

CENTRO UNIVERSITÁRIO FEEVALE

JONATAS CABERLON

UMA SOLUÇÃO, BASEADA EM FERRAMENTAS *ORACLE*,  
PARA O DESENVOLVIMENTO DE UM SISTEMA DE *BUSINESS*  
*INTELLIGENCE*

Novo Hamburgo, junho de 2007.

JONATAS CABERLON

UMA SOLUÇÃO, BASEADA EM FERRAMENTAS *ORACLE*,  
PARA O DESENVOLVIMENTO DE UM SISTEMA DE *BUSINESS*  
*INTELLIGENCE*

Centro Universitário Feevale  
Instituto de Ciências Exatas e Tecnológicas  
Curso de Ciência da Computação  
Trabalho de Conclusão de Curso

Professor orientador: Juliano Varella de Carvalho

Novo Hamburgo, junho de 2007.

## AGRADECIMENTOS

Gostaria de agradecer primeiramente aos meus pais que, com muita luta, puderam me dar a oportunidade de cursar uma faculdade.

Aos colegas de faculdade, por estarem comigo nos momentos mais difíceis do curso.

Aos colegas de trabalho, que de alguma forma, trouxeram contribuição a esse trabalho.

E ao professor-orientador Juliano, pelo seu brilhante apoio no desenvolvimento dessa primeira etapa do trabalho.

## RESUMO

O cenário globalizado atual exige uma intensa busca na excelência na atuação das empresas no mercado, assim como a adaptação às constantes mudanças de cenários econômicos. Para isso, o uso de sistemas de informação que auxiliem a tomada de decisão é de suma importância. As empresas, para reagir aos concorrentes, clientes, fornecedores, mudanças sociais e tecnológicas, constroem sistemas para auxiliar neste ambiente dinâmico. Portanto, a implantação de um Sistema de *Business Intelligence* poderá ser fundamental para o processo decisório da empresa. A Box Print, empresa de embalagens, possui todos os seus processos informatizados com um grande volume e variabilidade de dados. Porém, não há sistema a nível gerencial, que traga em um mesmo ambiente, consultas e relatórios que auxiliem a tomada de decisão de gerentes e diretores. Utilizando tecnologia Oracle, esse trabalho tem por objetivo desenvolver um sistema de Business Intelligence para um Data Mart comercial, que primará em facilitar a obtenção dos dados necessários para o processo decisório a nível gerencial.

Palavras-chave: *Business Intelligence*. OLAP. *Data Warehouse*. Oracle. Sistema de Apoio à Decisão (SAD)

## ABSTRACT

The current globalized scenery demands an intense search for the excellence in company's performance at the market as well as adaptation to the constant changes of economic scenes. Because of this, the use of systems of information that helps in the decision taking is of utmost importance. The companies, to react to the competitors, clients, suppliers, social and technologies changes, build systems to helps on this dynamic work's place. Therefore, the implantation of Business Intelligence System can be fundamental for this decision process of the company. Box Print, packaging company, has all of its computerized process with a great amount and variability of data. However, there is no system at a managerial level, that brings at same work's place, consultation and reports that helps the decision taking of managers and directors. Using Oracle technology, this final paper has as objective to develop a system of Business Intelligence for a Data Mart commercial, which will ease the attainment of necessary data to the decisive process at a managerial level.

Keywords: Business Intelligence. OLAP. Data Warehouse. Oracle. Decision Support Systems (DSS)

## LISTA DE FIGURAS

Figura 1.1 – Processo de tratamento de dados.....	22
Figura 1.2 – A questão da não-volatilidade.....	22
Figura 1.3 – Composição básica de uma tabela de fatos.....	26
Figura 1.4 – Exemplo de uma tabela de dimensão.....	27
Figura 1.5 – Exemplo de um esquema em estrela.....	29
Figura 1.6 – Exemplo de um esquema em <i>snowflake</i> .....	30
Figura 1.7 – Cubo de dados.....	32
Figura 1.8 – Níveis de granularidade.....	33
Figura 1.9 – Tabelas agregadas.....	34
Figura 1.10 – Exemplo das funcionalidades <i>Drill-down</i> e <i>Roll-up</i> .....	38
Figura 1.11 – <i>Data Mining</i> dentro do contexto de <i>Business Intelligence</i> .....	45
Figura 2.1 – Exemplo de um <i>wizard</i> na criação de tabelas no OWB.....	47
Figura 2.2 – <i>Oracle Warehouse Builder</i> .....	48
Figura 2.3 – Componentes do <i>Oracle Discoverer</i> .....	55
Figura 2.4 – <i>Discoverer Administrator</i> .....	56
Figura 2.5 – Estrutura do <i>Discoverer Desktop</i> .....	57
Figura 2.6 – Exemplo de consulta com tabela de referência cruzada.....	58
Figura 3.1 – Estrutura organizacional da empresa.....	68
Figura 3.2 – Organização do gerenciamento de vendas.....	69
Figura 3.3 – Modelagem dimensional.....	71
Figura 3.4 – Hierarquia principal dos dados do sistema.....	72
Figura 3.5 – Estrutura do sistema.....	74

## LISTA DE QUADROS

Quadro 1.1 – Comparação entre os dados de natureza operacional e informacional.	18
Quadro 1.2 – Diferenças entre DW e DM.....	35
Quadro 2.1 – Recursos das ferramentas de <i>Data Warehouse</i> .....	52
Quadro 2.2 – Comparativo de ferramentas <i>Data Warehouse</i> .....	53
Quadro 2.3 – Recursos das ferramentas OLAP.....	64
Quadro 2.4 – Comparação das ferramentas OLAP.....	65

## LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>
BD	Banco de Dados
BI	<i>Business Intelligence</i>
DM	<i>Data Mart</i>
DOLAP	<i>Desktop On-line Analytic Processing</i>
DW	<i>Data Warehouse</i>
ER	Entidade-relacionamento
ERP	<i>Enterprise Resource Planning</i>
ETL	<i>Extraction, Transformation and Loading</i>
HOLAP	<i>Hybrid On-line Analytic Processing</i>
HTML	<i>HyperText Markup Language</i>
J2EE	<i>Java 2Platform Enterprise Edition</i>
JCP	<i>Java Community Process</i>
JDBC	<i>Java Database Connectivity</i>
JOLAP	<i>Java On-line Analytic Processing</i>
MOLAP	<i>Multidimensional On-Line Analytic Processing</i>
ODBC	<i>Open Database Connectivity</i>
OLAP	<i>On-line Analytic Processing</i>
OLTP	<i>On-line Transactional Processing</i>



OWB	<i>Oracle Warehouse Builder</i>
PDF	<i>Portable Document Format</i>
RDL	<i>Report Definition Language</i>
ROLAP	<i>Relational On-Line Analytic Processing</i>
SAD	Sistema de Apoio à Decisão
SGBD	Sistema Gerenciador de Banco de Dados
SGBDR	Sistema Gerenciador de Banco de Dados Relacional
SQL	<i>Structured Query Language</i>
TI	Tecnologia da Informação
WOLAP	<i>Web On-line Analytic Processing</i>
XML	<i>EXtensible Markup Language</i>
XMLA	<i>EXtensible Markup Language for Analysis</i>

## SUMÁRIO

<b>INTRODUÇÃO .....</b>	<b>12</b>
<b>1 BUSINESS INTELLIGENCE (BI) .....</b>	<b>16</b>
1.1 Dados Operacionais x Dados Informativos .....	17
1.2 <i>Data Warehouse</i> .....	18
1.2.1 Histórico .....	19
1.2.2 Características do DW .....	21
1.2.2.1 Orientado a assuntos .....	21
1.2.2.2 Integrado .....	21
1.2.2.3 Não-Volátil .....	22
1.2.2.4 Variável com o tempo .....	23
1.2.3 Componentes do DW .....	23
1.2.4 Modelagem de Dados .....	24
1.2.4.1 Tabela de Fatos .....	25
1.2.4.1.1 Atributos Aditivos, semi-aditivos e não aditivos .....	26
1.2.4.2 Tabelas de Dimensões .....	27
1.2.4.3 Técnicas de Modelagem .....	28
1.2.4.3.1 <i>Star Schema</i> .....	28
1.2.4.3.2 <i>Snowflake</i> .....	30
1.2.4.4 Cubo de dados .....	31
1.2.4.5 Granularidade .....	32
1.2.4.5 Agregados .....	33
1.2.5 <i>Data Mart</i> (DM) .....	35
1.3 OLAP .....	36
1.3.1 Características .....	36
1.3.1.1 Funções Básicas .....	37
1.3.1.2 Operações OLAP .....	37
1.3.2 Arquiteturas OLAP .....	39
1.3.2.1 ROLAP .....	39
1.3.2.2 MOLAP .....	40
1.3.2.3 HOLAP .....	41
1.3.2.4 WOLAP .....	41
1.3.2.5 DOLAP .....	41
1.3.2.6 JOLAP .....	42
1.3.2.7 ROLAP x MOLAP, qual a melhor tecnologia OLAP? .....	42
1.4 <i>Data Mining</i> .....	44
<b>2 FERRAMENTAS .....</b>	<b>46</b>

2.1 Ferramentas DW .....	46
2.1.1 <i>Oracle Warehouse Builder</i> .....	46
2.1.1.1 A Ferramenta .....	46
2.1.1.2 Características Principais .....	47
2.1.1.2.1 Orientado a Projetos .....	47
2.1.1.2.2 Fontes de dados .....	48
2.1.1.2.3 Importação de metadados .....	49
2.1.2 <i>Oracle Express</i> .....	49
2.1.3 <i>Microsoft DTS</i> .....	50
2.1.4 <i>Business Information Warehouse</i> .....	51
2.1.5 Comparativo das ferramentas DW .....	52
2.2 Ferramentas OLAP .....	53
2.2.1 <i>Oracle Business Intelligence</i> .....	53
2.2.1.1 <i>Oracle Discoverer</i> .....	54
2.2.1.1.1 <i>Discoverer Administrador</i> .....	55
2.2.1.1.2 <i>Discoverer Desktop</i> .....	56
2.2.1.1.3 <i>Discoverer Plus e Discoverer Viewer</i> .....	60
2.2.1.2 <i>Oracle Spreadsheet Add-In</i> .....	60
2.2.1.3 <i>Oracle BI Beans</i> .....	60
2.2.2 <i>Microsoft Analysis Services</i> .....	61
2.2.3 <i>Pentaho BI</i> .....	62
2.2.4 <i>OpenI</i> .....	63
2.2.5 Comparativo das ferramentas OLAP .....	64
<b>3 ESTUDO DE CASO .....</b>	<b>66</b>
3.1 Estrutura da Empresa .....	66
3.1.1 Tecnologia .....	67
3.1.2 Estrutura Organizacional .....	67
3.2 Problemas Atuais e Necessidades .....	69
3.3 Proposta do sistema .....	70
3.3.1 Dados a serem utilizados .....	70
3.3.2 Interface gráfica e funcionalidades .....	72
3.4 Arquitetura do sistema .....	73
3.5 <i>Oracle</i> .....	74
3.5.2 Por que <i>Oracle</i> .....	74
<b>CONCLUSÃO.....</b>	<b>76</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>77</b>

## INTRODUÇÃO

Uma empresa que possui grande variabilidade e volume de dados tem consigo a possibilidade de obter as melhores condições para as tomadas de decisão, que muitas vezes resultam nos rumos que ela deve tomar. Mas para que essas informações sejam utilizadas de forma positiva, é necessário que as mesmas estejam devidamente organizadas e padronizadas, para evitar que elas sejam conflitantes ou duvidosas, gerando decisões erradas.

Uma das grandes dificuldades é usufruir dos dados de modo eficiente, pois a extração destes é sempre complexa, já que estão muitas vezes divididos em diversos locais da base, e muitas vezes da própria empresa, acarretando demora para agrupá-los, interpretá-los e extrair conclusões precisas.

Esta realidade está presente na empresa (Box Print) em que o projeto será realizado. Atualmente, o volume e a variabilidade de dados nesta corporação são muito grandes. Existem consultas e relatórios utilizados dentro do sistema transacional da empresa, que geram dados gerenciais. Além disso, planilhas paralelas muitas vezes são geradas, através de dados operacionais, para que se obtenham informações a nível gerencial que possibilitem a tomada de decisão.

No cenário atual da empresa, sempre que algum diretor ou gerente tem a necessidade de ver informações que lhe possibilitem a tomada de decisão, ele acaba designando um funcionário para garimpar, dentro do sistema transacional, dados de várias consultas e relatórios.

Essa coleta implica em uma dedicação e tempo muito grandes, além disso, há sempre a probabilidade de que as informações não sejam confiáveis, pois o funcionário designado para a tarefa não possui a visão de um gerente ou diretor, podendo cometer o erro de adicionar ou até mesmo retirar dados que seriam importantes para a tomada de decisão.

Para resolver os problemas descritos, surge então a necessidade de desenvolver um sistema de informações que possibilite aos gerentes e diretores obter a informação de forma mais macro, rápida e com celeridade, interpretá-la e tomar decisões precisas.

Segundo Poloni (2000), os sistemas de informação constituem-se de qualquer sistema usado para fornecer informações, independente da sua utilização. E, de acordo com Laudon & Laudon (1991) os sistemas são divididos em quatro tipos: Sistemas de Processamento de Transações, Sistemas de Base de Conhecimento, Sistemas Gerenciais e Sistema de Suporte Estratégico.

Dentre as alternativas acima, destaca-se o Sistema de Apoio a Decisão (SAD), integrado dentro de sistemas gerenciais, pois conforme Rezende (1999, p.50), “auxiliam o executivo em todas as fases de tomada de decisão, principalmente, nas etapas de desenvolvimento, comparação e classificação de riscos, além de fornecer subsídios para a escolha de uma boa alternativa em seus negócios”.

Porém, para que um SAD traga resultados positivos para o projeto, é necessário contar com uma tecnologia adequada ao desenvolvimento de um sistema de informação desta natureza.. Assim, surge a proposta de se trabalhar com o conceito de *Business Intelligence*.

Segundo Barbieri (2001), *BI-Business Intelligence*, pode ser entendido como a utilização de variadas fontes de informação para se definir estratégias de competitividade nos negócios.

Conforme já comentado, há grandes volumes de dados, mas existe a dificuldade de extração a partir dos mesmos, dificultando o processo de tomada de decisão. O BI traz como um dos seus objetivos principais justamente a definição de regras e técnicas para a formatação ideal desses volumes de dados, tendo em vista transformá-los em depósitos estruturados de informações, independentemente da sua origem. Além disso, cabe como objetivo do *BI* a visualização dos dados com o máximo de flexibilidade para o usuário

Conforme Fortulan e Filho apud Shim et al. (2002), os *Data Warehouses*, OLAP, *Data Mining* e *Web-SAD* surgiram no começo dos anos 90 como novas ferramentas para SAD, e formam a base dos sistemas de BI. A seguir, serão abordadas as ferramentas de *Data Warehouse (DW)* e OLAP, que farão parte desse projeto.

De acordo com Boghi e Shitsuka (2002), *DW* é um conjunto de tecnologias com o objetivo de converter uma grande quantidade de dados em informações utilizáveis. E para as organizações, o *DW* acaba transformando as fontes de dados operacionais em um ambiente que permite o uso estratégico dos dados.

Dentre suas principais características, o *DW* é em um banco de dados separado do banco de dados dos sistemas transacionais da empresa, que é desenhado para realizar tarefas analíticas utilizando dados de diferentes aplicações.

O *Data Mart (DM)* é uma espécie de *DW* em menor escala e com o escopo mais definido (vários *Data Marts* podem formar um *Data Warehouse*). Por ser menor, possibilita a análise multidimensional, com os cruzamentos de dados e visões previamente calculadas, com o objetivo de aumentar a velocidade na consulta das informações.

Bispo (1998) apresenta a OLAP como uma ferramenta capaz de efetuar análises de dados com visão multidimensional do negócio, comparando-os por diversos ângulos. Ou seja, os dados são agregados em várias dimensões<sup>1</sup> para que os analistas possam interagir com o meio e visualizar possíveis informações, de maneira ágil e consistente, que representam determinada situação sob o ponto de vista do usuário.

Portanto, considerando as tecnologias apresentadas, a proposta do trabalho parte da idéia de se construir um *Data Mart* de informações comerciais e desenvolver um sistema utilizando tecnologia OLAP para a visualização dos dados.

O projeto será realizado com ferramentas da *Oracle*, pois além de ser maior empresa de software empresarial do mundo (COMUNIQUE-SE, 2007), suas tecnologias já estão inseridas como padrão de desenvolvimento de aplicações na empresa.

Para a exibição do sistema, será utilizado o *Oracle Portal*, que é a ferramenta oficial da *Oracle* para criação de sites e portais corporativos. A ferramenta OLAP será o *Oracle Discoverer*. Ela permite ao usuário desenvolver consultas da sua necessidade, gráficos, explorações e consultas na *web*, a partir de bases *DW* ou *Data Marts*.

E por fim, os processos de modelagem e carga de todos os dados armazenados pela empresa, serão feitos com o auxílio do *Oracle Warehouse Builder*, ferramenta de projeto de

---

<sup>1</sup> Segundo Barbieri(2001), dimensões são os pontos de entrada específicos de uma estrutura dimensional de dados.

banco de dados e ETL (extração, transformação e carregamento) de todas as informações armazenadas pelas empresas.

Com isso, acredita-se que o sistema possa suprir as necessidades de se obter informações gerenciais através de via única, com rapidez e de forma confiável, auxiliando a tomada de decisões gerenciais.

Nesse trabalho de conclusão I, será realizado todo o estudo teórico sobre as tecnologias que serão utilizadas no projeto, além de uma explicação sobre as ferramentas escolhidas, além das outras existentes no mercado.

O trabalho está dividido em três capítulos. O primeiro aborda o embasamento teórico das tecnologias presentes no trabalho. O segundo aborda as diversas ferramentas de DW e OLAP. E por fim, o terceiro, que apresenta uma descrição do projeto proposto para o desenvolvimento do sistema.





## 1 *BUSINESS INTELLIGENCE* (BI)

Ter em mãos informações essenciais para tomada de decisão é cada vez mais importante no mundo competitivo dos negócios. A necessidade de excelência na atuação no mercado, bem como maior agilidade no processo de adaptação, á cada vez maior. Assim, lidar com informações para a tomada de decisão é um processo fundamental para a adaptação a essas constantes mudanças.

Fortulan e Filho apud Shim et al.(2002) dizem que os Sistemas de Apoio à Decisão são soluções computacionais desenvolvidas para apoiar a tomada de decisões complexas durante a resolução de problemas. Ferramentas clássicas de SAD compreendem componentes para gerenciamento de sofisticados bancos de dados, poderosas funções de modelagem e projetos de interface com o usuário, que permitem trabalhar interativamente com questões, relatórios e funções gráficas.

Porém, o termo SAD, segundo Fortulan e Filho apud Carlsson e Turban (2002), está sendo cada vez menos usado. Isso pode ser visto em artigos, revistas e até comerciais de sistemas. Segundo eles, no seu lugar tem sido cada vez mais freqüente o uso do termo *Business Intelligence*.

Segundo Barbieri (2001), BI, pode ser entendido como a utilização de variadas fontes de informação para se definir estratégias para a competitividade nos negócios da empresa. Ou ainda, BI “representa a habilidade de se estruturar, acessar e explorar informações, normalmente guardadas em DW/DM (*Data Warehouse/Data Mart*), com o objetivo de desenvolver percepções, entendimentos, conhecimentos, os quais podem produzir um melhor processo de tomada de decisão” (BARBIERI. 2001 p. 5).

Ainda, cresce o reconhecimento de que BI está se tornando cada vez mais um componente fundamental na chamada segunda geração dos sistemas ERP, que aponta a

necessidade de dar suporte não apenas ao processamento de transações operacionais, mas também ao processamento de análises (BARBIERI, 2001).

Segundo Serra (2002), o termo BI, na verdade vem sendo utilizado desde a década de 70, quando alguns produtos de *Business Intelligence* foram fornecidos para os analistas de negócios, porém exigiam interações com os usuários exaustivas e intensas, e não apresentavam respostas em tempo hábil para a tomada de decisões, além de possuir alto custo de implantação. Mas, com o surgimento dos bancos de dados relacionais, dos PC's e das interfaces gráficas, aliados ao aumento da complexidade dos negócios, surgiram os novos produtos direcionados aos analistas de negócios.

Segundo Serra (2002), algumas das principais características de um sistema de *Business Intelligence* são:

1. Extrair e integrar dados de múltiplas fontes;
2. Fazer uso da experiência do usuário no negócio;
3. Analisar dados contextualizados;
4. Procurar relações de causa e efeito;
5. Transformar os registros obtidos em informação útil para o conhecimento empresarial.

Conforme Fortulan e Filho apud Shim et al. (2002), os *Data Warehouses*, *OLAP*, *Data Mining* surgiram no começo dos anos 90 como novas ferramentas para SAD, e formam a base dos sistemas de BI. Estas, serão abordadas a seguir nesse capítulo.

### **1.1 Dados Operacionais x Dados Informativos**

Existem dois tipos de dados nas empresas: um é conhecido como dado operacional e o outro como dado informativo. Segundo Singn (2001), são considerados sistemas de dados operacionais todos os aplicativos que suportam de forma direta as funções críticas de negócio da empresa, e podem ser chamados também de sistemas de OLTP (*On-Line Transaction Processing*). Conforme Costa e Anciães (2001), os dados operacionais são parte da infraestrutura corporativa: são detalhados, atualizáveis e não-redundantes.

Já os dados informativos suportam o processo de tomada de decisões e possuem algumas diferenças em relação ao dado operacional. Enquanto o acesso operacional significa o acesso atual de instâncias específicas de dados, o acesso informativo implica em acessar grandes volumes de dados para análises elaboradas, a fim de planejar e tomar decisões. Esses tipos de dados estão relacionados à tecnologia OLAP, que será explicado nos próximos capítulos.

Os dois tipos de dados constituem fontes importantes para o estabelecimento dos conceitos do BI. (BARBIERI, 2001)

Quadro 1.1 – Comparação entre os dados de natureza operacional e informacional

<i>Características</i>	<i>Dados Operacionais</i>	<i>Dados Informativos</i>
Conteúdo	Valores correntes	Valores Sumariados, calculados, integrados de várias fontes
Organização dos dados	Por aplicação/sistema de informação	Por assuntos/negócios
Natureza dos dados	Dinâmica	Estática até a carga dos dados
Formato das Estruturas	Relacional, próprio para computação transacional	Dimensional, simplificado, próprio para atividades analíticas
Atualização dos dados	Atualização campo a campo	Acesso, sem atualização
Uso	Altamente estruturado, processamento repetitivo	Desestruturado, com processamento analítico/heurístico
Tempo de resposta	Otimizado para poucos segundos	Análises mais complexas, com tempos de respostas maiores

Fonte: BARBIERI, 2001

## 1.2 Data Warehouse

Segundo Inmon (1997), um *Data Warehouse* (DW) é um conjunto de dados baseado em assuntos, integrado, não volátil e variável em relação ao tempo, de apoio às decisões gerenciais. É um conceito que não é novo, pois foi originalmente usado como proposta de solução da IBM, sendo chamado de “*information warehouse*”.

