



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

Roger Pereira dos Santos

***DATA WAREHOUSE COM INTEGRAÇÃO DE BUSINESS
INTELLIGENCE***

**Assis
2015**



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

Roger Pereira dos Santos

DATA WAREHOUSE COM INTEGRAÇÃO DE BUSINESS INTELLIGENCE

Trabalho de Conclusão de Curso
apresentado ao Instituto Municipal de
Ensino Superior de Assis como requisito
do Curso de Graduação.

Orientador: Prof. Dr. Alex Sandro Romeo de Souza Poletto.

Área de Concentração: Informática.

**Assis
2015**

FICHA CATALOGRÁFICA

SANTOS, Roger Pereira dos.

Data Warehouse com Integração de Business Intelligence / Roger Pereira dos Santos. Fundação Educacional do Município de Assis– FEMA - Assis, 2015.

85 p.

Orientador: Dr. Alex Sandro Romeo de Souza Poletto.

Trabalho de Conclusão de Curso – Instituto Municipal de Ensino Superior de Assis - IMESA.

1. *Business Intelligence*. 2. *Data Warehouse*. 3. Banco de Dados Oracle. 4. Pentaho.

CDD: 001.6

Biblioteca da FEMA

DATA WAREHOUSE COM INTEGRAÇÃO DE BUSINESS INTELLIGENCE

ROGER PEREIRA DOS SANTOS

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis como requisito do Curso de Graduação, analisado pela seguinte comissão examinadora.

Orientador: Prof. Dr. Alex Sandro Romeo de Souza Poletto.

Analisador (1): Prof. Esp. Guilherme de Cleve Farto

**Assis
2015**

DEDICATÓRIA

Dedico este trabalho a minha família que sempre me apoia em todos os momentos, também dedico aos meus amigos, familiares e professores que puderam me passar um pouco do conhecimento que possuem. Apoiando-me com força necessária para que eu pudesse realizar meus sonhos e anseios.

AGRADECIMENTOS

Agradeço primeiramente a Deus, pois sem Ele, nada seria possível e não estaríamos aqui reunidos, aproveitando, juntos de momentos tão importantes, e também por ter me dado forças para chegar até aqui e creio que ainda irei muito além.

Agradeço aos meus pais Dalva e Isaias, batalhadores que fizeram de tudo para que eu pudesse estudar e ser de uma geração que mudaria muitas coisas em nossa família, e por estarem sempre comigo em todos os momentos. E ao meu irmão Rogério que sempre me apoiou em todas as decisões, inclusive com dúvidas em cálculos de física.

Agradeço também aos meus avós maternos Vicente e Vicência e paternos Alice e Aldemir (em memória), que sempre me apoiaram e incentivaram os estudos.

Agradeço também a todos os meus professores destes 4 anos de curso onde tive a oportunidade de conhecer e aprender com todos eles.

Ao meu orientador- amigo Prof. Dr. Alex Sandro Romeo de Souza Poletto, pela orientação, não somente durante este trabalho, mas por toda a minha vida acadêmica, por acreditar e tornar possível a construção de muitos de meus sonhos.

À professora e amiga Dr. Marisa Atsuko Nitto, por me ajudar em diversos momentos durante esse período do curso.

Aos amigos e todos que colaboraram direta e indiretamente, e apoiando nas decisões do dia-a-dia.

"The mind that opens to a new idea, never comes back to it's original size[...]"
Albert Einstein (1879-1955)

RESUMO

Devido ao alto grau de competitividade no mercado, os gestores tem percebido que a necessidade de uma boa e rápida tomada de decisão pode influenciar nos benefícios e ganhos de uma organização. A partir dessa percepção, e com o avanço da tecnologia, observa-se que cada vez mais as informações e dados são utilizados para que empresas possam se destacar, utilizando-os como uma forma de se manter na continua competitividade do mercado. Com isso, a implantação de *Data Warehouses* (DW) e ferramentas de *Business Intelligence* (BI), tornaram-se um requisito de negócio para grandes, médias e até pequenas empresas. Utilizando o DW para armazenamento de Dados Históricos, e o BI como suporte para melhores resultados gerando análises rápidas por meio de gráficos e relatórios.

Palavras Chave: 1 – *Business Intelligence*, 2 – *Data Warehouse*, 3 – Banco de Dados Oracle, 4 – Pentaho

ABSTRACT

Due to the high level of competitiveness in the market, managers have realized that a good and fast decision-making may influence in the benefits and earnings in a company. Combining this perception with the technology advancement, it's possible to notice that the use of information and data by the companies has increased, seeking a highlight position in the market and also remain competitive. Considering these points, the implant of Data Warehouses (DW) and Business Intelligence tools (BI), became a business requirement for big, medium and small sized companies. By using DW for Historical Data storage, and BI as a support tool to better results, a faster analysis through graphs and reports is generated.

Key Words: 1 – Business Intelligence, 2 – Data Warehouse, 3 – Banco de Dados Oracle, 4 – Pentaho

LISTA DE FIGURAS

Figura 1 – Proposta de Trabalho de Integração do BI com DW (autoria própria).	19
Figura 2 – Orientado por Assunto adaptado de (MACHADO, 2004).	24
Figura 3 – Integração de dados, adaptado de Machado (2004).	25
Figura 4 - Não Volátil no DW, adaptado de Machado (2004).	26
Figura 5 – Exemplo de Granularidade, adaptado de Machado (2004).	28
Figura 6 - Exemplo de detalhamento utilizando <i>Drill-Down</i>	30
Figura 7 - Exemplo de detalhamento utilizando <i>Drill-Up</i>	31
Figura 8 – Exemplo de DM adaptado de Machado (2004).	33
Figura 9 – Arquitetura de um ambiente de DW (SOARES, 1998).	35
Figura 10 – Modelo <i>Top Down</i> (SOARES, 1998).	37
Figura 11 – Exemplo de Arquitetura <i>Bottom Up</i> (SOARES, 1998).	39
Figura 12 – Exemplo de Tabela Fato.	41
Figura 13 – Exemplo de Tabela Dimensional.	42
Figura 14 – Exemplo de modelo estrela (MOREIRA, 2006).	43
Figura 15 – Exemplo do modelo floco de neve (MOREIRA, 2006).	44
Figura 16 – Processo de Business Intelligence (autoria própria).	48
Figura 17 - Exemplo do Processo de ETL.	56
Figura 18 - Exemplo de Edição e Visualização do Cubo no JPivot.	57
Figura 19 – Processo para construção de um <i>Data Warehouse</i> (autoria própria).	58
Figura 20 - Modelagem do banco de dados relacional.	66
Figura 21 - Script conexão PHP com Oracle (Quintiliano, 2015).	67
Figura 22 - Modelagem da tabela Fato_Vendas do DW.	68
Figura 23 - Modelagem da tabela Fato_Compras do DW.	69
Figura 24 - Conexão PDI Pentaho com Banco Relacional Oracle XE.	70
Figura 25 - Conexão PDI Pentaho com Banco Relacional Oracle ORCL.	71
Figura 26 - ETL Funcionário e criação da tabela DIM_FUNCIONARIO.	71
Figura 27 - ETL Usuário e criação da tabela DIM_USUARIO.	71
Figura 28 - ETL Fornecedor e criação da tabela DIM_FORNECEDOR.	72
Figura 29 - ETL Vendas a Praza e criação da tabela DIM_VENDASPRAZO.	72
Figura 30 - ETL Produto e criação da tabela DIM_PRODUTO.	72
Figura 31 - ETL Nota Venda e criação da tabela DIM_NOTAVENDA.	72
Figura 32 - ETL Item Venda e criação da tabela DIM_ITEMVENDA.	72
Figura 33 - ETL Nota Compra e criação da tabela DIM_NOTACOMPRA.	73
Figura 34 - ETL Item Compra e criação da tabela DIM_ITEMCOMPRA.	73
Figura 35 - ETL Cliente e criação da tabela DIM_CLIENTE.	73
Figura 36 - ETL Data e criação da tabela DIM_DATA.	73
Figura 37 - ETL e criação da tabela FATO_VENDAS.	74
Figura 38 - ETL e criação da tabela FATO_COMPRAS.	74
Figura 39 - Tabelas Dimensionais no Banco de Dados Oracle.	75
Figura 40 - Conexão do <i>Administration Console</i> com Oracle.	75
Figura 41 - Conexão do <i>Schema Workbench</i> com Oracle DW.	76
Figura 42 - Cubo Vendas criado no <i>Schema Workbench</i>	77
Figura 43 - Cubo Compras criado no <i>Schema Workbench</i>	77

