulta) \* 100

**Fim Se**

**Fim Para**

**Fim Para**

**Fim Para**

Seguindo o exemplo, o resultado produzido pelo Algoritmo de Preenchimento do Campo Porcentagem é:

Estatística	
Nome Fonte	Porcentagem
Companhia	50%
Empresa	92%

Tabela 5.51 - Porcentagem: Resultado Produzido pelo Algoritmo de Preenchimento do Campo Porcentagem

### Algoritmo de Preenchimento do Campo Subconjunto

Declare "Subconjunto" como Array de Inteiros

**Para** cada fonte da Estrutura de Dados Estatística **Faça**

**Para** cada Atributo Virtual da Estrutura de Dados Consulta **Faça**

**Se** Consulta.Fontes.Fonte = Estatística.Fonte **Então**

            Insira Número do Atributo Virtual em Subconjunto

**Fim Se**

**Fim Para**

**Fim Para**

**Para** cada Fonte do Array Subconjunto **Faça**

**Para** cada Fonte + 1 do Array Subconjunto **Faça**

**Se** Fonte <= Fonte + 1 **Então**

            Estatística[Fonte ].Subconjunto := Sim

**Fim Se**

**Senão**

**Se** Fonte >= Fonte + 1 **Então**

                Estatística[Fonte + 1].Subconjunto := Sim

**Fim Se**

**Fim Senão**

**Fim Para**

**Fim Para**

Para o nosso exemplo o resultado produzido pelo Algoritmo de Preenchimento do Campo Subconjunto é:

Estatística		
Nome Fonte	Porcentagem	Subconjunto
Companhia	50%	
Empresa	92%	

Tabela 5.52 - Porcentagem: Resultado Produzido pelo Algoritmo de Preenchimento do Campo Subconjunto

## 5.5.4 Seletor de Fontes

### Objetivo:

- Decidir para que(quais) fonte(s) deve(m) ser geradas subconsultas, baseado nos resultados levantados pelo Estimador de Fontes.

### Entrada:

- Estrutura de dados *Estatística*.

### Saída:

- Preenchimento do campo selecionada, referente à estrutura de dados *Estatística*.

### Algoritmos:

#### Algoritmo de Seleção de Fontes

**Para** cada Fonte da Estrutura de Dados Estatística **Faça**

**Se** Estatística.Porcentagem = 100 **Então**

        Estatística.Selecionada = ✓

**Fim Se**

**Fim Para**

**Para** cada Fonte da Estrutura de Dados Estatística **Faça**

**Se** Estatística.Subconjunto = Não **Então**

        Estatística.Selecionada = ✓

**Fim Se**

**Fim Para**

Para o nosso exemplo o resultado produzido pelo Algoritmo de Preenchimento do Campo Seleccionada é:

Estatística			
Nome Fonte	Porcentagem	Subconjunto	Seleccionada
Companhia	50%		✓
Empresa	92%		✓

Tabela 5.53 - Porcentagem: Resultado Produzido pelo Algoritmo de Preenchimento do Campo Seleccionada

## 5.5.5 Transformador de Consulta

### Objetivo:

Gerar as subconsultas a serem executadas, posteriormente, nas fontes de dados seleccionadas pelo Seletor de Fontes, utilizando as informações contidas no Metabanco de Dados. Para tal, o Transformador de Consultas mapeia as tabelas e atributos virtuais em tabelas e atributos, respectivamente, pertencentes às fontes de dados escolhidas, e, em seguida, gera as subconsultas necessárias para cada fonte. Após o mapeamento, a(s) subconsulta(s) fica(m) armazenada(s) no Metabanco de Dados, para que possa(m) ser executada(s) posteriormente em sua(s) fonte(s) correspondente(s).

### Entrada:

- Estrutura de dados *Consulta*;
- Estrutura de dados *Filtro*;
- Estrutura de dados *Junção*;
- Estrutura de dados *Estatística*.

### Saída:

- Descrição da consulta e subconsulta(s) geradas, a serem armazenadas no Metabanco de Dados.

### Algoritmos:

#### Algoritmo de Transformação de Consulta

**Para** cada Fonte da Estrutura de Dados Estatística **Faça**

**Se** Estatística.Selecionada = ✓ **Então**

        Cria Subconsulta(Fonte)

**Fim Se**

**Fim Para**

#### Algoritmo de Criação de Subconsulta (Fonte)

**Para** cada Atributo Virtual da Estrutura de Dados Consulta **Faça**

**Para** cada Fonte da Estrutura de Dados Fontes **Faça**

**Se** Consulta.Fontes.Fonte = Fonte **Então**

**Se** Consulta.Selecionado = ✓ **Então**

                Cria Select(SELECT, Atributo Virtual, Fonte)

**Fim Se**

**Se** Consulta.Filtro = ✓ **Então**

            Cria Filtro(WHERE, Atributo Virtual, Fonte)

**Fim Se**

**Se** Consulta.Juncao = ✓ **Então**

            Cria Junção(WHERE, Atributo Virtual, Fonte)

**Fim Se**

**Se** Consulta.OrderBy = ✓ **Então**

            Cria OrderBy(ORDERBY, Atributo Virtual, Fonte)

**Fim Se**

**Se** Consulta.GroupBy = ✓ **Então**

            Cria GroupBy(GROUPBY, Atributo Virtual, Fonte)

**Fim Se**

        Cria From(FROM, Atributo Virtual, Fonte)

**Fim Se**

**Fim Para**

**Fim Para**

```
Subconsulta = SELECT + FROM + WHERE + GROUPBY + ORDERBY
```

Para armazenar a descrição da consulta do nosso exemplo, o Integrador, mantendo interface com o Gerente de Metadados, utiliza a tabela *Consulta* do Metabanco de Dados, produzindo o seguinte comando SQL:

```
INSERT INTO Consulta VALUES (1, 'Empregados, com respectivos cargos, trabalhando em projetos controlado pelo departamento de Informática', 'Esta consulta recupera os empregados, com respectivos cargos, trabalhando em projetos gerenciados pelo departamento de Informática', 'Administrador', SYSDATE)
```

Tabela 5.54 – Comando SQL Referente ao Armazenamento da Descrição da Consulta

Para cada consulta produzida pelo Algoritmo de Transformação de Consulta, o Integrador formula um comando SQL que vai inseri-la no Metabanco de Dados, mais precisamente na tabela *Subconsulta*. No nosso exemplo, é gerada uma subconsulta para cada uma das fontes de dados – *Companhia* e *Empresa*. Os comandos SQL para armazenamento das subconsultas no Metabanco de Dados, podem ser vistos nas tabelas 5.55 e 5.56.

```
INSERT INTO Subconsulta VALUES (1, 'SELECT Empregado.Nm_Empregado, Empregado.Nm_Cargo FROM Empregado, Departamento WHERE Empregado.Cd_Departamento = Departamento.Cd_Departamento AND Departamento.Nm_Departamento = 'Informática' ORDER BY Empregado.Nm_Empregado', 1, 4)
```

Tabela 5.55 – Subconsulta Referente à Fonte de Dados *Companhia*

```
INSERT INTO Subconsulta VALUES (2, 'SELECT Funcionario.Nm_Funcionario,
Projeto.Nm_Projeto
FROM Funcionario, Departamento, Projeto, DepartamentoControlaProjeto,
FuncionarioTrabalhaProjeto
WHERE Funcionario.Cd_Departamento = Departamento.Cd_Departamento AND
Funcionario.Cd_Funcionario = FuncionarioTrabalhaProjeto.Cd_Funcionario AND
Projeto.Cd_Projeto = FuncionarioTrabalhaProjeto.Cd_Projeto AND Projeto.Cd_Projeto =
DepartamentoControlaProjeto.Cd_Projeto AND
DepartamentoControlaProjeto.Cd_Departamento = Departamento.Cd_Departamento AND
Departamento.Nm_Departamento = 'Informática' ORDER BY Projeto.Nm_Projeto,
Funcionario.Nm_Funcionario', 1, 5)
```

Tabela 5.56 – Subconsulta à Fonte de Dados *Empresa*

## 5.6 Combinador de Resultados

### Objetivo:

O Combinador de Resultados é um componente presente no Módulo de Execução de Consultas que desempenha as seguintes funções:

- Recuperar, a partir do Metabanco de Dados, as subconsultas referentes a uma determinada consulta selecionada pelo UG;
- Receber os resultados oriundos da execução das subconsultas nas fontes de dados locais e combina-los, eliminando quaisquer tipos de conflitos de esquema ou de dados que possam existir entre os conjuntos-resposta.

Este componente foi construído através de "scripts" inseridos em páginas "Web". Nos "scripts" estão contidas consultas ao Metabanco de Dados.

### Entrada:

Código da consulta selecionada pelo UG.

### Saída:

Conjunto-resposta final, livre de qualquer tipo de conflito, resultante da combinação do conjunto-resposta de cada fonte de dados local em questão.

### Algoritmos:

O algoritmo apresentado na seqüência obtém as subconsultas correspondentes a uma determinada consulta selecionada e remete-as as suas respectivas fontes de dados.

#### Algoritmo de Obtenção e Execução de Subconsultas

Código da Consulta = código da consulta selecionada pelo UG

Selecione, a partir do Metabanco de Dados, as subconsultas referentes à consulta identificada por Código da Consulta (Tabela 5.57)

**Para** cada subconsulta recuperada **Faça**

##### Caso Fonte Seja

Caso 0: Fonte0 = Execute a consulta na fonte de dados 0

Caso 1: Fonte1 = Execute a consulta na fonte de dados 1

Caso 2: Fonte2 = Execute a consulta na fonte de dados 2

...

Caso  $n$ : Fonten := Execute a consulta na fonte de dados  $n$

##### Fim Caso

**Fim Para**

De acordo com o nosso exemplo, para recuperar as subconsultas referentes à consulta *Empregados, com respectivos cargos, trabalhando em projetos controlado pelo departamento de Informática*, o Combinador de Resultados elabora a seguinte consulta:

```
SELECT Cd_Subconsulta, Vl_Subconsulta, Cd_Fonte
FROM Subconsulta
WHERE Subconsulta.Cd_Consulta = 1
ORDER BY Subconsulta.Cd_Subconsulta
```

Tabela 5.57 – Consulta que Recupera as Subconsultas Referentes a uma Consulta

Cada subconsulta recuperada é enviada ao Gerente de Metadados que, por sua vez, as remete as suas respectivas fontes de dados. No nosso exemplo, as subconsultas

das Tabelas 5.55 e 5.56 são recuperadas. Os resultados retornados por cada fonte de dados podem ser visualizados nas Tabelas 5.58 e 5.59.

Nm_Empregado	Cargo
Francisco M de Souza	DBA
Jorge C Silva	Desenvolvedor

Tabela 5.59 - Resultado Retornado pelo Banco de Dados *Companhia*

Nm_Funcionario	Nm_Projeto
Francisco Maturana de Souza	Alpha 1
Jorge Castro Silva	Alpha 1

Tabela 5.59 - Resultado Retornado pelo Banco de Dados *Empresa*

Após a execução de cada subconsulta em sua respectiva fonte dados, os conjuntos-resposta são retornados ao Gerente de Metadados que os devolve ao Combinador de Resultados. Este último aplica sobre os mesmos o Algoritmo de Resolução de Conflitos:

### Algoritmo de Resolução de Conflitos

**Para** cada conjunto-resposta Fonte **Faça**

**Para** cada nome de atributo **Faça**

        /\* Resolve conflitos de nome de atributo \*/

        Substituir o nome do atributo pelo nome atributo virtual correspondente

**Para** cada valor do atributo **Faça**

        /\* Resolve conflitos de expressões distintas e precisões diferentes \*/

**Se** valor do atributo deve ser mapeado **Então**

            Mapeie o valor para o novo valor

**Fim Se**

        /\* Resolve conflitos de tipo de dado \*/

**Senão**

**Se** tipo de dados do atributo diferente do tipo de dado do atributo virtual correspondente **Então**

**Caso** tipo de dado do atributo virtual **Seja**

                    Caso "Int": Converter valor do atributo para Inteiro

                    Caso "Char" ou "Varchar": Converter valor do atributo para String

                    Caso "Datetime" ou "Smalldatetime": Converter valor do atributo para

Data

                    Caso "Bit": Converter valor do atributo para Boolean

                    Caso "Money": Converter valor do atributo para Valor Monetário

**Fim Caso**

**Fim Se**

**Fim Senão**

**Fim Para**

**Fim Para**

**Fim Para**

O nome dos atributos contidos no conjunto-resposta retornado por cada fonte de dados deve ser novamente mapeado no nome do atributo virtual que os representa no esquema global. Por exemplo, no conjunto-resposta retornado pela fonte de dados *Empresa*, o nome do atributo que representa o nome dos funcionários (*Funcionario.Nm\_Funcionario*) deve ser mapeado no nome do atributo virtual que representa o nome dos empregados (*Empregado.Nm\_Empregado*).

Para obter o nome do atributo virtual, presente no esquema global *Organização*, que representa o atributo *Funcionario.Nm\_Funcionario* do esquema *Empresa*, o Combinador de Resultados remete a seguinte consulta ao Gerente de Metadados:

```
SELECT DISTINCT AV.Nm_Atributo_Virtual
FROM Atributo A, Atributo_Virtual AV, Tipo T, Atrib_Virt_Mapeado_Atributo AVMA
WHERE AV.Cd_Tipo = T.Cd_Tipo AND AVMA.Cd_Atributo_Virtual =
AV.Cd_Atributo_Virtual AND AVMA.Cd_Atributo = A.Cd_Atributo AND A.Nm_Atributo =
'Nm_Funcionario'
```

Tabela 5.60 – Obtenção do Nome do Atributo Virtual Referente ao Atributo  
*Funcionario.Nm\_Funcionario*

Como resultado, o Combinador de Resultados recebe a seguinte tabela:

Nm_Atributo_Vitual
Nm_Empregado

Tabela 5.61 – Nome do Atributo Virtual que Representa o Atributo do Esquema Local  
*Funcionario.Nm\_Funcionario*

Para resolver os Conflitos de Tipos de Dados, é necessário recuperar o tipo do atributo virtual do esquema global e do atributo do esquema local. Caso os mesmos sejam idênticos, então nenhum tipo de conversão necessita ser realizada nos dados referentes ao atributo. Caso contrário, os dados referentes ao atributo do esquema local devem ser convertidos para o tipo do atributo virtual.

No exemplo, o conjunto resposta retornado pelo esquema *Companhia* contém o atributo *Funcionario.Nm\_Funcionario*, que é do tipo Char. O atributo virtual (*Empregado.Nm\_Empregado*) que o representa no esquema global *Organização* é do tipo Varchar. Sendo assim, é necessária a realização de uma conversão de Char para Varchar nos dados referentes ao atributo do esquema local.