O *Data Warehouse* pode ser considerado, de forma clara e objetiva, a separação física entre os chamados sistemas de dados operacionais e os sistemas de suporte à decisão em uma organização (SINGN, 2001).

### 1.2.1 Histórico

“Na década de 70, os dados eram meros coadjuvantes de um processo de desenvolvimento de sistemas, onde o empirismo metodológico de muitos e a inspiração de alguns definiam os caminhos a trilhar.” (BARBIERI, 2001).

O princípio da gerência de dados era montada através de um modelo hierárquico, como nas empresas, dividida em níveis, que de certa forma permitiam que sistemas com um alto grau de complexidade pudessem ser implementados, porém com um foco muito mais tecnológico do que funcional.

Nessa década, o conceito de banco de dados estava surgindo e começava a promover visão de uma organização “baseada em dados”, em que o computador poderia atuar como coordenador central para atividades de toda a empresa. Nesta visão, o banco de dados tornou-se um recurso corporativo básico. A partir disso, as pessoas começaram a ver os computadores como uma verdadeira vantagem competitiva (DWBRASIL, 2003).

Com a evolução dos sistemas de informação, os anos 80 foram marcados com o surgimento da administração de dados, da modelagem dos mesmos, da engenharia de informação e da análise de dados. Com isso, surgiu o modelo relacional, que quebrava o paradigma da rigidez de estruturas hierárquicas, trazendo a flexibilidade das relações.

Nos anos 90 houve a grande união entre a informação e a comunicação, capitaneada pelo surgimento em grande escala da Internet. Juntamente com ela, os computadores pessoais começaram a proliferar tanto nos lares quanto nas organizações. Nas empresas, o conceito de *main-frame*<sup>1</sup> foi perdendo espaço e a nova arquitetura, conhecida como cliente-servidor<sup>2</sup>, promoveu a diminuição nos custos em aquisição de hardware, disponibilizou maior escalabilidade aos sistemas e flexibilidade aos usuários finais.

---

1 *Mainframe* é um computador de grande porte, dedicado normalmente ao processamento de um volume grande de informações (WIKIPEDIA, 2007).

2 Na arquitetura aplicação cliente-servidor o cliente liga-se a um servidor de aplicação ou sistema de base de dados. (WIKIPEDIA, 2007).

Com os computadores pessoais em mãos, novas aplicações e com bancos de dados pessoais e planilhas eletrônicas eram desenvolvidas pelos usuários. Assim, retiravam as informações do banco de dados central para analisar em suas estações de trabalho.

Com isso, os usuários poderiam analisar seus dados de forma em que pudessem visualizá-los ou manipulá-los da maneira que lhe fossem mais conveniente. Todavia, questões como controle, capacidade e integridade dos dados começaram a ser problemas com dificuldades de administração. Além disso, com o passar do tempo, as empresas foram crescendo, e com isso, a quantidade de dados também.

As necessidade da análise de dados passou a aumentar, tornando providencial as respostas mais rápidas, confiáveis e que melhor se adaptassem às necessidades de gerenciamento de negócios da empresa.

Esses problemas só foram trazendo a necessidade de construir mecanismos para controlar essa manipulação e visualização de dados em um ambiente distribuído.

Segundo Singn (2001) , os executivos passaram a expressar a necessidade de dados consistentes em suas empresas da seguinte forma: “Todos os nossos relatórios apresentam pequenas diferenças irritantes. Nós estamos gastando a primeira parte de cada reunião discutindo qual seria a informação correta. A proliferação de planilhas está contribuindo para o problema da inconsistência e imprecisão dos dados. Para que os nossos dados possam ser úteis, precisamos que estes sejam confiáveis”. Este problema de inconsistência de dados e a necessidade dos usuários de tomada de decisão foram o fator chave para o surgimento do conceito de *data warehouse*.

Segundo Harrison (1998), as empresas estão investindo milhões em *data warehouse*, pois nos últimos anos os DW's adquiriram muita força em função do sucesso dos Sistemas de Gerenciamento de Banco de Dados Relacionais (SGBDR) e também devido à redução dos custos e aumento do desempenho dos hardwares utilizados.

### **1.2.2 Características do DW**

Conforme Inmon (1997) um *Data Warehouse* deve apresentar as seguintes propriedades:

1. Orientado a assunto;
2. Integrado;
3. Não volátil;
4. Variável com o tempo.

As seções a seguir descrevem cada uma destas propriedades.

#### **1.2.2.1 Orientado a assuntos**

Conforme Harrison (1998) apud Inmon (1997), a questão do DW ser orientado em assuntos refere-se ao fato de que o mesmo está organizado de maneira a descrever o desempenho dos negócios. Enquanto isso, os bancos de dados operacionais são orientados para os processos de negócios.

Um exemplo dessa propriedade pode ser vista nesse projeto, onde se quer construir um DW focado em vendas. Questões como “quem foi o maior vendedor do ano”, ou “qual cliente apresenta mais problemas de devolução” podem ser respondidas nessa base voltada unicamente a vendas.

#### **1.2.2.2 Integrado**

Muitas vezes, para a criação do *Data Warehouse*, é necessário armazenar dados de diversas fontes, como planilhas, arquivos de texto, bancos de dados, entre outros. Isso deve ficar em um formato consistente, mas para isso surge a necessidade de que os dados sejam tratados antes de serem carregados para a estrutura definitiva.

Nesse processo, surgem algumas tarefas importantes como resolver conflitos de nomes, converter dados para um tipo de medida única ou ainda padronizar formatos de datas. A figura 1.1 mostra um exemplo de tratamento de dados:

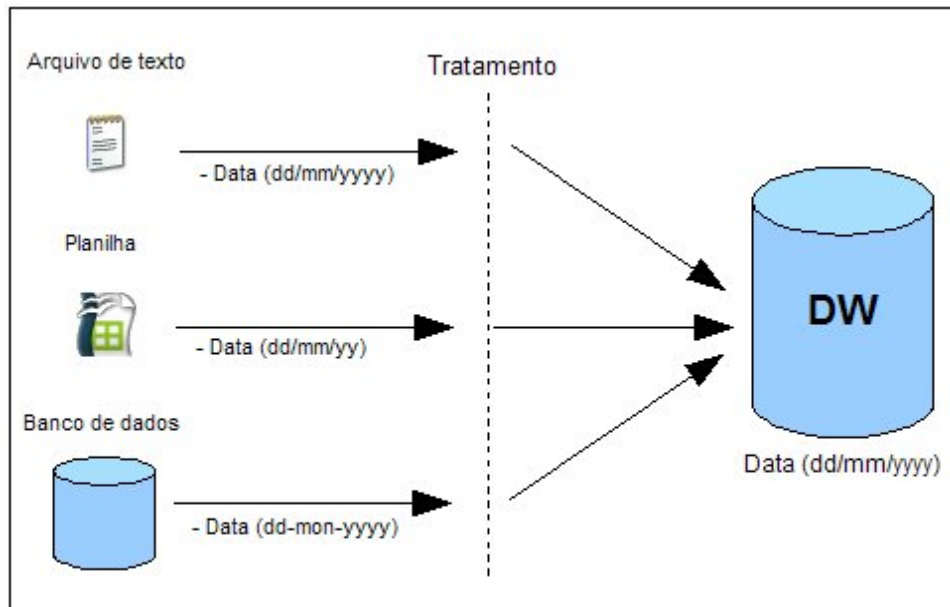


Figura 1.1 - Processo de tratamento de Dados  
Adaptada de Inmon (1997)

Conforme Inmon (1997), não adianta de nada carregar os dados operacionais para o DW sem fazer a integração. Caso os mesmos cheguem ao *data warehouse* de forma não-integrada, não será possível a sua utilização como base para uma visão corporativa.

### 1.2.2.3 Não-Volátil

A questão do DW ser não-volátil implica na questão de que nesse tipo de sistema só são permitidos dois tipos de operações: a carga de dados operacionais para a base e as consultas, ou seja, sem alteração de dados. Isso explica uma das diferenças com relação aos ambientes operacionais, no qual é permitido incluir, excluir, alterar e consultar dados. A Figura 1.2 ilustra essa diferença:

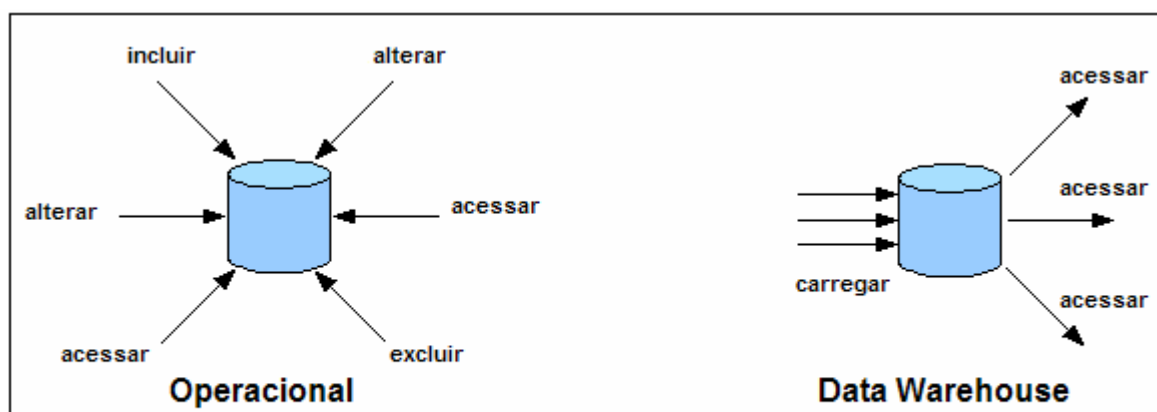


Figura 1.2 - A questão da não volatilidade  
Adaptada de Inmon (1997)

#### 1.2.2.4 Variável com o tempo

Em ambientes operacionais, os dados podem ser visualizados somente de forma atual, isentando a questão do histórico, ou seja, não sabendo como estavam os mesmos antes da última atualização. No *Data Warehouse*, essa possibilidade muda, pois nesse ambiente os dados históricos são precisos, ou seja, pode-se saber como estava determinado dado em algum determinado momento.

Segundo Inmon (1997), o dado quando carregado no DW recebe, na sua chave, uma unidade de tempo e nunca mais é atualizado. Este tipo de armazenamento permite que os analistas de negócios façam análises de tendências, podendo visualizar as variações de uma determinada informação ao longo do tempo.

#### 1.2.3 Componentes do DW

Segundo Singn (2001), um *data warehouse* sempre apresenta os seguintes componentes:

**Dados atuais:** refletem os acontecimentos mais recentes, que acabam sempre de grande interesse para a organização;

**Dados antigos:** são acessados com menor frequência e armazenados em nível de detalhe consistente com o detalhe dos dados atuais. Ocupam grande volume de espaço físico;

**Dados sumariados:** a maior parte das consultas executadas no *Data Warehouse* são feitas em níveis altos de sumariação. São divididos em altamente sumariados, que são compactados e de fácil acesso; e ligeiramente sumarizados que possuem um nível maior de detalhe;

**Metadados:** conforme Harrison (1998), a definição de metadados é “dados a respeito de dados”. Os mesmos são absolutamente críticos por descreverem os dados contidos dentro do DW e OLAP. Exemplos de metadados seriam: quando os dados foram atualizados pela última vez, ou qual o esquema de banco de dados, ou ainda, quais são as regras de agregação de dados.



### 1.2.4 Modelagem de Dados

Conforme Harrison (1998), o modelo entidade-relacionamento (ER)<sup>3</sup> produz um projeto otimizado para acessar dados registro por registro, no nível mais básico, que é o nível de transação ou ocorrências. Para esse tipo de sistema, são realizadas otimizações para uma eficiente criação, atualização e exclusão de registros específicos. E esses processos são realizados através da normalização.

De acordo com Bispo (1998), no modelo entidade-relacionamento, os dados são divididos em diversas tabelas, que se relacionam entre si, formando um diagrama. Seu formato muitas vezes dificulta a interpretação e análise, mas é importante para a eficiência e o desempenho na ambiente operacional, no qual não são necessárias consultas que utilizem recursos excessivos para sua realização.

Para consultas maiores, ou seja, que necessitem uma análise de universo maior de dados, o modelo dimensional surge como melhor alternativa, principalmente pelo fato de economizar nas junções de diversas tabelas, e armazenando dados que facilitem a análise das informações.

O modelo dimensional surgiu para atender sistemas de processamento analítico, com consultas para planejamento tático e estratégico da empresa. Atualmente a utilização desses sistemas pelo nível operacional das empresas vem crescendo, auxiliando o processo de tomada de decisões diárias. Normalmente, ele atende um pequeno número de usuários que realizam consultas planejadas (relatórios pré-definidos) e *ad-hoc* (HOKAMA et al., 2004).

A modelagem dimensional permite modelar logicamente dados para melhorar o desempenho de consultas de grande porte. Nesse tipo de modelo, o tempo de resposta é maior, mas, para melhor desempenho nas consultas, há redundância planejada dos dados, compensando os gastos com armazenamento e atualização das informações. O resultado é uma estrutura simples, com modelos que refletem o processo de análise de negócios.

Essa estrutura é composta basicamente pelas tabelas de fatos e tabelas de dimensões. A tabela de fatos traz o resultado da consulta, ou seja, valores de medição. As restrições, objeções e questionamentos ficam nas tabelas de dimensões, que trazem informações textuais sobre o valor medido na tabela de fatos. (BARBALHO, 2003).

---

3 Proposta por Peter P. Chen em 1976, A modelagem Entidade-Relacionamento (ER) envolve identificando, os elementos de importância na organização (entidades), suas propriedades (atributos) e como eles estão relacionados uns aos outros (relacionamentos). O modelo resultante da informação é independente de qualquer armazenamento de dados ou método de acesso (LOBO, 1998).

Os modelos dimensionais são compreensíveis, previsíveis, ampliáveis e resistentes ao grande uso de grupos de usuários de negócio, por se manter fiel à simplicidade, ter uma perspectiva voltada para as necessidades analíticas da empresa, e especialmente ao seu formato simétrico, em que todas as dimensões normalmente são iguais pontos de entrada na tabela de fatos (KIMBALL, 2002). Os modelos dimensionais são a base de muitos aprimoramentos de desempenho SGBD, inclusive agregações e métodos de indexação avançados.

“A modelagem de dados é seguramente um dos fatores críticos de sucesso num projeto de *Data Warehouse*, e pode representar a fronteira entre o sucesso e o seu fracasso” (BARBIERI, 2001, p.74) .

#### **1.2.4.1 Tabelas de fatos**

A tabela de fatos é a principal tabela de um modelo dimensional, onde as medições numéricas de interesse da empresa estão armazenadas (KIMBALL, 2002). A palavra "fato" representa uma medida dos processos que são modelados, como quantidades, valores e indicadores. Esse tipo de tabela registra os fatos que serão analisados e é composta por uma chave primária (formada por uma combinação única de valores de chaves de dimensão) e pelas métricas de interesse para o negócio.

De acordo com Hokama et al. (2004), a tabela de fatos é sempre esparsa, ou seja, possui um número relativamente pequeno de todas as combinações possíveis de valores de chaves. No contexto desse projeto, poderia ser dado como exemplo o fato de que um produto pode ser vendido por todos os representantes, comprado por todos os clientes, podendo ser faturado em qualquer data. Por isso pode-se concluir que é algo extremamente esparsa, pois uma porcentagem muito pequena de todas as combinações possíveis de representantes, clientes, produtos e datas de faturamento aparecerá na base.

Segundo Bispo (1998), uma tabela de fato contém vários fatos, correspondentes a cada uma de suas linhas, sendo que cada fato pode armazenar uma ou mais medidas numéricas, que constituirão os valores da análise dimensional. Esse tipo de tabela normalmente armazena muito mais linhas que as tabelas de dimensões, e devem receber atenção pois podem ter um volume muito grande.

Conforme Kimball (1998), não se deve preencher uma linha da tabela fato com zeros para representar que nada aconteceu (por exemplo, que não houve faturamento para um cliente em data específica), pois isso faria com que a tabela de fatos crescesse demais.

Na figura 1.3, é ilustrada a composição de uma tabela de fatos, onde cada linha está representando um fato e as colunas chaves são herdadas das tabelas de dimensões. A tabela contém ainda os valores das medidas para o modelo em questão.

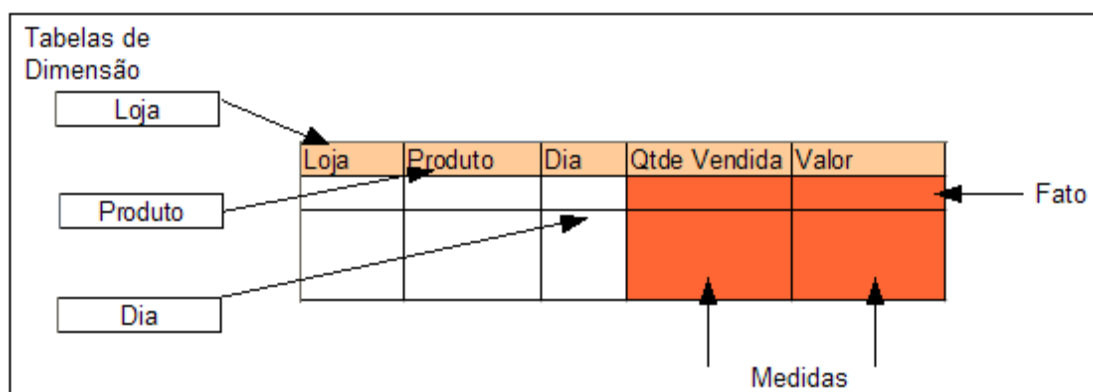


Figura 1.3 - Composição básica de uma tabela de fatos

Fonte: Barbieri, 2001

#### 1.2.4.1.1 Atributos Aditivos, Semi-aditivos e não aditivos

Conforme já mostrado, as tabelas de fatos armazenam as medições métricas do negócio. Essas métricas, segundo Barbieri (2001), estão definidas em três tipos: aditivas, semi-aditivas e não aditivas. A seguir uma descrição de cada uma delas:

**Aditivas:** Quando os valores são passíveis de serem somados em todas as dimensões;

**Semi-Aditivas:** Quando a soma ou qualquer outro tratamento estatístico tiver sentido apenas em uma dimensão;

**Não-Aditivas:** Quando o valor não puder ser somado em nenhuma dimensão, ou produzir qualquer valor sem sentido válido.

#### 1.2.4.2 Tabelas de Dimensões

As tabelas de dimensão são aquelas que armazenam as descrições textuais do negócio, onde cada uma dessas descrições textuais ajuda a definir um componente da

respectiva dimensão. Exemplo disso seria que cada registro da dimensão cliente refere-se a um cliente específico (KIMBALL, 1998).

Segundo Barbieri (2001), as tabelas de dimensão têm uma relação de 1:N com a tabela de fatos, logo, possuem um número de registros bem menor. Possuem inúmeras colunas de informação e uma chave primária, que acaba participando das tabelas de fatos, como parte de sua chave múltipla.

É muito importante que os atributos das tabelas de dimensão sejam preenchidos com valores descritivos ao invés de códigos sem sentido, criptografados ou abreviados (KIMBALL, 2002). Por exemplo, em uma tabela de dimensão Alimentos, o campo “perceível” deve ser preenchido com valores como “É perceível” ou “Não é perceível” ao invés de usar simplesmente “S” e “N”. Em um relatório com a listagem de milhares de produtos, um valor descritivo tem muito mais utilidade do que códigos. Ao invés de usar uma aplicação para decodificar esses códigos e mostrar uma descrição, é melhor armazenar essas descrições no banco de dados, tornando a informação disponível ao usuário independentemente de seu aplicativo de acesso aos dados (HOKAMA et al., 2004).

A figura a seguir ilustra um exemplo de uma tabela de dimensão:

Cliente
Cod_cliente
Razao_social
PF_PJ
Endereco
Cod_cidade
Cond_Pgto
Telefone
Fax
Email

Figura 1.4 - Exemplo de uma tabela de dimensão

### 1.2.4.3 Técnicas de Modelagem

Harrison (1998) considera importante que as empresas iniciem seus projetos *Data Warehouse* com a escolha de um modelo lógico que ofereça maior desempenho possível e versatilidade funcional antes de ponderar os benefícios de uma maior eficiência no armazenamento dos dados. Dentre os cinco projetos citados a seguir, o esquema em estrela (*star schema*), é o mais indicado para iniciar devido à simplicidade do projeto de banco de dados e facilidade de compreensão, inclusive por usuários não técnicos.

Segundo Harrison (1998), os cinco modelos de projeto de *Data Warehouse* são:

1. Esquema em estrela;
5. Esquema em estrela parcial;
6. Esquema de *fact table* (tabela de fatos) particionada;
7. Esquema de tabela dimensional particionada;
8. Esquema *snowflake*.

Será explicado a seguir como funciona o *star schema* e o *snowflake*, por serem as técnicas mais conhecidas e mais utilizadas.

#### **1.2.4.3.1 Star Schema**

Poe et al. (1998) explica que o esquema estrela é uma estrutura simples, com poucas tabelas e relacionamentos bem definidos e assemelha-se ao modelo de negócio, o que facilita a leitura e entendimento, não só pelos analistas, como por usuários finais não familiarizados com estruturas de banco de dados. Permite a criação de um banco de dados que facilita a execução de consultas complexas, podendo ser realizadas de modo eficiente e intuitivo pelo usuário.

Segundo Harrison (1998), a estrela possui quatro propriedades que a diferenciam dos demais modelos de DW.

1. Dentro de cada categoria, existe uma única tabela de fatos histórica simples, contendo detalhes e dados a nível de sumário, armazenados nos níveis de estrutura indicado em cada tabela de dimensão;
2. A chave primária da tabela de fatos contém somente uma coluna chave de cada dimensão;
3. Cada chave é uma chave gerada pelo sistema;
4. Cada dimensão é representada por única tabela de fatos, usando também uma chave gerada pelo sistema.