Figura 44 - Visualização do Cubo no JPivot dentro do PUC.....	78
Figura 45 – Proposta de Projetos Futuros (autoria própria).	80

LISTA DE TABELAS

Tabela 1 - Principais diferenças entre um BD operacional e um DW	29
Tabela 2 - Vantagens e Desvantagens de utilizar a arquitetura <i>Top Down</i>	37
Tabela 3 - Vantagens e Desvantagens de utilizar a arquitetura <i>BottomUp</i>	39
Tabela 4 - Versões de BD Oracle (ORACLE 2 a ORACLE 6.1)	49
Tabela 5 - Versões de BD ORACLE 7.0 a Oracle 7.3.....	50
Tabela 6 - Versões de BD Oracle ORACLE 8.0 a ORACLE 8i (8.1.7)	51
Tabela 7 - Versões de BD Oracle ORACLE 9i Release 1 a ORACLE 9i Release 2	52
Tabela 8 - Versões de BD Oracle ORACLE 7.0 a ORACLE 7.3.	53
Tabela 9 - Versões de BD Oracle ORACLE 11g.....	54
Tabela 10 - Vantagens e Desvantagens do Pentaho.....	56

LISTA DE ABREVIATURAS

BI	Business Intelligence
DW	Data Warehouse
DM	Data Marts
OLAP	On-Line Analytical Processing
OLTP	On-Line Transaction Processing
SI	Sistemas da Informação
ODS	Operational Data Store
ER	Entidade de Relacionamento
WS	Web Service
ETL	Extract, Transform and Load
PUC	Pentaho User Console
PAC	Pentaho Administration Console
PDI	Pentaho Data Integration
PSW	Pentaho Analysis Server
PRD	Pentaho Reporting Designers
PDM	Pentaho Data Mining
CDE	Community Dashboard Editor
PAD	Pentaho For Apache Hadoop
DBA	Administrador de Banco de Dados
BD	Banco de Dados
SGBD	Sistema Gerenciador de Banco de Dados
OLAP	On-line Analytic Processing
SIE	Sistemas de Informação Executivos

SUMÁRIO

1. INTRODUÇÃO	16
1.1. OBJETIVOS	17
1.1.1. OBJETIVO GERAL	17
1.1.1. OBJETIVOS ESPECÍFICOS	17
1.2. JUSTIFICATIVA	17
1.3. METODOLOGIA	18
1.4. PROPOSTA DO TRABALHO	19
1.5. ESTRUTURA DO TRABALHO	20
2. REVISÃO BIBLIOGRÁFICA	21
2.1. EVOLUÇÃO DOS DADOS E SISTEMAS	21
2.2. DATA WAREHOUSE	22
2.2.1. ORIENTADO A ASSUNTOS	24
2.2.2. INTEGRAÇÃO	25
2.2.3. NÃO VOLÁTIL	26
2.2.4. VARIAÇÃO DE TEMPO	27
2.2.5. GRANULARIDADE	27
2.2.6. DIFERENÇA ENTRE BANCOS DE DADOS CONVENCIONAIS E ANALÍTICOS	28
2.2.7. <i>ON-LINE ANALYTIC PROCESSING (OLAP)</i>	29
2.2.7.1. <i>Drill-Down</i>	30
2.2.7.2. <i>Drill-Up</i>	31
2.2.7.3. <i>Drill-Across</i>	31
2.3. DATA MINING	32
2.4. DATA MARTS	33
2.4.1. ARQUITETURAS DE <i>DATA WAREHOUSE</i>	34
2.4.1.1. Arquitetura <i>Top Down</i>	36
2.4.1.2. Arquitetura <i>Bottom Up</i>	38
2.4.2. MODELAGEM DIMENSIONAL	40
2.4.2.1. Tabela Fato	41
2.4.2.2. Tabela De Dimensão	42
2.4.2.3. Modelo <i>Star Schema</i>	43
2.4.2.4. Modelo <i>Snowflake</i>	44
2.5. <i>BUSINESS INTELLIGENCE</i>	45
2.6. BANCO DE DADOS ORACLE	49
2.6.1. EVOLUÇÃO HISTÓRICA DA ORACLE	49
2.7. PENTAHO	55
3. PROJETO PARA CONSTRUÇÃO DE UM <i>DATA WAREHOUSE</i>	58
3.1. PLANEJAMENTO DO <i>DATA WAREHOUSE</i>	59
3.2. LEVANTAMENTO DAS NECESSIDADES DA EMPRESA	59
3.3. MODELAGEM DIMENSIONAL	59
3.4. PROJETO FÍSICO DE BANCO DE DADOS	60
3.5. PROJETO DE ETL (<i>EXTRACT, TRANSFORM AND LOAD</i>)	60
3.5.1. FILTRO DE DADOS	60
3.5.2. INTEGRAÇÃO DE DADOS	61
3.5.3. CONDENSAÇÃO DE DADOS	61
3.5.4. CONVERSÃO DE DADOS	61
3.5.5. DERIVAÇÃO DE DADOS	62

3.6.	DESENVOLVIMENTO DE APLICAÇÕES.....	62
3.7.	VALIDAÇÃO E TESTES.....	62
4.	ESTUDO DE CASO	64
4.1.	CRIAÇÃO DA MODELAGEM DO BANCO RELACIONAL.....	65
4.2.	CRIAÇÃO DO MODELO FÍSICO.....	66
4.3.	CRIAÇÃO DA MODELAGEM DO DW VENDAS	68
4.4.	CRIAÇÃO DA MODELAGEM DO DW COMPRAS.....	69
4.5.	CRIAÇÃO DO ETL (EXTRACT, TRANSFORM AND LOAD).....	70
4.6.	CUBOS OLAP	75
4.7.	CUBO OLAP NO PENTAHOU USER CONSOLE.....	78
5.	CONSIDERAÇÕES FINAIS	79
	REFERENCIAS BIBLIOGRÁFICAS.....	81
	APÊNDICE A – SCRIPTS DE CRIAÇÃO DO BANCO RELACIONAL.....	83

1. INTRODUÇÃO

O uso de sistemas informatizados vem crescendo cada vez mais, podendo ser encontrado em um simples empreendimento. Em grandes empresas o uso de tais sistemas é de fundamental importância para seu crescimento contínuo e de sua qualidade em desempenho. Observa-se que as empresas buscam armazenar seus dados, e informações, já que para tornar-se competitiva ela precisa de informação.

A informação é considerada o bem mais valioso de uma empresa, já que por intermédio desses dados é possível conhecer o negócio da empresa, e assim, tomar decisões. Muitas empresas possuem enormes quantidades de dados e que não são utilizados pelas mesmas. Uma das maiores dificuldades dessas organizações é o controle das informações, de forma que estas possam contribuir na tomada de decisões, na melhoria do atendimento ao cliente e em seus processos.

Para auxiliar as organizações, existe o ambiente de *Data Warehouse* (DW), que segundo Kroenke (1999), é um grande repositório de dados, que contém todas as informações relevantes de uma empresa, de forma que os tomadores de decisões possam utilizar estas informações de forma rápida e precisa nas decisões estratégicas de negócio.

Nessa grande massa de dados, pode-se utilizar *Data Mining*, um processo de análise de dados utilizando técnicas para exploração, de forma a descobrir novos padrões e relações interessantes podendo representar informações de grande relevância. Devido ao grande montante de dados esses padrões dificilmente seriam descobertos com métodos mais tradicionais como consultas simples a base de dados ou relatórios.