Para recuperar o tipo de dados do atributo virtual do esquema global *Organização Empregado.Nm\_Empregado* e do atributo *Empregado.Nm\_Empregado* do esquema local *Companhia*, o Combinador de Resultados envia a seguinte consulta ao Gerente de Metadados:

```

SELECT DISTINCT T2.Nm_Tipo, T1.Nm_Tipo
FROM Atributo A, Atributo_Virtual AV, Tipo T1, Tipo T2, Tabela T, Tabela_Virtual TV,
Atrib_Virt_Mapeado_Atributo AVMA, Esquema_Global EG, Esq_Glob_Possui_Tab_Virt
EGPTV
WHERE A.Cd_Tabela = T.Cd_Tabela AND A.Cd_Tipo = T1.Cd_Tipo AND
AV.Cd_Tabela_Virtual = TV.Cd_Tabela_Virtual AND AV.Cd_Tipo = T2.Cd_Tipo AND
AVMA.Cd_Atributo = A.Cd_Atributo AND AVMA.Cd_Atributo_Virtual =
AV.Cd_Atributo_Virtual AND A.Nm_Atributo = 'Nm_Empregado' AND T1.Nm_Tipo <>
T2.Nm_Tipo AND EG.Nm_Esquema_Global = 'Organização' AND EG.Cd_Esquema_Global =
EGPTV.Cd_Esquema_Global AND EGPTV.Cd_Tabela_Virtual = TV.Cd_Tabela_Virtual AND
AV.Nm_Atributo_Virtual = 'Nm_Empregado' AND TV.Nm_Tabela_Virtual = 'Empregado'
AND T.Nm_Tabela = 'Empregado'

```

Tabela 5.62 – Obtenção do Tipo de Dados do Atributo Virtual *Empregado.Nm\_Empregado* e do Atributo do Esquema Local *Empregado.Nm\_Empregado*

Como resposta, o Combinador de Resultados recebe as seguintes informações:

Nm_Tipo	Nm_Tipo
Char	Varchar

Tabela 5.63 - Tipo de Dados do Atributo Virtual *Empregado.Nm\_Empregado* e do Atributo *Empregado.Nm\_Empregado* do Esquema *Companhia*

Ainda em relação ao nosso exemplo, o empregado *Francisco Maturana de Souza* é identificado como *Francisco M de Souza* no banco de dados *Companhia* e como *Francisco Maturana de Souza* no banco *Empresa*. Para resolver o Conflito de Expressões Diferentes, deve ser realizado um mapeamento de valores com base nas informações contidas no Metabanco de Dados, isto é, neste caso, o valor *Francisco M de Souza* deve ser mapeado para *Francisco Maturana de Souza*. Para obter tal mapeamento, o Combinador de Resultados envia a seguinte consulta ao Gerente de Metadados:

```

SELECT D1.Nm_Dado
FROM Atributo AS A, Atributo_Dado AS AD, Dado AS D1, Dado AS D2,
Dado_Mapeado_Dado AS DMD
WHERE A.Nm_Atributo = 'Nm_Empregado' AND A.Cd_Atributo = AD.Cd_Atributo AND
AD.Cd_Dado = D1.Cd_Dado AND D1.Cd_Dado = DMD.Cd_Dado_Mapeado AND
D2.Nm_Dado = 'Francisco M de Souza' AND D2.Cd_Dado = DMD.Cd_Dado

```

Tabela 5.64 – Obtendo o mapeamento do valor *Francisco M de Souza*

Como resposta, o Combinador de Resultados recebe a seguinte informação:

Nm_Dado
Francisco Maturana de Souza

Tabela 5.65 – Resultado do Mapeamento do valor *Francisco M de Souza*

Finalmente, após a resolução de todos os conflitos citados acima, o Combinador de Resultados dispõe dos seguintes conjuntos-resposta:

Nm_Empregado	Cargo
Francisco Maturana de Souza	DBA
Jorge C Silva	Desenvolvedor

Tabela 5.66 – Conjunto-Resposta do Banco de Dados *Companhia* Após a Eliminação de Conflitos

Nm_Empregado	Nm_Projeto
Francisco Maturana de Souza	Alpha 1
Jorge Castro Silva	Alpha 1

Tabela 5.67 – Conjunto-Resposta do Banco de Dados *Empresa* Após a Eliminação de Conflitos

A combinação de ambos os conjuntos-resposta, resulta nas informações contidas na Tabela 5.68.

Nm_Empregado	Cargo	Nm_Projeto
Francisco Maturana de Souza	DBA	Alpha 1
Jorge Castro Silva	Desenvolvedor	Alpha 1

Tabela 5.68 – Conjunto-Resposta Final

## 5.7 Conclusões

Neste capítulo, apresentamos em detalhes cada componente da Arquitetura BDRV. Para facilitar o entendimento utilizamos o exemplo 3 do Capítulo 4. Pudemos observar claramente o funcionamento dos algoritmos de cada componente. As entradas e saídas de cada um também foram analisadas. Em resumo, pudemos perceber o funcionamento individual de cada componente desde a formulação de uma consulta até a apresentação de seus resultados.

# Capítulo 6

## *Conclusões e Trabalhos Futuros*

### 7.1 Conclusões

Em nosso trabalho, apresentamos uma solução eficiente, chamada de Sistema BDRV, para integração de múltiplas fontes de dados, exclusivamente bancos de dados relacionais, baseada em uma abordagem híbrida (sob demanda e antecipada), proporcionando transparência em relação à localização de fontes de dados através de uma interface "Web".

O Sistema BDRV procurou incorporar pontos fortes de outras tecnologias de integração estudadas, das duas abordagens, assim como solucionar problemas presentes nestas mesmas arquiteturas.

Verificamos que a integração de múltiplos bancos de dados é uma atividade extremamente complexa, em virtude da presença de discrepâncias nos esquemas locais, sendo tema suficiente para várias dissertações de mestrado. Nosso propósito com relação a este ponto foi trabalhar somente com formas singelas de integração de dados, cujas soluções são já bem conhecidas.

Com relação aos objetivos *traçados* pelo Sistema BDRV, concluímos que:

#### **Desempenho**

Em sistemas inspirados na abordagem sob demanda (mediadores), o desempenho está diretamente relacionado com as três etapas de processamento de consultas (Seção 2.1): Seleção das Fontes de Dados; Tradução/Processamento da Consulta; Integração dos Resultados Parciais.

Em nosso Estudo de Caso (ver Anexo A), integramos três bancos de dados do Departamento de Sistemas de Computação (DSC). Embora todos os bancos de dados estejam em produção, nenhum deles armazena uma grande quantidade de dados. Calculamos uma média de 500 (quinhentos) registros para cada um.

Durante o desenvolvimento de nosso Estudo de Caso, calculamos a média da porcentagem de tempo utilizada por cada etapa de processamento de consulta desde o processo de geração e transformação de consultas até a obtenção de resultados. Em um sistema que integre um número maior de fontes de dados e que as mesmas apresentem uma quantidade maior de dados, os resultados podem ser diferentes. Os resultados obtidos foram os seguintes:

Etapa	Tempo (%)
Seleção de Fontes de Dados	15%
Tradução de Consulta	10%
Processamento de Consulta	25%
Integração de Resultados Parciais	50%

Tabela 6.1 – Tempos Aproximados de Cada Etapa de Processamento de Consultas

No Sistema BDRV as consultas são geradas antecipadamente, ou seja, as etapas de Seleção de Fontes de Dados e Tradução de Consulta não necessitam ser realizadas no momento em que uma consulta é acionada pelo Usuário Global (UG). No entanto, conforme podemos observar na Tabela 6.1, estas etapas representam apenas 25% do tempo total para o processamento de uma consulta. Concluímos que o fato do Sistema BDRV proporcionar a elaboração e tradução de consultas de modo antecipado não significa um ganho significativo no que se refere ao desempenho.

Alguns dos conflitos que o Sistema BDRV pretende solucionar, citados no Capítulo 3, como por exemplo, Nome de Atributo, Tipo de Dados, Expressões e Precisões Diferentes, somente podem ser solucionados após as subconsultas serem enviadas as suas respectivas fontes de dados e as mesmas retornarem seus conjuntos-resposta.

Para cada dado pertencente a um conjunto-resposta retornado por uma fonte de dados, todos os conflitos citados anteriormente são analisados. Para solucionar o conflito de Expressões Diferentes, por exemplo, é necessária, para cada dado, a

realização de uma consulta ao Metabanco de Dados (ver Seção 5.7). Independente do fato de que o dado em questão deva ou não ser mapeado em outro dado, esta consulta necessariamente é realizada. Observe que o mesmo acontece para a resolução dos outros conflitos: Nome de Atributo, Tipo de Dados e Precisões Diferentes.

Em termos quantitativos, suponhamos que uma consulta é elaborada e transformada em subconsultas destinadas a 3 (três) fontes de dados (uma subconsulta para cada uma). Imaginemos também que cada fonte de dados retorne uma média de 30 (trinta) registros e que cada registro contenha uma média de 5 (cinco) atributos.

Pelo anteriormente mencionado, para resolver os conflitos de Nome de Atributo, Tipo de Dados, Expressões e Precisões Diferentes são necessárias a execução de  $3$  (fontes de dados)  $\times$   $30$  (total de registros retornados por uma fonte de dados)  $\times$   $5$  (número de atributos contidos em cada registro) = 450 consultas auxiliares ao Metabanco de Dados.

Justificamos o fato de que a etapa de Integração de Resultados Parciais representa metade do tempo gasto no processamento de uma consulta (Tabela 6.1) pelo excessivo e inaceitável número de consultas auxiliares ao Metabanco de Dados. O algoritmo Combinador de Resultados (Seção 5.7) representa uma potencial perda de desempenho no que diz respeito ao processamento de consultas no Sistema BDRV.

Na tecnologia de "Data Warehousing", inspirada na abordagem antecipada para integração de dados, os dados de cada fonte são extraídos antecipadamente, traduzidos, filtrados e integrados apropriadamente, e armazenados em um "Data Warehouse" (DW). Quando uma consulta é submetida, a mesma é avaliada diretamente no DW, sem a necessidade, portanto, de ter acesso às várias fontes de dados em tempo real.

Como DWs podem ser considerados fontes de dados locais no Sistema BDRV, algumas consultas podem ser respondidas unicamente pelos mesmos, sem a necessidade de consultar outras fontes, visto que já armazenam um resumo pré-computado de dados contidos em outras fontes de dados. Isto também implica em um ganho de desempenho.

## Abordagem Híbrida

Diferentemente de outras tecnologias de integração de fontes de dados, no Sistema BDRV, as abordagens sob demanda e antecipada não são tratadas de forma ortogonal.

Considerar DWs como fontes de dados locais não acarreta nenhuma desvantagem, pelo contrário, obtém-se ganho de desempenho no processamento de consultas, conforme observarmos anteriormente.

## Interface "Web"

Diferentemente de outras tecnologias de integração de fontes de dados, concluímos que a tecnologia "Web" apresenta recursos suficientes para proporcionar transparência em relação à localização de fontes de dados locais e, portanto, não existe razão para tratar ambas as tecnologias de maneira ortogonal.

No Sistema BDRV, diferentemente da execução de consultas, a elaboração de consultas não é realizada através de uma interface "Web". Admitimos que não existe razão para tratar ambas as etapas utilizando interfaces distintas.

## 7.2 Trabalhos Futuros

Dadas as dificuldades enfrentadas para a construção manual de esquemas globais, um trabalho que pretendemos encaminhar é a construção de um módulo *Gerador de Esquema Global Virtual*, para gerar automaticamente um banco de dados virtual, a partir de esquemas locais de bancos de dados já existentes.

Outro aperfeiçoamento no Sistema BDRV é o suporte a atualizações de fontes locais, a partir de um esquema global virtual.

Um outro trabalho interessante é oferecer a oportunidade de elaboração e execução de consultas através da "Web".

Finalizando, um outro trabalho futuro seria o desenvolvimento de um componente Combinador de Resultados mais eficiente do que o atual, no que se trata da rapidez na resolução de conflitos e integração de conjuntos-resposta.

- Tentaremos minimizar o problema da potencial ineficiência na obtenção de resultados, no que diz respeito ao tempo de resposta de uma consulta, encontrado nos SBDMs, realizando o processo de tradução de consultas de forma antecipada.

Algumas das vantagens encontradas nos SBDMs a serem incorporadas na Arquitetura BDRV são:

- Visão personalizada do esquema global, ou seja, baseada nas necessidades de informação de cada usuário global;
- Transparência em relação à localização das fontes.

A Arquitetura BDRV também apresenta suas desvantagens, dentre as quais podemos destacar:

- possibilidade de integração de um único tipo de fonte de dados: bancos de dados;
- os bancos de dados locais devem necessariamente ser baseados no modelo relacional;
- não permite a realização de atualizações sobre os dados armazenados nos bancos de dados locais;
- impossibilidade de realização de consultas "ad-hoc", por parte dos usuários globais;

potencial ineficiência na obtenção de resultados, no que diz respeito ao tempo de resposta de uma consulta, em razão de sua arquitetura puramente distribuída.

# Bibliografia

- [1] Berners-Lee, T., Connolly, D., "HyperText Markup Language Specification", IETF HTML Working Group, 1994
- [2] R. Elmasri and S. Navathe, "Fundamentals of Database Systems", Benjamin/Cummings Publishing Co., 3<sup>rd</sup> Edition, 1999
- [3] Berners-Lee T., Cailliau R., Loutonen A., "The World Wide Web", Communications of the ACM, Vol. 37(8), 1994, pp. 76-82
- [4] Gravano L., Papakinstatinou Y., "Mediating and Metasearching on the Internet", ACM Sigmod Record (<http://www.acm.org/sigmod/record>)
- [5] Chung, Soon M., Mah, Pyeong S., "Schema Integration for Multidatabases Using Relational and Object-Oriented Model", ACM, 1995 (<http://www.acm.org/>)
- [6] Ceri, S., Pelagatti, G., "Distributed Databases: Principles and Systems", 1984, McGraw-Hill, New York
- [7] Berners-Lee, T., "Hypertext Transfer Protocol", Internet Working Draft, 5 November 1993
- [8] Lima, I., "O Ambiente Web Banco de Dados: Funcionalidades e Arquiteturas de Integração", Dissertação de Mestrado, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Agosto de 1997
- [9] Silberschatz, A., Korth, Henry F., Sudarshan, S., "Sistema de Banco de Dados", São Paulo, Makron Books, 1999
- [10] McLeod, D., Heimbigner, D., "A Federated Architecture for Data Base Systems", 1980, In Proceedings of the AFIPS National Computer Conference, vol. 39, AFIPS Press, Arlington, Va.
- [11] Frank, M., "Database and the Internet", DBMS On-line Magazine, December, 1995

- [12] Batini, C., Lenzerini, M., "A Comparative Analysis of Methodologies for Database Schema Integration", ACM Computing Surveys, Vol. 18, No. 14, December 1986
- [13] Wiederhold, G., "Mediators in the Architecture of Future Information Systems", IEEE Computer, 25(3):38-49, 1992
- [14] Bright, M. W., Hurson, A. R., Pakzad, Simin H., "A Taxonomy and Current Issues in Multidatabase Systems", IEEE Computer, 25(3):50-60, 1992
- [15] Motro, A., "Superviews: Virtual Integration of Multiple Databases", IEEE Transactions on Software Engineering, vol. SE-13, No. 7, July 1987
- [16] Bodorik, P., Riordon, J. S., "System Integration in Multidatabases", SIGSMALL/PC Symposium 1990: 160-163
- [17] Ram, S., "Heterogeneous Distributed Database Systems", ACM Computing Surveys, Vol.22, No. 3 (Sept. 1990), pp. 237-266
- [18] Kimball, R., "The Data Warehouse Toolkit - Técnicas para construção de DWs Dimensionais", Makron Books, 1998
- [19] Glossbrenner, A., Glossbrenner, E., "Internet: truques espertos", Axccl Books, 1994
- [20] Wells, D., "Wrappers", Survey (<http://www.objs.com/survey/wrap.htm>)
- [21] Kim, W., "Modern Database Systems: the object model, interoperability, and beyond", Addison-Wesley, 1995
- [22] Kim, W., Seo, J., "Classifying Schematic and Data Heterogeneity in Multidatabase Systems", IEEE Computer, December, 1991
- [23] Sheth, A., Larson, J., "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases", ACM Computing Surveys, Vol. 22, N 3, September, 1990
- [24] Papakonstantinou, Y., Abiteboul, S., Molina, G., "Object Fusion in Mediator Systems", Proceedings of the 22<sup>nd</sup> International Conference on Very Large Data Bases, Mumbai (Bombay), India, 1996
- [25] Stein, D., "How to Set Up and Maintain a Web Site",

- Addison-Wesley, 1995
- [26] Gardner, Stephen R., "Building the Data Warehouse", Communications of the ACM, Vol 41, No 9, Setembro 1998.
- [27] Landers, T., Rosenberg, R., "An Overview of Multibase", Tutorial: Distributed Database Management, IEEE Computer Society 575, 556-579
- [28] Kimball, Ralph, Reeves, Laura, Margy, Ross, Thornthwaite, Warren, "The Data Warehouse Lifecycle Toolkit : Experts Methods for Designing, Developing and Deploying Data Warehouses", John Wiley & Sons Inc., 1998.
- [29] Inmon, W.H., "Como Construir o Data Warehouse", Tradução da Segunda Edição, Editora Campus, 2a Edição, 1997.
- [30] Widom, Jennifer, "Research Problems in Data Warehousing", Proc. 4th Int'l Conference on Information and Knowledge Management (CIKM), Novembro 1995.
- [31] Dayal, U., Hwang, H., "View Definition and Generalization for Database Integration in Multidatabase System", IEEE Trans. Softw. Eng. SE-10, 6 (Nov), 628-644
- [32] Thomsen, E., "OLAP Solutions: Building Multidimensional Information Systems", John Wiley & Sons Inc., USA, 1997.
- [33] Chiavenato, I., "Teoria Geral da Administração", Volume 2, 4ª Edição, Makron Books, 1993
- [34] Elmagarmid, A., Pu, C., "Guest Editors' Introduction to the Special Issue on Heterogeneous Databases", ACM Computing Surveys, Vol. 22, No. 3, September 1990
- [35] Hull, R., Zhou, G., "A Framework for Supporting Data Integration Using the Materialized and Virtual Approaches", SIGMOD '96 6/96 Montreal, Canada
- [36] Berners-Lee, T., Masinter, L., McCahiel, M., "Uniform Resource Locators (URL)", RFC 1738, University of Minnesota, 1994
- [37] Weissinger, A., "ASP in a Nutshell", O'Reilly & Associates, Paperback, 2nd edition, Published July 2000

# Anexo A

## *Estudo de Caso*

Nosso estudo de caso envolve a integração de três fontes de dados, localizadas no Departamento de Sistemas e Computação (DSC), da Universidade Federal da Paraíba (UFPB), Campus II. Todas as fontes armazenam basicamente dados de nosso ambiente acadêmico.