A seguir, uma ilustração de um modelo em estrela:

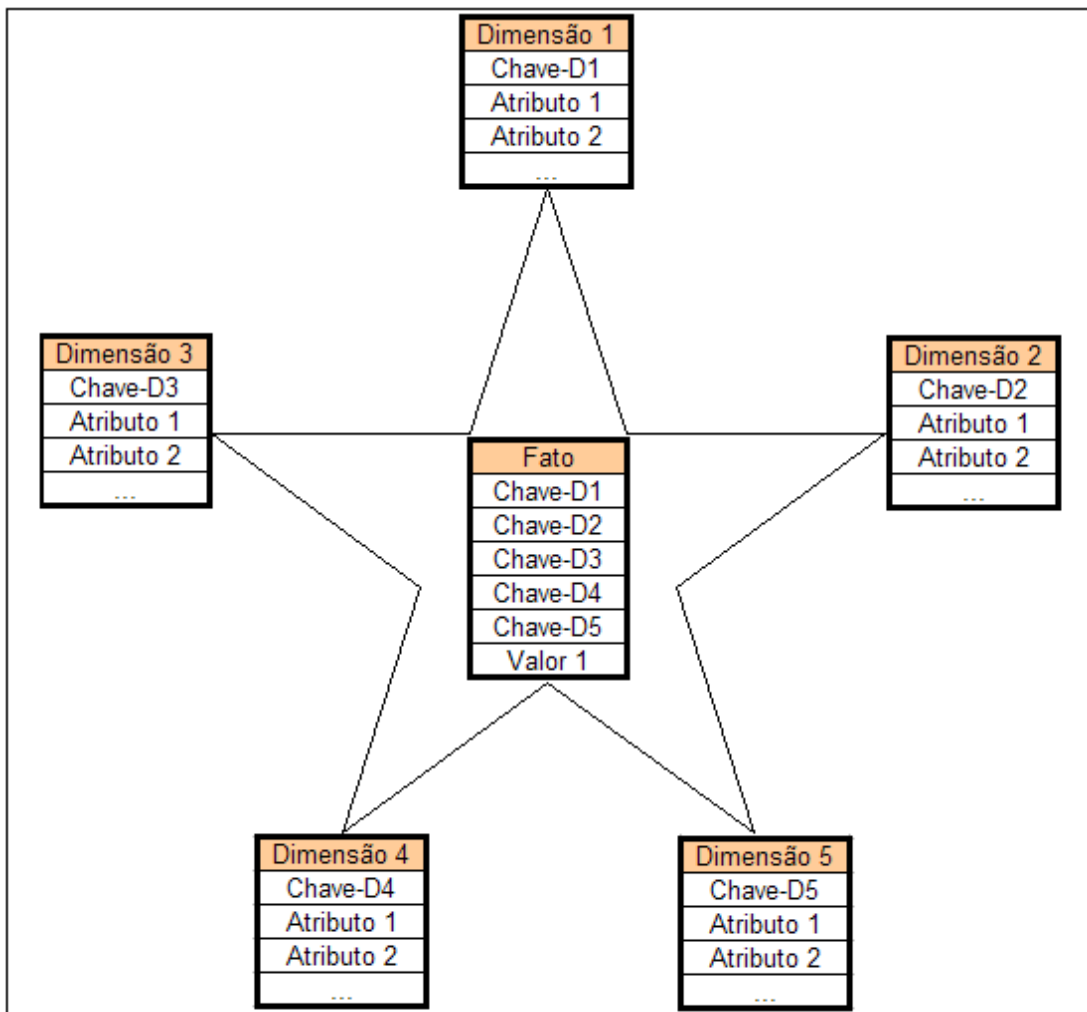


Figura 1.5 - Exemplo de um esquema em estrela  
 Fonte: adaptada de HOKAMA et al (2004)

Como pode ser observado, o termo estrela está ligado à disposição das tabelas no modelo, que exibe uma tabela central, a de fatos, e diversas ligadas a ela, sendo as tabelas de dimensões.

Esse tipo de modelo oferece várias vantagens, incluindo um desempenho maior através do uso de chaves geradas pelo sistema, que reduzem o tamanho do índice. Segundo Harrison (1998), o uso de uma tabela única por dimensão e de uma tabela de fatos simples por categoria assegura que as definições dos metadados possam ser usadas novamente, independentemente do nível de sumário ou fatos.

### 1.2.4.3.2 Snowflake

Segundo Machado (2000), o esquema *snowflake* (flocos de neve) é uma variação do *star schema*, no qual todas as tabelas de dimensão são normalizadas na terceira forma normal, ou seja, são retirados das tabelas os campos que são funcionalmente dependentes de outros campos que não são chaves.

Na figura 1.6, pode-se visualizar que em *snowflake*, as tabelas de dimensão têm uma conexão lógica com as tabelas de fatos, através das chaves primárias. Há também tabelas menores, chamadas de extensões, que são usadas para armazenar descrições e decodificações para chaves e códigos nas tabelas maiores.

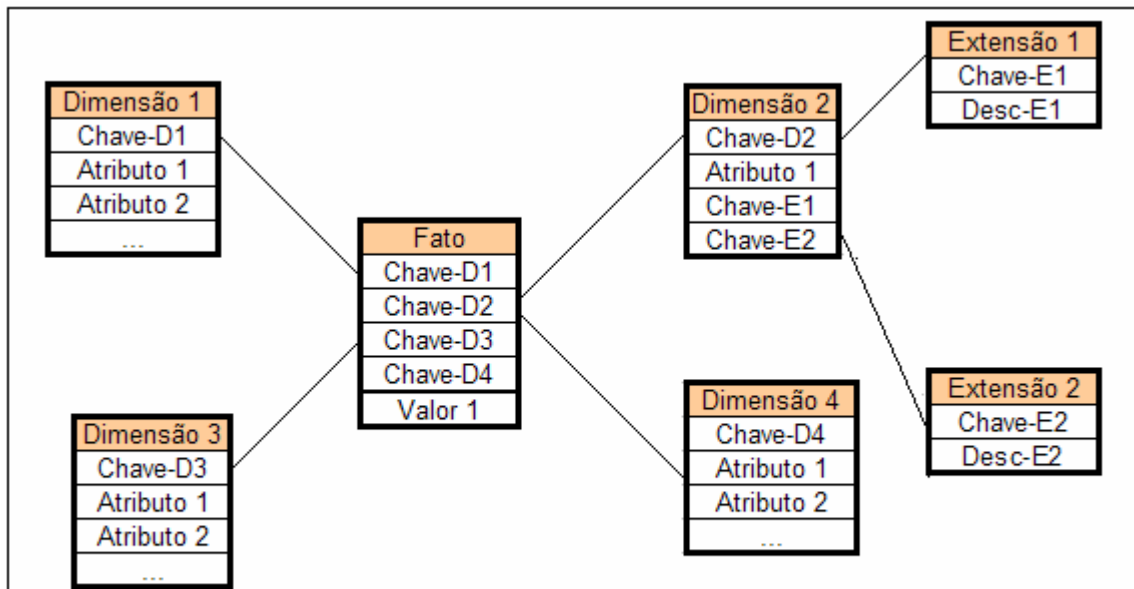


Figura 1.6 - Exemplo de um esquema *snowflake*

Fonte: adaptada de HOKAMA et al (2004)

Segundo Singn (2001), o uso do esquema *snowflake* traz como desvantagens o aumento da complexidade da estrutura de dados, dificultando a compreensão do modelo por parte de usuários que trabalham diretamente com a estrutura física das tabelas. E essa complexidade pode trazer ainda, uma diminuição de performance nos processos que envolvem esse tipo de esquema

Porém, o uso do *snowflake* pode ser indispensável em alguns casos em que, por exemplo, o uso de esquemas em estrela necessite espaço em disco muito maior ou suas tabelas dimensionais sejam muito grandes, onde assim causariam problemas de performance no sistema.

Kimball (1996) sugere aos projetistas "bem-intencionados" a resistirem a idéia de transformar *star schemas* em *snowflake*, em função da complexidade deste tipo de estrutura sobre o usuário final, enquanto que o ganho em termos de espaço de armazenamento seria pouco relevante.

#### 1.2.4.4 Cubos de Dados

Segundo Kimball (1998), através dos cubos de dados, quase todos os tipos de dados de negócio podem ser representados. Nessas estruturas, as células do cubo representam os valores medidos enquanto os lados representam as dimensões dos dados. Quando um cubo apresenta mais de três dimensões, são chamados de hipercubos.

Cubo é a estrutura multidimensional de dados que expressa a forma na qual os tipos de informações se relacionam entre si. É formado pela tabela de fatos e pelas tabelas de dimensão que a circundam e representam possíveis formas de visualizar e consultar os dados. O cubo armazena todas as informações relacionadas a um determinado assunto, de maneira a permitir que sejam montadas várias combinações entre elas, resultando na extração de várias visões sobre o mesmo tema.(HOKAMA et al. 2004, p. 49).

Conforme Gray e Watson (1999), os cubos são os principais objetos de um OLAP. São desenvolvidos com tecnologia que permite rápido acesso aos dados, sendo que normalmente são compostos por sub-conjuntos de um *Data Warehouse* e organizados e sumariados dentro de estruturas multidimensionais definidas por dimensões e medidas. Essas estruturas podem resultar em diversas matrizes esparsas que permitem trabalhar simultaneamente com diversos cenários definidos por combinações de dados, como produtos, clientes, períodos, etc. Esses cubos podem ser armazenados em modelos de bancos de dados ROLAP (OLAP Relacional), MOLAP (OLAP Multidimensional) ou HOLAP (OLAP Híbrido), que serão explicados nos próximos capítulos.

Na figura 1.7, pode-se visualizar a composição de um cubo, formado pelas células que compõe as medições (valores faturados) e as laterais representando dimensões de período, clientes e produtos.



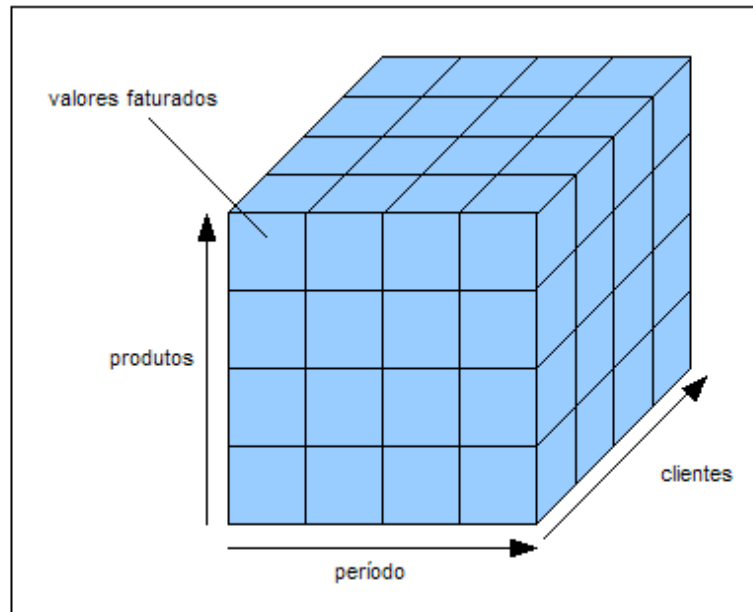


Figura 1.7 - Cubo de dados

Uma das técnicas que faz análise desse tipo de estrutura é o *slice-dice* (INMON et al., 1999), que significa cortar o cubo de dados em fatias permitindo rotacionar os lados do mesmo em qualquer sentido, possibilitando a combinação de quaisquer dimensões e a obtenção de informações correspondentes sobre vários enfoques. Essa técnica, dentre outras, será explicada com mais detalhes no capítulo relacionado a OLAP.

#### 1.2.4.5 Granularidade

Segundo Inmon (1996), a granularidade refere-se ao nível de detalhe ou sumariação contida nas unidades de um DW. Quanto maior o detalhamento, menor a granularidade, quanto menor o detalhamento, maior a granularidade. O exemplo na figura 1.8 mostra isso. Caso seja visualizado o total de vendas dentro do mês, haverá um baixo nível de detalhes, mas uma granularidade alta (Figura 1.8[a]). Se esses valores forem divididos em dias, o nível de detalhe aumentará, porém a granularidade não (Figura 1.8[b]).

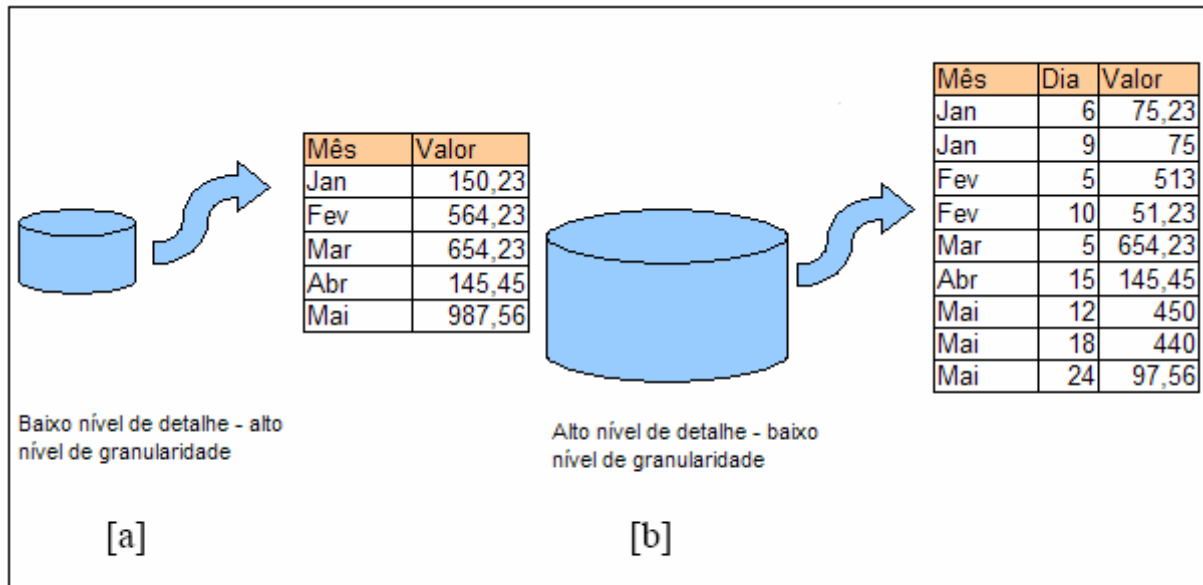


Figura 1.8 - Níveis de granularidade

Fonte: adaptada de Inmon (1996)

A mais importante questão de projeto que o desenvolvedor de *data warehouse* precisa enfrentar diz respeito à definição da granularidade do *data warehouse*. Quando a granularidade de um *data warehouse* é apropriadamente estabelecida, os demais aspectos de projeto e implementação fluem tranqüilamente; quando ela não é estabelecida, todos os outros aspectos se complicam (INMON, 1997, p.143).

Os níveis de granularidade dependem muito do total de linhas a serem carregados para o *Data Warehouse*. Por exemplo, se para um horizonte de um ano, o total de linhas for em torno de 10.000, qualquer técnica funcionará. Caso for maior, supondo-se 1.000.000 de linhas, seria necessário utilizar mais níveis de granularidade, os chamados “níveis duais de granularidade”. Esses níveis permitem que se processe eficientemente a enorme quantidade de solicitações e atenda a qualquer questão que possa ser respondida. Essa é a melhor de todas as situações e deveria ser a opção de projeto padrão (INMON, 1997).

Inmon (1997) exemplifica esse tipo de método com um sistema bancário. Em um sistema desses, pode-se armazenar, por exemplo, os lançamentos individuais em contas correntes nos últimos 60 dias, e armazenar o histórico resumido desses lançamentos nos últimos 5 anos, com o valor total de lançamentos sumariados por mês. Tanto os dados resumidos, quanto os detalhados estarão disponíveis para o usuário.

### 1.2.3.5.1 Agregados

Dentro das aplicações transacionais, o número de registros a serem recuperados normalmente é reduzido. Ao contrário de ambientes DW, onde as consultas são bastante

intensivas (*data-intensive*). As consultas em sistemas de apoio a decisão, segundo Meredith e Khader(1996), podem ser divididas em *data-intensive*, que acessam um número grande de linhas e colunas, não importando se são recuperadas através de índices ou buscas por toda a tabela, e *data-selective*, que mesmo com poucas colunas tem critérios de seleção complexos.

Índices podem auxiliar nas consultas *data-selective*, mas para melhorar o desempenho de consultas do tipo *data-intensive* são necessárias outras técnicas. Dentre elas o particionamento, que reduz o número de dados a serem varridos pelo sistema e a agregação, que pré-calcula as medidas necessárias para que o sistema acesse menos registros.

Conforme Kimball (1998), os agregados são sumários armazenados, que devem existir juntamente com os registros base do sistema, criados para aumentar o desempenho das consultas. Esses ganhos de performance podem chegar muitas vezes até a um fator de 100 a 1000 vezes maior.

“O agregado é um registro de tabela de fatos que representa o resumo de nível básico da tabela de fatos. Um registro da tabela de fatos agregado está sempre associado a um ou mais registros de tabela de dimensões agregadas” (KIMBALL, 1998, p. 191).

Um dos problemas levantados nos agregados é a questão de que essa solução agride de certa forma os processos tradicionais de ausência de redundância, estabelecidos nos preceitos dos projetos de banco de dados. Além de ocuparem mais espaço, pela razão de exigirem uma coleção de tabelas de fatos ou de dimensões, que passam a serem dedicadas ao armazenamento de dados pré-processados (BARBIERI, 2001).

A figura 1.9 mostra como podem ser utilizados agregados através de tabelas de fatos. A tabela FatoVenda tem duas agregações, a “agregada1”, onde os dados estão sumariados por representante e a “agregada2”, onde estão sumariados por cliente.

<b>FatoVenda</b>	<b>Agregada1</b>	<b>Agregada2</b>
Cod_data	Cod_data	Cod_data
Cod_Cliente	Cod_Representante	Cod_Cliente
Cod_Representante	Cod_Produto	Cod_Produto
Cod_Produto	Valor1	Valor1
Valor1	Valor2	Valor2
Valor2	Valor3	Valor3
Valor3		

Figura 1.9 - Tabelas agregadas  
Fonte: adaptada de Hokama et al (2004)

### 1.2.5 *Data Mart* (DM)

Segundo Barbieri (2001), *Data Mart* (DM) é um depósito de dados que atende a áreas específicas da empresa e objetiva auxiliar o processo decisório gerencial. Sobre DW e DM, Barbieri (2001, p.50) cita: “ambos podem ser definidos como espécies do mesmo tipo, ficando a diferença entre os dois centrada no escopo do projeto e nos limites de suas abrangências”.

Na definição de Serra (2002), *Data Mart* é um pequeno *Data Warehouse* que fornece suporte à decisão para um pequeno grupo de pessoas. Pelo fato de ser mais focado em uma área específica, o *Data Mart* é atrativo por possuir baixo custo e tempo menor de implementação, com crescentes avanços tecnológicos. Segundo Serra (2002, p. 136), “os *Data Marts* podem servir como veículo de teste para empresas que desejam explorar os benefícios do *Data Warehouse*”.

Apesar das vantagens de custo e desempenho, como os DM's diferem de departamento para departamento, o desenvolvimento de DM's independentes, sem um planejamento global acarreta a fragmentação de dados de uma organização e inibe a utilização de informações de forma integrada na corporação, podendo fazer surgir e proliferar à falta de integração e compartilhamento de dados entre os sistemas transacionais (FORTULAN e FILHO apud GRAY E WATSON, 1998).

O quadro 1.2, aponta as diferenças básicas entre *Data Warehouse* e *Data Marts*:

Quadro 1.2 – Diferenças entre DW e DM.

<i>Data Warehouse</i>	<i>Data Mart</i>
- Utilização altamente imprevisível, aplicações não estruturadas, analíticas;	- Tipo de <i>Data Warehouse</i> em que os dados estão mais próximos aos usuários;
- Tempo de resposta pode ser de segundos ou minutos;	- Menores e mais fáceis de serem gerenciados;
- Dados relacionais;	- Tomada de decisão pode ser em nível departamental;
- Informações organizadas por área de análise, normalmente históricas;	- Dados relacionais ou multidimensionais.
- Usuários finais: gerência, consumidores de informação.	

Fonte: Adaptado de Serra(2002)

De acordo com Poe et al. (1998), as organizações podem escolher esse tipo de arquitetura, os *data marts*, para o início do desenvolvimento de um projeto piloto, limitado a

uma área de negócios específica, provendo uma oportunidade de aprendizagem e futura integração em um projeto global único que seria um DW corporativo. Outras características marcantes dos DM's são a rapidez na implementação, o baixo custo, o controle local em vez de centralizado e a redução do tempo de resposta a consultas, tornando a questão do custo-benefício muito favorável, contrastando com o longo trabalho de modelagem, tempo de desenvolvimento e recursos financeiros exigidos pelo DW.

Fortulan e Filho apud Gray e Watson (1998) reforçam essas idéias ao afirmar que os altos custos de implementação de um DW limitam o seu uso por empresas de grande porte, as quais muitas vezes não estão dispostas a correr riscos no investimento em algo que não se tem certeza do sucesso e, conseqüentemente, o retorno do investimento, tornando os *data marts*, uma alternativa reduzida e de baixo custo.

### **1.3 OLAP**

A sigla OLAP (*On-Line Analytic Processing*) significa Processamento Analítico *On-Line*, sendo o inverso do conceito de OLTP (Processamento de Transações *On-Line*). Segundo Kimball (1998), OLAP é um termo inventado para descrever uma abordagem dimensional para suporte à decisão. Porém, a filosofia OLAP, contudo, ainda necessita de critérios mais específicos para ser aceita como padrão de comparação para sistemas de suporte à decisão.

Harrison (1998) explica que OLAP é um rótulo, antes mesmo de ser considerada uma tecnologia, aplicando-se a todas as funcionalidades analíticas requeridas para a criação de informações que sejam úteis a partir de dados de DW. OLAP permite exercer as mais diversas funcionalidades de análise dos dados através das dimensões do *Data Warehouse*.

#### **1.3.1 Características**

O OLAP, segundo Santos (2000), é um paradigma muito poderoso para análise estratégica de dados de bancos de dados de grandes organizações. Dentre as principais características deste tipo de análise, as que podem ser consideradas chave são: grandes volumes de dados; apoio explícito para a dimensão temporal; apoio para vários tipos de agregação; análise de longo alcance, no qual tendências de dimensões globais são mais importantes que detalhes de atributos de dados individuais.