E para complementar o DW, pode-se incrementar, usando a tecnologia de *Business Intelligence* (BI) que se refere ao processo de coleta, organização, análise, compartilhamento e monitoramento de informações que oferecem suporte a gestão de negócios, tornando assim, um banco de informações completo e organizado com a capacidade de auxiliar as empresas nas tomadas de decisão estratégicas de longo prazo.

Todo o conteúdo do banco de dados entra na empresa por meio de um *Web Service*, que por sua vez é usado como base de saída e consulta de informações. A criação desse ambiente é um fator de grande importância, já que é o consumo dele que resultará em uma boa análise do negócio, além de que um sistema web é extremamente vantajoso para empresa e clientes que precisam acessar dados de diversos lugares e a qualquer momento.

1.1. OBJETIVOS

1.1.1. Objetivo Geral

O objetivo deste trabalho consiste em integrar as tecnologias de *Data Warehouse* ao *Business Intelligence*, no qual visa gerar resultados para gestores, mostrando as vantagens de uma boa decisão quando.

1.1.1. Objetivos Específicos

Os objetivos específicos que figuram nessa pesquisa são:

- Complementar e dispor o conhecimento em *Data Warehouse* e *Business Intelligence* utilizando um bom referencial teórico;
- Desenvolver uma análise sobre os aspectos práticos dessas tecnologias, apresentando estudos de casos;
- Ter contato com a ferramenta de banco de dados *Oracle* voltada para *Data Warehouse* e a ferramenta *Pentaho* para *Business Intelligence*, inclusive criar um pequeno banco relacional que será a base para a criação do DW e também para a integração de dados usando a ferramenta PDI do *Pentaho* para criar o processo de ETL.

1.2. JUSTIFICATIVA

Este projeto visa fornecer uma ferramenta que permite aos profissionais de pequenas, médias e até grandes empresas, avaliar padrões e informações que possam dar maior consistência as decisões estratégicas. Utilizando um *Web Service* para servir as informações e com as técnicas de *Data Warehouse*,

Data Mining e *Business Intelligence* no meio comercial, possibilitando que os empresários possam definir estratégias futuras para suas empresas.

1.3. METODOLOGIA

A metodologia utilizada para este projeto de pesquisa será do tipo exploratória, tendo em vista a busca de material bibliográfico científico sobre a preocupação apresentada, bem como o desenvolvimento de estudo de caso focalizado mais na parte relacionada ao banco de dados e inteligência dos negócios.

Para atender aos objetivos estabelecidos, definiu-se que, primeiramente deve-se obter o conhecimento necessário para o desenvolvimento deste trabalho investigando as partes que constituem os ambientes de *Data Warehouse*, *Business Intelligence* e *Data Mining*, por intermédio de pesquisa na literatura da área.

Em seguida, serão verificadas as características de cada uma destas partes, estruturas, componentes e técnicas de aplicação, com o objetivo de obter os conceitos envolvidos nestas para que seja possível definir os elementos do *Data Warehouse* e do *Business Intelligence*, bem como das aplicações de *Data Mining*. Com isso foi possível coletar informações relativas para o trabalho, sendo possível realizar um bom embasamento teórico para auxiliar o trabalho de conclusão de curso.

Como banco de dados será utilizado o Oracle para compor os *Data Marts*, que são um conjunto de dados orientado para o assunto, integrado, não volátil, variante ao tempo, no apoio de decisões gerenciais que tem como foco a organização como um todo, englobando várias áreas de assunto, conseqüentemente criando *Data Marts* que se relacionam entre si, tornando-se a estrutura do *Data Warehouse*.

Para a análise será utilizado o *Pentaho* para gerar os relatórios e gráficos do *Business Intelligence*.

Ao final do projeto com os dados levantados será feito a implantação de um DW e um BI em uma empresa de metais e ferragens e empresas do ramo de construção civil para projetos futuros.

1.4. PROPOSTA DO TRABALHO

Demonstrar a integração do *Business Intelligence* em um *Data Warehouse*, utilizando todo o referencial teórico apresentando e mostrando técnicas de DW e BI, utilizando também de exemplo(s) do uso das ferramentas: Oracle e Pentaho.

- Oracle para criação do DW.
- Pentaho para integração de dados (ETL) e elaboração de cubos de análise, além de mostrar resultados finais com gráficos para Dashboards.

A imagem a seguir demonstra como é o processo de integração do *Data Warehouse* e do conceito de *Business Intelligence* sendo implantado utilizando as ferramentas que compõem o Pentaho.

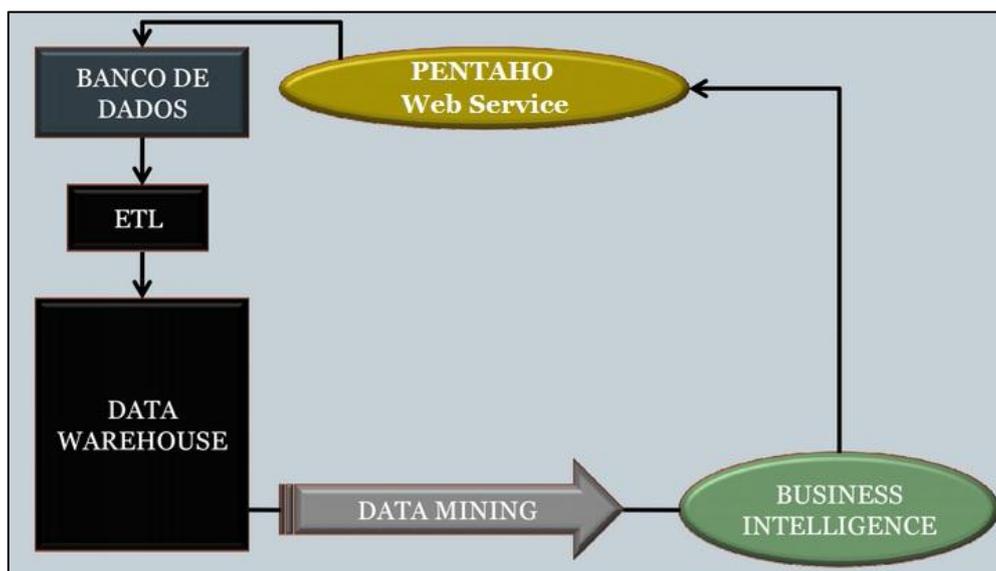


Figura 1 – Proposta de Trabalho de Integração do BI com DW (autoria própria).

1.5. ESTRUTURA DO TRABALHO

O trabalho está inicialmente estruturado em seis capítulos, sendo o primeiro essa introdução, contendo a ideia principal do projeto, os objetivos, justificativas, motivação e os métodos utilizados para as pesquisas e estruturação do trabalho.

Capítulo 2 – Revisão Bibliográfica: capítulo que contém a definição de *Data Warehouse*, as propriedades que compõem este ambiente, bem como conceitos relacionados à *Data Mining*, *Data Marts* e *Business Intelligence*.

Capítulo 3 – Projeto para Construção de Data Warehouse: capítulo que demonstra as principais preocupações e como deve ser estruturado um projeto de implantação de um *Data Warehouse*.

Capítulo 4 – Estudo de Caso: neste capítulo serão apresentados alguns estudos de casos para elucidar o conhecimento proporcionado pelo referencial teórico elencado nesta pesquisa.

Capítulo 5 – Considerações Finais: tratará dos resultados e as conclusões sobre os assuntos que nortearam o trabalho.

Referências: parte que relaciona todas as referências consultadas durante a elaboração do projeto.

2. REVISÃO BIBLIOGRÁFICA

Este capítulo tem como objetivo apresentar as características de um *Data Warehouse* assim como *Business Intelligence*, e os conceitos relacionados aos dois assuntos, também mostra um pouco sobre banco de dados Oracle e a ferramenta de BI Pentaho.