Este estudo de caso pretende discutir detalhes de características já abordadas, além de outras ainda não citadas de forma mais objetiva e direcionadas ao sistema. A efetiva implementação a ser apresentada foi bastante útil também para levantar muitos problemas funcionais. Na realidade, a motivação para este trabalho ocorreu durante o desenvolvimento da aplicação.

### **6.1 Descrição da Aplicação**

O objetivo principal da aplicação é propiciar aos professores, alunos e funcionários do DSC, um sistema de consulta à informações de âmbito acadêmico. A aplicação foi desenvolvida no Laboratório de Sistemas de Informação (LSI), pertencente ao DSC. As fontes estão localizadas em terminais da rede NT instalada no LSI.

As fontes utilizadas para integração em nossa aplicação são as seguintes:

- Sistema de Seminários do DSC - BDSEMINAR

Descrição: sistema de controle de seminários do DSC. Neste sistema, encontram-se armazenados todos os seminários apresentados no DSC nos anos de 1998 a 2000. Informações sobre cada palestrante também podem ser encontradas.

Tipo da fonte: Banco de Dados SQL Server 6.5

Localização: máquina *lsi\_sql*

- "Datawarehouse" da COPIN - DW COPIN

Descrição: este sistema, construído especialmente para nossa aplicação, corresponde a um pequeno "Data Warehouse" cujos dados são extraídos a partir de dados armazenados no SYSCOPIN (ver a seguir).

Tipo da Fonte: Banco de Dados SQL Server 6.5

Localização: máquina *lsi\_station1*

- Sistema de Informações da COPIN - SYSCOPIN

Descrição: é o sistema de informações da Coordenação de Pós-Graduação em Informática (COPIN). Armazena dados relativos a professores, alunos, teses, artigos, disciplinas, entre outros.

Tipo da Fonte: Banco de Dados SQL Server 6.5

Localização: máquina *lsi\_pdc*

- Metabanco de Dados

Tipo da Fonte: Banco de Dados SQL Server 6.5

Localização: máquina *lsi\_pdc*

Como servidor "Web", escolhemos o "Internet Information Server" (IIS 4.0), da Microsoft, em virtude de estarmos utilizando a tecnologia "Active Server Pages" (ASP). Este encontra-se instalado na máquina *lsi\_pdc*.

A descrição detalhada de cada banco de dados utilizado em nossa aplicação pode ser encontrada no Anexo B.

## 6.2 Elaboração de Consulta

Imagine que estamos interessados em obter alguns dados elementares a respeito dos alunos da Copin cujo período de entrada seja 981. Os dados a serem obtidos são: *nome, orientador e e-mail*.

O primeiro passo é selecionar o esquema global no qual se deseja elaborar a consulta. Neste caso, devemos escolher o esquema global *COPIN*, conforme podemos observar na Figura 6.1.

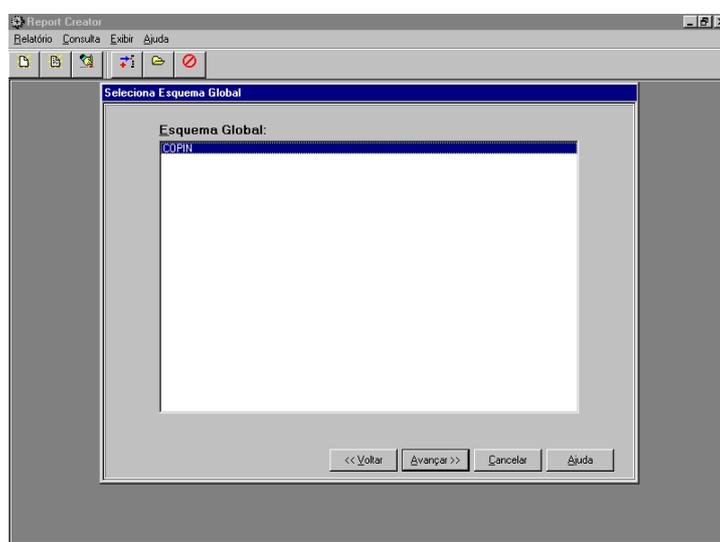
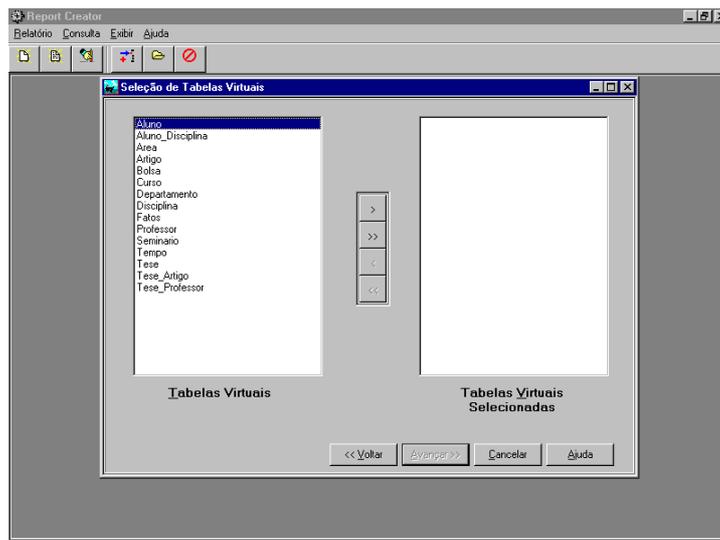
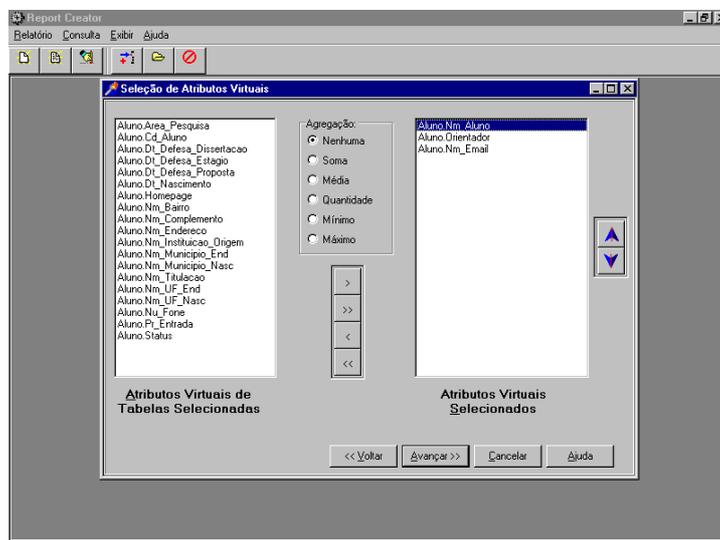


Figura 6.1 - Seleção do Esquema Global *COPIN*

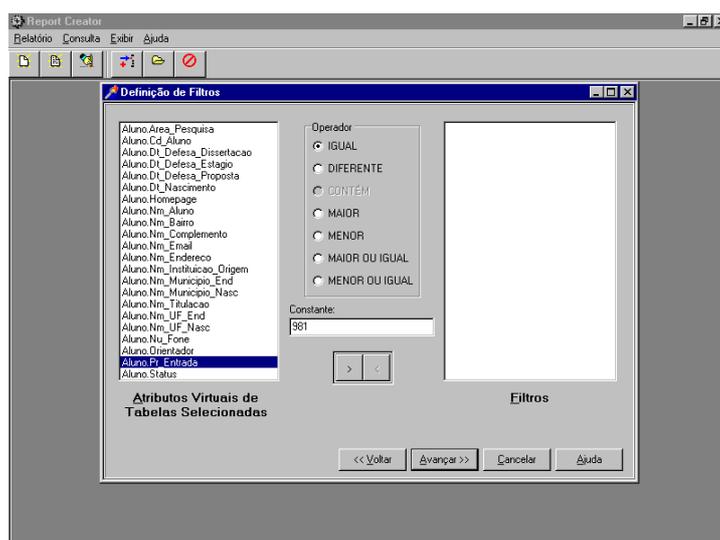
O segundo passo é a seleção das tabelas virtuais a serem incluídas na consulta. De acordo com a consulta que desejamos elaborar, devemos selecionar apenas a tabela virtual *Aluno*, conforme podemos verificar na Figura 6.2.

Figura 6.2 - Seleção da Tabela Virtual *Aluno*

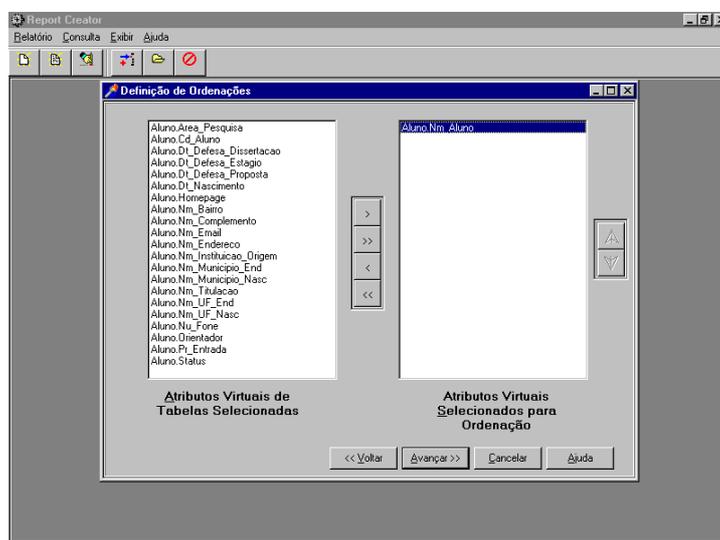
O terceiro passo é a seleção dos atributos virtuais que desejamos obter como resposta. Baseados em nossa consulta podemos observar que é necessário a seleção dos atributos virtuais *Aluno.Nm\_Aluno*, *Aluno.Orientador* e *Aluno.Nm\_Email*. Na Figura 6.3, é possível acompanhar esta etapa.

Figura 6.3 - Seleção dos Atributos Virtuais *Aluno.Nm\_Aluno*, *Aluno.Orientador* e *Aluno.Nm\_Email*

O quarto passo é a definição de filtros em atributos virtuais pertencentes às tabelas virtuais selecionadas. De acordo com nossa consulta precisamos definir o filtro: *Aluno.Pr\_Entrada IGUAL 981*. Na Figura 6.4, é possível observar a definição do filtro.

Figura 6.4 - Definição do Filtro *Aluno.Pr\_Entrada IGUAL 981*

O quinto passo é a definição de ordenações em atributos pertencentes às tabelas virtuais selecionadas no passo três. De acordo com nossa consulta precisamos definir apenas uma ordenação no atributo *Aluno.Nm\_Aluno*. Este passo pode ser visto na Figura 6.5.

Figura 6.5 - Definição da Ordenação *Aluno.Nm\_Aluno*

O sexto passo é a definição de agrupamentos, que na nossa consulta não será necessária.

Finalmente, o sétimo passo é simplesmente fornecer algumas informações em relação à consulta, conforme podemos observar na Figura 6.6.

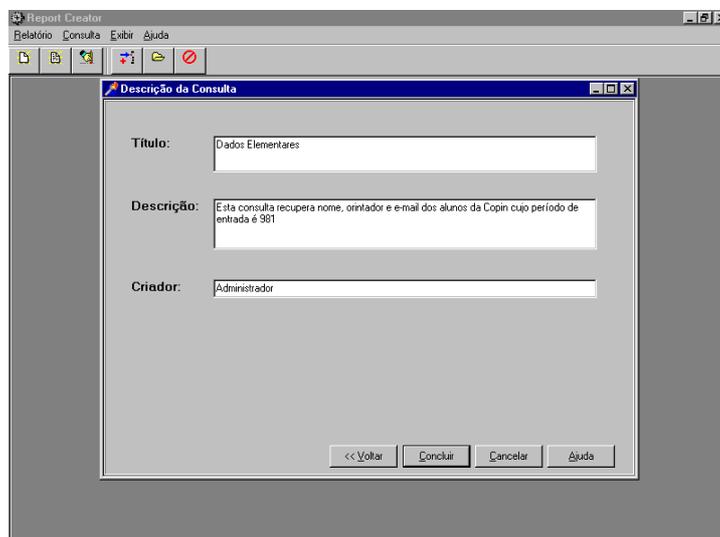


Figura 6.6 - Descrição da Consulta

Após concluir a elaboração da consulta, esta é repassada para o Integrador que a transforma, gerando uma ou mais subconsultas, a serem executadas em bancos de dados locais, neste caso, BDSEMINAR, DWCOPIIN ou SYSCOPIN.

A consulta SQL, baseada no esquema global *COPIN*, correspondente ao conjunto de passos que o UDC acabou de percorrer é a seguinte:

```
SELECT Aluno.Nm_Aluno, Aluno.Orientador, Aluno.Nm_Email  
FROM Aluno  
WHERE Aluno.Pr_Entrada = 981  
ORDER BY Aluno.Nm_Aluno
```

Vamos, agora, acompanhar detalhadamente a transformação desta consulta em subconsultas, analisando o funcionamento dos componentes do Integrador.

## 6.2.1 Transformação de Consulta

As seleções de tabelas virtuais, atributos virtuais, filtros, ordenações e agrupamentos, realizadas através da IDC, são enviadas para o Integrador, em forma de conjuntos:

- Tabelas Virtuais = {Aluno}
- Atributos Virtuais = {{Aluno, Nm\_Aluno, Ø}, {Aluno, Orientador, Ø}, {Aluno, Nm\_Email, Ø}}
- Junções = Ø
- Filtros = {{Aluno, Pr\_Entrada, IGUAL, 981}}
- Ordenações = {{Aluno, Nm\_Aluno}}
- Agrupamentos = Ø

As informações são recebidas pelo Integrador, mais precisamente pelo Analisador de Consulta, que estrutura as informações, conforme podemos observar nas Tabelas 6.39, 6.40 e 6.41.

Tabela Virtual	Atributo Virtual	Selecionado	Função	Filtro	Junção	OrderBy	GroupBy
Aluno	Nm_Aluno	✓				✓	
Aluno	Orientador	✓					
Aluno	Nm_Email	✓					
Aluno	Pr_Entrada			✓			

Tabela 6.39 - Tabelas e Atributos Virtuais Selecionados

Tabela Virtual	Atributo Virtual	Operador	Constante
Aluno	Pr_Entrada	=	981

Tabela 6.40 - Filtros

Tabela Virtual	Atributo Virtual	Operador	Tabela Virtual de Junção	Atributo Virtual de Junção

Tabela 6.41 - Junções

O Coletor de Fontes, utilizando o Metabanco de Dados, seleciona as possíveis fontes que podem atender a cada atributo virtual selecionado, conforme podemos visualizar na Tabela 6.42.

Tabela Virtual	Atributo Virtual	Fonte
Aluno	Nm_Aluno	BDSEMINAR
		DWCOPIN
		SYSCOPIN
Aluno	Orientador	DWCOPIN
Aluno	Nm_Email	BDSEMINAR
		SYSCOPIN
Aluno	Pr_Entrada	DWCOPIN
		SYSCOPIN

Tabela 6.42 - Fontes

As informações produzidas pelo Estimador de Fontes podem ser vistas na Tabela 6.43(a). Como podemos observar, nenhuma das fontes atende, sozinha, a todos os atributos virtuais incluídos na consulta. Além disso, os atributos virtuais atendidos pela fonte BDSEMINAR são, na verdade, um subconjunto dos atributos virtuais atendidos pela fonte SYSCOPIN. Sendo assim, as únicas fontes selecionadas para responder à consulta são: DWCOPIN e SYSCOPIN, como podemos ver na Tabela 6.43(a).