Segundo Harrison (1998), o OLAP possui capacidades computacionais que podem ser divididas entre as quatro a seguir:

1. Gerar consultas e relatórios, proporcionando formatações até mesmo a cargo do próprio usuário final;
2. Análise multidimensional, permitindo acessar qualquer dimensão do DW;
3. Análise estatística, possibilitando calcular médias e outras formas medidas mais sofisticadas como regressão, correlação, fatoração e agrupamentos;
4. *Data Mining*, fazendo identificação de modelos e relações e algoritmos de aprendizado para trabalhar com previsões.

#### **1.3.1.1 Funções Básicas**

De acordo com Harrison (1998) os aplicativos ou ferramentas OLAP em geral executam cinco funções básicas:

1. Interface – as telas e métodos usados para direcionar instruções internas a outras funções baseadas nas seleções dos usuários;
2. Consulta - a lógica do aplicativo usada para gerar o código SQL;
3. Processo – a lógica do aplicativo que executa a análise de dados no conjunto de resultados retornado pela consulta ao banco de dados;
4. Formato – a lógica do aplicativo requerida para rotular propriamente linhas e colunas de dados e criar um arquivo padrão;
5. Exibição – apresentação do arquivo formatado, como relatório ou gráfico, para visualização pelo usuário.

#### **1.3.1.2 Operações OLAP**

Como já explicado, o OLAP permite ao usuário navegar nas mais diversas dimensões do *data warehouse*. Isso possibilita a solicitação às mais variadas consultas, de acordo com as sumariações desejadas, ficando a cargo deste usuário a definição de cada uma das visões que melhor se adequar à resposta esperada.

Kimball (1996), Inmon (1997, 1999), Lopes (2006) trazem como algumas das principais operações a serem realizadas através da tecnologia OLAP:

**Pivoting:** Serve para adicionar, remover ou rearranjar as dimensões das tabelas, utilizando simplesmente o mouse, arrastando e soltando o botão.

**Slice and dice:** É a descrição da habilidade de fatiar e cortar o cubo separando partes de um cubo. O uso do *slice and dice* permite rotacionar os lados de um cubo de dados (dimensões) em qualquer sentido, possibilitando assim, a combinação de quaisquer dimensões e a obtenção de informações correspondentes sobre o enfoque desejado.

**Drill down e roll-up(ou drill-up):** *Drill-down* serve para que usuário tenha a possibilidade de ter uma visão mais detalhada de um conjunto de dados, navegando por exemplo em uma hierarquia de dados pré-definida. *Roll-up* é o processo inverso do *drill-down*, ou seja, exibe os dados de uma forma mais macro, mais agrupados ou sumariados. A figura 1.10 representa os processos de *drill-down* e *roll-up* em um exemplo de hierarquia de tempo.

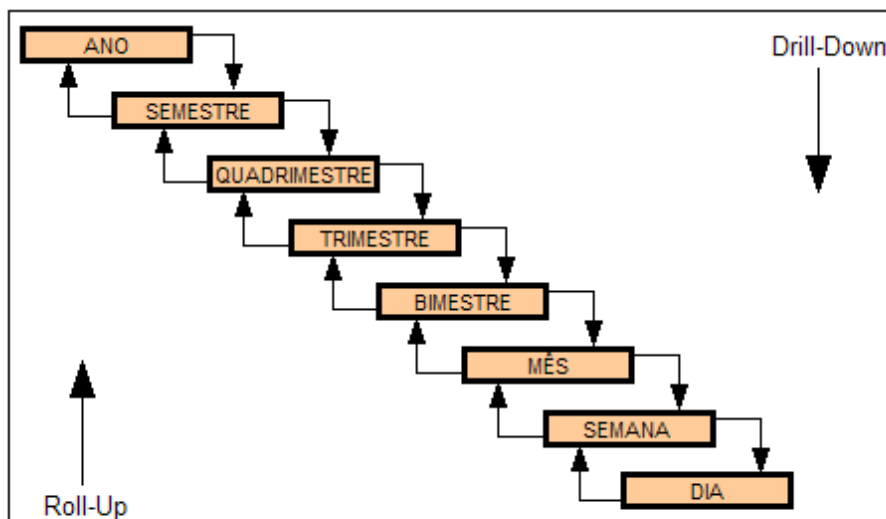


Figura 1.10 - Exemplo das operações de *Drill-Down* e *Roll-Up*

**Drill-across e Drill-through:** São variações das operações de *drill-down* e *roll-up*. *Drill-across* diz respeito a navegação em uma dimensão avançando os níveis intermediários, enquanto o *drill-through* está relacionado a navegar a um ponto de detalhe menor que o existente na dimensão.

**Tabelas cruzadas:** as tradicionais planilhas eletrônicas. A diferença reside no fato de que os dados são apresentados em planilhas com mais de duas dimensões, normalmente quatro ou mais.

**Consultas *ad-hoc*:** segundo Inmon (1997) são consultas com acesso casual único e tratamento dos dados segundo parâmetros nunca antes utilizados, geralmente executados de forma interativa e heurística. De forma mais sucinta, consultas *ad-hoc* estão ligadas ao fato de o usuário gerar consultas de acordo com suas necessidades, relacionando informações de uma maneira única e própria.

**Geração de *queries*:** possibilidade de o usuário montar *queries* próprias, através de uma ferramenta amigável e transparente, sobre a qual é necessário um conhecimento mínimo de informática por parte do usuário, a fim de obter as informações que deseja. Basicamente consiste em gerar o acesso ao *Data Warehouse/Data Mart* para obter a informação e analisá-la (LOPES, 2006).

Além dessas operações básicas, diversas ferramentas possuem funções que atuam como indicadores. Exemplos disso são semáforos e sinalizadores, além dos indicadores de tendências, relatórios de exceção, previsões, projeções, simulações, entre outras (WAGNER, 2003).

### 1.3.2 Arquiteturas OLAP

Segundo Anzanello (2002), de acordo com o método de armazenamento de dados para a aplicação OLAP, será elaborada a arquitetura ideal para a aplicação. Os principais métodos para esse fim são MOLAP, ROLAP, DOLAP E HOLAP, complementados por uma tendência chamada JOLAP. Destes, cada um tem uma função específica e deve ser utilizada quando melhor atender às necessidades de análise pela ferramenta de OLAP.

#### 1.3.2.1 ROLAP

Segundo Kimball (1998), o ROLAP (Relacional OLAP - *Relational OLAP*) constitui-se de um conjunto de interfaces de usuário e aplicações que dá ao banco de dados relacional características dimensionais.



Com relação ao funcionamento de ROLAP, Bispo (1998), afirmam que as ferramentas ROLAP podem efetuar o processamento para realizar as consultas ou outras funções em um ambiente dimensional de duas formas:

1. Efetuar o processamento dos dados no servidor da base de dados, ou seja. O servidor OLAP gera os comandos SQL em múltiplos passos e as tabelas temporárias necessárias para o devido processamento das consultas;
2. Executar comandos SQL para recuperar os dados, mas fazer todo o processamento, incluindo junções e agregações, no servidor OLAP.

De acordo com Fernandes (2004), o desenvolvimento de aplicações ROLAP pressupõe que não é criada mais nenhuma estrutura fixa de dados, para além da estrutura de dados do *data warehouse*. As *queries* (consulta a um banco de dados) são elaboradas na própria ferramenta ROLAP e sempre que o utilizador pede um relatório contendo determinada *query*, esta é lançada no motor de dados que suporta o *Data Warehouse*, que enviará posteriormente uma resposta à ferramenta ROLAP a fim de esta apresentar os referidos resultados.

De acordo com Figueiredo (1998), uma das vantagens na utilização de uma solução ROLAP consiste na utilização de uma tecnologia estabelecida, de arquitetura aberta e padronizada como é a relacional, beneficiando-se da diversidade de plataformas, escalabilidade e paralelismo de *hardware*.

### **1.3.2.2 MOLAP**

Segundo Kimball (1998), MOLAP (*Multidimensional OLAP*), também chamado de Banco de Dados Multidimensional, constitui-se de um conjunto de interfaces de usuário, aplicações e banco de dados, com tecnologia proprietária, que possui características eminentemente dimensionais. Esse tipo de banco de dados armazena os dados em cubos que podem conter múltiplas dimensões, adicionando ainda a variável de tempo às dimensões.

Através da análise multidimensional, é possível comparar qualquer parte do negócio com qualquer outra parte, definindo diversos tipos de análise, sempre que necessários. Isso, sem a obrigatoriedade de ter que projetar um novo banco de dados para cada análise realizada. Assim há a possibilidade de gerar inúmeras análises de forma intensa, em um curto espaço de tempo (BISPO, 1998).

Por ser uma tecnologia complexa, principalmente quanto as questões de armazenamento e visualização de dados, o MOLAP exige uma trabalho muito maior com todo o processo de engenharia de banco de dados, tamanho, estruturação, processo de carga, indexação, otimização, entre outros (HOKAMA et al., 2004).

### **1.3.2.3 HOLAP**

Segundo Bispo (1998), os desenvolvedores passaram a utilizar o OLAP de uma forma que, para atender as necessidades dos clientes, pudessem maximizar as vantagens e minimizar as desvantagens dessa tecnologia. Para isso, o OLAP passou a ser utilizado de maneira híbrida, ou seja, utilizando MOLAP e ROLAP.

Usufruindo dessas duas tecnologias OLAP pode-se utilizar, por exemplo, da vantagem de alta performance do MOLAP com a melhor escalabilidade do ROLAP. E segundo Bispo (1998), as ferramentas atuais são inteligentes o suficiente para definir qual tecnologia será utilizada, de acordo com a atividade a ser realizada, proporcionando assim o melhor desempenho possível.

### **1.3.2.4 WOLAP**

Conforme Gerber (1998), já há uma migração da tecnologia OLAP para o ambiente de internet, e está sendo chamada de WOLAP ou *Web OLAP*. De acordo com Carickhoff (1997) as facilidades desse tipo de ambiente são: possibilidade de plataformas independentes para dar suporte a usuários distantes, aplicações de softwares colaborativos, facilidades de aprendizado e de manutenção. As dificuldades do WOLAP são as limitações dos recursos da *Internet*, em relação as interfaces e facilidades do ambiente de cliente-servidor.

### **1.3.2.5 DOLAP**

Segundo Cielo (2000), DOLAP (*Desktop On-Line Analytical Processing*) são ferramentas que disparam uma instrução SQL, de um cliente qualquer, para o servidor e recebem o microcubo de informações de volta para ser analisado no cliente.

Uma das vantagens no uso dessa arquitetura é a redução de tráfego na rede, visto que todo o processamento OLAP acontece na máquina cliente, causando maior agilidade na análise, além do servidor de banco de dados não ficar sobrecarregado, evitando problemas de escalabilidade.

A desvantagem é que o tamanho do micro-cubo não pode ser muito grande, caso isso ocorra, a análise passa a ser demorada e/ou a máquina do cliente pode não suportar em função de sua configuração.

### 1.3.2.6 JOLAP

Segundo Anzanello (2002), o JOLAP (*On-Line Analytic Processing*) pode ser considerada uma tendência, por se tratar ainda de um esforço da *Java Community Process* (JCP)<sup>4</sup> em projetar uma API Java<sup>5</sup> para servidores e aplicações OLAP, aderentes ao ambiente *Java 2Platform Enterprise Edition* (J2EE). Seu projeto trabalha na questão de poder suportar a criação e manutenção de dados e metadados OLAP, independente de fornecedor.

A JOLAP não possui uma linguagem de consulta multidimensional definida em sua especificação. Ao invés disso, possui vários métodos, classes e modelos de consultas disponíveis (HYPERION, SUN, 2003).

JOLAP é baseada em uma forte generalização, orientada a objeto e nos conceitos de OLAP. Este modelo suporta conceitos referentes a três áreas que são chave para as aplicações OLAP: metadados, dados e pesquisas. As especificações não descrevem a estratégia de implementação. Este projeto está em fase de amadurecimento e conta com a participação de fornecedores como a IBM, *Oracle*, *Microsoft*, *Sun*, entre outros (ANZANELLO, 2002).

### 1.3.2.7 ROLAP x MOLAP, qual a melhor tecnologia OLAP?

Essa é uma dúvida que acaba gerando muitas discussões, pois cada uma delas tem suas vantagens e desvantagens. Porém, é necessário analisar muito bem o contexto em que será utilizada a tecnologia OLAP para decidir qual delas usar.

Bispo apud Gentia e Software (1998), fez levantamentos de pontos importantes entre os dois tipos de tecnologias OLAP que podem auxiliar na tomada de decisão de qual delas usar:

**Volume de dados:** Ferramentas ROLAP podem gerenciar volumes maiores que MOLAP, mas o desempenho e manutenção podem se tornar limitados. Mas o que vem

---

4 O *Java Community Process* tem o objetivo de desenvolver e rever as especificações da tecnologia Java. JCP.

5 *Java Technology*. Disponível em: < <http://java.sun.com>>. Acesso em: 06 Jun. 2007

ocorrendo é a criação de subconjuntos de volume de dados grandes, tornando vantajosos ambos os tipos de tecnologia;

**Aplicações de constantes atualizações:** O processo de atualização de dados é bastante semelhante entre os dois tipos de tecnologia;

**Análise real de dados operacionais:** Não é recomendável efetuar análises complexas de forma multidimensional em bancos transacionais. Mas caso isso seja necessário, os tempos de resposta entre ROLAP e MOLAP são equivalentes;

**Disponibilidade dos dados:** Os dados em ROLAP podem ser acessados por um horizonte maior de ferramentas. Mas para que os tempos de respostas sejam aceitáveis, os dados são armazenados em tabelas relacionais de certa forma numerosas, e às vezes não muito compreensíveis para o usuário;

**Velocidade na carga de dados:** a carga dos dados implica em uma série de processos, que vai desde a leitura dos dados, até uma validação final dos mesmos. Tudo isso ocorre mantendo-se o banco de dados *on-line* e consistente. Portanto, é mais provável que o processo se torne mais rápido no MOLAP do que no ROLAP.

**Performance em cálculos e recuperação de dados:** O ROLAP necessita de mais processamento para executar as tarefas do MOLAP, porém um ROLAP bem projetado pode atingir um desempenho equiparável ao de um MOLAP.

**Análise multidimensional:** Mesmo ferramentas ROLAP que utilizam indexação para simular bancos multidimensionais não atingem a mesma performance de ferramentas MOLAP. Com relação aos cálculos multidimensionais, as ferramentas ROLAP superaram limitações nesse ponto através de múltiplos passos de SQL, mas mesmo assim não tem a mesma quantidade de cálculos que o MOLAP.

**Análises simples:** MOLAP não são necessários para consultas mais simples, que podem ser geradas através das ferramentas OLAP ou geradores de relatórios que acompanham os bancos de dados relacionais.

**Plataformas de servidores:** Alguns produtos ROLAP possuem maior disponibilidade em plataformas diferentes do que a maioria dos produtos MOLAP;

**Custo:** MOLAP implica em bancos de dados especiais e licenças mais caras. Porém o fato de necessitar de menos espaço em disco, menos processamento e menos refinação, fazem com que o custo do MOLAP possa se tornar acessível mesmo com as taxas embutidas na compra desse tipo de tecnologia.

Gorla (2003), também fez um estudo nessa comparação e acabou chegando em algumas conclusões. A indicação para utilizar MOLAP seria para usuários que consigam pré-definir todas as suas necessidades de informação e que não precisem dos dados numa base de atualização diária. Sendo assim MOLAP poderia ser usado naqueles sistemas em que a atualidade dos dados não é uma questão vital para o negócio e se admite uma defasagem entre os dados reais e os das análises que pode ir de um dia a uma semana, ou mais.

Com relação a ROLAP, Gorla (2003) recomenda para usuários que precisem analisar informações de mercado com elevado nível de variação e que tenham uma constância maior de atualização. Ainda, nos casos em que a volatilidade dos dados é constante, os sistemas ROLAP apresentam melhores respostas.

#### **1.4 Data Mining**

De acordo com Harrison (1998), o conceito de *Data Mining* está relacionando a exploração e análise de grandes quantidades de dados, por meios automáticos ou semi-automáticos, buscando descobrir regras significativas.

O princípio do *Data Mining* está na idéia de que, ao invés de o usuário definir o problema, selecionar os dados e as ferramentas para analisar tais dados, as ferramentas de *Data Mining* pesquisam automaticamente os mesmos, à procura de anomalias e possíveis relacionamentos, identificando assim problemas que não tinham sido identificados pelo usuário. Ou melhor, as ferramentas dessa tecnologia analisam, descobrem problemas ou oportunidades escondidas nos relacionamentos dos dados, e diagnosticarão o comportamento dos negócios, requerendo a mínima intervenção do usuário, assim ele se dedicará somente a buscar conhecimento e produzir mais vantagens competitivas (Bevilacqua e Bitu, 2003).

O *Data Mining* no contexto de *Business Intelligence* pode ser exemplificado através da figura 1.11:

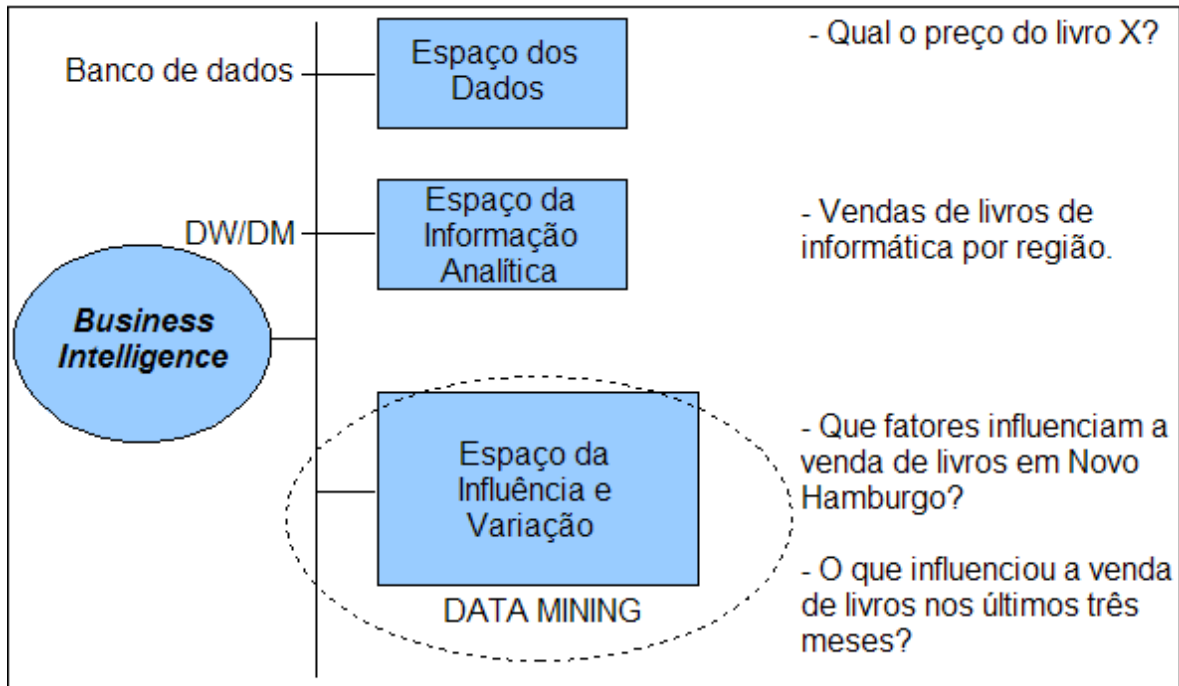


Figura 1.11 - *Data Mining* dentro do contexto de *Business Intelligence*

Fonte: Adaptada de Barbieri, 2001

Visto os principais conceitos de BI, no próximo capítulo serão apresentadas algumas ferramentas dessa tecnologia, sendo focadas em OLAP e *Data Warehouse*.

## 2 FERRAMENTAS

Nesse capítulo, será realizado um estudo das principais ferramentas de *Business Intelligence*. Será feita uma divisão em duas seções. Na primeira serão abordadas as principais ferramentas de *Data Warehouse* existentes no mercado, e na segunda, as principais ferramentas de OLAP.

### 2.1 Ferramentas DW

As ferramentas que serão mostradas a seguir foram analisadas através de manuais técnicos, sites relacionados e trabalhos acadêmicos que utilizaram algumas das ferramentas, que são: *Oracle Warehouse Builder*, *Oracle Express*, *Microsoft DTS* e *Business Information Warehouse*.

Ao final do capítulo será feito um breve comparativo entre as ferramentas DW descritas, apontando as principais diferenças entre elas, em um quadro com os pontos fundamentais que compõem esse tipo de produto.

#### 2.1.1 Oracle Warehouse Builder

A seguir, será explicado com um nível maior de detalhes o funcionamento da ferramenta *Oracle Warehouse Builder* (OWB). Esse detalhamento maior foi realizado por se tratar da ferramenta que será utilizada para apoiar o processo de criação do repositório de dados do projeto.

##### 2.1.1.1 A ferramenta

O *Oracle Warehouse Builder* é uma ferramenta de *Business Intelligence* da empresa *Oracle Corporation*, desenvolvida em 1999 e que hoje está na versão 10g. Ela traz uma

solução integrada para modelar e desenvolver *data warehouses*, *data marts* e aplicações de *Business Intelligence*.

Ela possui componentes que efetuam o processo de ETL e, além disso, é considerada o ponto central da integração com as ferramentas do pacote *Oracle BI*, provendo integração com utilitários de consultas *ad-hoc* e funcionalidades de bancos relacionais (ORACLE, 2003).

### 2.1.1.2 Características principais

Conforme Marques (2002), o OWB é um aplicativo, de interface *Java*, que opera em múltiplas camadas: interface com o usuário, gerador de código, integradores, interface de aplicativo *Java* e o repositório de metadados. Essas ferramentas acabam tornando um produto de referência no mercado na questão de carregamento e manutenção do DW.

Uma das vantagens do OWB em relação às outras ferramentas de ETL é que a maioria das atividades necessárias para se modelar um *warehouse* são feitas através de assistentes, os chamados *wizards*, que orientam todos os passos necessários para se definir o *data warehouse* (GONÇALVES, 2002). E isso, traz um grande ganho de produtividade, além de facilitar o uso para o desenvolvedor,. A figura 2.1 mostra um exemplo das telas de *wizard* existentes no OWB, o assistente de criação de uma tabela.