2.1. EVOLUÇÃO DOS DADOS E SISTEMAS

Na década de setenta (70), Seymuor Pappert (matemático e proeminente educador estadunidense nascido na África do Sul), afirmou que os dados e correlatos seriam responsáveis por uma grande revolução na forma que a sociedade entenderia os dados e sistemas. Os dados eram tratados em segundo plano no processo de desenvolvimento de sistemas, sendo utilizadas para as tomadas de decisões, metodologias e inspirações de algumas pessoas (BARBIERI, 2001).

Na década seguinte (80), apareceram novos movimentos metodológicos que mudaram o conceito anterior, dando uma posição de mais destaque para os dados. Nesse período surgiram os conceitos de Administração de Banco de Dados conhecidos hoje como DBA, Modelagem de Dados, da Engenharia da Informação e da Análise dos Dados.

A partir dessa década a forma de desenvolvimento de sistemas, que tinham características estáticas nas suas estruturas hierárquicas, foi substituída pelo modelo relacional que ofereceu um recurso de flexibilidade das relações entre os programas (INMON, 1997).

Com a flexibilização das relações, deu-se início ao processo de integração dos dados dos bancos dimensionais, onde os registros contidos nesses ambientes eram transformados em matrizes bidimensionais (INMON, 1997).

Na década de noventa (90), surgiu o conceito de *Data Mining* (Mineração dos dados) que tem por objetivo oferecer recursos para mineração de dados para auxiliar durante a busca de informações na identificação de padrões de correlação que são geralmente difíceis de detectar com as análises

convencionais. Em seguida surgiu o conceito de *Business Intelligence* (BI) que tem por objetivo auxiliar no processo de tomadas de decisões (BARBIERI, 2001).

A seguir abordado o conceito de DW para explicar as características desse conceito. Os conceitos de BI serão explicados mais adiante neste trabalho.

2.2. DATA WAREHOUSE

Um DW é um armazém de dados que é utilizado para guardar informações de uma empresa e mantê-lo disponível e acessível para consultas posteriores. Essas consultas devem fornecer resultados e claros, para realmente auxiliar na competitividade das empresas no mercado de trabalho (MACHADO, 2004).

Segundo BARBIERI (2001):

O DW é um banco de dados, destinado a sistemas de apoio à tomada de decisão e cujos dados foram armazenados em estruturas lógicas dimensionais, possibilitando o seu processamento analítico por ferramentas especiais (*Online Analytical Processing* e *Data Mining*).

Então o principal objetivo do DW é fornecer informações que auxiliem os administradores das empresas nas suas tomadas de decisões

Singh (2001) menciona que o DW integra dados de várias fontes de informações incompatíveis em um repositório de dados único e consolidado. Permite-se, assim, que sejam retiradas informações que se transformarão em conhecimento após uma análise precisa e consistente dos administradores da empresa.

As principais justificativas de se implantar a tecnologia de DW em uma organização é a existência de alguns fatores na empresa (MACHADO, 2004), tais como:

- A existência e utilização de várias plataformas de hardware e software;
- As constantes alterações nos sistemas transacionais corporativos;

- As dificuldades acentuadas na recuperação de dados históricos em períodos superiores ao ano atual em que estão sendo operadas as informações;
- A existência de programas de fornecedores diferentes;
- A falta de padronização e integração dos dados existentes dos programas utilizados pelas empresas;
- A falta de documentação e segurança no armazenamento dos dados;
- A dificuldade de reunir informações dos diversos sistemas existentes nas empresas;

Como resultado da implementação de um DW, têm-se a obtenção das seguintes variações do conhecimento que podem ser obtidos do mesmo (MACHADO, 2004).

- Informações disponíveis para auxiliar na gestão da empresa;
- Visão das variações do comportamento dos concorrentes e clientes;
- Agilidade de ferramentas para apoio a tomada de decisões;
- Segurança de que as informações utilizadas nas análises realmente auxiliaram nas decisões;
- Uma visão mais abrangente dos indicadores de tendências de mercado;
- Recursos mais abrangentes para análise de negócios;
- Através das tecnologias da informação, atender as necessidades e expectativas dos administradores;

Todas as informações e conhecimentos retirados de um DW podem proporcionar um ganho para as empresas em relação a custos e benefícios, economia de tempo e, principalmente, aumento na produtividade das empresas. Pois, através da análise dos dados obtidos do DW os empresários podem fazer novos negócios para melhorar suas empresas e tornarem-nas mais competitivas no mercado de trabalho (MACHADO, 2004).

Segundo INMON (1997):

Um DW é um conjunto de dados baseado em assuntos, integrado, não-volátil, variável em relação ao tempo e de apoio à tomadas de decisões gerenciais.

Quando se opta pela construção de um DW deve-se levar em consideração e analisar muito bem as características afirmadas por Inmon. Para melhor entendimento, a seguir serão explicadas e exemplificadas essas características.

Lembrando que não somente essas características, mas a granularidade do DW também deve ser analisada com cuidado, sendo explicada após as características de Inmon.

2.2.1. Orientado a Assuntos

Os dados do DW devem ser agrupados pelos assuntos mais importantes ou pelos processos principais da empresa. Geralmente são processos que demonstram o desempenho e os indicadores da mesma. A Figura 1 ilustra um exemplo dessa característica.

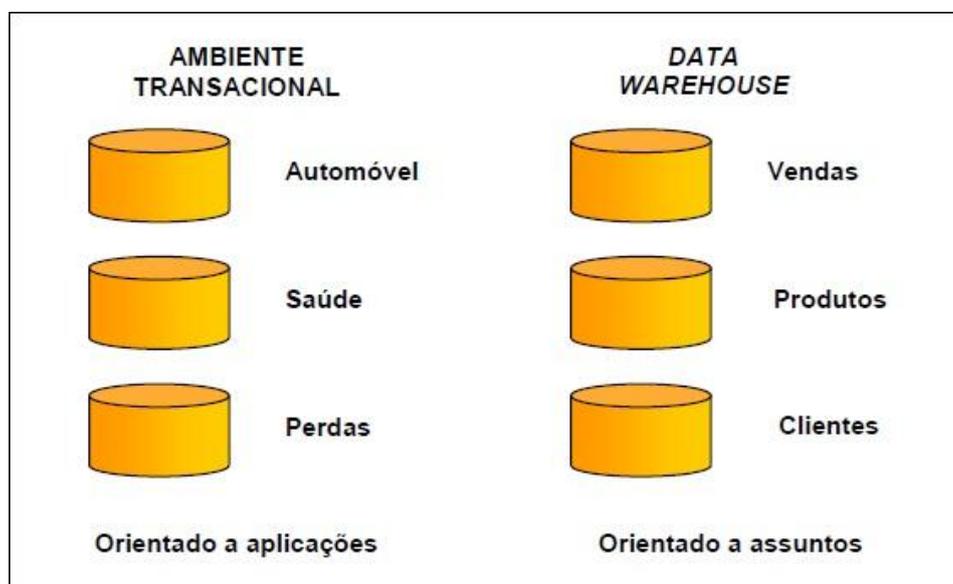


Figura 2 – Orientado por Assunto adaptado de (MACHADO, 2004).

A Figura 1 mostra a diferença entre um modelo transacional e o DW. No primeiro modelo (lado esquerdo), o banco de dados é orientado conforme as

aplicações existentes na empresa, já no segundo modelo (lado direito), tem-se o banco de dados dividido pelos principais assuntos.

2.2.2. Integração

A integração é referente à forma com que os dados são convertidos, padronizados em um único formato e armazenados na base de dados original ou transacional (INMON, 1996) e (MACHADO, 2004). Os programas normalmente são desenvolvidos, cada um usando uma forma diferente de armazenar seus dados, causando incoerência dos dados no momento de uma junção de bancos de dados, como exemplifica a Figura 2.