Fonte	Porcentagem	Subconjunto
BDSEMINAR	50%	✓
DWCOPIN	75%	
SYSCOPIN	75%	

Tabela 6.43(a) - Estatística

A etapa seguinte é realizada pelo Seletor de Fontes, que decide qual(is) das três fontes deve(m) atender a consulta. São elas: DWCOPIN e SYSCOPIN (ver Tabela 6.43(b))

Fonte	Porcentagem	Subconjunto	Selecionada
BDSEMINAR	50%	✓	
DWCOPIN	75%		✓
SYSCOPIN	75%		✓

Tabela 6.43(b) - Estatística

O último módulo a entrar em ação é o Transformador de Consultas, cujo papel é gerar as subconsultas para as fontes de dados selecionadas anteriormente. As subconsultas geradas para os bancos de dados DWCOPIN e SYSCOPIN são, respectivamente:

```
SELECT Aluno.Nm_Aluno, Aluno.Orientador
FROM Aluno
WHERE Aluno.Pr_Entrada = 981
ORDER BY Nm_Aluno
```

```
SELECT Aluno.Nm_Aluno, Aluno.Nm_Email
FROM Aluno
WHERE Aluno.Pr_entrada = 981
ORDER BY Nm_Aluno
```

Após a geração das subconsultas, as mesmas são armazenadas no Metabanco de Dados para posterior execução em suas respectivas fontes de dados.

## 6.2.2 Execução de Consulta

Para executar uma consulta é necessário apenas que o UG selecione a consulta desejada e clique no "link" rotulado pelo nome da consulta. Seguindo o nosso exemplo, o UG deve clicar na consulta intitulada *Dados Elementares*, conforme podemos observar na Figura 6.7.

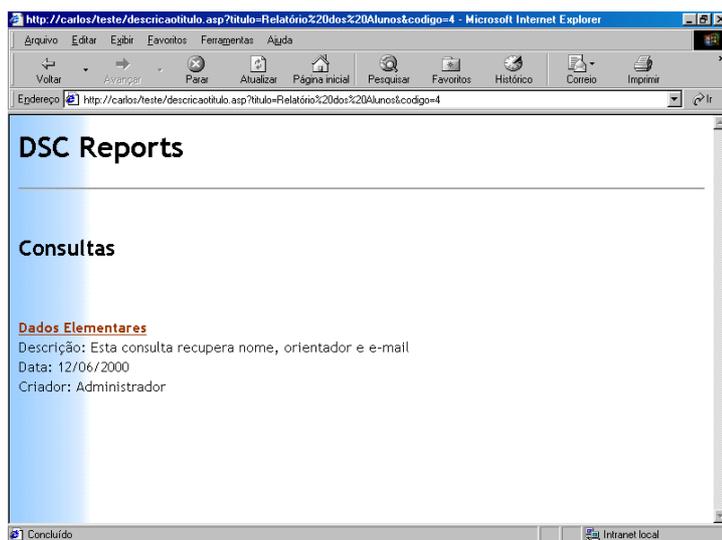


Figura 6.7 - Seleção da Consulta *Dados Elementares*

Um clique no nome da consulta resulta na recuperação, a partir do Metabanco de Dados, e posterior execução das subconsultas correspondentes à mesma em suas respectivas fontes de dados - DWCOPIIN e SYSCOPIN. Cada fonte de dados retorna seu conjunto-resposta, que é combinado e apresentado. Neste exemplo, durante a combinação de resultados, foram detectados e solucionados conflitos de dados do tipo Expressões Diferentes. Por exemplo, um dos registros recuperados em função da subconsulta submetida ao banco de dados DWCOPIIN foi:

Nm_Aluno	Orientador
...	
Carlos Eduardo Santos Pires	Marcus Costa Sampaio
...	

Tabela 6.44 - Resultados retornados pelo banco de dados DWCOPIIN

O registro que armazena o mesmo aluno no banco de dados SYSCOPIN foi recuperado da seguinte forma:

Nm_Aluno	Nm_Email
...	
Carlos E S Pires	carlos@dsc.ufpb.br
...	

Tabela 6.45 - Resultados retornados pelo banco de dados SYSCOPIN

Desta forma, quando combinados os conjuntos-resposta, o nome do aluno em DWCOPIIN foi mapeado para o nome contido no banco de dados SYSCOPIN, ou seja, *Carlos E S Pires* foi mapeado em *Carlos Eduardo Santos Pires*, como podemos ver na Figura 6.8.



Nm_Aluno	Orientador	Nm_Email
Adriano Wágner de Araújo Bezerra	Fátima Turnell	wagner@dsc.ufpb.br
Antonio Augusto Coutinho	Jacques Sauve	coutinho@dsc.ufpb.br
Carlos Eduardo Santos Pires	Marcus Costa Sampaio	carlos@dsc.ufpb.br
Flávia Estelita Coelho	Jacques Sauve	flavia@dsc.ufpb.br
Gilene Fernandes	Ulrich Schiel	glene@dsc.ufpb.br
Ivette Kafure	Bernardo Lula Júnior	ivette@dsc.ufpb.br
Juliano Varella de Carvalho	Marcus Costa Sampaio	varella@dsc.ufpb.br
Júlio César Oliveira Gomes	Ulrich Schiel	julio@dsc.ufpb.br
Marinaldo Nunes da Silva	Ulrich Schiel	nunes@dsc.ufpb.br
Raíssa Dantas Freira	Jacques Sauve	raissa@dsc.ufpb.br
Starch Melo	Jacques Sauve	starch@dsc.ufpb.br

Figura 6.8 - Resultado Final da Consulta *Dados Elementares*

Um fato bastante observado durante o desenvolvimento de nossa aplicação foi o excesso de tarefas no servidor "Web", em função de que, para resolver conflitos de dados, cada dado necessitava ser tratado individualmente. Para solucionar Conflitos de Expressões Diferentes, é necessário o envio de uma consulta pelo Combinador de Resultados no intuito de verificar se um determinado dado deve ou não ser mapeado em outro. No exemplo apresentado anteriormente, o aluno *Carlos E S Pires* deve ser mapeado no aluno *Carlos Eduardo Santos Pires*. O Combinador de Resultados somente pode ter ciência disto através de uma consulta ao Metabanco de Dados, via Gerente de Metadados.

### 6.3 Conclusões

Neste capítulo, apresentamos uma aplicação desenvolvida utilizando o Sistema BDRV. Nesta aplicação, integramos três bancos de dados relacionais distintos, apresentando a alunos, professores e funcionários do DSC um esquema global único, onde os mesmos podem elaborar consultas e obter resultados. Esta aplicação foi bastante útil na tentativa de detecção e solução de problemas funcionais.

# Anexo B

## *Estudo de Caso – Esquemas das Fontes de Dados Locais e do Metabanco de Dados*

### Notação:

Símbolo Semântica

# Chave Primária

\* Chave Estrangeira

### B.1 Fonte de Dados Local *BDSEMINAR*

Tabela: Autor

Descrição: descreve os dados dos palestrantes

Atributo	Tipo	Descrição
#Cd_Autor	Int	Código que identifica unicamente um palestrante
Nome_Autor	Varchar	Nome do palestrante
E_Mail	Varchar	Endereço eletrônico
Tipo	Varchar	Indica se é ALUNO ou PROFESSOR

Tabela B.1 - Intenção da Tabela *Autor*

Tabela: Seminário

Tabela Referenciada: Autor (Cd\_Autor)

Descrição: descreve os dados relativos aos seminários

Atributo	Tipo	Descrição
#Cd_Seminario	Int	Código que identifica unicamente um seminário
Nome_Seminario	Varchar	Título do seminário
Descricao	Text	Descrição do seminário
Data	Smalldatetime	Data de apresentação do seminário
Horario	Datetime	Horário de apresentação do seminário
*Cd_Palestrante	Int	Código que identifica unicamente um palestrante

Tabela B.2 - Intenção da Tabela *Seminário*

Tabela: Autor

Cd_Autor	Nome_Autor	E_Mail	Tipo
1	Carlos Eduardo Santos Pires	carlos@dsc.ufpb.br	Aluno
2	Marcus Costa Sampaio	sampaio@dsc.ufpb.br	Professor
3	Edeyson Gomes	edeyson@dsc.ufpb.br	Aluno
4	Eduardo Jorge	emjorge@dsc.ufpb.br	Aluno
5	Ulrich Schiel	ulrich@dsc.ufpb.br	Professor
6	Maria Izabel Cavalcanti Cabral	izabel@dsc.ufpb.br	Professor
7	Jacques Sauve	jacques@dsc.ufpb.br	Professor
8	Ivette Kafure	ivette@dsc.ufpb.br	Aluno
9	Júlio Cezar Oliveira Gomes	julio@dsc.ufpb.br	Aluno
10	André Barbosa Rocha	andre@dsc.ufpb.br	Aluno
11	Eliane Cristina de Araújo	eliane@dsc.ufpb.br	Aluno
12	Carlos A C Alves	caca-pa@dsc.ufpb.br	Aluno
13	Raissa Dantas Freire	raissa@dsc.ufpb.br	Aluno
14	Andréa de Farias Barros	raissa@dsc.ufpb.br	Aluno
15	Flavia Estelita S Coelho	flavia@dsc.ufpb.br	Aluno
16	Antonio Augusto Coutinho	coutinho@dsc.ufpb.br	Aluno
17	Juliano Varella de Carvalho	varella@dsc.ufpb.br	Aluno
18	Starch Melo	coutinho@dsc.ufpb.br	Aluno
19	Angel Anibal	angel@dsc.ufpb.br	Aluno

Tabela B.3 - Extensão da Tabela *Autor*

Tabela: Seminário

Cd_Seminario	Nome_Seminario	Descricao	Data	Horario	Cd_Palestrante
1	Servidores de BD e a World Wide Web		5 Mai 1999 0:00	1 Jan 1900 15:00	1
2	Metamodelagem: Uma Introducao		20 Jan 1999 0:00	1 Jan 1900 15:00	9
3	Anarquismo, sociedade e informática		27 Jan 1999 0:00	1 Jan 1900 14:00	5
4	Tutorial em Bancos de Dados Temporais		3 Fev 1999 0:00	1 Jan 1900 14:00	10
5	Unified Modeling Language - UML - APRENDA AGORA OU SE ARREPENDA PARA SEMPRE!!!		3 Mar 1999 0:00	1 Jan 1900 14:00	9
6	Os Limites do conhecimento Científico		3 Mar 1999 0:00	1 Jan 1900 15:00	11
7	Projeto para Redes Multimidia		17 Mar 1999 0:00	1 Jan 1900 14:00	18
8	Conhece a Colômbia?		24 Mar 1999 0:00	1 Jan 1900 14:00	8
9	Projeto para redes multimidia		24 Mar 1999 0:00	1 Jan 1900 15:00	12
10	Por que ler os clássicos?		7 Abr 1999 0:00	1 Jan 1900 15:00	2
11	Amor, Sexo and Rock&Roll: Como Fazer uma Intranet com Java e uma Arquitetura em 3 Camadas		19 Mai 1999 0:00	1 Jan 1900 14:00	7
12	Diretrizes Curriculares de Cursos da Área de Computação e Informática		2 Jun 1999 0:00	1 Jan 1900 15:00	6
13	Uma API para o desenvolvimento de aplicações de gerência de faltas		11 Ago 1999 0:00	1 Jan 1900 14:00	13
14	Avaliação de Mecanismos de Busca - Pro-Copin		18 Ago 1999 0:00	1 Jan 1900 14:00	12
15	Proposta de Mecanismos para transmissão de fluxo CBR sobre Ethernet		15 Set 1999 0:00	1 Jan 1900 14:00	12
16	Suporte a tempo real em Ethernet		22 Set 1999 0:00	1 Jan 1900 14:00	12
17	Arquitetura AMADEUS para a gerência de		22 Set	1 Jan 1900	14

	dispositivos com servidores Web embutidos		1999 0:00	15:00	
18	Música: sua essência e relação com política		29 Set 1999 0:00	1 Jan 1900 14:00	5
19	Orientação a objetos na modelagem conceitual: avanço ou regresso		29 Set 1999 0:00	1 Jan 1900 15:00	5
20	Jasmine: um SGBD Orientado a Objeto		6 Out 1999 0:00	1 Jan 1900 14:00	3
21	Oracle 8.0: um SGBD Objeto-Relacional		6 Out 1999 0:00	1 Jan 1900 15:00	4
22	Um framework para o desenvolvimento de aplicações de gerência de faltas - Parte II		13 Out 1999 0:00	1 Jan 1900 14:00	13

Tabela B.4 - Extensão da Tabela *Seminário*

## B.2 Fonte de Dados Local *DWCOPIN*

Tabela: Aluno

Descrição: descreve os dados dos alunos de pós-graduação da COPIN

Atributo	Tipo	Descrição
#Cd_Aluno	Int	Código que identifica unicamente um aluno
Nm_Aluno	Varchar	Nome do aluno
Pr_Entrada	Int	Período de entrada
Orientador	Varchar	Nome do orientador
Dt_Defesa_Estagio	Smalldatetime	Data de defesa do estágio supervisionado
Dt_Defesa_Proposta	Smalldatetime	Data de defesa da proposta de dissertação
Dt_Defesa_Dissertacao	Smalldatetime	Data de defesa da dissertação
Area_Pesquisa	Varchar	Nome da área de concentração
Status	Varchar	Indica se o aluno é MESTRE ou MESTRANDO

Tabela B.5 - Intenção da Tabela *Aluno*

Tabela: Disciplina

Descrição: descreve os dados das disciplinas de pós-graduação da COPIN

Atributo	Tipo	Descrição
#Cd_Disciplina	Int	Código que identifica unicamente uma disciplina
Nm_Disciplina	Varchar	Nome da disciplina
Creditos	Int	Número de créditos
Tp_Disciplina	Int	Tipo da disciplina: 0 (Básica) e 1 (Obrigatória)
Area_Pesquisa	Varchar	Nome da área de concentração

B.6 - Intenção da Tabela *Disciplina*

## Tabela: Fatos

Tabelas Referenciadas: Aluno (Cd\_Aluno), Professor (Cd\_Professor), Tempo (Cd\_Tempo) e Disciplina (Cd\_Disciplina)

Descrição: descreve fatos relacionados com as tabelas aluno, professor, tempo e disciplina

Atributo	Tipo	Descrição
#*Cd_Aluno	Int	Código que identifica unicamente um aluno
#*Cd_Professor	Int	Código que identifica unicamente um professor
#*Cd_Tempo	Int	Código que identifica unicamente um instante no tempo
#*Cd_Disciplina	Int	Código que identifica unicamente uma disciplina
Resultado	Char	Conceito atribuído ao aluno com relação a uma disciplina cursada, ministrada por um certo professor, em determinado período
Ft_Artificial	Int	Fato artificial

Tabela B.7 - Intenção da Tabela *Fatos*

## Tabela: Professor

Descrição: descreve os dados dos professores de pós-graduação da COPIN

Atributo	Tipo	Descrição
#Cd_Professor	Int	Código que identifica unicamente um professor
Nm_Professor	Varchar	Nome do professor
Area_Pesquisa	Varchar	Nome da área de concentração
Nm_Email	Varchar	Endereço eletrônico

Tabela B.8 - Intenção da Tabela *Professor*

## Tabela: Tempo

Descrição: descreve as granularidades de tempo

Atributo	Tipo	Descrição
#Cd_Tempo	Int	Código que identifica unicamente uma granularidade de tempo
Semestre	Int	Indica o semestre do ano: primeiro (1) e segundo (2)
Ano	Int	Ano