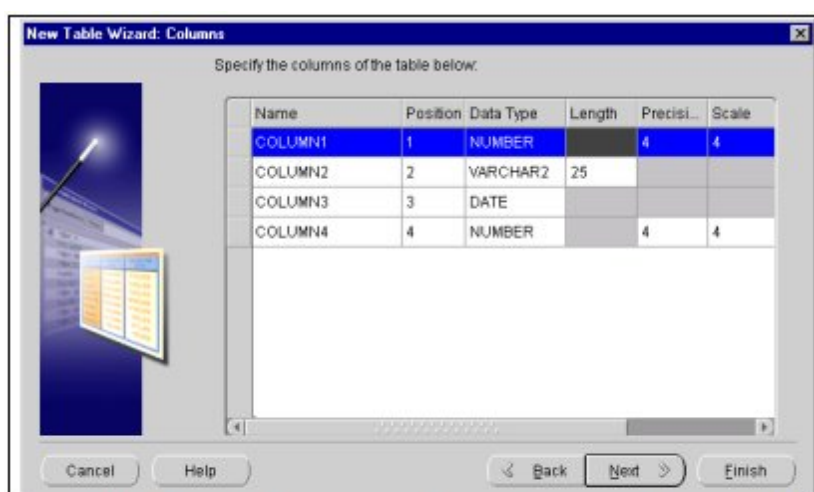


Figura 2.1 - Exemplo de um *wizard* na criação de tabelas no OWB  
Fonte: ORACLE (2003)

#### 2.1.1.2.1 Orientado a Projetos

Sempre que for iniciada uma nova modelagem e construção de um *warehouse* é necessário que se crie um novo projeto. Por isso, pode-se afirmar que o OWB é uma



ferramenta orientada a projeto. As informações existentes em cada projeto são armazenadas no próprio repositório do OWB através de um SGBD *Oracle*. Segundo Marques (2002), as informações armazenadas no repositório de metadados podem ser utilizadas fora da ferramenta, possibilitando uma integração com o *Oracle Business Intelligence*.

Na figura 2.2, pode-se observar a tela principal do OWB, mostrando um novo projeto, denominado de “MY\_PROJECT”, com os principais itens que o compõem.

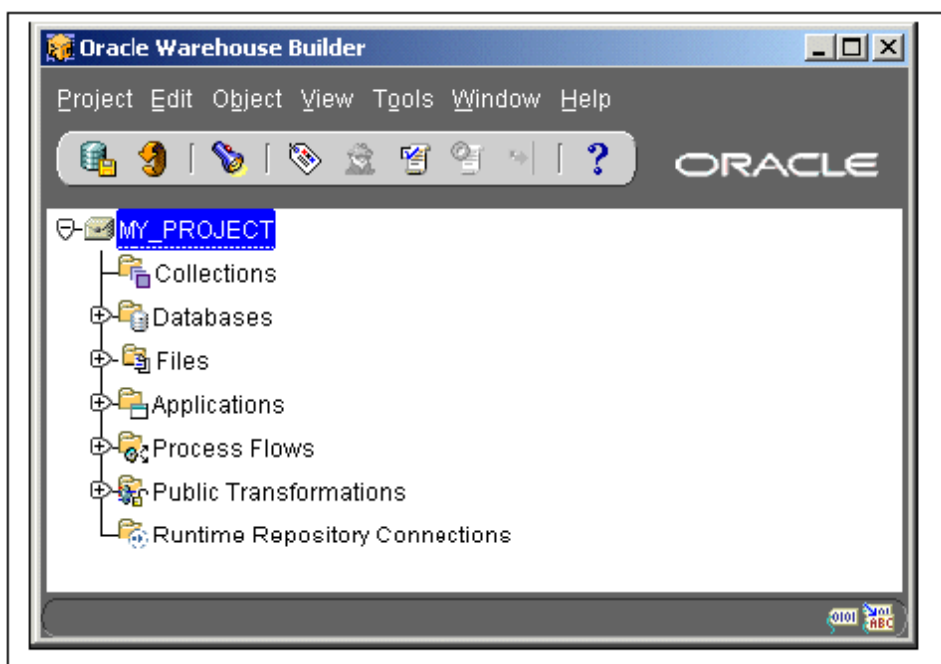


Figura 2.2 - *Oracle Warehouse Builder*  
Fonte: Oracle (2003)

#### 2.1.1.2.2 Fontes de dados

Conforme a *Oracle* (2003), o OWB pode criar módulos de dados de diferentes fontes. As fontes são:

1. Bancos de dados *Oracle*;
2. Bancos de dados não-*Oracle*;
3. Arquivos de texto;
4. Aplicações em SAP.

Conforme Gonçalves (2002), uma restrição existente no *Oracle Warehouse Builder* é que existem apenas três interfaces de integração que são disponibilizadas na própria ferramenta: interface com bancos de dados *Oracle*, arquivos de texto e aplicações SAP.

As outras ferramentas de interface que podem ser utilizadas são *gateways* desenvolvidos pela *Oracle* para acessar outros bancos de dados. Porém, para adquirir esses produtos há um custo adicional. Segundo a *Oracle*, *drivers* ODBC também podem ser utilizados para prover conexões com bancos de dados relacionais de outros fornecedores.

Com relação aos tipos de dados oferecidos pela ferramenta, o OWB trabalha com objetos dimensionais e relacionais. Dentro dos objetos relacionais estão incluídas tabelas, *views*<sup>1</sup>, *materialized views*<sup>2</sup> e *sequencies*<sup>3</sup>. Os objetos multidimensionais incluem dimensões e cubos de dados (ORACLE, 2003).

#### **2.1.1.2.3 Importação de Metadados**

Uma das opções que merece destaque na ferramenta é a possibilidade do usuário poder importar metadados de estruturas de banco de dados transacionais e/ou de *data warehouses* já existentes. Esta técnica, que é conhecida na maioria das ferramentas CASE como engenharia reversa, tendo como principal objetivo documentar bases de dados que foram construídas sem a utilização de uma ferramenta CASE. Além disso, a importação de metadados no OWB facilita principalmente a modelagem das bases transacionais que são utilizadas para a extração de dados, pois com a importação dos dados para o DW, o usuário não necessita definir de forma manual todos os objetos que serão utilizados na extração (GONÇALVES, 2002).

#### **2.1.2 Oracle Express**

Adquirida pela *Oracle* em 1995 da *Information Resource Inc.*, não é apenas uma ferramenta que desempenha o papel de *data warehouse*. Além de trabalhar com o processo de ETL, o *Oracle Express* atua como ferramenta de OLAP. Na versão *Oracle Express Server*,

- 
- 1 *Views* são apresentações personalizadas de dados através de um SQL consultando em uma ou mais tabelas (ORACLE, 2003).
  - 2 *Materialized Views* são tabelas de *views* atualizáveis, que possuem dados vindos através de uma consulta SQL de uma ou mais tabelas (ORACLE, 2003).
  - 3 *Sequences* são objetos do banco de dados que geram listas de valores únicos (ORACLE, 2003).

atua como servidor de banco de dados multidimensional. Na versão *Oracle Personal Express*, desempenha a função de DOLAP, executando na versão cliente.

Segundo Barbieri (2001), como MOLAP, o *Oracle Express* armazena os dados como uma planilha multidimensional, com aspectos de otimização de espaços para valores esparsos, estratégias especiais de indexação, que se diferem daquelas usadas nos gerenciadores relacionais, nos quais as linhas são armazenadas com as suas chaves adjacentes dos dados.

A ferramenta trabalha nos três estilos de cliente-servidor: completa, média e *web*. Na primeira, os dados podem ser transferidos do servidor para o cliente, onde as consultas rodarão na versão local, o *Oracle Personal Express*. Na média, o cliente acessa os dados através do servidor, e a terceira, o cliente utiliza o *browser* como único instrumento para o acesso dos dados no servidor.

### 2.1.3 Microsoft DTS

O *Microsoft Data Transformation Services* (DTS) é um produto da empresa *Microsoft*, composta de um conjunto de ferramentas gráficas e objetos programáveis com o objetivo de extrair, transformar e consolidar dados de fontes diversas em simples ou múltiplo destino.

Conforme Barbieri (2001), a manipulação de dados pode ser efetuada através de passos, agrupados em pacotes, que podem ser automatizados e ter também o seu fluxo controlado, por desvios e condicionantes. Dentro de um passo, é possível executar um SQL<sup>4</sup>, um *script* em *Java*, *Perl* ou *Visual Basic*, qualquer programa externo ou ainda recuperar um outro pacote de ações, inclusive um envio de *e-mail*.

Segundo Youness (2000), o DTS permite referenciar *scripts* do banco de dados, procedimentos armazenados e programas externos responsáveis pela execução da validação dos dados que serão migrados ao DW.

Segundo Peterson et al. (1999), as principais funcionalidades apresentadas pelo *Microsoft DTS* são:

1. Suporte a múltiplas fontes de dados como fonte e destino;

---

4 SQL é uma sintaxe, criada pela IBM, usada para a definição e manipulação de dados em um banco de dados relacional (WIKIPEDIA, 2007)

2. Operação de manipulação sobre cada registro de dado extraído da fonte usando funções scripts *VBScript*<sup>5</sup>, *JScript*<sup>6</sup>, ou *PerlScript*<sup>7</sup>;
3. Importação de dados em arquivo texto para o *SQL Server*;
4. Transferência de objetos do *SQL Server* de um servidor para outro;
5. Transferência do esquema e dados de um banco relacional para o *SQL Server*;
6. Registro de como os dados foram manipulados;

#### **2.1.4 Business Information Warehouse**

O *Business Information Warehouse* é o produto da empresa SAP, que tem por objetivo a disponibilização de um ambiente separado do produto de ERP<sup>8</sup> da mesma empresa, o ERP-SAP R/3, para o tratamento exclusivo de informações gerenciais. É uma ferramenta que já vem com alguns itens pré-configurados, como *templates*<sup>9</sup> de relatórios, modelos definidos, além de diversos procedimentos de ETL (BARBIERI, 2001).

A ferramenta é composta por três módulos: Explorador de Negócios (*Business Explorer*), Servidor BW (*BW Server*) e os Extratores de dados.

No *Explorer*, o objetivo são a análise e produção de relatórios, que possui um vasto repertório de relatórios pré-formatados, ou relatórios dimensionais, além de possibilitar manipulação de dados através de planilhas do Microsoft Excel<sup>10</sup>.

Os extratores de dados atuam de duas formas: para bases SAP e bases não-SAP. Com relação às bases SAP, os dados são carregados transparentemente até o repositório de dados DW, através de processos já disponibilizados pela ferramenta. Para bases não-SAP, a solução

---

5 *Visual Basic Scripting Edition*. É um subconjunto da linguagem de programação do *Visual Basic* da Microsoft, otimizado para a programação relacionada à Web (WIKIPEDIA, 2007).

6 *Jscript* é a denominação dada pela Microsoft para a linguagem de *scripts Java* chamada JavaScript. MACHADO., 2000)

7 *PerlScript* é um *script* ActiveX desenvolvido pela empresa *ActiveState* (WIKIPEDIA, 2007).

8 Os ERPs são sistemas de informação integrados adquiridos na forma de pacotes comerciais de *software* com a finalidade de dar suporte à maioria das operações de uma empresa (BERVIAN apud SOUZA e ZWICKER, 2000).

9 *Template* é um documento sem conteúdo, com apenas a apresentação visual (apenas cabeçalhos por exemplo) e instruções sobre onde e qual tipo de conteúdo deve entrar a cada parcela da apresentação (WIKIPEDIA, 2007).

10 Ferramenta para criação e edição de planilhas de cálculo do pacote *Office* da *Microsoft* (MICROSOFT, 2007)

é utilizar ferramentas de terceiros, que estão disponíveis no mercado, porém para bases Oracle, a SAP possui um mecanismo de extração nativa.

O *BW Server* é o componente que efetivamente realiza os processamentos das consultas e demais solicitações. Ele trata os dados multidimensionais através de *drill-down*, *roll-up* e *drill-across*. Possui ainda um *cache* dos relatórios mais utilizados, que pode acabar otimizando as requisições repetidas (BARBIERI, 2001).

### 2.1.5 Comparativo das ferramentas DW

No quadro 2.1 estão alguns dos recursos principais das ferramentas de *Data Warehouse* que foram apresentadas nesse trabalho. Os itens foram retirados de uma relação feita por Gonçalves (2002). Nesse levantamento, alguns itens como desempenho e escalabilidade eram analisados. Porém, estes itens não foram analisados nesse projeto pelo fato de que esse tipo de análise não seja o escopo principal, mas uma apuração de recursos que deveriam ser testados em cada uma das ferramentas.

Quadro 2.1 – Recursos das ferramentas de *Data Warehouse*

<i>Nr</i>	<i>Descrição</i>
1	Extrair dados de diversas fontes
2	Executar em ambiente multi-plataforma
3	Permitir compactação de dados
4	Permitir criptografia de dados
5	Conter módulo para administração
6	Permitir gerenciamento centralizado
7	Conter repositório de metadados
8	Permitir <i>schedule</i> de tarefas
9	Disponibilizar interface gráfica para montar cargas de dados
10	Disponibilizar uma linguagem para transformação dos dados
11	Gerar código através de diagramas gráficos
12	Permitir importar metadados

Fonte: Adaptado de Gonçalves (2002)

Assim, o seguinte quadro pode ser desenvolvido, através de manuais técnicos e casos de uso, comparando os recursos das ferramentas descritas nesse trabalho:

Quadro 2.2 – Comparativo de ferramentas *Data Warehouse*

<b>Ferramenta /Recurso</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>
Oracle Warehouse Builder	X	X			X	X	X	X	X		X	X
Oracle Express	X	X			X	X	X	X	X		X	X
Microsoft DTS	X				X	X	X	X	X		X	X
Business Information Warehouse	X	X			X	X	X		X		X	X

Com relação ao item 2 não estar presente na ferramenta *Microsoft DTS*, deve-se ao fato da mesma não ser executada em sistemas operacionais não-*Windows*. Os itens 3 e 4 e 10 não foram localizados em nenhuma das documentações das ferramentas, portanto, chegou-se a conclusão de que as funcionalidades não estão presentes. O mesmo ocorre no item 8 para a ferramenta *Business Information Warehouse*, no qual não foi identificado nenhum recurso de *schedule* de tarefas

## 2.2 Ferramentas OLAP

Nessa seção serão mostradas algumas das ferramentas de análise OLAP disponíveis no mercado. As ferramentas a serem estudadas são: *Oracle Business Intelligence*, *Microsoft Analysis Services* e as ferramentas *open source: Pentaho e OpenI*.

Assim como realizado com o DW, ao final do capítulo será feito um breve comparativo entre as ferramentas de OLAP descritas, trazendo as principais diferenças entre elas, num quadro com os pontos fundamentais que compõem esse tipo de produto.

### 2.2.1 Oracle Business Intelligence

Como o próprio nome já indica, o *Oracle Business Intelligence (Oracle BI)* é a solução para *Business Intelligence* da *Oracle Corporation*. É um produto bastante robusto, que atende a toda uma gama de requisitos analíticos das empresas, que inclui consultas *ad-hoc*, geração de relatórios e análise, ETL e desenvolvimento de aplicativos de BI. Suas principais ferramentas são (ORACLE BI, 2005):

**Oracle Discoverer** – Consulta, geração de relatórios e análise com recursos de formatação de dados;

**Oracle Spreadsheet Add-In** – Funcionalidade para acessar dados do DW através de planilhas do *Microsoft Excel*;

**Oracle BI Beans** – Desenvolvimento personalizado de aplicativos de BI;

**Oracle Warehouse Builder** – ferramenta de ETL já explicada anteriormente.

Com essas ferramentas, o *Oracle Business Intelligence* torna-se um produto completo, com acesso integrado ao banco de dados, eliminando necessidades de integração entre componentes de BI de fornecedores diferentes.

Dentre as ferramentas que compõem o pacote *Oracle Business Intelligence*, o projeto utilizará a *Oracle Discoverer*, explicada com detalhes a seguir.

#### **2.2.1.1 Oracle Discoverer**

O *Oracle Discoverer* é a ferramenta de análise OLAP do pacote *Oracle BI*, que é baseada na arquitetura ROLAP e atende às necessidades de inteligência empresarial, como consultas *ad-hoc*, criação de relatórios e gráficos, explorações e consultas na *web*, através de DW ou *Data Marts*. Seu acesso aos bancos de dados é feito através do SQL Net, no caso de um SGBD *Oracle*, ou via ODBC para outros SGBD's (ORACLE 2005a).

Além de atender bancos relacionais, a última versão do produto, 10g, traz a opção de análise em ambientes multidimensionais, desde que esteja baseada em um banco de dados *Oracle* na versão *Enterprise Edition*, com a opção OLAP instalada. A figura 2.3 ilustra um esquema mostrando a estrutura da ferramenta *Oracle Discoverer*, mostrando seus componentes principais (ORACLE 2005a).

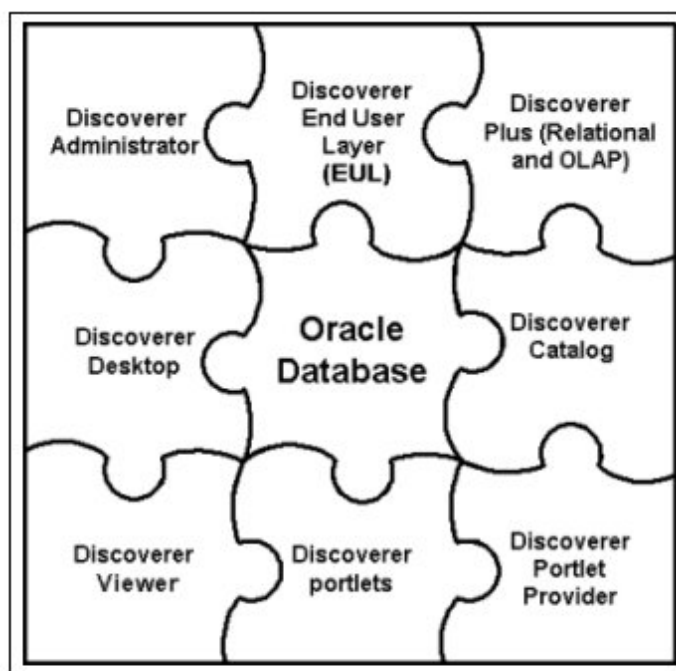


Figura 2.3 - Componentes do *Oracle Discoverer*  
Fonte: *Oracle* (2005a)

#### 2.2.1.1.1 Discoverer Administrator

O *Discoverer Administrator* é o componente responsável por gerenciar o ambiente de dados que será utilizado para as consultas dos usuários finais. Nele, há uma camada lógica de metadados chamada *End User Layer* (EUL), onde são criadas as hierarquias de dados, tabelas sumariadas, lista de valores e as áreas de negócios, chamadas de *Business Areas* (ORACLE 2005a).

Como o próprio nome aponta, as áreas de negócios são conjuntos de informações (tabelas ou visões) relacionadas entre si e agrupadas logicamente, com intuito de atender a uma finalidade comum de negócios e necessidades de informação do usuário. Por exemplo, poderia ser criada dentro de uma EUL, uma área de negócios voltada para um DM de vendas e outra para um DM relacionado aos dados de produção (ORACLE 2005a).

Cada *Business Area* pode ser composta de um ou mais *folders*, que são um conjunto de dados retirados do BD, que podem ser uma tabela, *view*, junções de tabelas ou até mesmo um SQL personalizado. Cada *folder* é composto de itens, que podem ser colunas da tabela ou *view*, ou ainda fórmulas desenvolvidas para um fim específico. Os itens podem ter sua máscara e nomes alterados, de forma que, quando visualizados pelo usuário final, estejam da forma mais amigável possível. Na figura 2.4 há uma representação do *Discoverer Administrator*, composta por uma *Business Area* com alguns *folders* (ORACLE 2005a).



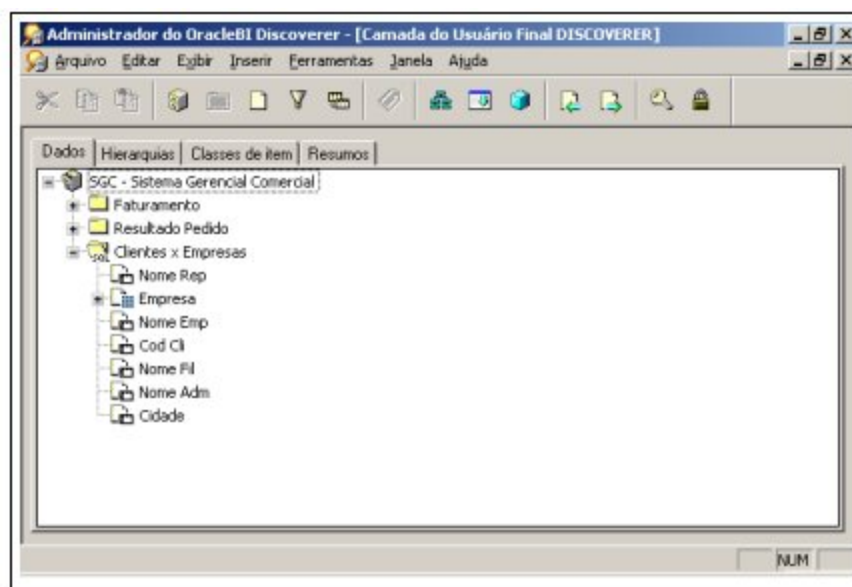


Figura 2.4 - *Discoverer Administrator*  
 Fonte: Adaptado de Oracle (2005a)

No *Administrator*, existem funcionalidades para dar permissão aos usuários para que visualizem ou não determinada *Business Area*. Ou ainda, permitir que certos usuários criem ou compartilhem consultas baseadas em determinadas *Business Areas* (ORACLE 2005a).

Uma das funções do *Administrator* que merece destaque é a criação de hierarquias. As hierarquias podem ser criadas através de uma lógica de dados ou datas. Por exemplo, é possível criar uma hierarquia que possui como nível máximo o cliente, num nível mais baixo o produto e em seguida o pedido. Ou ainda uma hierarquia com ano, semestre, mês e dia. Essas montagens servirão para a utilização do *drill-down* e *roll-up* nas consultas do usuário final (ORACLE 2005a).

#### 2.2.1.1.2 *Discoverer Desktop*

Esse é o componente responsável pela interação com o usuário final, que possibilita desenvolver ou mesmo consultar relatórios baseados nas *Business Areas* disponibilizadas a ele através do *Discoverer Administrator*. É uma aplicação que deve ser instalada na máquina do cliente e pode rodar em *Windows*, *Solaris*, *Linux* e *HP-UX*, mas não pode ser rodada na web (ORACLE, 2005b).

A ferramenta possui uma interface bastante semelhante a um programa de planilha de cálculo, pois assim como, por exemplo, um arquivo do *Microsoft Excel*, um documento do

*Discoverer*, que é chamado de *workbook*, pode conter diversas páginas (*worksheets*) com assuntos diferentes (ORACLE, 2005b). A estrutura que compõe a ferramenta pode ser visualizada na figura 2.5.