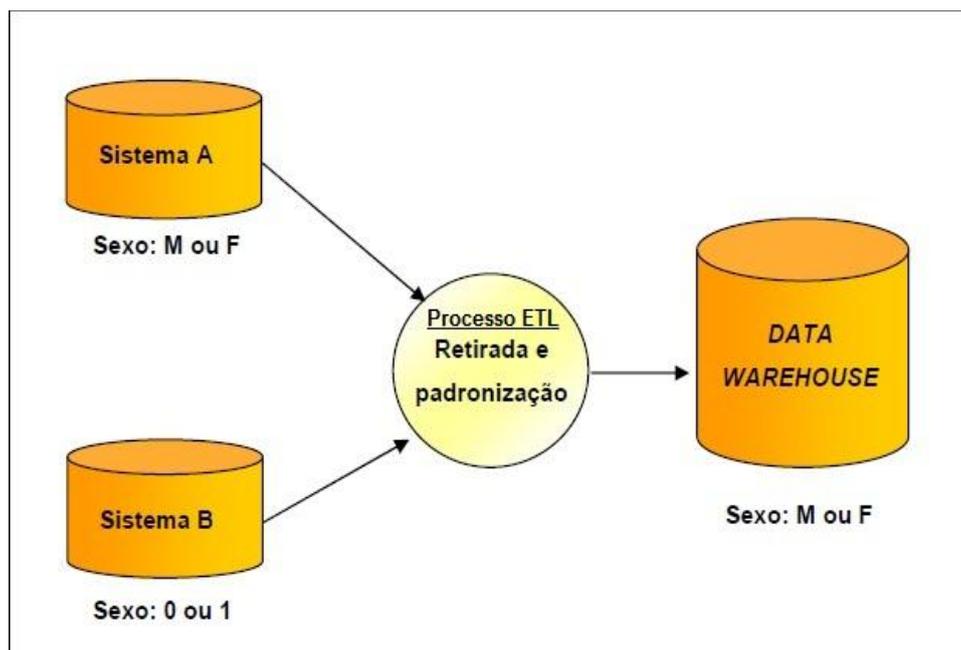


Figura 3 – Integração de dados, adaptado de Machado (2004).

Os dados referentes ao sexo dos clientes dos Sistemas A e B, são guardados em suas bases de formas diferentes. No Sistema A, o armazenamento do dado é feito como sendo a letra “F” para o sexo feminino e “M” sendo a letra para representar o sexo masculino. No Sistema B, este dado é armazenado o sexo feminino como “1” e o masculino como “0”.

Os dados provenientes dos Sistemas A e B devem passar por um processo de padronização e limpeza, antes de serem inseridos no DW. Deve ser adotada uma forma padrão de armazenamento, como exemplificado na Figura 2, o sexo do tipo feminino será armazenado como “F” e o masculino como “M”.

Cada desenvolvedor de DW deve prestar bastante atenção a essa característica, pois se esses dados não forem padronizados, podem causar problemas no carregamento do DW, dificultado o trabalho e atrasando o projeto.

2.2.3. Não Volátil

Os dados são carregados para o DW em grande quantidade após a sua limpeza e padronização sendo acessados constantemente. Mas, a atualização desses dados normalmente não ocorre (INMON, 1996).

Conforme Machado (2004) é permitido apenas dois tipos de ações a serem executadas no DW, o de inserção de informações e o de consultas. As operações de atualização e de retirada dos dados não podem ocorrer, para evitar inconsistências nos dados do DW. A Figura 3 exemplifica a característica.

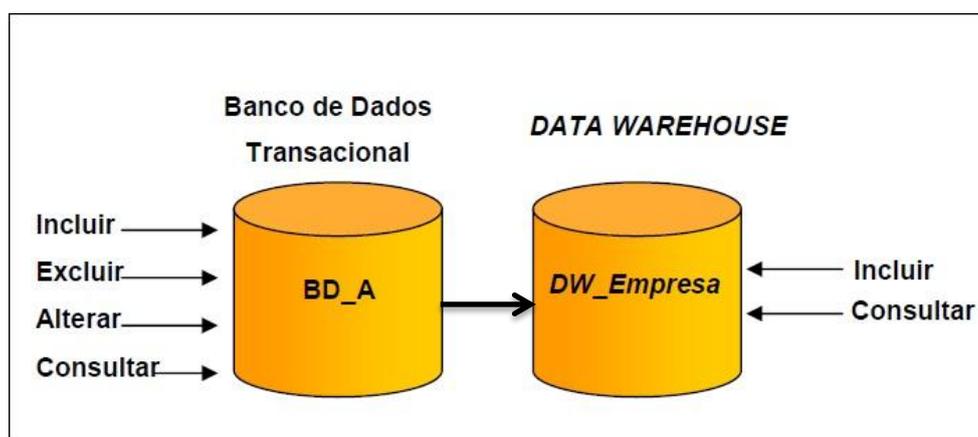


Figura 4 - Não Volátil no DW, adaptado de Machado (2004).

A Figura 3 mostra os tipos de consultas que podem ocorrer em um Banco de Dados Transacional e em um DW. A exclusão e a alteração dos dados no DW

não são permitidas para que se possam garantir um histórico correto das transações efetuadas nos sistemas de origem dos dados.

2.2.4. Variação de Tempo

Segundo Singh (2001), os sistemas transacionais não guardam dados históricos por muito tempo, por falta de espaço para manter um bom nível de desempenho dos sistemas.

Conforme Inmon (1996) e Machado (2004), o tempo em que os bancos transacionais normalmente guardam suas informações são de 60 a 90 dias, após esse tempo os BD passam por atualizações, perdendo os históricos dos dados.

O DW deve ser projetado de forma a garantir esse histórico dos dados, em que a cada nova carga feita no DW, deve ser guardada a data em que foi realizada a mesma. E, além de guardar essa data, deve-se utilizar o campo data como um dado “chave”. Com isso, garantem-se os dados históricos das empresas.

2.2.5. Granularidade

Inmon (1996) define a granularidade como sendo o aspecto mais importante do DW. Essa característica diz respeito ao nível de detalhamento ou de resumo das informações que irão conter no DW. Ainda segundo Inmon (1996) e Machado (2004) quanto mais detalhes existir no banco, mais baixo será o nível de granularidade, e quanto menos detalhes existir no banco, mais alto será o nível de granularidade. A Figura 4 mostra as diferenças das granularidades.

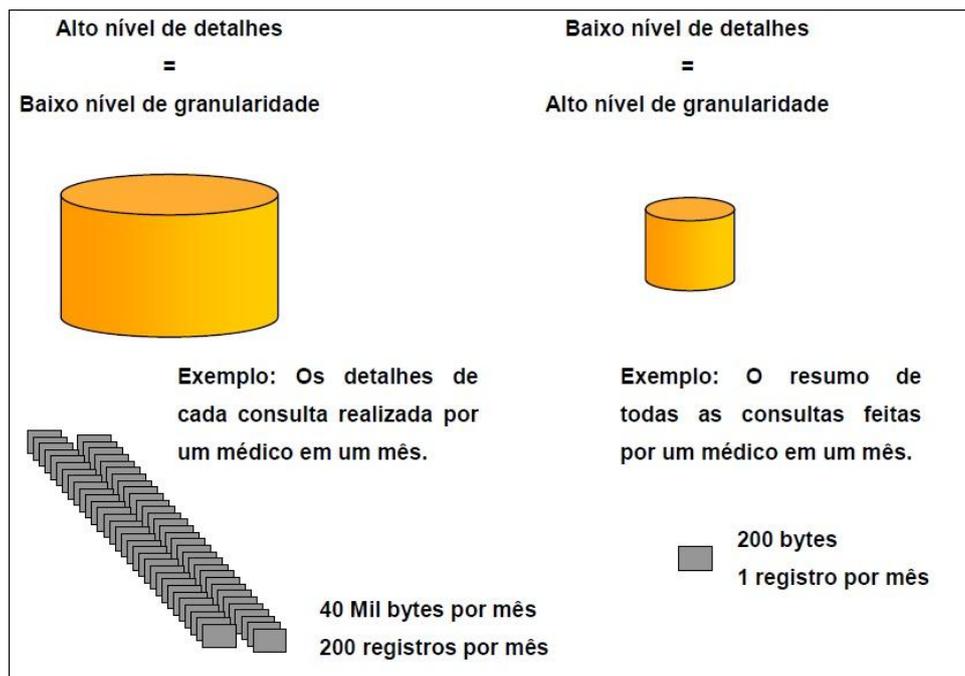


Figura 5 – Exemplo de Granularidade, adaptado de Machado (2004).