Tabela B.9 - Intenção da Tabela *Tempo*

Tabela: Aluno

Cd_Aluno	Nm_Aluno	Pr_Entrada	Orientador	Dt_Defesa_Estagio	Dt_Defesa_Proposta	Dt_Defesa_Dissertacao	Área_Pesquisa	Status
1	Carlos Eduardo Santos Pires	981	Marcus Costa Sampaio	30 Dez 1997 0:00	16 Mar 1999 0:00		Banco de Dados	Mestrando
2	Juliano Varella de Carvalho	981	Marcus Costa Sampaio	31 Jan 1997 0:00	14 Mar 1999 0:00		Banco de Dados	Mestrando
3	Marinaldo Nunes da Silva	981	Ulrich Schiel	12 Ago 1998 0:00	8 Mar 1999 0:00		Banco de Dados	Mestrando
4	Gilene Fernandes	981	Ulrich Schiel	12 Jan 1997 0:00	8 Mar 1999 0:00		Banco de Dados	Mestrando
5	Júlio Cezar Oliveira Gomes	981	Ulrich Schiel	21 Abr 1999 0:00	8 Mar 1999 0:00		Banco de Dados	Mestrando
6	Adriano Wagner de A Bezerra	981	Fatima Turnell	24 Set 1995 0:00	23 Mai 1999 0:00		Banco de Dados	Mestrando
7	Cristine Pires da Costa	991	Marcus Costa Sampaio	22 Jul 1999 0:00			Banco de Dados	Mestrando
8	Ana Isabel Pinheiro da Silva	961	Marcus Costa Sampaio	8 Nov 1995 0:00	12 Dez 1996 0:00	4 Jul 1998 0:00	Banco de Dados	Mestre
9	Starch Melo	981	Jacques Sauve	22 Dez 1997 0:00	23 Mai 1999 0:00		Redes de Computadores	Mestrando
10	André Barbosa Rocha	971	Ulrich Schiel	1 Nov 1996 0:00	10 Mar 1998 0:00		Banco de Dados	Mestrando
11	Ivette Kafure	981	Bernardo Lula	3 Mar 1995 0:00	4 Fev 1999 0:00		Interfaces Homem-Maquina	Mestrando
12	Giovanni Almeida Santos	991	Jacques Sauve	27 Abr 1999 0:00			Redes de Computadores	Mestrando
13	Marcio Farias de Souza	971	Marcus Costa Sampaio	12 Set 1996 0:00	1 Ago 1998 0:00	7 Jul 1999 0:00	Banco de Dados	Mestre
14	Eduardo Jorge	991	Marcus Costa Sampaio	19 Out 1995 0:00			Banco de Dados	Mestrando
15	Edeyson Gomes	991	Marcus Costa Sampaio	13 Mai 1994 0:00			Banco de Dados	Mestrando
16	Gustavo Vasconcellos Cavalcante	961	Francisco V Brasileiro	7 Mar 1995 0:00	4 Fev 1997 0:00	13 Set 1999 0:00	Sistemas Distribuidos	Mestre
17	Algeir Prazeres Sampaio	961	Jose Homero F. Cavalcanti	11 Jun 1995 0:00	16 Abr 1997 0:00	17 Set 1999 0:00	Inteligencia Artificial	Mestre

Tabela B.10 - Extensão da Tabela *Aluno*

Tabela: Disciplina

Cd_Disciplina	Nm_Disciplina	Creditos	Tp_Disciplina	Area_Pesquisa
1	Banco de Dados	4	Básica	Banco de Dados
2	Banco de D Avançado	2	Específica	Banco de Dados
3	Redes de C e Protocolos	4	Básica	Redes de Computadores
4	IA	4	Básica	Inteligência Artificial
5	Data Mining	2	Específica	Banco de Dados
6	Data Warehousing	2	Específica	Banco de Dados
7	Sistemas Especialistas	2	Específica	Inteligência Artificial
8	Projeto de I Homem-Máquina	2	Específica	Interface Homem-Máquina
9	Lógica Clássica	2	Básica	Inteligência Artificial
10	Redes de Petri	4	Básica	Redes de Petri
11	Sistemas Operacionais I	4	Básica	Sistemas Distribuídos
12	Sistemas Operacionais II	2	Específica	Sistemas Distribuídos

Tabela B.11 - Extensão da Tabela *Disciplina*

Tabela: Fatos

Cd_Professor	Cd_Aluno	Cd_Disciplina	Cd_Tempo	Resultado	Ft_Artificial
1	1	1	37	A	1
1	1	6	38	A	1
1	2	1	37	A	1
1	2	6	38	A	1
1	3	1	37	A	1
1	3	6	38	A	1
1	4	1	37	A	1
1	4	6	38	A	1
1	5	1	37	A	1
1	6	1	37	A	1
1	6	6	38	A	1
1	7	1	39	A	1
1	8	2	33	A	1
1	8	2	34	B	1
1	8	6	34	B	1
1	10	1	35	A	1
1	10	6	36	A	1
1	12	1	39	A	1
1	13	1	35	A	1
1	13	6	36	A	1
1	14	1	39	A	1
1	15	1	39	A	1
1	16	2	33	B	1
1	16	2	34	A	1
1	16	6	34	A	1
1	17	2	33	B	1
1	17	2	34	B	1
1	17	6	34	A	1
2	1	2	38	A	1
2	3	2	38	A	1
2	4	2	38	A	1
2	5	2	38	A	1
2	6	2	38	A	1
2	10	2	36	A	1
2	13	2	36	A	1
3	9	11	37	A	1
3	12	11	39	A	1
4	9	3	37	A	1
4	12	3	39	A	1
5	1	4	37	B	1

5	3	4	37	C	1
5	4	4	37	B	1
5	5	4	37	C	1
5	10	4	35	B	1
5	13	4	35	B	1
6	16	11	33	B	1
6	17	11	33	B	1
7	10	11	35	C	1
7	13	11	35	B	1
7	16	12	34	B	1
7	17	12	34	B	1
8	1	8	38	A	1
8	3	8	38	B	1
8	4	8	38	B	1
8	5	8	38	B	1
8	6	8	38	B	1
8	10	8	36	A	1
8	11	8	38	A	1
9	2	9	37	A	1
9	3	9	37	A	1
9	4	9	37	B	1
9	8	8	34	B	1
9	8	9	33	B	1
9	9	9	37	B	1
9	10	9	35	A	1
9	11	9	37	A	1
9	13	8	34	B	1
9	13	9	35	A	1
9	16	8	34	C	1
9	16	9	33	C	1
9	17	9	33	C	1
10	8	4	33	A	1
10	16	4	33	A	1
10	17	4	33	A	1

Tabela B.12 - Extensão da Tabela *Fatos*

Tabela: Professor

Cd_Professor	Nm_Professor	Area_Pesquisa	Nm_Email
1	Marcus Costa Sampaio	Banco de Dados	sampaio@dsc.ufpb.br
2	Ulrich Schiel	Banco de Dados	ulrich@dsc.ufpb.br
3	Francisco Villar Brasileiro	Sistemas Distribuídos	fubica@dsc.ufpb.br
4	Jacques Sauve	Redes de Computadores	jacques@dsc.ufpb.br
5	Giuseppe Mongiovi	Inteligência Artificial	giuseppe@dsc.ufpb.br
6	Pedro Sérgio Nicoletti	Redes de Computadores	peter@dsc.ufpb.br
7	Maria Izabel Cavalcanti Cabral	Redes de Computadores	izabel@dsc.ufpb.br
8	Maria de Fátima Vieira Turnell	Interface Homem-Máquina	turnellm@dee.ufpb.br
9	Bernardo Lula Júnior	Interface Homem-Máquina	bernardo@dsc.ufpb.br
10	Homero Cavalcanti	Inteligência Artificial	homero@dsc.ufpb.br
11	Edilson Ferneda	Inteligência Artificial	edilson@dsc.ufpb.br

Tabela B.13 - Extensão da Tabela *Professor*

Tabela: Tempo

Cd_Tempo	Semestre	Ano
1	1	1980
2	2	1980
3	1	1981
4	2	1981
5	1	1982
6	2	1982
7	1	1983
8	2	1983
9	1	1984
10	2	1984
11	1	1985
12	2	1985
13	1	1986
14	2	1986
15	1	1987
16	2	1987
17	1	1988
18	2	1988
19	1	1989
20	2	1989
21	1	1990
22	2	1990
23	1	1991
24	2	1991
25	1	1992
26	2	1992
27	1	1993
28	2	1993
29	1	1994
30	2	1994
31	1	1995
32	2	1995
33	1	1996
34	2	1996
35	1	1997
36	2	1997
37	1	1998
38	2	1998
39	1	1999

Tabela B.14 - Extensão da Tabela *Tempo*

### B.3 Fonte de Dados Local SYSCOPIN

Tabela: Aluno

Descrição: descreve os dados dos alunos de pós-graduação da COPIN

Atributo	Tipo	Descrição
#Mt_Aluno	Int	Código que identifica unicamente um aluno de mestrado
Nm_Aluno	Varchar	Nome do aluno
Pr_Entrada	Varchar	Período de entrada
Dt_Nascimento	Smalldatetime	Data de nascimento
Nm_Municipio_Nasc	Varchar	Município de nascimento
Nm_UF_Nasc	Char	Estado de nascimento
Nm_Email	Varchar	Endereço eletrônico
Nm_Endereco	Varchar	Endereço residencial
Nm_Complemento	Varchar	Complemento do endereço
Nm_Bairro	Varchar	Bairro onde reside
Nm_Municipio_End	Varchar	Município onde reside
Nm_UF_End	Char	Estado onde reside
Nu_Fone	Int	Telefone do aluno
Nm_Titulacao	Varchar	Titulação do aluno
Nm_Instituicao_Origem	Varchar	Instituição em que foi titulado
Homepage	Varchar	Endereço da homepage

Tabela B.15 - Intenção da Tabela *Aluno*

Tabela: Aluno\_Disciplina

Tabelas Referenciadas: Aluno (Mt\_Aluno) e Disciplina (Cd\_Disciplina)

Descrição: descreve as matrículas em disciplinas e o resultado da avaliação dos alunos

Atributo	Tipo	Descrição
#*Mt_Aluno	Int	Código que identifica unicamente um aluno
#*Cd_Disciplina	Int	Código que identifica unicamente uma disciplina
Pr_Disciplina	int	Período da matrícula do aluno
Conceito	Char	Conceito da disciplina
Situacao	Varchar	Indica se Aprovado, Reprovado por Notas, Reprovado por Faltas, Trancado, Dispensado ou Matriculado

Tabela B.16 - Intenção da Tabela *Aluno\_Disciplina*

Tabela: Área

Tabela Referenciada: Curso (Cd\_Curso)

Descrição: descreve as áreas de pesquisa do curso de pós-graduação da COPIN

Atributo	Tipo	Descrição
#Cd_Area	Int	Código que identifica unicamente uma área de concentração
Nm_Area	Varchar	Nome da área de concentração
*Cd_Curso	Int	Código que identifica unicamente um curso

Tabela B.17 - Intenção da Tabela *Área*

Tabela: Artigo

Tabela Referenciada: Area (Cd\_Area)

Descrição: descreve os artigos publicados

Atributo	Tipo	Descrição
#Cd_Artigo	Int	Código que identifica unicamente um artigo
Nm_Artigo	Varchar	Título do artigo
Tp_Artigo	Varchar	Indica se o artigo é de RESUMO, de COMUNICAÇÃO ou COMPLETO
Nm_Idioma	Varchar	Idioma em que foi publicado
Tp_Publicacao	Varchar	Tipo da publicação
Nm_Publicacao	Varchar	Nome da publicação em que o artigo encontra-se inserido
Dt_Publicacao	Smalldatetime	Data de publicação
*Cd_Area	Int	Código que identifica unicamente uma área de concentração

Tabela B.18 - Intenção da Tabela *Artigo*

Tabela: Bolsa

Tabela Referenciada: Aluno (Mt\_Aluno)

Descrição: descreve os dados relativos às bolsas da pós-graduação da COPIN

Atributo	Tipo	Descrição
#Cd_Bolsa	Int	Código que identifica unicamente uma bolsa
Ds_Bolsa	Varchar	Descrição da bolsa
Nm_Orgao_Financiador	Varchar	Nome do órgão financiador
Vl_Bolsa	Money	Valor da bolsa
Dt_Inicio	Smalldatetime	Data de início da bolsa
Dt_Termino	Smalldatetime	Data de término da bolsa
*Mt_Aluno	int	Código que identifica unicamente um aluno de mestrado

Tabela B.19 - Intenção da Tabela *Bolsa*

Tabela: Curso

Tabela Referenciada: Departamento (Cd\_Departamento)

Descrição: descreve as informações sobre o curso de pós-graduação

Atributo	Tipo	Descrição
#Cd_Curso	Int	Código que identifica unicamente um curso
Nm_Curso	Varchar	Nome do curso
Qt_Creditos	Int	Quantidade mínima de créditos
Nu_Horas	Int	Carga horária do curso
*Cd_Departamento	Int	Código que identifica unicamente um departamento

Tabela B.20 - Intenção da Tabela *Curso*

## Tabela: Departamento

Descrição: descreve as informações sobre os departamentos que tem curso de pós-graduação

Atributo	Tipo	Descrição
#Cd_Departamento	Int	Código que identifica unicamente um departamento
Nm_Departamento	Varchar	Nome do departamento
Nm_Email	Varchar	Endereço eletrônico do departamento
Nm_Instituicao	Varchar	Nome da instituição a qual o departamento pertence

Tabela B.21 - Intenção da Tabela *Departamento*

## Tabela: Disciplina

Tabelas Referenciadas: Professor (Mt\_Professor), Area (Cd\_Area) e Curso (Cd\_Curso)

Descrição: descreve os dados relativos às disciplinas da COPIN

Atributo	Tipo	Descrição
#Cd_Disciplina	Int	Código que identifica unicamente uma disciplina
Nm_Disciplina	Varchar	Nome da disciplina
Nu_Creditos	Int	Número de créditos
Ds_Disciplina	Varchar	Descrição da disciplina
Tp_Disciplina	Varchar	Indica se a disciplina é BÁSICA ou ESPECÍFICA
*Cd_Curso	Int	Código que identifica unicamente um curso
*Cd_Area	Int	Código que identifica unicamente uma área de concentração
*Mt_Professor	Int	Código que identifica unicamente um professor

Tabela B.22 - Intenção da Tabela *Disciplina*

## Tabela: Professor

Tabelas Referenciadas: Area (Cd\_Area) e Departamento (Cd\_Departamento)

Descrição: descreve os dados relativos aos professores da COPIN

Atributo	Tipo	Descrição
#Cd_Professor	Int	Código que identifica unicamente um professor
Nm_Professor	Varchar	Nome do professor
Nm_Email	Varchar	Endereço eletrônico
*Cd_Área	Int	Código que identifica unicamente uma área de concentração
Tp_Professor	Varchar	Indica se o professor é PERMANENTE, VISITANTE ou PARTICIPANTE
*Cd_Departamento	Int	Código que identifica unicamente um departamento

Tabela B.23 - Intenção da Tabela *Professor*

Tabela: Tese

Tabelas Referenciadas: Area (Cd\_Area) e Aluno (Mt\_Aluno)

Descrição: descreve todas as teses da COPIN

Atributo	Tipo	Descrição
#Cd_Tese	Int	Código que identifica unicamente um professor
Nm_Titulo	Varchar	Nome do professor
Nm_Área_Concentracao	Varchar	Indica se o professor é permanente, visitante ou participante
Dt_Inicio	Smalldatetime	Endereço eletrônico
Dt_Termino	Smalldatetime	Código que identifica unicamente um departamento
Dt_Defesa	Smalldatetime	Código que identifica unicamente uma área de concentração
Nu_Pagina	Int	Número de páginas da tese
Nm_Idioma	Varchar	Idioma em que a tese se encontra
Nm_Resumo	Text	Resumo da tese
Nm_Projeto_Pesquisa	Varchar	Indica o projeto de pesquisa no qual a tese está envolvida
*Mt_Aluno	Int	Código que identifica unicamente um aluno
*Cd_Área	Int	Código que identifica unicamente a área de pesquisa seguida pela tese

Tabela B.24 - Intenção da Tabela *Tese*

Tabela: Tese\_Artigo

Tabelas Referenciadas: Tese (Cd\_Tese) e Artigo (Cd\_Artigo)

Descrição: descreve as publicações realizadas

Atributo	Tipo	Descrição
#*Cd_Tese	Int	Código que identifica unicamente uma tese
#*Cd_Artigo	Int	Código que identifica unicamente um artigo

Tabela B.25 - Intenção da Tabela *Tese\_Artigo*

Tabela: Tese\_Professor

Tabelas Referenciadas: Tese (Cd\_Tese) e Professor (Cd\_Professor)

Descrição: descreve o tipo de participação de um professor em uma tese

Atributo	Tipo	Descrição
#*Cd_Tese	Int	Código que identifica unicamente uma tese
#*Cd_Professor	Int	Código que identifica unicamente um professor
Tp_Status_Prof	Varchar	Indica se o professor é ORIENTADOR ou CO-ORIENTADOR