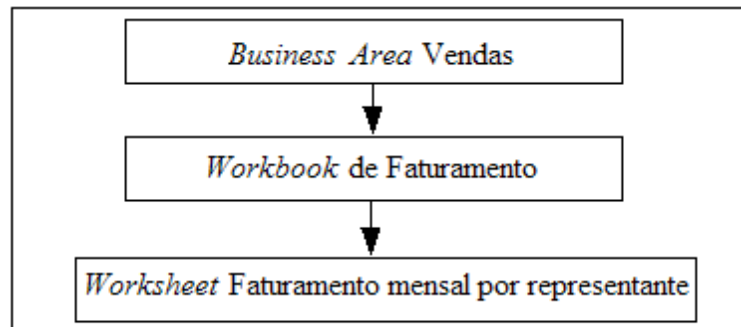


Figura 2.5 - Estrutura do *Discoverer Desktop*  
Fonte: Adaptado de Meira e Souza (2002)

A *worksheet* possui dois tipos principais de visualização de dados: tabela e tabelas com referência cruzada. Na tabela os itens são exibidos em linhas e colunas, como se estivesse visualizando uma tabela diretamente do banco de dados. A tabela com referência cruzada permite uma visão multidimensional dos dados, na qual o usuário pode cruzar linhas e colunas, com identificações de dados. Cada formato permite ainda a inclusão dos chamados itens de página, que são parâmetros que permitem novas consultas em uma mesma *worksheet* sem uma nova requisição ao banco de dados (ORACLE, 2005b). A figura 2.6 mostra uma consulta formatada em tabela de referência cruzada, com itens de página.

Area de Trabalho do Oracle Business Intelligence Discoverer - [SGCD0100 - Sistema Gerencial Comercial]

Sistema Gerencial Comercial  
Faturamento por Representante

Data: 11/05/07  
Pág.: 1 à 1

Itens de página: Ano: 2007 | Segmento: <tudo> | Tipo de Pedido: <tudo> | Mercado: <tudo> | Administrador: 2-EDUARDO

Representante	Valor Faturado						
	Mês	> 01/2007	> 02/2007	> 03/2007	> 04/2007	> 05/2007	Total
> 21-REPRES. REICHERT LTDA		753.666,05	818.058,10	798.731,34	897.461,95	428.283,20	3.686.200,64
> 9-OBMO REPRESENTACOES LTDA		452.753,02	434.840,63	724.294,30	533.704,98	224.601,64	2.370.194,57
> 99-OBRETO.		683.456,85	504.303,78	435.340,53	388.727,92	203.119,11	2.214.948,19
> 59-JEISON ANDRE DE AZEVEDO BARBOSA		371.191,66	382.066,65	497.983,05	461.985,86	181.290,58	1.894.517,80
> 1-WERNER VIDEO COM. E REPRESENT. LTDA ME		442.654,92	385.464,10	422.254,67	356.075,29	217.792,52	1.824.241,50
> 10-B.S.COM. REPRESENT. LTDA		351.484,59	344.797,15	422.353,98	270.317,21	120.041,06	1.508.993,99
> 28-DINI.COM. E REPRESENT. LTDA		266.051,40	174.571,77	311.867,44	354.238,84	156.627,63	1.263.357,00
> 12-MONTEBAL REPRESENT. LTDA		118.676,78	185.698,13	260.071,29	377.172,53	113.328,87	1.054.947,63
> 6-MARCOS ROSENSTENZEL		15.074,13	12.142,13	27.982,94	471.490,67	22.355,76	549.045,63
> 76-PAULA RAMOS REPRESENTAÇÕES LTDA ME			89.673,14	103.077,64	71.748,08	8.613,87	273.112,73
> 35-INEL REPRESENT. LTDA		93.223,30	47.988,12	47.598,43	8.872,90	25.536,49	223.219,32
> 19-REMA.COM. E REPRESENT. LTDA			65.150,42	49.555,21	24.023,94		138.729,57
> 17-RODRIGO REPRESENTACOES LTDA		60.886,97	6.514,57	33.241,32	8.562,85	1.023,14	110.228,85
> 57-TOBIAS BUNDCHEN		2.588,97	11.191,35	15.589,92	16.516,68	3.901,49	49.698,41
> 61-MARCELO STEYER		723,92	5.578,65	10.636,06	7.060,13	807,79	24.806,55
Total		3.612.432,64	3.467.948,69	4.150.578,12	4.247.959,83	1.707.323,15	17.186.242,43

Linhas

Colunas

Figura 2.6 - Exemplo de consulta com tabela de referência cruzada

O usuário pode criar sua consulta de forma bastante simples, bastando selecionar os dados que deseja visualizar da *Business Area* desejada. Assim o usuário não precisa conhecer conceitos de banco relacional ou garimpar dados em diversos locais diferentes. A ferramenta possui uma série de funcionalidades que ajudam a enriquecer a consulta, dentre elas (ORACLE, 2005b):

**Gráficos:** é possível desenvolver gráficos através das consultas criadas pelo usuário;

**Exportar dados:** pode-se exportar os dados para uma planilha em *Excel* ou nos mais diversos formatos de arquivo, como pdf<sup>11</sup>, html<sup>12</sup>, ou arquivo de texto formatado;

**Totais:** há ferramentas de totalização de dados, que incluem funções de sumariação, percentual, valor máximo e valor mínimo em um conjunto de dados;

11 Criado pela Adobe, o formato PDF (*Portable Document Format*) é uma especificação disponível publicamente usada por entidades de padronização do mundo inteiro para a distribuição e a troca mais seguras e confiáveis de documentos eletrônicos (ADOBE, 2007).

12 HTML (*HyperText Markup Language*, que significa Linguagem de Marcação de Hipertexto) é uma linguagem de marcação utilizada para produzir páginas na *Web* (WIKIPEDIA, 2007).

**Itens calculados:** o usuário pode criar itens através de cálculos que podem ser entre colunas existentes na consulta;

**Condições:** o usuário pode criar suas próprias condições para o resultado da pesquisa. Nesse componente, por exemplo, o usuário pode informar que deseja visualizar somente dados que estejam com o valor maior que o indicado por ele. É possível criar condições dinâmicas, que devem ser parametrizadas a cada re-consulta;

**Formatação de dados:** pode-se formatar a máscara, tamanho e cor da fonte dos dados;

**Exceções:** funcionalidade que permite ao usuário destacar dados dentro da consulta. Exemplo: em uma consulta de faturamento por cliente, o usuário pode indicar que deseja visualizar na cor vermelha todos aqueles clientes que possuem faturamento inferior a R\$ 10.000,00.

Com relação a análise OLAP, o *Discoverer* possui as seguinte funcionalidades:

***Drill-down e Roll-up:*** através de hierarquias criadas no *Administrator*, é possível navegar nos diversos níveis de dados;

***Slice e Dice:*** permite que as consultas possam ser visualizadas de forma particionada ou fatiada com base nas dimensões;

***Pivoting:*** permite, através de clicar com o mouse, transformar linhas em colunas e vice-versa;

***Sorts:*** o usuário pode ordenar a consulta pela informação que desejar;

**Filtros:** através dos componentes de condição e parametrização, é possível filtrar somente os dados, exibindo somente o desejado.

O usuário possui ainda condições de agendar as suas consultas, ou seja, executá-las em determinado horário para que quando ele necessite das informações, não tenha a necessidade de aguardar que as mesmas sejam buscadas do banco de dados. Além disso, há a possibilidade de salvar os *workbooks* tanto no banco de dados quanto em arquivos, que podem ser compartilhados com outros usuários (ORACLE, 2005b).

### **2.2.1.1.1 Discoverer Plus e Discoverer Viewer**

O Discoverer Plus é nada mais do que a versão Web do Discoverer Desktop. A ferramenta apresenta as mesmas funcionalidades da versão desktop, porém roda através de um *browser* (ORACLE, 2005c).

O *Discoverer Viewer* é a ferramenta que permite ao usuário apenas visualizar as consultas desenvolvidas através do *Discoverer Desktop* e *Discoverer Plus*. A *Discoverer Viewer* ainda possibilita ao usuário ajustar seus relatórios usando parâmetros específicos e permite a conversão e publicação de relatórios gerados usando diferentes formatos como HTML, PDF ou XML (ORACLE, 2005d).

### **2.2.1.2 Oracle Spreadsheet Add-In**

O *Oracle Spreadsheet Add-In* é um componente que permite ao usuário acessar os dados do OLAP Oracle através do *Microsoft Excel*. Quando o *Spreadsheet* é instalado no cliente, é gerado um item de menu dentro do *Excel* chamado "*OracleBI*". Nele, os dados podem ser selecionados e carregados dentro da planilha, permitindo ao usuário analisar e formatá-los da maneira que desejar. Para a seleção de dados não é necessário nenhum tipo de conhecimento técnico da estrutura do banco dados, pois os dados virão formatados de acordo com as *Business Areas* criadas no *Discoverer Administrator*.

O componente está disponível somente para usuários que possuam *Windows* e em banco de dados *Oracle* que possuam a opção OLAP instalada.

### **2.2.1.3 Oracle BI Beans**

O *Oracle BI Beans* é um componente integrado a ferramenta da *Oracle JDeveloper*, que é utilizada para desenvolvimento de aplicações em Java. No *BI Beans*, os desenvolvedores podem desenvolver aplicações personalizadas de *Business Intelligence*, incluindo as mais diversas funcionalidades OLAP.

*Oracle BI Beans* inclui *beans*<sup>13</sup> de apresentação (gráfico e *crosstab*), beans de dados (construtores de consulta e cálculos) e serviços de persistência, que podem ser implantados tanto em aplicações cliente HTML quanto em aplicações cliente *Java*.

---

<sup>13</sup> *Beans* são componentes reutilizáveis da plataforma Java para criação de aplicações mais sofisticadas (JAVA, 2007)

### 2.2.2 Microsoft Analysis Services

O *Microsoft Analysis Services* é a ferramenta OLAP da *Microsoft Corporation*, que vem acoplada ao banco de dados da mesma empresa, o *SQL-Server*. Oferece uma visão integrada de seus dados empresariais para relatórios, análise OLAP, *scorecards* de indicadores chave de desempenho chave (*key performance indicator - KPI*) e *data mining* (MICROSOFT, 2007). É uma ferramenta que suporta MOLAP, ROLAP, HOLAP e DOLAP. Conforme Barbieri (2001), é dividida em cinco componentes básicos:

1. *PivotTable Service*, que pode ser conectado via *Excel*, *Office 2000* e versões posteriores ou ferramentas de terceiros;
2. Métodos para acesso aos dados, baseado no padrão OLE/DB-OLAP<sup>14</sup>;
3. Máquina para serviços e armazenamento OLAP;
4. Gerenciador OLAP;
5. *DTS- Data Transformation Service* (já explicado anteriormente).

A ferramenta é dividida em quatro camadas principais: camada cliente, camada servidora, camada de gerência e acesso e camada de retaguarda.

A camada cliente contém os componentes que fazem interfaces com o usuário e podem ser as próprias planilhas *Excel* e produtos de terceiros. Um dos recursos que merece destaque nessa camada é o *PivotTable Service*, que manipula os dados e os armazena em cubos ou fatias, permitindo que as solicitações sejam feitas no próprio cliente, sem a necessidade de acesso constante ao servidor, formando um DOLAP.

Na camada servidora está a máquina de serviços OLAP, onde as requisições são recebidas e são realizados os acessos e tratamentos das mesmas. Os dados podem ser armazenados tanto em esquema em estrela (*star schema*) como em esquema *snowflake*. Um dos diferenciais da ferramenta está relacionado ao tratamento dos cubos de dados. No *Analysis Services* é possível particionar um cubo lógico em diversos cubos físicos, inclusive em diferentes servidores. Além disso, é possível criar cubos virtuais, uma espécie de *view* multidimensional, que pode ser formado pela junção de diversos cubos (BARBIERI, 2001).

---

14 O *Microsoft OLE DB para OLAP* é um conjunto de objetos e interfaces que estendem a habilidade do OLE DB para prover acesso ao armazenamento de dados multidimensionais (MICROSOFT, 2007).

A camada de gerência, chamada também de *OLAP Manager* é aquela que gerencia o ambiente OLAP, tendo interfaces para a criação de tabelas e definições de agregados. Um dos seus componentes principais é o *Cube Browser*, que permite ao usuário acessar de forma interativa os dados das tabelas, cubos e metadados. Sua linguagem padrão para acesso de dados é o MDX<sup>15</sup>.

A camada de retaguarda é o DTS (*Data Transformation Services*), ferramenta de ETL, que já foi explicada no capítulo 2.1.3, no qual é voltado apenas a ela.

### **2.2.3 Pentaho BI**

A seguir será apresentada a ferramenta *Pentaho BI*. As informações da mesma foram retiradas do seu site oficial ([www.pentaho.org](http://www.pentaho.org)).

O projeto *Pentaho BI* é uma iniciativa da comunidade de *Open Source* para fornecer às empresas uma solução que supra as necessidades de sistemas de *Business Intelligence*. Pelo fato de ser *Open Source*, inovações e novas implementações são constantes na ferramenta, agregando funcionalidades e inserindo melhorias.

Na plataforma *Pentaho BI*, o elemento central na arquitetura da *Pentaho Open BI Suite*, é centrada em processos cujo controle central é realizado através de um mecanismo de *workflow*. Este mecanismo utiliza definições de processos para determinar os processos de BI que serão executados na plataforma. Os processos podem ser personalizados e novos podem ser facilmente incorporados.

A ferramenta integra *workflow*, regras de negócios, notificação e entrega de informações, escalonamento, auditoria, integração de aplicações navegação por conteúdo, interfaces ao usuário, ferramentas de projeto e administração de relatórios, análise, *dashboards* e componentes de mineração de dados.

A arquitetura do *Pentaho* é dividida em Servidor *Pentaho*, e o *Design Studio*, baseado no *Eclipse*<sup>16</sup>.

---

15 A MDX é uma linguagem de expressão multidimensional definida na especificação OLE DB para OLAP (MICROSOFT, 2007)

16 *Eclipse* - Ferramenta de desenvolvimento de aplicações JAVA - [www.eclipse.org](http://www.eclipse.org)

O Servidor *Pentaho* roda em acordo com o padrão de servidor web *J2EE Server*, tais como *Apache*<sup>17</sup>, *JBOSS AS*<sup>18</sup>, *WebSphere*<sup>19</sup>, *WebLogic*<sup>20</sup> e *Oracle AS*<sup>21</sup>. Os conteúdos apresentados ao usuário podem ser em XML, HTML, ou mostrados na tela pelos JSR-168 *portlets*<sup>22</sup>. O servidor *Pentaho Server* fornece ainda os mecanismos e componentes para gerar relatórios, análises, regras de negócios, e-mail e fluxos de processos.

O *Pentaho Studio*, que é baseada no *Eclipse*, provê ferramentas amigáveis de projeto de relatórios, *dashboards* e visões analíticas; um processo de projeto de *workflow*; editor de regras de negócios; console de mineração de dados para preparação de dados; e ferramentas de modelagem OLAP. Essa ferramenta é multi-plataforma, desenvolvida em *Java* e é instalada na máquina dos desenvolvedores e dos administradores do sistema.

Dentre suas principais vantagens, o *Pentaho* traz uma fácil personalização de suas funcionalidades e integração com sistemas externos. Além claro, de possuir um custo bastante menor que a maioria das soluções de *Business Intelligence* existentes no mercado.

#### 2.2.4 *OpenI*

A seguir será tratada da ferramenta de *Business Intelligence OpenI*, no qual todas as informações referentes a ela foram retiradas do seu site oficial ([www.openi.org](http://www.openi.org)).

O *OpenI* é uma ferramenta de análise OLAP, *open-source*, que é disponibilizada como uma aplicação *Web J2EE*, e suporta servidores OLAP compatíveis com o padrão XMLA, incluindo o *Microsoft Analysis Services* e *Mondrian*. Possui a implementação padrão executando no servidor *Web Apache Tomcat*. Ainda pode publicar relatórios analíticos baseados na *Web* a partir de três tipos de servidores de dados: Servidores OLAP, servidores de bancos de dados relacionais e servidores de *data mining*.

O *OpenI* está dividido em três categorias principais:

**Componente de conexão:** A função dos conectores é "falar" o idioma nativo das fontes de dados de análise. Para os bancos de dados relacionais, o *OpenI* utiliza o JDBC. Para

---

17 *Apache* – [www.apache.org](http://www.apache.org)

18 *JBOSS AS* – [www.jboss.org](http://www.jboss.org)

19 *WebSphere* - [www.ibm.com/software/websphere](http://www.ibm.com/software/websphere)

20 *WebLogic*- [www.bea.com](http://www.bea.com)

21 *Oracle AS* - [www.oracle.com/appserver/index.html](http://www.oracle.com/appserver/index.html)

22 JSR-168 *Portlets* são aplicações em tecnologia *Java* baseadas em componentes *web*, gerenciado por um *portlet container* que processa requests e gera conteúdo dinâmico (OLIVEIRA, 2004).



os servidores OLAP, é utilizado o XMLA como o protocolo padrão de comunicação. Para servidores RDBMS, é utilizado o padrão ODBC. Para os serviços de data mining, o *OpenI* é integrado com a plataforma *open source* de *data mining* chamada *The R project*<sup>23</sup>.

**Componente de Relatório:** O *OpenI* utiliza linguagens específicas de definição de relatório (RDL - *Report Definition Language*) para definir e explorar os relatórios criados na plataforma. Para os relatórios em base de dados relacionais, ele utiliza a RDL .jrmxl do *JasperReports*<sup>24</sup>. Para relatórios OLAP e *data mining*, ele implementa sua própria RDL baseada em XML.

**Componente de interface com o usuário:** O componente de interface com o usuário do *OpenI* reúne outros projetos em uma única plataforma, tornando-a amigável aos usuários não-técnicos, como os analistas de negócio. São utilizados componentes dos projetos *Jpivot*<sup>25</sup> e *Jfreechart*<sup>26</sup>, unificando-os como um consistente *framework Web* de navegação.

### 2.2.5 Comparativo das ferramentas OLAP

Baseado em Klein (1999), pode-se criar o seguinte quadro com recursos importantes nas ferramentas OLAP:

Quadro 2.3 – Recursos das ferramentas OLAP

<i>Nr</i>	<i>Característica</i>	<i>Descrição</i>
1	Múltiplas camadas	Camada do usuário final, camada do administrador, camada para Web e camada de metadados
2	Interface amigável	Navegação, apresentação visual alternativa, guia para formulação de consultas etc.
3	Múltiplos usuários	Possuir uma arquitetura do tipo cliente/servidor
4	Flexibilidade de consulta	Operações de “ <i>rollup</i> ”, “ <i>drill</i> ”, “ <i>slice e dice</i> ”, “ <i>ranking</i> ”, “filtros”, “ <i>pivoting</i> ”, alinhamento de dimensões, manipulação de hierarquias assimétricas e incompletas etc.
5	Manipulação de dados não-convencionais	Capacidade de a ferramenta lidar, além dos dados convencionais (numéricos, datas e <i>strings</i> ), com dados não-convencionais no cubo.
6	Funções Matemáticas	Agregações, cálculos procedurais e com cruzamento de dimensões, precedência de fórmulas, matrizes esparsas etc.

23 *The R project* - <http://www.rproject.org>

24 *JasperReports* - <http://jasperreports.sourceforge.net>

25 *JPivot* - <http://jpivot.sourceforge.net>

26 *JfreeChart* - <http://www.jfree.org/jfreechart>

- |   |                               |   |
|---|-------------------------------|---|
| 7 | Manipulação de hierarquias    | Estabelecimento de hierarquias nas dimensões, possibilitando que um membro pai represente a consolidação de membros filhos. |
| 8 | Funcionalidades de manutenção | Gerenciamento de atualização das tabelas sumariadas   |
| 9 | Funcionalidades para WEB      | Aumento da disponibilidade do ambiente através da <i>Web</i>  |

Assim, com as ferramentas apresentadas nesse trabalho, foi possível levantar uma análise das mesmas, baseado em manuais técnicos e estudos de caso, criando-se o seguinte quadro de comparação entre elas:

Quadro 2.4 – Comparação das ferramentas OLAP

<b><i>Ferramenta/recurso</i></b>	<b><i>1</i></b>	<b><i>2</i></b>	<b><i>3</i></b>	<b><i>4</i></b>	<b><i>5</i></b>	<b><i>6</i></b>	<b><i>7</i></b>	<b><i>8</i></b>	<b><i>9</i></b>
<i>Oracle Business Intelligence</i>	X	X	X	X	X	X	X	X	X
<i>Microsoft Analysis Services</i>	X	X	X	X		X	X	X	X
<i>Pentaho</i>	X	X	X	X		X	X		X
<i>OpenI</i>		X	X	X	X		X		X

Com relação aos itens faltantes, esses não possuem nenhum tipo referências de sua existência nas documentações oficiais das nas ferramentas analisadas.

## 3 ESTUDO DE CASO

Na concepção de Simon (2006), o estudo de caso é uma técnica de estudo, em que é realizada uma pesquisa sobre um caso particular, para tirar conclusões sobre princípios gerais daquele caso específico.

Nesse capítulo será realizado um estudo de caso da aplicação de um sistema de *Business Intelligence* para a construção de um *Data Mart* comercial na empresa Box Print, utilizando ferramentas *Oracle*.

O capítulo passará por uma explicação da empresa em que o projeto será aplicado, sua estrutura, forma de organização e como a tecnologia está inserida nela. A seguir será mostrado o cenário atual das informações disponíveis para a tomada de decisão, quais suas deficiências e problemas para a sua busca e análise.

Após, será tratado de forma mais ampla o porquê da utilização de tecnologias *Oracle*. E por fim, será mostrada uma montagem de como a solução a ser aplicada funcionará, através de uma arquitetura do sistema e de um modelo dos dados a serem incorporados.

### **3.1 Estrutura da Empresa**

A Box Print é uma empresa do ramo de embalagens, que fabrica produtos na área de cartões diversos, microondulados e ondulados, para os setores de calçados, eletrodomésticos, farmacêuticos, gêneros alimentícios, congelados, autopeças, cutelaria, cosméticos, perfumaria e outros. Atende, além do mercado interno, clientes localizados na América Latina e Estados Unidos.

A empresa possui quatro unidades fabris, todas localizadas no Rio Grande do Sul, e um escritório central de vendas, localizado na cidade de São Paulo-SP. A unidade matriz

localiza-se em Campo Bom-RS, e é nesse local que a estrutura principal de tecnologia está instalada.