A Figura 4 mostra um exemplo da granularidade para as consultas médicas efetuadas por um médico durante um mês de trabalho, na qual podem ser observadas as diferenças entre o menor grau de granularidade e o maior grau. O primeiro arquivará uma maior quantidade de informações na base do DW e o segundo guardará uma quantidade bem menor na base de dados. A definição de qual grau de granularidade utilizar é definida pelos desenvolvedores, conforme as necessidades dos especialistas do domínio, durante o planejamento do DW.

2.2.6. Diferença entre Bancos de Dados Convencionais e Analíticos

Segundo Mata (2005), sistemas transacionais são utilizados para armazenar as operações diárias de uma empresa e é a base original que fornecem as informações para os sistemas analíticos.

Conforme Dill (2002) os ambientes operacionais são usados para oferecer um desempenho satisfatório para os clientes. Podem ser identificadas facilmente as transações que são executadas neste ambiente e os horários em que a

base é mais utilizada, facilitando o gerenciamento e o dimensionamento da base na sua carga.

Em um ambiente analítico de DW os processos executados são mais complexos, por trabalharem com uma quantidade grande de informações. A forma de retirar informações da base de dados de um DW não é padronizada e seus horários de maior fluxo de acesso não são conhecidos, dificultando a gerencia e a dimensão da carga dos dados da base, conforme afirma Dill (2002).

As principais diferenças entre os tipos de bancos de dados são apresentadas na Tabela 1.

Banco de dados Operacional	Banco de Dados Analítico de <i>Data Warehouse</i>
Baseado em objetivos operacionais.	Baseado em registros históricos.
Operações de Consulta, Inclusão, Alteração e Exclusão são permitidos.	São permitidos apenas operações de Inclusão e Consulta.
Operações acessam poucos registros.	Operações acessam muitos registros.
Estrutura Normalizada.	Estrutura permite redundância.
Usuários que acessam são os operadores.	Usuários que acessam são os responsáveis pelas tomadas de decisões.
Modelagem Relacional dos Dados.	Modelagem Dimensional dos Dados.
Atualização realizada em tempo real.	Atualização realizada periodicamente.

Tabela 1 - Principais diferenças entre BD operacional e DW (DILL, 2002).

2.2.7. On-Line Analytic Processing (OLAP)

O OLAP é um conjunto de ferramentas que fornecem condições para acessar e explorar os dados contidos em um DW (MACHADO, 2004). Os dados adquiridos da base do DW geralmente são retirados para responder as perguntas dos gerentes, executivos e analistas.

As ferramentas OLAP fornecem aplicações em que os usuários finais possam efetuar operações básicas de consultas e construir seus relatórios gerenciais. As principais ferramentas são: *Drill Down*, *DrillUp* e *Drill Across* (MACHADO,

2004). Essas operações são formas de mostrar os resultados finais para um usuário.

2.2.7.1. Drill-Down

Segundo Barbieri (2001) e Machado (2004) o *Drill Down* está diretamente ligado ao aumento do nível de detalhe da informação, buscando dados mais detalhados diminuindo a granularidade da consulta. Geralmente esse tipo de consulta é utilizado para juntar resultados de uma consulta.

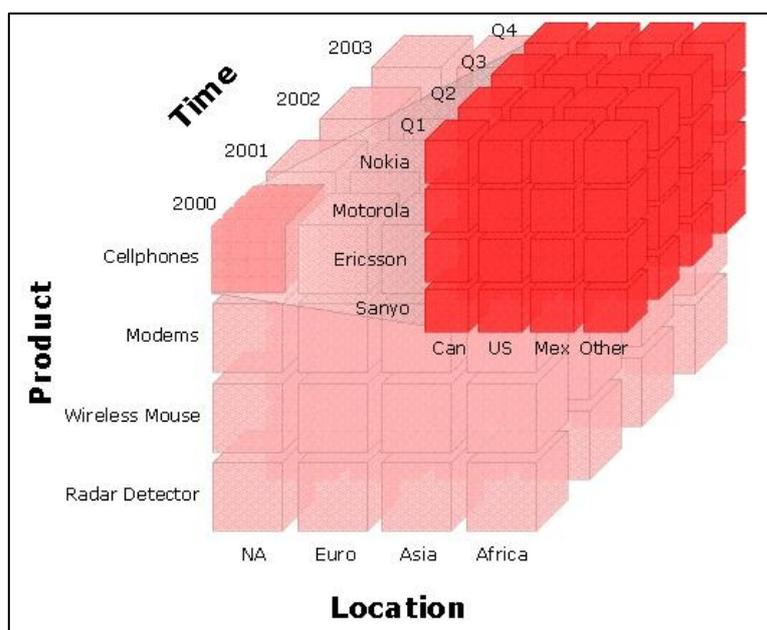


Figura 6 - Exemplo de detalhamento utilizando *Drill-Down*.

A Figura 5 demonstra por meio do cubo como aumenta o nível de detalhamento de um dado, nesse caso mostrou aumentou no nível de detalhamento dos dispositivos móveis (*Cellphones*) exibindo resultados por marcas.

2.2.7.2. Drill-Up

O *DrillUp*, ao contrário do *Drill Down*, diminui o nível de detalhe da informação, buscando dados com menos detalhes da base do DW, aumentando assim, o grau de granularidade das informações consultadas (BARBIERI, 2001; MACHADO, 2004).



vendedor	Measures
	• val_venda
All dimensao vendedor.vendedors	4.864.177
Ana Maria	649.694
André Luiz	723.361
Antônio Carlos	968.031
Carlos Brandão	335.496
José Augusto	350.807
João Carlos	701.097
Maria Izabel	351.632
Solange Brandão	784.059

Slicer: [num_ano=2012]

Figura 7 - Exemplo de detalhamento utilizando *Drill-Up*

A Figura 6 mostra uma análise feita no JPivot que é uma ferramenta para analisar cubos do Pentaho, essa análise mostra que o nível de detalhamento é menor, isso é confirmado pelas métricas (*measures*) onde tem-se apenas o total que cada vendedor.

2.2.7.3. Drill-Across

Conforme Machado (2004) o conceito de *Drill-Across* está relacionado diretamente com o fato de pular um nível intermediário dentro de uma mesma dimensão numa consulta. Como por exemplo, numa dimensão tempo do DW, em que ele possui os atributos dia, mês, ano; bimestre, semestre e ano, quando na realização de uma consulta para um relatório, faz-se uma pesquisa utilizando os dias e no mesmo relatório consulta dados do ano inteiro; essa consulta passou do nível de detalhamento máximo exemplificado pelo 'dia' para o nível de detalhamento mínimo que seria o 'ano'.

2.3. DATA MINING

Data mining, ou mineração de dados trata-se do processo de análise de dados utilizando-se de técnicas para exploração, de forma a descobrir novos padrões e relações interessantes podendo representar informações de grande relevância. Devido ao grande montante de dados esses padrões dificilmente seriam descobertos com métodos mais tradicionais como consultas a base de dados ou relatórios.

Os padrões podem ser definidos como sendo uma afirmação sobre uma distribuição de probabilidade, podendo ser expressos na forma de regras, sejam elas por formulas e funções, entre outras.

Os conceitos de garimpagem de dados estão relacionados com a nova tendência (para aplicações comerciais) de se buscar correlações escondidas em altos volumes de dados, nem sempre evidentes, principalmente no tratamento cotidiano dos sistemas de informação. (BARBIERI, 2001)

O interesse existente por este tipo de informação se dá principalmente ao fato de que as instituições estão coletando e armazenando cada vez mais dados e como consequência do baixo valor de meios de armazenamento e computadores e também do aumento da capacidade de ambos.