Tabela B.26 - Intenção da Tabela *Tese\_Professor*

Tabela: Aluno

Mt_Aluno	Nm_Aluno	Pr_Entrada	Dt_Nascimento	Nm_Municipio_Nasc	Nm_UF_Nasc	Nm_Email	Nm_Endereco	Nm_Complemento	Nm_Bairro	Nm_Municipio_End	Nm_UF_End	Nu_Fone	Nm_Titulacao	Nm_Instituicao_Origem	Homepage
1	Carlos E S Pires	981	6 Jul 1975 0:00	Recife	PE	carlos@dsc.ufpb.br	Rua Ana Firmino da Costa, 37		Catole	Campina Grande	PB	3373508	Bach. em Ciência da Computação	UFPB	www.dsc.ufpb.br/~carlos
2	Gilene Fernandes	981	5 Mar 1974 0:00	Natal	RN	gilene@dsc.ufpb.br	Rua Indios Cariris, 354	Apto 103	Centro	Campina Grande	PB	93125123	Bach. em Ciência da Computação	UFPB	
3	André B Rocha	971	11 Set 1974 0:00	Campina Grande	PB	andre@dsc.ufpb.br	Av Rio Branco, 562		Prata	Campina Grande	PB	9729596	Bach. em Ciência da Computação	UFPB	www.dsc.ufpb.br/~andre
4	Adriano W de A Bezerra	981	9 Dez 1973 0:00	Campina Grande	PB	wagner@dsc.ufpb.br	Rua Franklin Araujo, 98		Alto Branco	Campina Grande	PB	9722410	Bach. em Ciência da Computação	UFPB	
5	Antonio A Coutinho	981	15 Jan 1974 0:00	Joao Pessoa	PB	coutinho@dsc.ufpb.br	Av Rodrigues Alves, 23		Alto Branco	Campina Grande	PB	3333062	Bach. em Ciência da Computação	UFPB	www.dsc.ufpb.br/~coutinho
6	Cristine P da Costa	991	22 Mai 1976 0:00	Recife	PE	cristine@dsc.ufpb.br	Av Vigario Calixto, 53	Apto 303	Catole	Campina Grande	PB	3311762	Bach. em Ciência da Computação	UFPB	www.dsc.ufpb.br/~coutinho
7	Eduardo Jorge	991	20 Set 1973 0:00	Salvador	BA	emjorge@dsc.ufpb.br	Av Rodrigues Alves, 445	Apto 303	Prata	Campina Grande	PB	3215252	Bach. em Ciência da Computação	UFBA	
8	Raissa D Freire	981	8 Fev 1975 0:00	Campina Grande	PB	raissa@dsc.ufpb.br	Rua Alemanha, 78		Nacoes	Campina Grande	PB	3216733	Bach. em Ciência da Computação	UFPB	www.dsc.ufpb.br/~raissa
9	Edeyson Gomes	991	24 Abr 1972 0:00	Salvador	BA	edeyson@dsc.ufpb.br	Av Rodrigues Alves, 445	Apto 303	Prata	Campina Grande	PB	3215252	Bach. em Ciência da Computação	UFBA	
10	Marinaldo N da Silva	981	5 Jan 1974 0:00	Guarabira	PB	nunes@dsc.ufpb.br	Av Floriano Peixoto, 87	Apto 203	Centro	Campina Grande	PB	93125677	Bach. em Ciência da Computação	UFPB	
11	Starch Melo	981	26 Mai 1973 0:00	Campina Grande	PB	starch@dsc.ufpb.br	Rua Saint Patrick, 95		Centenario	Campina Grande	PB	3216344	Bach. em Ciência da Computação	UFPB	
12	Márcio F de Souza	971	13 Jul 1974 0:00	Campina Grande	PB	marcio@dsc.ufpb.br	Rua Pres Vargas, 78		Alto Branco	Campina Grande	PB	3216622	Bach. em Ciência da Computação	UFPB	
13	Ivette Kafure	981	30 Out 1973 0:00	Cali	CO	ivette@dsc.ufpb.br	Av Rodrigues Alves, 9879		Prata	Campina Grande	PB	3215167	Bach. em Ciência da Computação	UFCA	
14	Algeir P Sampaio	961	1 Abr 1970 0:00	Campina Grande	PB	algeir@dsc.ufpb.br	Av Rodrigues Alves, 13		Prata	Campina Grande	PB	3221789	Bach. em Ciência da Computação	UFPB	
15	Juliano Varella	981	5 Fev 1976 0:00	Pelotas	RS	varella@dsc.ufpb.br	Rua Indios Cariris, 854	Apto 104	Centro	Campina Grande	PB	3225123	Bach. em Ciência da Computação	UFPEL	
16	Júlio C O Gomes	981	9 Jun 1974 0:00	Patos	PB	julio@dsc.ufpb.br	Rua Antenor Navarro, 223	Apto 203	Centro	Campina Grande	PB	93128976	Bach. em Ciência da Computação	UFPB	
17	Ana I P da Silva	961	13 Ago 1972 0:00	Sousa	PB	isabel@dsc.ufpb.br	Rua Bartolomeu Dias, 784	Apto 103	Jardim Paulistano	Campina Grande	PB	3413457	Bach. em Ciência da Computação	UFPB	
18	Giovanni A Santos	991	20 Set 1974 0:00	Esperanca	PB	giovanni@dsc.ufpb.br	Rua Pedro Castro, 68		Centro	Campina Grande	PB	3215568	Bach. em Ciência da Computação	UFPB	
19	Gustavo V Cavalcante	961	5 Mar 1972 0:00	Campina Grande	PB	gustavo@dsc.ufpb.br	Rua Padre Manuel Dias,	Travessa Sul	Centro	Campina Grande	PB	3216679	Bach. em Ciência da	UFPB	

20	Flávia E Coelho	981	12 Jul 1974 0:00	Campina Grande	PB	flavia@dsc.ufpb.br	45 Rua Sergipe, 12		Liberdade	Campina Grande	PB	3417819	Computação Bach. em Ciência da Computação	UFPB	
----	--------------------	-----	---------------------	----------------	----	--------------------	--------------------------	--	-----------	----------------	----	---------	--	------	--

Tabela B.27 - Extensão da Tabela *Aluno*

Tabela: Aluno\_Disciplina

Mt_Aluno	Cd_Disciplina	Pr_Disciplina	Conceito	Situacao
1	1	981	A	Aprovado
1	2	982	A	Aprovado
1	6	982	A	Aprovado
1	8	982	A	Aprovado
2	1	981	A	Aprovado
2	2	981	A	Aprovado
2	6	982	A	Aprovado
2	8	982	B	Aprovado
3	1	971	A	Aprovado
3	2	972	A	Aprovado
3	6	972	A	Aprovado
3	8	972	B	Aprovado
4	1	981	A	Aprovado
4	2	981	A	Aprovado
4	6	982	A	Aprovado
4	8	982	B	Aprovado
6	1	991	A	Aprovado
7	1	991	A	Aprovado
9	1	991	A	Aprovado
10	1	981	A	Aprovado
10	2	981	A	Aprovado
10	6	982	A	Aprovado
10	8	982	B	Aprovado
11	6	981	A	Aprovado
12	1	971	A	Aprovado
12	2	972	A	Aprovado
12	6	972	A	Aprovado
12	8	972	A	Aprovado
13	8	982	A	Aprovado
14	1	961	B	Aprovado
14	6	962	B	Aprovado
14	8	962	B	Aprovado
15	1	981	A	Aprovado
15	6	982	A	Aprovado
16	1	981	A	Aprovado
16	2	981	A	Aprovado
16	8	982	B	Aprovado
17	1	961	B	Aprovado
17	2	962	B	Aprovado
17	6	962	C	Aprovado
17	8	962	B	Aprovado
18	1	991	A	Aprovado
18	6	991	A	Aprovado
19	1	961	B	Aprovado
19	6	962	B	Aprovado
19	8	962	C	Aprovado

Tabela B.28 - Extensão da Tabela *Aluno\_Disciplina*

Tabela: Area

Cd_Area	Nm_Area	Cd_Curso
1	Banco de Dados	1
2	Redes de Computadores	1
3	Arquiteturas Dedicadas	1
4	Redes de Petri	1
5	Interface Homem-Máquina	1
6	Inteligencia Artificial	1
7	Sistemas Distribuídos	1
8	Software Design	1

Tabela B.29 - Extensão da Tabela Área

Tabela: Artigo

Cd_Artigo	Nm_Artigo	Tp_Artigo	Nm_Idioma	Tp_Publicacao	Nm_Publicacao	Dt_Publicacao	Cd_Area
1	Utilizando Técnicas de Data Mining para Reconhecimento de Caractes Manuscritos	Artigo Completo	Português	Congresso da SBBB	Anais do XIV Congresso da SBBB	10 Set 1999 0:00	1

Tabela B.30 - Extensão da Tabela Artigo

Tabela: Bolsa

Cd_Bolsa	Ds_Bolsa	Nm_Orgao_Financiador	Vl_Bolsa	Dt_Inicio	Dt_Termino	Mt_Aluno
1	Bolsa Financiada pela CAPES	CAPES	723.00	1 Jul 1998 0:00	1 Jul 2000 0:00	1
2	Bolsa Financiada pela CAPES	CAPES	723.00	1 Set 1998 0:00	1 Set 2000 0:00	2
3	Bolsa Financiada pelo CNPq	CNPq	723.00	1 Mar 1997 0:00	1 Mar 1999 0:00	3
4	Bolsa Financiada pela CAPES	CAPES	723.00	1 Mar 1998 0:00	1 Mar 2000 0:00	4
5	Bolsa Financiada pela CAPES	CAPES	723.00	1 Mar 1998 0:00	1 Mar 2000 0:00	5
6	Bolsa Financiada pela CAPES	CAPES	723.00	1 Out 1999 0:00	1 Out 2001 0:00	6
7	Bolsa Financiada pela CAPES	CAPES	723.00	1 Mar 1999 0:00	1 Mar 2001 0:00	7
8	Bolsa Financiada pelo CNPq	CNPq	723.00	1 Mar 1998 0:00	1 Mar 2000 0:00	8
9	Bolsa Financiada pela CAPES	CAPES	723.00	1 Mar 1999 0:00	1 Mar 2001 0:00	9
10	Bolsa Financiada pela CAPES	CAPES	723.00	1 Mar 1998 0:00	1 Mar 2000 0:00	10
11	Bolsa Financiada pelo CNPq	CNPq	723.00	1 Mar 1998 0:00	1 Mar 2000 0:00	11
12	Bolsa Financiada pela CAPES	CAPES	723.00	1 Mar 1997 0:00	1 Mar 1999 0:00	12
13	Bolsa Financiada pela CAPES	CAPES	723.00	1 Mar 1998 0:00	1 Mar 2000 0:00	13
14	Bolsa Financiada pelo CNPq	CNPq	723.00	1 Mar 1996 0:00	1 Mar 1998 0:00	14
15	Bolsa Financiada pela CAPES	CAPES	723.00	1 Mar 1998 0:00	1 Mar 2000 0:00	15

16	Bolsa Financiada pela CAPES	CAPES	723.00	1 Mar 1998 0:00	1 Mar 2000 0:00	16
17	Bolsa Financiada pela CAPES	CAPES	723.00	1 Mar 1996 0:00	1 Mar 1998 0:00	17
18	Bolsa Financiada pelo CNPq	CNPq	723.00	1 Mar 1999 0:00	1 Mar 1999 0:00	18
19	Bolsa Financiada pelo CNPq	CNPq	723.00	1 Mar 1996 0:00	1 Mar 1998 0:00	19
20	Bolsa Financiada pela CAPES	CAPES	723.00	1 Mar 1998 0:00	1 Mar 2000 0:00	20

Tabela B.31 - Extensão da Tabela *Bolsa*

Tabela: Curso

Cd_Curso	Nm_Curso	Qt_Creditos	Nu_Horas	Cd_Departamento
1	Bach. em Ciência da Computação	180	220	1
2	Bach. em Engenharia Elétrica	220	280	2
3	Bach. em Engenharia de Minas	160	200	3

Tabela B.32 - Extensão da Tabela *Curso*

Tabela: Departamento

Cd_Departamento	Nm_Departamento	Nm_Email	Nm_Instituicao
1	DSC	dsc@dsc.ufpb.br	UFPB
2	DEE	dee@dee.ufpb.br	UFPB
3	DMG	dmg@dmg.ufpb.br	UFPB
4	DEC	dec@dec.ufpb.br	UFPB
5	DEM	dem@dem.ufpb.br	UFPB

Tabela B.33 - Extensão da Tabela *Departamento*

Tabela: Disciplina

Cd_Disciplina	Nm_Disciplina	Nu_Creditos	Ds_Disciplina	Tp_Disciplina	Cd_Curso	Cd_Area	Mt_Professor
1	Banco de Dados	4	Conceitos de Bancos de Dados	Básica	1	1	1
2	Banco de Dados Avançado	2	Conceitos Avançados de Bancos de Dados	Específica	1	1	2
3	Redes de Computadores e Protocolos	4	Conceitos de Redes de Computadores	Básica	1	2	6
4	Inteligência Artificial	4	Noções básicas de Inteligência Artificial	Básica	1	6	5
5	Data Mining	2	Principais técnicas de Data Mining	Específica	1	1	5
6	Data Warehousing	2	Principais técnicas de Data Warehousing	Específica	1	1	1
7	Sistemas	4	Conceitos de	Básica	1	6	5

	Especialistas		Sistemas Especialistas				
8	Projeto de Interfaces Homem-Máquina	2	Estudo de interfaces de interação homem-máquina	Específica	1	5	8
9	Lógica Clássica	4	Estudo geral de Lógica Clássica	Básica	1	6	9
10	Redes de Petri	4	Noções de Redes de Petri	Básica	1	4	14
11	Sistemas Operacionais I	4	Conceitos de Sistemas Operacionais	Básica	1	7	4
12	Sistemas Operacionais II	2	Conceitos de Sistemas Operacionais Distribuídos	Específica	1	7	3

Tabela B.34 - Extensão da Tabela *Disciplina*

## Tabela: Professor

Cd_Professor	Nm_Professor	Nm_Email	Cd_Area	Tp_Professor	Cd_Departamento
1	Marcus C Sampaio	sampaio@dsc.ufpb.br	1	Permanente	1
2	Ulrich Schiel	ulrich@dsc.ufpb.br	1	Permanente	1
3	Francisco V Brasileiro	fubica@dsc.ufpb.br	7	Permanente	1
4	Jacques Sauve	jacques@dsc.ufpb.br	2	Permanente	1
5	Giuseppe Mongiovi	giuseppe@dsc.ufpb.br	6	Permanente	1
6	Pedro S Nicoletti	peter@dsc.ufpb.br	2	Permanente	1
7	Maria I C Cabral	izabel@dsc.ufpb.br	2	Permanente	1
8	Maria de F V Turnell	turnellm@dsc.ufpb.br	5	Permanente	2
9	Bernardo L Júnior	bernardo@dsc.ufpb.br	5	Permanente	1
10	Homero Cavalcanti	homero@dsc.ufpb.br	6	Permanente	1
11	Edilson Ferneda	edilson@dsc.ufpb.br	6	Permanente	1
12	Cláudio Baptista	baptista@dsc.ufpb.br	1	Permanente	1
13	Marcelo Barros	barros@dsc.ufpb.br	8	Permanente	1
14	Ângelo Perkusic	angelo@dsc.ufpb.br	4	Permanente	1