### 3.1.1 Tecnologia

Todos os processos da empresa são informatizados. São mais de oitocentos funcionários que lidam diariamente com os sistemas desenvolvidos pelo setor de TI da empresa, que possui uma equipe de oito pessoas.

A plataforma padrão de aplicação e desenvolvimento de sistemas é *Oracle*. Ela está inserida desde os SGBD's até as aplicações para o usuário final. A empresa possui um total de seis SGBD's que possuem as seguintes funções:

**Banco de produção:** atende aos diversos sistemas transacionais da empresa. Dentre eles pode-se destacar o sistema comercial, compras, administrativo, controle de estoque, qualidade e controle e planejamento de produção. Atualmente possui a versão 9i do SGBD Oracle

**Banco de desenvolvimento:** banco de dados utilizado pelo setor de TI para o desenvolvimento de aplicações e novos sistemas. É uma cópia do banco de produção, que é atualizada sempre que necessário. Possui a versão 10g do SGBD Oracle.

**Banco de modelagem:** possui o repositório dos modelos ER desenvolvidos através da ferramenta *Oracle Designer*, dos sistemas transacionais da empresa. Possui a versão 10g do SGBD Oracle.

**Sistema de clientes:** possui os dados a serem acessados através do sistema *BoxNet*, utilizado por clientes e representantes comerciais para consultar informações de pedidos e notas fiscais. Possui a versão 10g do SGBD *Oracle*.

**Site Corporativo e Intranet:** banco de dados da ferramenta *Oracle Portal*, no qual é utilizado para armazenar dados do sistema de *intranet* e do site da empresa ([www.boxprint.com.br](http://www.boxprint.com.br)). Possui a versão 10g do SGBD *Oracle*.

**Sistema de representantes:** banco de dados com informações relacionadas ao sistema utilizado pelos representantes da empresa para enviar e receber orçamentos e pedidos. Esta máquina intermedia a comunicação entre os computadores utilizados pelos

representantes e o banco de produção, contendo informações do que é enviado ou recebido por ambas as pontas existentes no sistema. Cabe ainda, ressaltar que o sistema utilizado pelos representantes possui um SGBD Oracle 8i instalado em suas respectivas máquinas e as aplicações também são desenvolvidas com tecnologia *Oracle*.

### 3.1.2 Estrutura Organizacional

A empresa possui no seu maior cargo o diretor-presidente da empresa, seguido pelos diretores de suas respectivas áreas, passando pelos gerentes, que por fim gerenciam setores específicos dentro da organização. A figura abaixo pode demonstrar resumidamente a estrutura organizacional da empresa Box Print, com detalhamento no setor de TI.

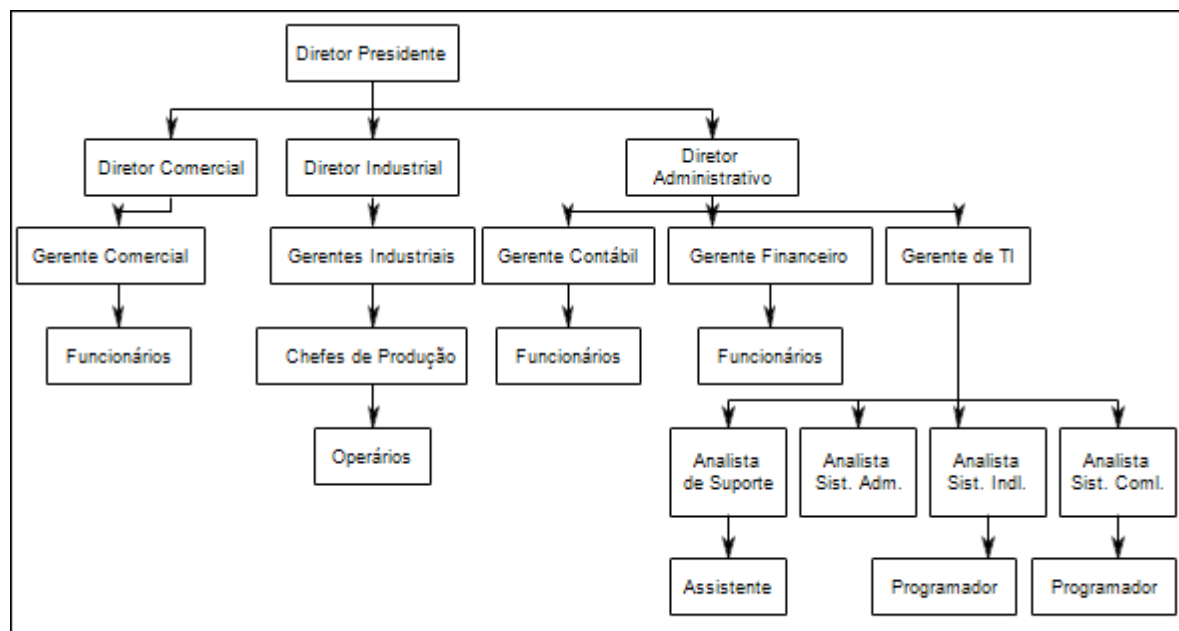


Figura 3.1 - Estrutura Organizacional da Empresa

O setor comercial, que é o escopo do trabalho, possui uma estrutura organizacional que inicia no seu cargo principal com o diretor comercial. Ele tem em uma de suas responsabilidades coordenar o trabalho dos administradores de venda. Estes por sua vez, são os responsáveis em gerenciar um grupo determinado de representantes de venda, que possui clientes específicos a serem atendidos. Essa estrutura pode ser resumidamente representada na figura 3.2.

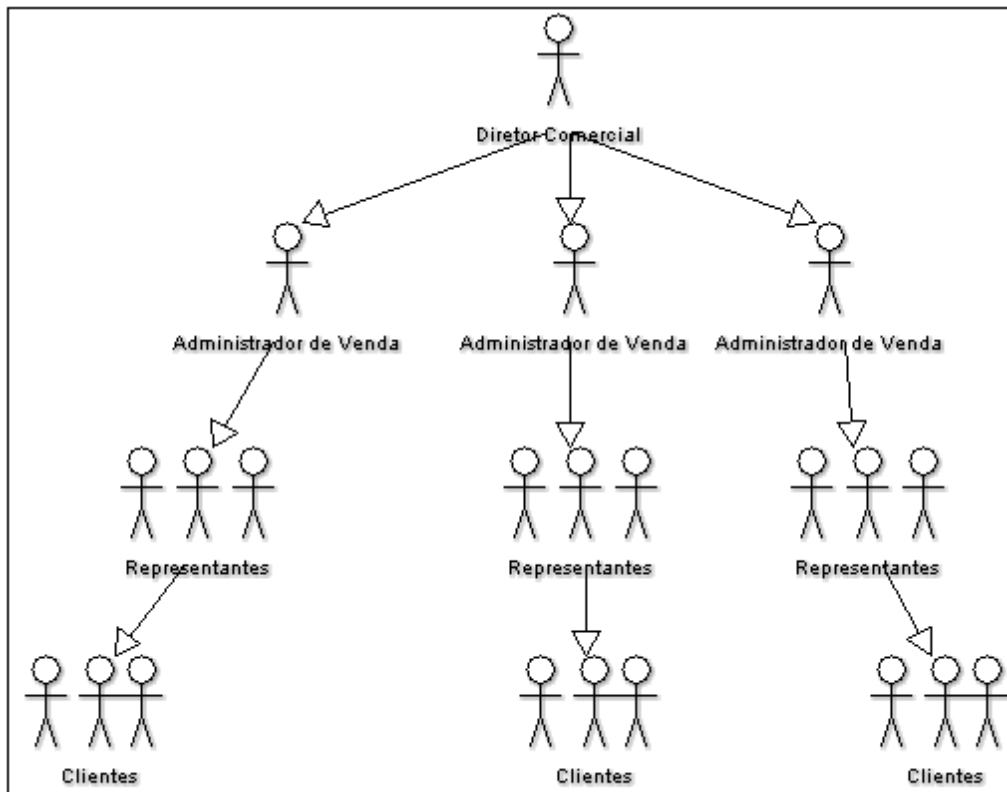


Figura 3.2 - Organização do gerenciamento de vendas

### 3.2 Problemas Atuais e Necessidades

A tecnologia *Oracle* foi incorporada na Box Print no ano de 2000, através de uma migração de um antigo sistema transacional desenvolvido em *Cobol*<sup>1</sup>. Desde então, a base de dados desse sistema está se tornando cada vez mais sólida, com um volume de dados cada vez maior, tornando-se assim uma fonte muito rica para as mais diversas requisições e, por conseqüência, ambiente adequado para aplicação de tecnologias que buscam informação e conhecimento..

Porém a empresa não possui atualmente, um sistema que reúna esse grande volume de dados para fins de análise gerencial que auxiliem a tomada de decisão. O que existe são diversos relatórios e consultas, desenvolvidos para fins específicos, através das ferramentas *Oracle Forms Builder* (desenvolvimento de programas) e *Oracle Reports Builder*. (desenvolvimento de relatórios). Essas aplicações estão disponíveis através dos sistemas transacionais da empresa.

1 COBOL - Linguagem originada de um aprimoramento do ARGOL, criada na década de 70 e batizada em homenagem a Blaise Pascal (WIKIPEDIA, 2007). Site Oficial: <http://www.cobolportal.com>

Porém, pelo fato de serem aplicações que possibilitam a análise gerencial e são executadas através do banco de dados transacional, acabam tendo uma geração dos resultados para o usuário de forma demorada, consumindo muito recurso do servidor, prejudicando muitas vezes o desempenho normal do mesmo.

Muitas vezes, essas formatações de informações existentes atualmente na empresa não atendem totalmente as necessidades do usuário. Com isso, muitas vezes o gerente ou diretor acaba designando um funcionário para garimpar, dentro do sistema transacional, dados de várias consultas e relatórios, implicando em uma dedicação e tempo muito grandes. Além disso, há sempre a probabilidade de que as informações possam não ser confiáveis, uma vez que o usuário, que não possui uma visão gerencial, pode cometer o erro de adicionar ou até mesmo retirar dados que seriam importantes para a tomada de decisão.

E essa realidade está localizada em todos os setores da empresa, inclusive no setor comercial. É exatamente nessa área que este trabalho de desenvolvimento de um sistema de *Business Intelligence* será realizado. Essa escolha foi tomada baseando-se no fato de que apesar de a empresa possuir relatórios gerenciais nos diversos setores, o setor comercial possui um número mínimo desse tipo de informação e seria um ponto inicial para um futuro sistema gerencial totalmente integrado, envolvendo todas as áreas da empresa. Além do que, esse setor é extremamente estratégico para a empresa, pois através de uma análise gerencial, possibilitará um mapeamento do desempenho de administradores de venda e representantes.

### **3.3 Proposta do sistema**

Para atender as necessidades de análise de informações gerenciais, surge a proposta de desenvolver um sistema de *Business Intelligence*, através das ferramentas da *Oracle* já explicadas anteriormente.

#### **3.3.1 Dados a serem utilizados**

A proposta é utilizar como fonte de dados o atual sistema transacional da empresa, e através das funcionalidades da ferramenta *Oracle Warehouse Builder*, construir a modelagem dimensional para um *Data Mart* comercial e um processo de carga que atualizá o repositório DM através desses dados transacionais.

Os dados a serem incorporados no DM serão aqueles provenientes do módulo de faturamento e do módulo de devolução. Para cada um deles será criado um modelo dimensional, através do conceito de *star schema* e que podem ser cruzados para análises mais complexas de informação. Na figura 3.3, está representado graficamente uma idéia inicial de como será a modelagem dimensional do sistema, com as tabelas de fatos “faturamento” e “devoluções”.

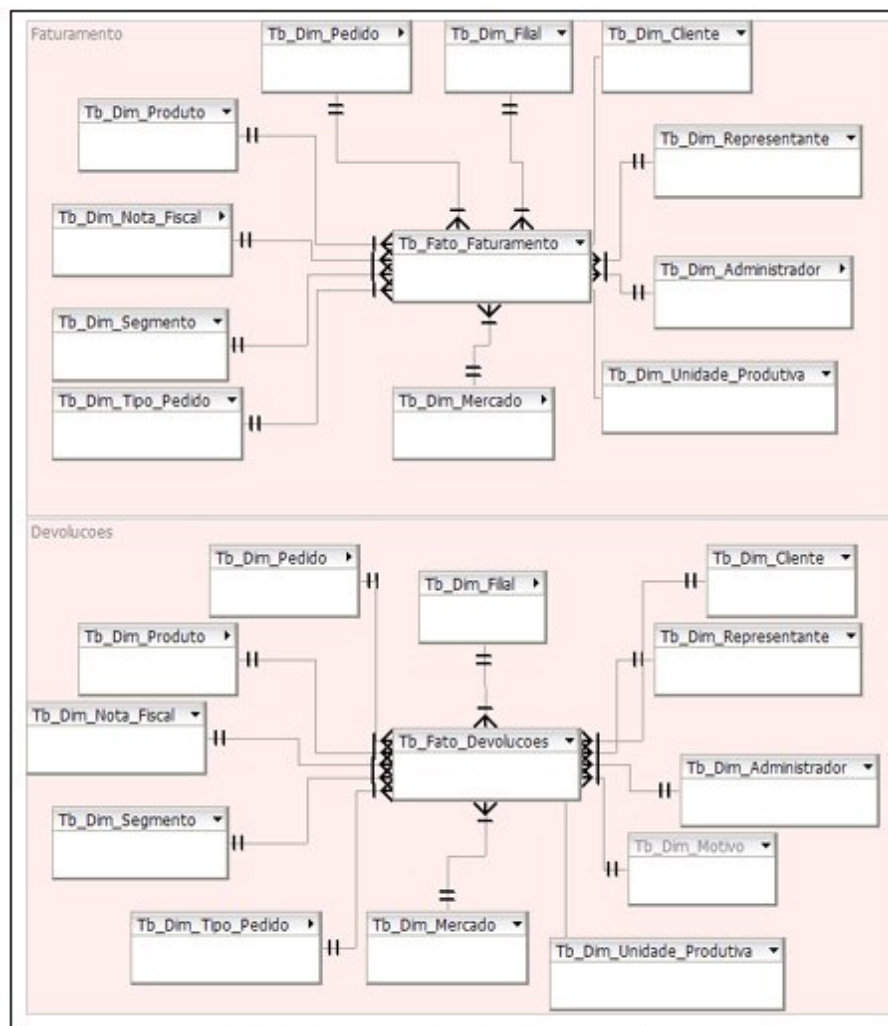


Figura 3.3 - Modelagem Dimensional

Como pode ser observado na figura 3.3, dentre os dados existentes estão as tabelas de administradores, representantes, clientes, filiais, produtos, notas fiscais e pedidos. É baseado nesses dados que será desenvolvida a hierarquia principal do sistema, com o administrador coordenando os representantes, que atendem determinados clientes e filiais, que



possuem seus produtos os quais serão faturados através de pedidos. Ela pode ser representada graficamente na figura 3.4.

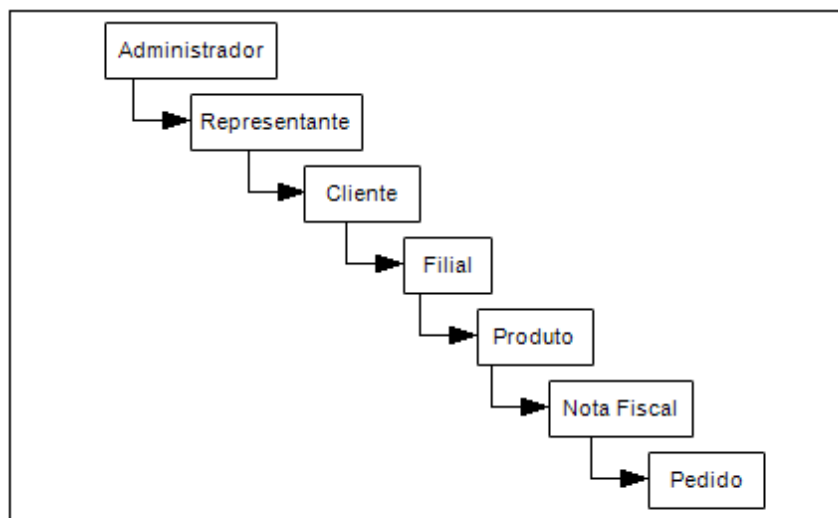


Figura 3.4 - Hierarquia Principal dos dados do sistema

### 3.3.2 Interface gráfica e funcionalidades

Inicialmente, o sistema estará dividido em dois perfis de usuários, diretor comercial e administradores de venda. Cada um deles acessará informações e telas específicas. O diretor comercial terá a possibilidade de consultar os dados em sua totalidade, ou seja, poderá gerenciar todos os administradores de venda, passando pelos representantes, seguindo a hierarquia já explicada anteriormente. Os administradores de venda visualizarão unicamente os dados da sua equipe, ou seja, seus representantes de venda e respectivas informações.

O acesso será feito através da *intranet* da empresa, sistema *web* desenvolvido através da ferramenta *Oracle Portal*, que já possui uma política de conteúdo para cada usuário ou grupo de usuários. A idéia é de que ao efetuar o *login* na *intranet*, o *link* do sistema de BI fique habilitado, para que, a partir deste ponto, o usuário faça acesso ao mesmo.

Ao entrar no sistema, cada usuário terá uma tela inicial com os dados mais relevantes para ele. Por exemplo, o administrador de vendas terá em sua tela inicial um resumo do faturamento de seus representantes ou ainda um *ranking* dos cinco maiores clientes de sua competência. Além disso, o usuário terá um menu personalizado para o seu perfil, com variados formatos de consultas de faturamento, devoluções ou cruzamento de ambos.

Essas consultas serão desenvolvidas e visualizadas através da ferramenta *Oracle Discoverer*, já explicada anteriormente. Além das funcionalidades de OLAP que essa ferramenta possui, como o *drill-down*, *roll-up*, *pivoting*, entre outras, o sistema trará as seguintes funcionalidades para o usuário:

1. Imprimir as consultas;
2. Salvar as consultas em um arquivo de planilha de cálculo ou HTML;
3. Gerar gráficos;
4. Enviar as consultas por *e-mail*;

À medida que os usuários forem se ambientando com o sistema, e principalmente com as funcionalidades do *Discoverer*, uma das possibilidades que surge é de que os mesmos possam vir a desenvolver e formatar as suas consultas de acordo com as suas necessidades, podendo abri-las dentro do próprio ambiente do sistema ou compartilhar com outros usuários.

### **3.4 Arquitetura do sistema**

Conforme já explicado anteriormente, os dados serão carregados dentro de um repositório de dados, provenientes do sistema transacional da empresa. Esse repositório será visualizado através de consultas feitas em *Oracle Discoverer*, e publicadas em um ambiente *web* através do *Oracle Portal*. Esse esquema pode ser visualizado graficamente através da figura 3.5

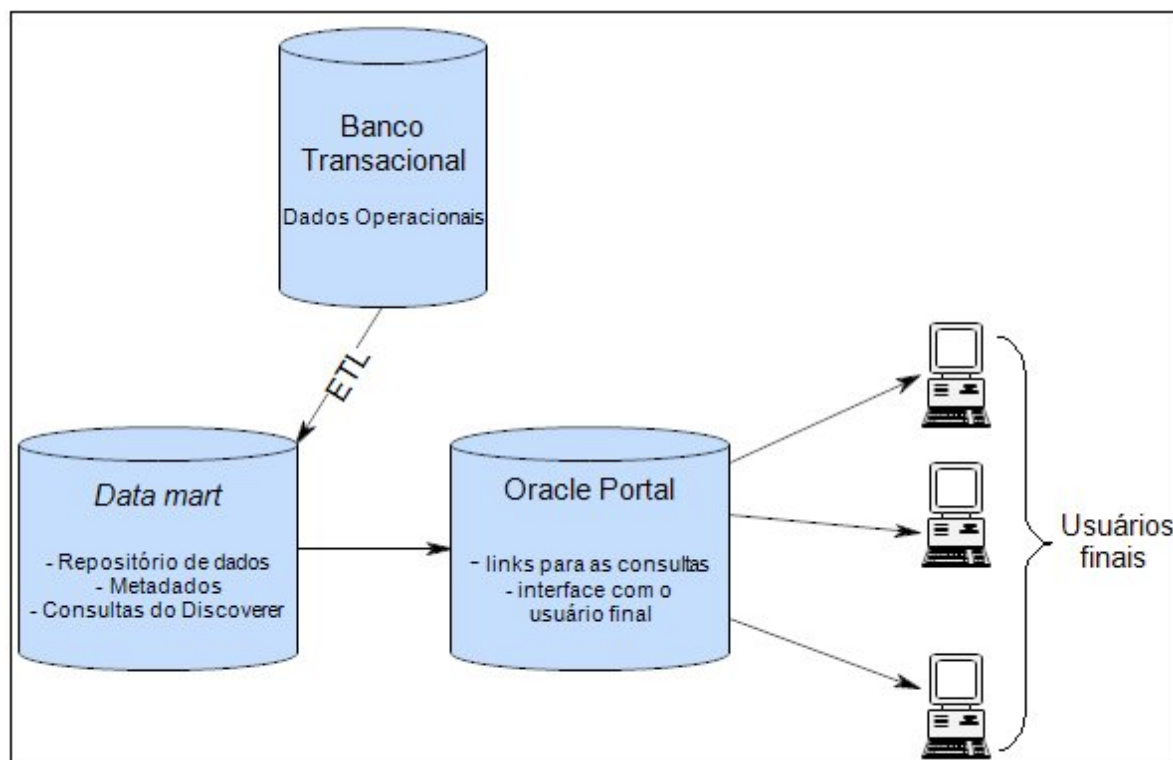


Figura 3.5 - Estrutura do Sistema

### 3.5 Oracle

Nesse capítulo será tratado um pouco da *Oracle*, ou seja, que empresa é essa, e o que ela traz de vantagens para os sistemas de informação e para o esse projeto. Todas as informações relacionadas a empresa e sobre seus produtos foram retiradas do site oficial da empresa ([www.oracle.com](http://www.oracle.com)).

#### 3.5.1 Por que Oracle

A escolha de utilizar ferramentas *Oracle* para o desenvolvimento do sistema deu-se ao fato de que essa tecnologia é a padrão da empresa em que o projeto será implementado. Pois já é uma tecnologia aprovada e homologada dentro da empresa, e a continuação da sua utilização nesse projeto foi dada como natural em função dessa aprovação.

Pois, como as bases de dados existentes são *Oracle*, é possível ter uma integração extremamente mais segura e com muito mais recursos disponíveis quando as ferramentas que processam as informações dessas bases são desenvolvidas pela mesma empresa.