Com a maior utilização do DW, tende a aumentar a quantidade de informações disponíveis. Conforme já mencionado anteriormente, métodos tradicionais de análise de dados, não são apropriados para grandes volumes de dados, pois podem criar relatórios informativos sobre os dados, mas não conseguem analisar o conteúdo destes relatórios a fim de obter conhecimentos importantes.

Para Barbieri (2001), o DW é uma forma de se capitalizar em cima de informações, na tentativa de descobrir padrões de comportamento de clientes ou estilos de ações fraudulentas em cartões de crédito, seguradoras etc.

A técnica de mineração busca algo a mais que apenas interpretação dos dados existentes, almejando principalmente a realização de previsões com possíveis fatos e correlações não explicadas em um DW ou DM.

No fundo, com as técnicas de *Data Mining* esperasse identificar atributos e indicadores capazes de melhor definir uma situação específica.

Barbieri (2001) cita o exemplo de uma empresa de seguros no qual as ferramentas de OLAP responderiam perguntas como: “Qual o valor médio de pagamentos de seguros de vida para não fumantes, na região sul do estado, em agosto de determinada data?”. O uso das ferramentas de *Mining* para o exemplo acima nos trariam melhores atributos de clientes, capazes de ajudarem como previsores de possíveis acidentes de automóvel.

2.4. DATA MARTS

O termo *Data Mart* significa um depósito de dados relacionados a um setor específico a um setor específico da empresa, criados para auxiliar no processo decisório dessa empresa. Assim, pode-se dizer que um DM é um DW setorial. Segue as mesmas características de um DW geral, mas tendo os seus dados voltados apenas para um setor. Portanto, um DW é formado por vários DM's (BARBIERI, 2001).

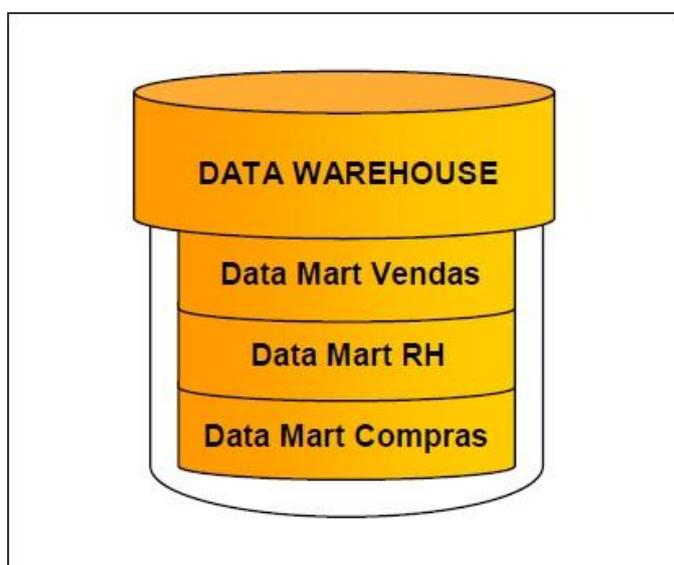


Figura 8 – Exemplo de DM adaptado de Machado (2004).

A Figura 7 exibe um DW que possui três DM setoriais, dos setores de Vendas, Recursos Humanos e Compras. Com a construção dos DM's pode-se construir um DW com uma maior facilidade. A partir de um DW também existe a possibilidade desenvolver os DM's com uma maior rapidez. Cada forma de programar depende da necessidade da empresa.

Antes do desenvolvimento de um DW ou um DM, deve ser feito uma análise das necessidades da empresa, definindo a qual arquitetura e o modelo a ser utilizado na implementação do projeto. Os tipos de arquitetura mais utilizados serão descritas na seção na seção seguinte.

2.4.1. Arquiteturas de *Data Warehouse*

Conforme os estudos de Machado (2004):

A arquitetura de um DW inclui, além de estruturas de dados, mecanismos de comunicação, processamento e apresentação da informação para o usuário.

Normalmente, são utilizados conjuntos de ferramentas que auxiliam desde o processo de retirada e limpeza dos dados até o processamento dos relatórios gerados a partir de um DW.

Na arquitetura do DW podem ser analisados os principais componentes, como os papéis exercidos por pessoas, as ferramentas que podem ser utilizadas, os dados e os processos que podem conter no mesmo. A Figura 7 mostra um exemplo de uma arquitetura de um ambiente de DW.

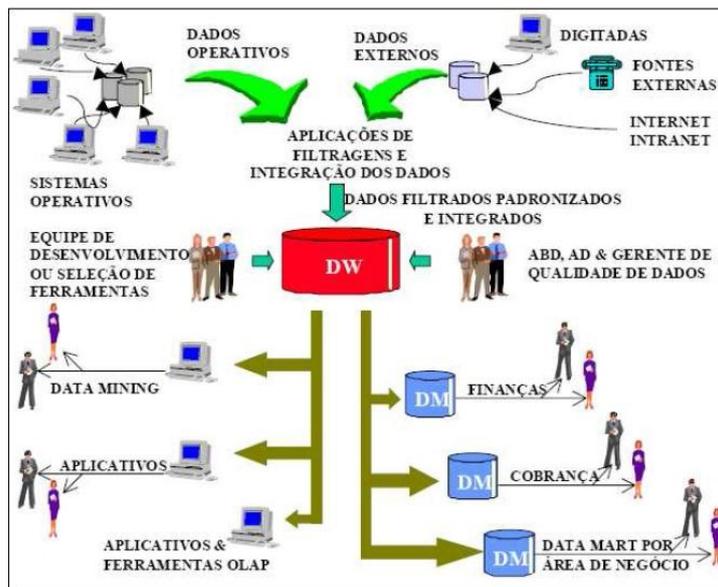


Figura 9 – Arquitetura de um ambiente de DW (SOARES, 1998).

A Figura 8 apresenta algumas fontes de informação para carregamento de um DW. Os dados geralmente estão armazenados em bases diferentes em formatos distintos, com níveis de detalhamento e sem seguirem um padrão no armazenamento dos dados, sendo necessário o uso de aplicações que executam a filtragem e integração dos dados.

Para fazer a análise do que será implementado no banco do DW e quais as informações que devem ser retiradas do mesmo, é necessário conter na equipe de desenvolvimento, profissionais que são responsáveis pelas cargas dos dados junto ao DW. Tendo o conhecimento das bases de origem e do modelo do DW. Também deve conter especialistas do domínio, gerentes, executivos que utilizarão o DW nas suas decisões e administradores da rede e bancos de dados, para garantir que a comunicação do DW com suas bases de origem estejam disponíveis durante as cargas do DW.

Finalizando o processo de análise, o DW é carregado, dando início ao processo de extração dos dados da base, através de ferramentas de Data Mining, implementação de aplicativos desenvolvidos pelos programadores da empresa ou através de aplicativos e ferramentas OLAP.

Com o DW carregado têm-se a possibilidade de criação de DM com uma maior facilidade, mas a criação dos DM depende das necessidades da empresa. No

processo de criação dos DM o DW seria dividido por áreas gerenciais da empresa para auxiliar os setores na busca de suas próprias informações.

Existem duas abordagens de implementações principais para desenvolvimento de um DW, são elas: *Top Down* e *BottomUp* (Machado, 2004). A escolha de qual arquitetura utilizar é definida normalmente no início do projeto, mas pode ser mudada durante a análise e implementação do DW.

A escolha depende do tempo disponível para o desenvolvimento do projeto, quais os retornos serão adquiridos após o término do mesmo, quanto tempo levará para o início da aquisição de conhecimento a partir do DW implementado, a satisfação dos administradores da empresa e os recursos financeiros, operacionais, computacionais, entre outros, disponíveis na empresa (Machado, 2004).

Na sequencia será apresentado o conceito das duas arquiteturas.