Tabela B.35 - Extensão da Tabela *Professor*

Tabela: Tese

Cd_Tese	Nm_Titulo	Nm_Area_Concentracao	Dt_Inicio	Dt_Termino	Dt_Defesa	Nu_Pagina	Nm_Idioma	Nm_Resumo	Nm_Projeto_Pesquisa	Mt_Aluno	Cd_Area
1	VBDR - Uma Arquitetura Híbrida para a Integração de Múltiplos Bancos de Dados	Integração SGBD-Web	9 Mar 1998 0:00	9 Mar 2000 0:00		0	Português			1	1
2	Título não definido	Interface	9 Mar 1998 0:00	9 Mar 2000 0:00		0	Português			2	5
3	Título não definido	Data Warehousing	9 Mar 1997 0:00	9 Dez 1999 0:00		0	Português			3	1
4	Título não definido	Avaliação de Interfaces	9 Mar 1998 0:00	9 Set 2000 0:00		0	Português			4	5
5	Um Sistema Web-Based para Gerência de Redes	Redes de Computadores	9 Mar 1998 0:00	9 Mar 2000 0:00		0	Português			5	4
6	Título não definido		9 Mar 1999 0:00	9 Mar 2001 0:00		0	Português			6	1
7	Título não definido		9 Mar 1999 0:00	9 Mar 2001 0:00		0	Português			7	1
8	Um framework para o desenvolvimento de aplicações de gerência de faltas	Redes de Computadores	9 Mar 1998 0:00	9 Set 2000 0:00		0	Português			8	4
9	Título não definido		9 Mar 1999 0:00	9 Mar 2001 0:00		0	Português			9	1
10	Título não definido	Banco de Dados	9 Mar 1998 0:00	9 Mar 2000 0:00		0	Português			10	1
11	Uma Metodologia para o Projeto de Redes Multimidia	Redes de Computadores	9 Mar 1998 0:00	9 Mar 2000 0:00		0	Português			11	2
12	Título não definido	Data Warehousing	9 Mar 1997 0:00	9 Dez 1999 0:00		0	Português			12	1
13	Título não definido	Interface	9 Mar 1998 0:00	9 Mar 2000 0:00		0	Português			13	5
14	Título não definido		9 Mar 1996 0:00	9 Mar 1997 0:00	17 Set 1999 0:00	88	Português			14	6
15	Um Sistema para o Reconhecimento de Caracteres Manuscritos	Data Mining	9 Mar 1998 0:00	9 Mar 2000 0:00		0	Português			15	1

16	Título não definido		9 Mar 1998 0:00	9 Mar 2000 0:00		0	Português			16	1
17	Um Browser Personalizado para Bancos de Dados Relacionais	Interface	9 Mar 1996 0:00	9 Mar 1997 0:00	20 Jul 1999 0:00	103	Português			17	5
18	Título não definido		9 Mar 1999 0:00	9 Mar 2001 0:00		0	Português			18	4
19	Título não definido		9 Mar 1996 0:00	9 Mar 1997 0:00	19 Set 1999 0:00	101	Português			19	7
20	Métodos para o Projeto de Redes com Requisitos de Confiabilidade	Redes de Computadores	9 Mar 1998 0:00	9 Set 2000 0:00		0	Português			20	4

Tabela B.36 - Extensão da Tabela *Tese*

Tabela: Tese\_Artigo

Cd_Tese	Cd_Artigo
15	1

Tabela B.37 - Extensão da Tabela *Tese\_Artigo*

Tabela: Tese\_Professor

Cd_Tese	Cd_Professor	Tp_Status_Prof
1	1	Orientador
2	2	Orientador
3	1	Orientador
4	8	Orientador
5	4	Orientador
6	1	Orientador
7	1	Orientador
8	4	Orientador
9	1	Orientador
10	2	Orientador
11	4	Orientador
12	1	Orientador
13	9	Orientador
14	10	Orientador
15	1	Orientador
15	5	Co-Orientador
16	2	Orientador
17	1	Orientador
18	4	Orientador
19	3	Orientador
20	4	Orientador

Tabela B.38 - Extensão da Tabela *Tese\_Professor*

## B.4 Metabanco de Dados

Tabela: Atributo

Cd_Atributo	Nm_Atributo	Cd_Tipo	Ds_Atributo	Flag_Chave_Primary	Flag_Chave_Estrangeira	Cd_Tabela	Cd_Atributo_Referencia
1	Cd_Autor	3		1	0	1	
2	Nome_Autor	6		0	0	1	
3	E_Mail	6		0	0	1	
4	Tipo	6		0	0	1	
5	Cd_Seminario	3		1	0	2	
6	Nome_Seminario	6		0	0	2	
7	Descricao	6		0	0	2	
8	Data	5		0	0	2	
9	Horario	2		0	0	2	
10	Cd_Palestrante	3		0	1	2	1
11	Cd_Aluno	3		1	0	3	
12	Nm_Aluno	6		0	0	3	
13	Pr_Entrada	3		0	0	3	
14	Orientador	6		0	0	3	
15	Dt_Defesa_Estagio	5		0	0	3	
16	Dt_Defesa_Proposta	5		0	0	3	
17	Dt_Defesa_Dissertacao	5		0	0	3	
18	Area_Pesquisa	6		0	0	3	
19	Status	6		0	0	3	
20	Cd_Disciplina	3		1	0	4	
21	Nm_Disciplina	6		0	0	4	
22	Creditos	3		0	0	4	
23	Tp_Disciplina	6		0	0	4	
24	Area_Pesquisa	6		0	0	4	
25	Cd_Aluno	3		1	1	5	11
26	Cd_Professor	3		1	1	5	31
27	Cd_Tempo	3		1	1	5	35
28	Cd_Disciplina	3		1	1	5	20
29	Ft_Artificial	3		0	0	5	
30	Resultado	6		0	0	5	
31	Cd_Professor	3		1	0	6	
32	Nm_Professor	6		0	0	6	

33	Area_Pesquisa	6		0	0	6	
34	Nm_Email	6		0	0	6	
35	Cd_Tempo	3		1	0	7	
36	Semestre	3		0	0	7	
37	Ano	3		0	0	7	
38	Mt_Aluno	3		1	0	8	
39	Nm_Aluno	6		0	0	8	
40	Pr_Entrada	6		0	0	8	
41	Dt_Nascimento	5		0	0	8	
42	Nm_Municipio_Nasc	6		0	0	8	
43	Nm_UF_Nasc	1		0	0	8	
44	Nm_Email	6		0	0	8	
45	Nm_Endereco	6		0	0	8	
46	Nm_Complemento	6		0	0	8	
47	Nm_Bairro	6		0	0	8	
48	Nm_Municipio_End	6		0	0	8	
49	Nm_UF_End	1		0	0	8	
50	Nu_Fone	3		0	0	8	
51	Nm_Titulacao	6		0	0	8	
52	Nm_Instituicao_Origem	6		0	0	8	
53	Homepage	6		0	0	8	
54	Mt_Aluno	3		1	1	9	38
55	Cd_Disciplina	3		1	1	9	86
56	Situacao	6		0	0	9	
57	Conceito	1		0	0	9	
58	Pr_Disciplina	3		0	0	9	
59	Cd_Area	3		1	0	10	
60	Nm_Area	6		0	0	10	
61	Cd_Curso	3		0	1	10	77
62	Cd_Artigo	3		1	0	11	
63	Nm_Artigo	6		0	0	11	
64	Tp_Artigo	6		0	0	11	
65	Nm_Idioma	6		0	0	11	
66	Tp_Publicacao	6		0	0	11	
67	Nm_Publicacao	6		0	0	11	
68	Dt_Publicacao	5		0	0	11	
69	Cd_Area	3		0	1	11	59
70	Cd_Bolsa	3		1	0	12	
71	Ds_Bolsa	6		0	0	12	

72	Nm_Orgao_Financiador	6		0	0	12	
73	VI_Bolsa	4		0	0	12	
74	Dt_Inicio	5		0	0	12	
75	Dt_Termino	5		0	0	12	
76	Mt_Aluno	3		0	1	12	38
77	Cd_Curso	3		1	0	13	
78	Nm_Curso	6		0	0	13	
79	Qt_Creditos	3		0	0	13	
80	Nu_Horas	3		0	0	13	
81	Cd_Departamento	3		0	1	13	82
82	Cd_Departamento	3		1	0	14	
83	Nm_Departamento	6		0	0	14	
84	Nm_Email	6		0	0	14	
85	Nm_Instituicao	6		0	0	14	
86	Cd_Disciplina	3		1	0	15	
87	Nm_Disciplina	6		0	0	15	
88	Nu_Creditos	3		0	0	15	
89	Ds_Disciplina	6		0	0	15	
90	Tp_Disciplina	6		0	0	15	
91	Cd_Curso	3		0	1	15	77
92	Cd_Area	3		0	1	15	59
93	Mt_Professor	3		0	1	15	94
94	Cd_Professor	3		1	0	16	
95	Nm_Professor	6		0	0	16	
96	Tp_Professor	6		0	0	16	
97	Nm_Email	6		0	0	16	
98	Cd_Departamento	3		0	1	16	82
99	Cd_Area	3		0	1	16	59
100	Cd_Tese	3		1	0	17	
101	Nm_Titulo	6		0	0	17	
102	Nm_Area_Concentracao	6		0	0	17	
103	Dt_Inicio	5		0	0	17	
104	Dt_Termino	5		0	0	17	
105	Dt_Defesa	5		0	0	17	
106	Nu_Pagina	3		0	0	17	
107	Nm_Idioma	6		0	0	17	
108	Nm_Resumo	6		0	0	17	
109	Nm_Projeto_Pesquisa	6		0	0	17	
110	Mt_Aluno	3		0	1	17	38

111	Cd_Area	3		0	1	17	59
112	Cd_Artigo	3		1	1	18	62
113	Cd_Tese	3		1	1	18	100
114	Cd_Professor	3		1	1	19	94
115	Cd_Tese	3		1	1	19	100
116	Tp_Status_Prof	6		0	0	19	

Tabela B.39 - Extensão da Tabela *Atributo*

Tabela: Atributo\_Dado

Cd_Atributo	Cd_Dado
12	1
12	15
12	41
12	43
12	45
12	47
12	49
12	51
12	53
12	55
12	57
12	59
21	18
21	20
21	22
21	24
21	26
21	28
21	30
21	32
21	34
21	36
21	38
21	40
32	61
32	63
32	65
32	67
32	69
32	71
39	2
39	16
39	42
39	44
39	46
39	48
39	50
39	52
39	54
39	56
39	58
39	60
87	17
87	19
87	21
87	23
87	25
87	27
87	29
87	31
87	33
87	35
87	37
87	39
95	62
95	64
95	66

95	68
95	70
95	72

Tabela B.40 - Extensão da Tabela *Atributo\_Dado*

Tabela: Atributo\_Virtual

Cd_Atributo_Virtual	Nm_Atributo_Virtual	Cd_Tipo	Ds_Atributo_Virtual	Flag_Chave_Primary	Flag_Chave_Estrangeira	Cd_Tabela_Virtual	Cd_Atributo_Virtual_Referencia
1	Cd_Aluno	3		1	0	2	
2	Nm_Aluno	6		0	0	2	
3	Nm_Email	6		0	0	2	
4	Cd_Seminario	3		1	0	1	
5	Nome_Seminario	6		0	0	1	
6	Descricao	6		0	0	1	
7	Data	5		0	0	1	
8	Horario	2		0	0	1	
9	Cd_Aluno	3		0	1	1	1
10	Pr_Entrada	3		0	0	2	
11	Orientador	6		0	0	2	
12	Dt_Defesa_Estagio	5		0	0	2	
13	Dt_Defesa_Proposta	5		0	0	2	
14	Dt_Defesa_Dissertacao	5		0	0	2	
15	Area_Pesquisa	6		0	0	2	
16	Status	6		0	0	2	
17	Cd_Disciplina	3		1	0	3	
18	Nm_Disciplina	6		0	0	3	
19	Creditos	3		0	0	3	
20	Tp_Disciplina	6		0	0	3	
21	Area_Pesquisa	6		0	0	3	
22	Cd_Aluno	3		1	1	4	1
23	Cd_Professor	3		1	1	4	28
24	Cd_Tempo	3		1	1	4	32
25	Cd_Disciplina	3		1	1	4	17
26	Ft_Artificial	3		0	0	4	
27	Resultado	6		0	0	4	
28	Cd_Professor	3		1	0	5	
29	Nm_Professor	6		0	0	5	
30	Area_Pesquisa	6		0	0	5	
31	Nm_Email	6		0	0	5	
32	Cd_Tempo	3		1	0	6	
33	Semestre	3		0	0	6	
34	Ano	3		0	0	6	
35	Dt_Nascimento	5		0	0	2	

36	Nm_Municipio_Nasc	6		0	0	2	
37	Nm_UF_Nasc	1		0	0	2	
38	Nm_Endereco	6		0	0	2	
39	Nm_Complemento	6		0	0	2	
40	Nm_Bairro	6		0	0	2	
41	Nm_Municipio_End	6		0	0	2	
42	Nm_UF_End	1		0	0	2	
43	Nu_Fone	3		0	0	2	
44	Nm_Titulacao	6		0	0	2	
45	Nm_Instituicao_Origem	6		0	0	2	
46	Homepage	6		0	0	2	
47	Mt_Aluno	3		1	1	7	1
48	Cd_Disciplina	3		1	1	7	17
49	Situacao	6		0	0	7	
50	Conceito	1		0	0	7	
51	Pr_Disciplina	3		0	0	7	
52	Cd_Area	3		1	0	8	
53	Nm_Area	6		0	0	8	
54	Cd_Curso	3		0	1	8	70
55	Cd_Artigo	3		1	0	9	
56	Nm_Artigo	6		0	0	9	
57	Tp_Artigo	6		0	0	9	
58	Nm_Idioma	6		0	0	9	
59	Tp_Publicacao	6		0	0	9	
60	Nm_Publicacao	6		0	0	9	
61	Dt_Publicacao	5		0	0	9	
62	Cd_Area	3		0	1	9	52
63	Cd_Bolsa	3		1	0	10	
64	Ds_Bolsa	6		0	0	10	
65	Nm_Orgao_Financiador	6		0	0	10	
66	Vl_Bolsa	4		0	0	10	
67	Dt_Inicio	5		0	0	10	
68	Dt_Termino	5		0	0	10	
69	Mt_Aluno	3		0	1	10	1
70	Cd_Curso	3		1	0	11	
71	Nm_Curso	6		0	0	11	
72	Qt_Creditos	3		0	0	11	
73	Nu_Horas	3		0	0	11	
74	Cd_Departamento	3		0	1	11	75

75	Cd_Departamento	3		1	0	12	
76	Nm_Departamento	6		0	0	12	
77	Nm_Email	6		0	0	12	
78	Nm_Instituicao	6		0	0	12	
79	Ds_Disciplina	6		0	0	3	
80	Cd_Curso	3		0	1	3	70
81	Cd_Area	3		0	1	3	52
82	Mt_Professor	3		0	1	3	28
83	Tp_Professor	6		0	0	5	
84	Cd_Departamento	3		0	1	5	75
85	Cd_Area	3		0	1	5	52
86	Cd_Tese	3		1	0	13	
87	Nm_Titulo	6		0	0	13	
88	Nm_Area_Concentracao	6		0	0	13	
89	Dt_Inicio	5		0	0	13	
90	Dt_Termino	5		0	0	13	
91	Dt_Defesa	5		0	0	13	
92	Nu_Pagina	3		0	0	13	
93	Nm_Idioma	6		0	0	13	
94	Nm_Resumo	6		0	0	13	
95	Nm_Projeto_Pesquisa	6		0	0	13	
96	Mt_Aluno	3		0	1	13	1
97	Cd_Area	3		0	1	13	52
98	Cd_Artigo	3		1	1	14	55
99	Cd_Tese	3		1	1	14	86
100	Cd_Professor	3		1	1	15	28
101	Cd_Tese	3		1	1	15	86
102	Tp_Status_Prof	6		0	0	15	
103	Cd_Professor	3		0	1	1	28

Tabela B.41 - Extensão da Tabela *Atributo\_Virtual*

Tabela: Atrib\_Virt\_Mapeado\_Atributo

Cd_Atributo_Virtual	Cd_Atributo
4	5
5	6
6	7
7	8
8	9
9	10
1	1
1	11
1	38
2	2
2	12
2	39
10	13
10	40
11	14
12	15
13	16
14	17
15	18
16	19
17	20
17	86
18	21
18	87
19	22
19	88
20	23
20	90
21	24
22	25
23	26
24	27
25	28
26	29
27	30
28	1
28	31
28	94
29	2
29	32
29	95
30	33
31	3
31	34
31	97
32	35
33	36
34	37
35	41
36	42
37	43
3	3
3	44
38	45
39	46
40	47
41	48
42	49

43	50
44	51
45	52
46	53
47	54
48	55
49	56
50	57
51	58
52	59
53	60
54	61
55	62
56	63
57	64
58	65
59	66
60	67
61	68
62	69
63	70
64	71
65	72
66	73
67	74
68	75
69	76
70	77
71	78
72	79
73	80
74	81
75	82
76	83
77	84
78	85
79	89
80	91
81	92
82	93
83	96
84	98
85	99
86	100
87	101
88	102
89	103
90	104
91	105
92	106
93	107
94	108
95	109
96	110
97	111
98	112
99	113
100	114
101	115
102	116
104	10