Acredita-se que a utilização de ferramentas de outros fabricantes para as tarefas criação da estrutura de dados até a sua visualização por parte do usuário traria menos opções de funcionalidades, sem contar que a integração com tecnologias diferentes é sempre um

obstáculo a mais no desenvolvimento de um projeto do porte desse que está sendo proposto nesse trabalho.

Além do mais, pelo simples fato de *Oracle* ser uma tecnologia paga, há um serviço de suporte técnico muito eficaz, pois caso surja alguma dificuldade com uso da ferramenta ou problemas nas mesmas, o atendimento ao cliente sempre traz soluções que acabam auxiliando na resolução dessas dificuldades e problemas.

## CONCLUSÃO

Através de tudo que foi apresentado nessa primeira etapa do trabalho de conclusão, pode-se perceber que a Box Print é uma empresa com grande volume de dados, tendo assim condições de obter melhores condições para as tomadas de decisão, mas que não possui um método eficaz para que isso seja efetivado.

Baseado nisso, esse trabalho mostrou o conceito de *Business Intelligence* e suas tecnologias envolvidas, buscando apresentar uma solução para que as dificuldades em obter informações gerenciais sejam sanadas.

Dentro de BI, foram exploradas principalmente as tecnologias de *Data Warehouse* e OLAP e através delas, foi apresentado um embasamento teórico desses conceitos, mostrando de que maneira podem ser usados no desenvolvimento de um sistema para tomada de decisão.

Foi realizado ainda, um levantamento das principais ferramentas existentes no mercado que se enquadram no conceito de *Business Intelligence*. Dentre elas, as escolhidas foram as desenvolvidas pela *Oracle*, por além de apresentarem inúmeros recursos, fazem parte da tecnologia padrão da empresa em que o projeto será implementado.

Nessa empresa, já foi realizada a análise de requisitos inicial, e através dela, o projeto de BI será realizado utilizando os dados competentes à área comercial, por se tratar da área com maior carência de informações gerenciais.

Assim, a continuação desse projeto, passando pela modelagem completa dos dados, desenvolvimento da solução e sua implantação serão tarefas que, conforme o cronograma previsto, estarão no trabalho de conclusão II.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ADOBE. *Adobe PDF*. 2007. Disponível em [www.adobe.com/br/products/acrobat/adobepdf.html](http://www.adobe.com/br/products/acrobat/adobepdf.html). Acesso em 10 jun. 2007.
- ANZANELLO, Cynthia Aurora. **OLAP: conceitos e utilização**. Porto Alegre: 2002. 7p. Artigo - Instituto de Informática, UFRGS. Disponível em: [http://www.inf.ufrgs.br/~clesio/cmp151/cmp15120021/artigo\\_cynthia.pdf](http://www.inf.ufrgs.br/~clesio/cmp151/cmp15120021/artigo_cynthia.pdf). Acesso em 02 mai. 2007.
- APACHE. *Apache Software Foundation*. Disponível em [www.apache.org](http://www.apache.org). Acesso em 10 jun. 2007.
- AUDY, Rejane Blomberg; ENDRES, Ana Cristina Trois; MALVEZZI, Maria Luiza Falsarella. **Case BI - Informações Gerenciais em Hospital de Referência**. Porto Alegre: 2003. 11p. Artigo – Hospital de Clínicas de Porto Alegre. Disponível em <http://inf.unisul.br/~ines/workcomp/cd/pdfs/2420.pdf>. Acesso em 15 mar. 2007
- BARBALHO, Patrícia. Descubra o *Data Warehouse*: produtividade e rapidez. **Revista SQL Magazine**. Rio de Janeiro: nº 3, ano 1, p. 34 – 38, 2003
- BARBIERI, Carlos. **Business Intelligence: modelagem e tecnologia**. Rio de Janeiro: Axcel Books, 2001. 424 p.
- BERVIAN, Andreia Eliana. **Crítérios para a decisão de personalização de processos na implantação de Sistemas ERP**. São Leopoldo: 2004. 69 p. Trabalho de conclusão de curso apresentado como requisito parcial para a obtenção do título de Bacharel em Informática- Universidade do Vale do Rio dos Sinos. Disponível em [http://inf.unisinos.br/alunos/arquivos/TC\\_AndreiaBervian.pdf](http://inf.unisinos.br/alunos/arquivos/TC_AndreiaBervian.pdf). Acesso em 10 jun. 2007

BEVILACQUA, J. F.; BITU Y. A. **Business Intelligence (BI) e a abordagem de Gestão Balanced Scorecard (BSC) na Organização**. Brasília: 2003. 54p. Monografia - Pós Graduação MBA em Sistemas de Informação – Universidade Católica de Brasília. Disponível em [http://www.eln.gov.br/Conhecimento/GestaodoConhecimento/Monografia\\_MBA\\_180703.pdf](http://www.eln.gov.br/Conhecimento/GestaodoConhecimento/Monografia_MBA_180703.pdf). Acesso em 15 mar. 2007

BISPO, Carlos Alberto Ferreira. **Uma análise da nova geração de sistemas de apoio à decisão**. São Carlos: 1998. 160 p. Dissertação de Mestrado - Escola de Engenharia de São Carlos, Universidade de São Paulo. Disponível em: [http://www.teses.usp.br/teses/disponiveis/18/18140/tde-04042004-152849/publico/dissertacao\\_carlos.pdf](http://www.teses.usp.br/teses/disponiveis/18/18140/tde-04042004-152849/publico/dissertacao_carlos.pdf). Acesso em: 10 mar. 2007.

BOGHI, C.; SHITSUKA R.. **Sistemas de Informação: Um Infoque Dinâmico**. São Paulo-SP: Érica, 2002. 288p.

CARICKHOFF, Rich. **A New Face For OLAP: OLAP on the Web Shows Almost the Same Flexibility as Its Client/Server Counterpart**. 1997. <http://www.dbmsmag.com/9701i08.html>. Acesso em 04 mai. 2007.

CARLSSON, C.; TURBAN, E. **DSS: directions for the next decade**. *Decision Support Systems*, v. 33, n. 2, p. 105-110, 2002.

CIELO, Ivã. **Arquiteturas OLAP**. 2000. Disponível em <http://www.datawarehouse.inf.br/artigos/olap2.asp>. Acesso em 04 mai. 2007.

COMUNIQUE-SE. **Comunique-se – O Portal da comunicação**. 2007. Disponível em [http://www.comunique-se.com.br/produtos/saladeimprensa/oracle/show.asp?\\_mat=32295&\\_ed=602&\\_tar=PK&\\_sec=om](http://www.comunique-se.com.br/produtos/saladeimprensa/oracle/show.asp?_mat=32295&_ed=602&_tar=PK&_sec=om). Acesso em 15 jul. 2007

COSTA, André Fernandes; ANCIÃES, Felipe Curvello. **Aspectos de criação e carga de um ambiente de Data Warehouse**. Rio de Janeiro: 2001. 111 p. Projeto de Diplomação (Bacharelado em Informática) – Departamento de Ciência da Computação, Universidade Federal do Rio de Janeiro. Disponível em: <http://dataware.nce.ufrj.br:8080/dataware/publicacoes/dataware/fisico/projetosFinais/datawarehousing/COSTA-ANCIÃES-2001.pdf>. Acesso em: 19 abr. 2007.

- DWBRASIL. **Histórico do Data Warehouse**. 2003. Disponível em [http://www.dwbrasil.com.br/html/artdw\\_hist.html](http://www.dwbrasil.com.br/html/artdw_hist.html). Acesso em 19 abr. 2007.
- ECLIPSE. **Eclipse Foundation**. Disponível em: <http://www.eclipse.org>. Acesso em: 10 jun 2007.
- FERNANDES, Sérgio Antônio de Sousa. **O projeto de Data Warehouses: A tecnologia ROLAP versus MOLAP**. Lisboa-Portugal: 2004. 13 p. Artigo – Instituto Superior Técnico – Lisboa – Portugal. Disponível em: [http://berlin.inesc-id.pt/cadeiras/pfsi/PFSI2003/SEMINARIO/pdfs/Data\\_Warehouse-Sergio\\_Fernandes.pdf](http://berlin.inesc-id.pt/cadeiras/pfsi/PFSI2003/SEMINARIO/pdfs/Data_Warehouse-Sergio_Fernandes.pdf). Acesso em: 29 abr. 2007.
- FIGUEIREDO, Adriana Maria C.M. MOLAP x ROLAP: Embate de Tecnologias para Data warehouse. **Developer's Magazine**. Rio de Janeiro: nº 18 p. 24, 1998
- FORTULAN, Marcos Roberto. Filho; Eduardo Vila Gonçalves. **Uma Proposta de Aplicação de um Business Intelligence no Chão-de-Fábrica**. São Carlos-SP: 2005. 12p. Artigo – G & P: Gestão e Produção. Disponível em: [www.scielo.br/pdf/gp/v12n1/a06v12n1.pdf](http://www.scielo.br/pdf/gp/v12n1/a06v12n1.pdf) . Acesso em 01 mai. 2007
- GENTIA SOFTWARE. **OLAP for enterprise**. Disponível em [www.gentia.com/products/gseolap.exe](http://www.gentia.com/products/gseolap.exe). Acesso em 11 mar. 2007
- GERBER C., **Excavate your data. Datamation**. 1998. Disponível em <http://itmanagement.earthweb.com/PlugIn/workbench/datamine/stories/excav.htm>. Acesso em 04 mai. 2007.
- GONÇALVES, Marcio Elias. **Uma Ferramenta de Extração de Dados para Data Warehouse Baseada em Agentes Distribuídos**. Florianópolis: 2002. 112 p. Dissertação de Mestrado - Universidade Federal de Santa Catarina. Disponível em <http://www.datainfo.inf.br/marcio/download/artigos/mestrado.pdf>. Acesso em 06 mai. 2007.
- GORLA, Nasimhaiah. **Features to consider in a Data Warehouse System**. 2003. Disponível em <http://portal.acm.org/citation.cfm?id=948389>. Acesso em 04 mai. 2007
- GRAY, Paul, WATSON, Hugh J. **Decision Support in the Data Warehouse**. New Jersey: Prentice Hall PTR, 1998. 416 p.



GRAY, P.; WATSON, H. J. *The new DSS: data warehouses, OLAP, MDD and KDD*. 1999. Disponível em: <<http://hsb.baylor.edu/ramsover/ais.ac.96/papers/graywats.htm>>. Acesso em: 01 mai. 2007.

HARRISON, Thomas H. *Intranet Data Warehouse*. Tradução: Daniel Vieira. São Paulo: Berkeley Brasil, 1998. 359 p.

HOKAMA, Daniele Del Bianco et al. **A modelagem de dados no ambiente *Data Warehouse***. São Paulo: 2004. 121 p. Projeto de Diplomação (Bacharelado em Sistemas de Informação) – Faculdade de Computação e Informática, Universidade Presbiteriana Mackenzie. Disponível em: <<http://meusite.mackenzie.com.br/rogerio/tgi/2004ModelagemDW.pdf>>. Acesso em: 29 abr. 2007.

HYPERION SOLUTIONS, SUN MICROSYSTEMS. *Java™ OLAP Interface (JOLAP), Proposed Final Draft*, 2003. Disponível em: <http://jcp.org/aboutJava/communityprocess/first/jsr069/index.html>. Acesso em: 04 Mai 2007.

INMON, Willian H. *Building the Data Warehouse*. New York: John Wiley & Sons. 1996. 401 p.

INMON, Willian H. **Como construir um *Data Warehouse***. Tradução da Segunda Edição. Rio de Janeiro-RJ: Campus, 1997. 388 p.

INMON, W.H, WELCH, J. D., GLASSEY, K. L. **Gerenciando *Data Warehouse***. São Paulo: Makron Books, 1999. 375p.

JAVA. *Java Technology*. Disponível em: <http://java.sun.com>. Acesso em: 10 jun. 2007.

JBOSS. *JBOSS Community driven*. Disponível em [www.jboss.org](http://www.jboss.org). Acesso em 10 jun. 2007.

JCP. *Java Community Process*. Disponível em: <http://www.jcp.org>. Acesso em: 10 jun. 2007.

JFREECHART. *A free Java chart library*. Disponível em: <http://www.jfree.org/jfreechart>. Acesso em: 10 jun. 2005.

JPIVOT. *A JSP based OLAP*. Disponível em: <http://jpivot.sourceforge.net>. Acesso em 10 jun. 2007.

KIMBALL, Ralph. **Data Warehouse Toolkit**. São Paulo: Makron Books, 1996. 388 p.

KIMBALL, Ralph, REEVES, Laura, ROSS, Margy, THORNTHWAITE, Warren. **The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing and Deploying Data Warehouses**. New York: John Wiley & Sons, Inc., 1998. 771 p.

KIMBALL, Ralph, ROSS, Margy. **The Data Warehouse Toolkit (Segunda Edição)**. Rio de Janeiro: Editora Campus, 2002. 494 p.

KLEIN, Cc LAWRENCE ZORDAM. **A Tecnologia Relacional - Objeto em um Ambiente de Data Warehouse**. Rio de Janeiro: 1999. 191 p. Tese Submetida como Requisito Parcial para a Obtenção do Grau de Mestre em Ciências em Sistemas e Computação - Instituto Militar de Engenharia. Disponível em <http://dataware.nce.ufrj.br:8080/dataware/publicacoes/dataware/fisico/teses/datawarehousing/KLEIN-1999.pdf>. Acesso em 15/05/2007

LAUDON, K. C; LAUDON, J. P. **Sistemas de Informação com internet**. 4. ed. Rio de Janeiro: LTC, 1999. 408 p.

LOBO, Adilton. **Modelagem de um estudo de caso utilizando a ferramenta entidade/relacionamento, baseada no modelo de Peter Chen**. Joinville, SC: 1998. Disponível em: <http://pages.udesc.br/~r4al/artentre.htm>. Acesso em: 06 jun. 2007.

LOPES, Alison Zille. **As tendências e preocupações da Tecnologia da Informação no Brasil**. Lavras – MG: 2006. 68 p. Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para a obtenção do título de Bacharel em Ciência da Computação - Departamento de Ciência da Computação da Universidade Federal de Lavras. Disponível em [http://www.comp.ufla.br/monografias/ano2006/As\\_tendencias\\_e\\_preocupacoes\\_da\\_tecnologia\\_da\\_informacao\\_no\\_Brasil.pdf](http://www.comp.ufla.br/monografias/ano2006/As_tendencias_e_preocupacoes_da_tecnologia_da_informacao_no_Brasil.pdf). Acesso em 27 abr. 2007

MACHADO, Felipe N. R. **Projeto de Data Warehouse: uma visão multidimensional**. São Paulo: Érica, 2000. 251 p.

MARQUES, Valmir Ferreira. **Analisando os Dados de Melhoramento Genético da Raça Nelore com Data Warehousing e Data Mining**. São Carlos-SP: 2002. 134 p. Dissertação de

Mestrado na Área de Ciências da Computação e Matemática Computacional – Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo. Disponível em <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-28022003-161537/publico/Dissertacao-Valmir.pdf>. Acesso em 06 mai. 2007.

MEIRA, Carlos Alberto Alves; SOUZA, Tatiana Aparecida Lima de. *Oracle Discoverer: Guia do Usuário*. 2002. Campinas-SP: 2002. 54 p. Ministério da Agricultura, Pecuária e Abastecimento. Disponível em <http://www.cnptia.embrapa.br/modules/tinycontent3/content/2002/doc21.pdf>. Acesso em 15 mar. 2007.

MEREDITH, Mary E; KHADER Aslam. *Divide and Aggregate: desingning large warehouses*. 1996. Disponível em <http://www.dbpd.com/vault/khader.htm>. Acesso em 01 mai. 2007

MICROSOFT. *Analysis Services*. 2005. Disponível em <http://www.microsoft.com/brasil/sql/technologies/analysis/default.mspx>. Acesso em 08 mai. 2007.

MICROSOFT. *Microsoft Corporation*. 2007. Disponível em [www.microsoft.com](http://www.microsoft.com). Acesso em 10 jun. 2007.

MONDRIAN. *Pentaho Analysis Services: Mondrian Project*. 2007. Disponível em <http://mondrian.pentaho.org/>. Acesso em 14 mai. 2007

OPENI. *OpenI*. 2005. Disponível em <http://portuguese.osstrans.net/software/openi.html>, Acesso em 08 mai. 2007.

OLIVEIRA, Eric C M. *Tecnologia de Portais Java e a JSR 168*. 2004. Disponível em [http://www.linhadecodigo.com.br/artigos.asp?id\\_ac=494](http://www.linhadecodigo.com.br/artigos.asp?id_ac=494). Acesso em 10 jun. 2007.

ORACLE. *Oracle® Business Intelligence Discoverer Desktop - User's Guide - 10g Release 2 (10.1.2.1) for Windows*. 2005. Disponível em [http://download-east.oracle.com/docs/pdf/B13917\\_03.pdf](http://download-east.oracle.com/docs/pdf/B13917_03.pdf), Acesso em 12 mai. 2007

ORACLE. *Oracle® Business Intelligence Discoverer Administrator - User's Guide - 10g Release 2 (10.1.2.1) for Windows*. 2005. Disponível em [http://download-east.oracle.com/docs/pdf/B13917\\_04.pdf](http://download-east.oracle.com/docs/pdf/B13917_04.pdf), Acesso em 12 mai. 2007

ORACLE. *Oracle® Business Intelligence Discoverer Plus - User's Guide - 10g Release 2 (10.1.2.1) for Windows*. 2005. Disponível em [http://download-east.oracle.com/docs/pdf/B13915\\_04.pdf](http://download-east.oracle.com/docs/pdf/B13915_04.pdf), Acesso em 12 mai. 2007

ORACLE. *Oracle® Business Intelligence Discoverer Viewer - User's Guide - 10g Release 2 (10.1.2.1) for Windows*. 2005. Disponível em [http://download-east.oracle.com/docs/pdf/B13987\\_04.pdf](http://download-east.oracle.com/docs/pdf/B13987_04.pdf), Acesso em 12 mai. 2007

ORACLE. *Oracle® Warehouse Builder - User's Guide - 10 g Release 1 (10.1)*. 2003. Disponível em [http://download-east.oracle.com/docs/pdf/B12146\\_01.pdf](http://download-east.oracle.com/docs/pdf/B12146_01.pdf), Acesso em 19 mar. 2007

ORACLE. *Oracle apresenta nova versão de ferramenta de design de banco de dados*. 2006. Disponível em [http://www.oracle.com/global/br/corporate/press/2006\\_oct/oracle\\_nova\\_ferramenta\\_banco\\_dados.html](http://www.oracle.com/global/br/corporate/press/2006_oct/oracle_nova_ferramenta_banco_dados.html). Acesso em 23 mar. 2007

ORACLE. *Oracle Corporation*. 2007. Disponível em [www.oracle.com](http://www.oracle.com). Acesso em 29 mai. 2007.

ORACLE AS. *Oracle Application Server*. Disponível em [www.oracle.com/appserver/index.html](http://www.oracle.com/appserver/index.html) . Acesso em 10 jun. 2007.

ORACLE BI. *Oracle Business Intelligence*. 2006. Disponível em <http://www.oracle.com/bi/index.html>. Acesso em 23 mar. 2007.

PENTAHO. *Open Source Business Intelligence*. Disponível em: <http://www.pentaho.org>. Acesso em: 18 mai. 2007

PETERSON et al. *Microsoft Olap Unleashed*. Sams Publishing, 1999. 949p.

POE, Vidette, KLAUER, Patricia, BROBST, Stephen. *Building a Data Warehouse for Decision Support*. New Jersey. Prentice-Hall, Inc, 1998. 285 p.

POLONI, E. G. F. **Administrando sistemas de informação**. São Paulo: Futura, 2000. 272 p.

REZENDE, D. A. **Engenharia de software e sistemas de informação**. RJ: Brasport, 1999. 344 p.

SANTOS, Rodrigo Gonçalves. **Utilização de técnicas de Data Mining na Busca de conhecimento na web**. Pelotas-RS: 2000. 119 p. Monografia apresentada ao Curso de Bacharelado em Informática do Instituto de Física e Matemática da Universidade Federal de Pelotas, como requisito parcial à obtenção do título de Bacharel em Informática - Instituto de Física e Matemática da Universidade Federal de Pelotas. Disponível em <http://www.ufpel.edu.br/prg/sisbi/bibct/acervo/info/2000/Mono-RodrigoSantos.pdf>. Acesso em 02 mai. 2007.

SERRA, Laércio. **A essência do Business Intelligence**. São Paulo: Berkeley, 2002. 296 p.

SHIM, J. P. Et al . **Past, present, and future of decision support technology**. *Decision Support System*, v. 33, n. 2, p. 111-126, 2002.

SIMON, Inre. Um Estudo de Caso: A Produção e Disseminação da Literatura Acadêmica. Disponível em: <<http://www.ime.usp.br/~is/ddt/mac339-01/aulas/www.linux.ime.usp.br/hvila/mac339/tema8.html>. Acesso em 12 jun. 2007.

SINGN, Harry S. **Data Warehouse: Conceitos, Tecnologias, Implementação e Gerenciamento**. São Paulo: Makron Books, 2001. 382 p.

SOUZA, César Alexandre de; ZWICKER, Ronaldo. Ciclo de vida de sistemas ERP. **Caderno de pesquisas em administração**, São Paulo, v. 1, n. 11, 1 trim./2000.

SPRAGUE, R. H.; WATSON, H. J. **Sistemas de Apoio à Decisão: Colocando a Teoria em Prática**. Rio de Janeiro: Campus, 1991. 498 p.

THE R PROJECT. **The R Project for Statistical Computing**. Disponível em: <http://www.rproject.org>. Acesso em: 10 jun. 2007.

WAGNER, Cláudio Arruda. **Estudo para implantação de um Data Warehouse em um ambiente empresarial**. Florianópolis: 2003. 99 p. Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para obtenção do grau de Mestre em

Ciência da Computação - Universidade Federal de Santa Catarina. Disponível em <http://teses.eps.ufsc.br/defesa/pdf/16068.pdf>. Acesso em 03 mai. 2007.

WEBLOGIC. *WebLogic – Business Software*. Disponível em [www.bea.com](http://www.bea.com). Acesso em 10 jun. 2007.

WEBSPHERE. *IBM WebSphere Software*. Disponível em [www.ibm.com/software/websphere](http://www.ibm.com/software/websphere). Acesso em 10 jun.2007.

WIKIPEDIA. A enciclopédia livre. 2007. Disponível em <http://pt.wikipedia.org>. Acesso em 10 jun. 2007.

YOUNESS, S. *Professional Data Warehousing with SQL Server 7.0 and OLAP Services*. Wrox Press Ltd, 2000. 604 p.