103	10
4	5
5	6
6	7
7	8
8	9
9	10
103	10
1	1
1	11
1	38
2	2
2	12
2	39
10	13
10	40
11	14
12	15
13	16
14	17
15	18
16	19
17	20
17	86
18	21
18	87
19	22
19	88
20	23
20	90
21	24
22	25
23	26
24	27
25	28
26	29
27	30
28	1
28	31
28	94
29	2
29	32
29	95
30	33
31	3
31	34
31	97
32	35
33	36
34	37
35	41
36	42
37	43
3	3
3	44
38	45
39	46
40	47
41	48
42	49
43	50

44	51
45	52
46	53
47	54
48	55
49	56
50	57
51	58
52	59
53	60
54	61
55	62
56	63
57	64
58	65
59	66
60	67
61	68
62	69
63	70
64	71
65	72
66	73
67	74
68	75
69	76
70	77
71	78
72	79
73	80
74	81
75	82
76	83
77	84
78	85
79	89
80	91
81	92
82	93
83	96
84	98
85	99
86	100
87	101
88	102
89	103
90	104
91	105
92	106
93	107
94	108
95	109
96	110
97	111
98	112
99	113
100	114
101	115
102	116
104	10

Tabela B.42 - Extensão da Tabela *Atrib\_Virt\_Mapeado\_Atributo*

Tabela: Banco

Cd_Banco	Nm_Banco	Cd_Gerenciador
1	BDSEMINAR	2
2	DWCOPIN	2
3	SYSCOPIN	2

Tabela B.43 - Extensão da Tabela *Banco*

Tabela: Consulta

Cd_Consulta	Nm_Consulta	Ds_Consulta	Nm_Criador	Dt_Criacao
0	Dados Elementares	Esta consulta recupera o nome e a área de pesquisa do Professor Marcus Costa Sampaio	Administrador	25 Mai 2000 0:00
1	Seminários Anuais	Esta consulta recupera título, data e horário dos seminários apresentados pelo Professor Marcus Costa Sampaio	Administrador	25 Mai 2000 0:00
2	Quantidade de disciplinas ministradas na Copin nos últimos dois anos	Esta consulta recupera a quantidade de disciplinas ministradas pelo Professor Marcus Costa Sampaio, na Copin, nos últimos dois anos	Administrador	26 Mai 2000 0:00
3	Número atual de orientandos	Esta consulta recupera a quantidade de orientandos do Professor Marcus Costa Sampaio	Administrador	26 Mai 2000 0:00
4	Relação das teses com respectivos alunos orientados do ano de 97 até o ano atual	Esta consulta recupera o título da tese e nome do aluno das teses defendidas de 1997 até o ano atual	Administrador	26 Mai 2000 0:00
5	Dados Elementares	Esta consulta recupera nome, número de créditos, descrição, tipo e área de pesquisa da disciplina Banco de Dados	Administrador	26 Mai 2000 0:00
6	Quantidade de vezes em que a disciplina Banco de Dados foi oferecida desde o ano de 1997	Esta consulta recupera a quantidade de vezes em que a disciplina Banco de Dados foi oferecida desde o ano de 1997	Administrador	26 Mai 2000 0:00
7	Média de alunos que cursaram a disciplina desde o ano de 1997	Esta consulta recupera a média de alunos que cursaram a disciplina desde o ano de 1997	Administrador	26 Mai 2000 0:00
8	Dados Elementares	Esta consulta recupera nome, e-mail, titulação, instituição de origens e datas de defesa de estágio, proposta de dissertação e dissertação do aluno Carlos Eduardo Santos Pires	Administrador	26 Mai 2000 0:00
9	Dados da Bolsa e Orientador	Esta consulta recupera alguns atributos relativos à bolsa do aluno Carlos Eduardo Santos Pires, além do nome de seu orientador	Administrador	26 Mai 2000 0:00
10	Seminários Anuais	Esta consulta recupera título, data e horário dos seminários apresentados pelo Aluno Carlos Eduardo Santos Pires	Administrador	29 Mai 2000 0:00

Tabela B.44 - Extensão da Tabela *Consulta*

Tabela: Dado

Cd_Dado	Nm_Dado
1	Carlos Eduardo Santos Pires
2	Carlos E S Pires
15	Juliano Varella de Carvalho
16	Juliano Varella
17	Banco de Dados
18	Banco de Dados
19	Banco de Dados Avançado
20	Banco de D Avançado
21	Redes de Computadores e Protocolos
22	Redes de C e Protocolos
23	Inteligência Artificial
24	IA
25	Data Mining
26	Data Mining
27	Data Warehousing
28	Data Warehousing
29	Sistemas Especialistas
30	Sistemas Especialistas
31	Projeto de Interfaces Homem-Máquina
32	Projeto de I Homem-Máquina
33	Lógica Clássica
34	Lógica Clássica
35	Redes de Petri
36	Redes de Petri
37	Sistemas Operacionais I
38	Sistemas Operacionais I
39	Sistemas Operacionais II
40	Sistemas Operacionais II
41	André Barbosa Rocha
42	André B Rocha
43	Adriano Wagner de Araújo Bezerra
44	Adriano W de A Bezerra
45	Cristine Pires da Costa
46	Cristine P da Costa
47	Marinaldo Nunes da Silva
48	Marinaldo N da Silva
49	Márcio Farias de Souza
50	Márcio F de Souza
51	Algeir Prazeres Sampaio
52	Algeir P Sampaio
53	Júlio Cezar Oliveira Gomes
54	Júlio C O Gomes
55	Ana Isabel Pinheiro da Silva
56	Ana I P da Silva
57	Giovanni Almeida Santos
58	Giovanni A Santos
59	Gustavo Vasconcellos Cavalcante
60	Gustavo V Cavalcante
61	Marcus Costa Sampaio
62	Marcus C Sampaio
63	Francisco Villar Brasileiro
64	Francisco V Brasileiro
65	Pedro Sérgio Nicoletti
66	Pedro S Nicoletti
67	Maria Izabel Cavalcanti Cabral
68	Maria I C Cabral
69	Maria de Fátima Vieira Turnell
70	Maria de F V Turnell

71	Bernardo Lula Júnior
72	Bernardo L Júnior

Tabela B.45 - Extensão da Tabela *Dado*

Tabela: Dado\_Mapeado\_Dado

Cd_Dado	Cd_Dado_Mapeado
2	1
16	15
18	17
20	19
22	21
24	23
26	25
28	27
30	29
32	31
34	33
36	35
38	37
40	39
42	41
44	43
46	45
48	47
50	49
52	51
54	53
56	55
58	57
60	59
62	61
64	63
66	65
68	67
70	69
72	71

Tabela B.46 - Extensão da Tabela *Dado\_Mapeado\_Dado*

Tabela: Esquema\_Global

Cd_Esquema_Global	Nm_Esquema_Global	Ds_Esquema_Global
1	COPIN	Integra os BDs BDSEMINAR, DWCOPIIN e SYSCOPIIN

Tabela B.47 - Extensão da Tabela *Esquema\_Global*

Tabela: Esq\_Glob\_Possui\_Tab\_Virt

Cd_Esquema_Global	Cd_Tabela_Virtual
1	1
1	2
1	3
1	4
1	5
1	6
1	7
1	8
1	9
1	10
1	11
1	12
1	13
1	14
1	15

Tabela B.48 - Extensão da Tabela *Esq\_Glob\_Possui\_Tab\_Virt*

Tabela: Gerenciador

Cd_Gerenciador	Nm_Gerenciador
1	Interbase
2	SQL Server
3	Oracle

Tabela B.49 - Extensão da Tabela *Gerenciador*

Tabela: Relatório

Cd_Relatorio	Nm_Relatorio	Nm_Assunto	Ds_Relatorio	Nm_Criador	Dt_Criacao
1	Relatório do Professor Marcus Costa Sampaio	Atividades acadêmicas do Professor Marcus Costa Sampaio	Este relatório exhibe dados importantes do professor Marcus Costa Sampaio com relação as suas atividades acadêmicas no curso de Mestrado em informática da UFPB/CCT/Copin	Administrador	25 Mai 2000 0:00
2	Relatório da Disciplina Banco de Dados	Disciplina Banco de Dados	Este relatório exhibe dados importantes a respeito da disciplina Banco de Dados, oferecida pela UFPB/CCT/Copin	Administrador	25 Mai 2000 0:00
3	Relatório do Aluno Carlos Eduardo Santos Pires	Dados do aluno Carlos Eduardo Santos Pires (Copin)	Este relatório exhibe dados importantes a respeito do aluno Carlos Eduardo Santos Pires, do curso de Mestrado em Informática da UFPB/CCT/Copin	Administrador	25 Mai 2000 0:00

Tabela B.50 - Extensão da Tabela *Relatório*

Tabela: Subconsulta

Cd_Subconsulta	VI_Subconsulta	Cd_Consulta	Cd_Banco
1	SELECT Professor.Nm_Professor, Area.Nm_Area FROM Professor, Area WHERE Professor.Nm_Professor LIKE '%Marcus%' AND Professor.Cd_Area = Area.Cd_Area	0	2
2	SELECT Seminario.Nome_Seminario, Seminario.Data, Seminario.Horario, Autor.Nome_Autor FROM Seminario, Autor WHERE Autor.Nome_Autor LIKE '%Marcus%' AND Seminario.Cd_Palestrante = Autor.Cd_Autor ORDER BY Seminario.Nome_Seminario	1	0
3	SELECT Professor.Nm_Professor, COUNT(Fatos.Ft_Artificial) FROM Professor, Fatos, Tempo, Disciplina WHERE Professor.Nm_Professor LIKE '%Marcus%' AND Tempo.Ano >= 1997 AND Fatos.Cd_Tempo = Tempo.Cd_Tempo AND Fatos.Cd_Professor = Professor.Cd_Professor AND F	2	1
4	SELECT Professor.Nm_Professor FROM Professor, Disciplina WHERE Professor.Nm_Professor LIKE '%Marcus%' AND Professor.Cd_Professor = Disciplina.Mt_Professor GROUP BY Professor.Nm_Professor	2	2
5	SELECT Aluno.Orientador, COUNT(Aluno.Cd_Aluno) FROM Aluno WHERE Aluno.Orientador LIKE '%Marcus%' AND Aluno.Status = 'Mestrando' GROUP BY Aluno.Orientador	4	1
6	SELECT Aluno.Nm_Aluno FROM Aluno, Professor, Tempo, Fatos WHERE Professor.Nm_Professor LIKE '%Marcus%' AND Tempo.Ano >= 1997 AND Tempo.Ano <= 2000 AND Aluno.Cd_Aluno = Fatos.Cd_Aluno AND Fatos.Cd_Tempo = Tempo.Cd_Tempo AND Professor.Cd_Professor = Fatos.C	5	1
7	SELECT Tese.Nm_Titulo, Aluno.Nm_Aluno FROM Tese, Aluno, Professor, Tese_Professor WHERE Professor.Nm_Professor LIKE '%Marcus%' AND Tese_Professor.Tp_Status_Prof = 'Orientador' AND Tese_Professor.Cd_Tese = Tese.Cd_Tese AND Tese.Mt_Aluno = Aluno.Mt_Aluno AN	5	2
8	SELECT Disciplina.Nm_Disciplina, Disciplina.Creditos, Disciplina.Tp_Disciplina, Disciplina.Area_Pesquisa FROM Disciplina WHERE Disciplina.Nm_Disciplina = 'Banco de Dados'	6	1
9	SELECT Disciplina.Nm_Disciplina, Disciplina.Nu_Creditos, Disciplina.Ds_Disciplina, Disciplina.Tp_Disciplina FROM Disciplina WHERE Disciplina.Nm_Disciplina = 'Banco de Dados'	6	2
10	SELECT Disciplina.Nm_Disciplina, COUNT(Fatos.Ft_Artificial) FROM Disciplina, Fatos, Tempo WHERE Disciplina.Nm_Disciplina = 'Banco de Dados' AND Tempo.Ano >= 1997 AND Fatos.Cd_Disciplina = Disciplina.Cd_Disciplina AND Fatos.Cd_Tempo = Tempo.Cd_Tempo GROUP	7	1
11	SELECT Disciplina.Nm_Disciplina, AVG(Fatos.Ft_Artificial) FROM Disciplina, Fatos, Tempo WHERE Disciplina.Nm_Disciplina = 'Banco de Dados' AND Tempo.Ano >= 1997 AND Fatos.Cd_Tempo = Tempo.Cd_Tempo AND Fatos.Cd_Disciplina = Disciplina.Cd_Disciplina GROUP B	8	1
12	SELECT Aluno.Nm_Aluno, Aluno.Dt_Defesa_Estagio, Aluno.Dt_Defesa_Proposta, Aluno.Dt_Defesa_Dissertacao FROM Aluno WHERE Aluno.Nm_Aluno LIKE '%Carlos E%'	9	1
13	SELECT Aluno.Nm_Aluno, Aluno.Nm_Email, Aluno.Nm_Titulacao, Aluno.Nm_Instituicao_Origem FROM Aluno WHERE Aluno.Nm_Aluno LIKE '%Carlos E%'	9	2
14	SELECT Aluno.Nm_Aluno, Aluno.Orientador FROM Aluno WHERE Aluno.Nm_Aluno LIKE '%Carlos E%'	10	1
15	SELECT Aluno.Nm_Aluno, Bolsa.Ds_Bolsa, Bolsa.Nm_Orgao_Financiador, Bolsa.Dt_Inicio, Bolsa.Dt_Termino FROM Aluno, Bolsa WHERE Aluno.Nm_Aluno LIKE '%Carlos E%' AND Bolsa.Mt_Aluno = Aluno.Mt_Aluno	10	2
16	SELECT Seminario.Nome_Seminario, Seminario.Data, Seminario.Horario, Autor.Nome_Autor FROM Seminario, Autor	11	0

	WHERE Autor.Nome_Autor LIKE '%Carlos E%' AND Seminario.Cd_Palestrante = Autor.Cd_Autor ORDER BY Seminario.Nome_Seminario		
--	--	--	--

Tabela B.51 - Extensão da Tabela *Subconsulta*

Tabela: Tabela

Cd_Tabela	Nm_Tabela	Cd_Banco
1	Autor	1
2	Seminario	1
3	Aluno	2
4	Disciplina	2
5	Fatos	2
6	Professor	2
7	Tempo	2
8	Aluno	3
9	Aluno_Disciplina	3
10	Area	3
11	Artigo	3
12	Bolsa	3
13	Curso	3
14	Departamento	3
15	Disciplina	3
16	Professor	3
17	Tese	3
18	Tese_Artigo	3
19	Tese_Professor	3

Tabela B.52 - Extensão da Tabela *Tabela*

Tabela: Tabela\_Virtual

Cd_Tabela_Virtual	Nm_Tabela_Virtual	Ds_Tabela_Virtual
1	Seminario	
2	Aluno	
3	Disciplina	
4	Fatos	
5	Professor	
6	Tempo	
7	Aluno_Disciplina	
8	Area	
9	Artigo	
10	Bolsa	
11	Curso	
12	Departamento	
13	Tese	
14	Tese_Artigo	
15	Tese_Professor	

Tabela B.53 - Extensão da Tabela *Tabela\_Virtual*

Tabela: Tab\_Virt\_Mapeada\_Tabela

Cd_Tabela_Virtual	Cd_Tabela
1	2
2	1
2	3
2	8
3	4
3	15
4	5
5	1
5	6
5	16
6	7
7	9
8	10
9	11
10	12
11	13
12	14
13	17
14	18
15	19

Tabela B.54 - Extensão da Tabela *Tab\_Virt\_Mapeada\_Tabela*

Tabela: Tab\_Virt\_Referencia\_Tab\_Virt

Cd_Tabela_Virtual_Referencia	Cd_Tabela_Virtual_Referenciada
1	2
1	5
3	5
3	8
3	11
4	2
4	3
4	5
4	6
5	8
5	12
7	2
7	3
8	11
9	8
10	2
11	12
13	2
13	8
14	9
14	13
15	5
15	13

Tabela B.55 - Extensão da Tabela *Tab\_Virt\_Referencia\_Tab\_Virt*

Tabela: Tipo

Cd_Tipo	Nm_Tipo
1	Char
2	Datetime
3	Int
4	Money
5	Smalldatetime
6	Varchar
7	Bit

Tabela B.56 - Extensão da Tabela *Tipo*