2.4.1.1. Arquitetura *Top Down*

Essa foi a primeira arquitetura proposta para o desenvolvimento em um ambiente de DW. Também é conhecida como arquitetura padrão. Consiste na extração dos dados do DW geral para os DM's setoriais, necessitando de que sua escolha seja definida no início do projeto e precisa ter um conhecimento da visão geral da empresa (MACHADO, 2004).

Essa arquitetura inicia o seu processo com a extração, a transformação e a integração dos dados externos ou bancos independentes, passando essas informações para um *Operational Data Store*(ODS). Esse processo de construção do ODS é opcional, mas a sua construção diminuiria os esforços na construção o DW final, pois todos os esforços quanto a integração dos dados seriam depositados no ODS. Finalizando a construção do ODS, passaria para a transferência das informações para o DW (MACHADO, 2004).

A partir do DW carregado podem ser extraídos os dados e metadados para os DM's. Terminando todo o processo de carregamento e preenchimento do DW e dos DM's, são utilizadas ferramentas OLAP, de *Data Mining* ou mesmo aplicações desenvolvidas pelos próprios programadores componentes da

equipe, para retirar as informações dos bancos para auxiliar nas decisões da empresa. A Figura 9 ilustra um exemplo dessa arquitetura.

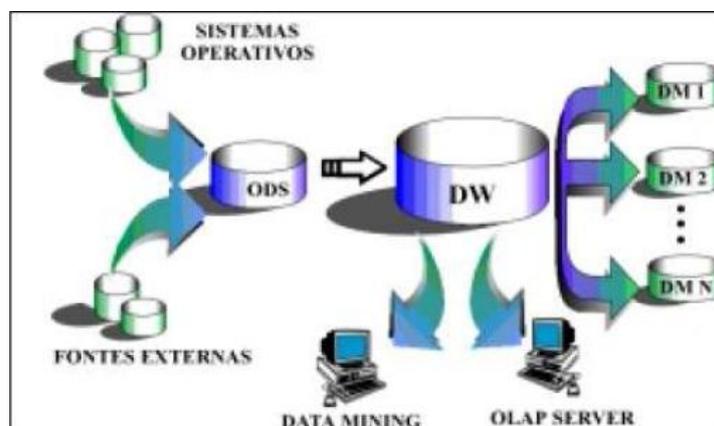


Figura 10 – Modelo *Top Down* (SOARES, 1998).

Esse modelo de arquitetura possui suas vantagens e desvantagens, segundo Machado (2004), segue a Tabela 2:

Vantagens	Desvantagens
Herança da arquitetura do DW pelos DM's.	Implementação muito longa, por envolver todos os setores da empresa.
Visão da empresa como um todo.	Alta taxa de riscos, por não existir garantias de resultados nesse tipo de ambiente.
Concentração dos dados e "metadados" em um único repositório e simplicidade na manutenção.	Heranças de cruzamentos funcionais, que exige ao final do projeto, profissionais altamente capacitados para garantir a manutenção do sistema.
Regras de extração, limpeza e integração centralizados facilitando a manutenção.	A demora na construção do DW e a falta de retorno rápido causa ansiedade e expectativas nos usuários.

Tabela 2 - Vantagens e Desvantagens de utilizar a arquitetura *Top Down* (MACHADO, 2004).

Essa arquitetura exige que seus programadores tenham conhecimento de todo o processo de funcionamento da empresa, tornando a sua implementação mais demorada, além de exigir que a empresa mantenha na equipe de funcionários,

profissionais altamente qualificados para garantir a manutenção do DW. Mas, em contra partida, após o DW finalizado, a retirada de DM's se torna simples, pois seria apenas uma cópia do DW geral, mas sendo apenas de um departamento e facilitando a sua manutenção por se localizar fisicamente em um mesmo lugar. Com o DW mantêm-se uma visão da empresa por completo e concentra os trabalhos de atualizações apenas em um local.

2.4.1.2. Arquitetura *Bottom Up*

As dificuldades encontradas na implementação *Top Down*, em que requer um tempo maior para ser concluído, possui uma demora em apresentar resultados e o custo de construção terminava caro. Após a análise das informações citadas anteriormente, pode-se perceber que a utilização da arquitetura *Bottom Up* passou a ser mais usada, por ser uma arquitetura que objetiva a construção de um DW incremental, iniciando pelos DM's e evoluindo para a construção do DW (MACHADO, 2004).

Os DM's permitem que seus planejamentos e seus desenhos possam ser feitos sem necessitar de uma visão da corporação inteira, bastando apenas conhecer o fluxo existente em uma área da empresa, implementando DW incremental.

Segundo Machado (2004, p. 54) essa arquitetura inicia o seu processo com a extração, a transformação e a integração dos dados externos ou bancos independentes, passando essas informações para um ou mais DM's. Finalizando a construção dos DM's, passaria para a transferência das informações para um DW geral.

A partir dos DM's carregados podem ser extraídos os dados e metadados para o DW. Terminando todo o processo de carregamento e preenchimento dos DM's e do DW, são utilizadas ferramentas para retirar as informações dos bancos para auxiliar nas decisões da empresa. A Figura 10 ilustra um exemplo dessa arquitetura.

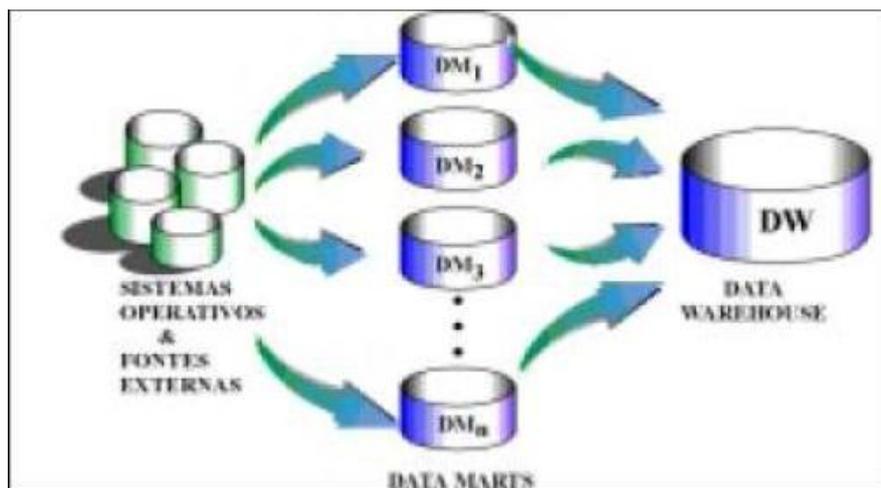


Figura 11 – Exemplo de Arquitetura *Bottom Up* (SOARES, 1998).

Este modelo de arquitetura possui algumas vantagens e desvantagens, segundo Machado (2004, p. 55), sendo apresentadas algumas na Tabela 3:

Vantagens	Desvantagens
Implementação direcionada, permitindo um desenvolvimento rápido.	Perigo dos DM's se tornarem DM's independentes, que dificultam e as vezes inviabilizam a integração com o DW geral.
Fornecer um retorno rápido para o usuário, por ser desenvolvida mais rápido.	Desafio de possuir e manter uma visão do negócio como um todo. Esse controle dificulta a extração e combinação das fontes que serão utilizadas no DW.
Manutenção do enfoque da empresa, fazendo com que toda a atenção dos desenvolvedores fique em uma determinada área da empresa.	Dificulta a administração das equipes de desenvolvimento, que normalmente são divididas para desenvolver DM's em paralelo.
Herança das informações pelo DW de forma incremental, reduzindo os riscos no projeto.	A "maldição do sucesso" começa a perseguir os desenvolvedores do projeto. Enquanto uns usuários ficam felizes com seu DM atualizado, outros ficam aguardando que o seu DM seja atualizado, fazendo com que seus desenvolvedores tenham que administrar políticas de atualizações e de recursos.

Tabela 3 - Vantagens e Desvantagens de utilizar a arquitetura *BottomUp* (MACHADO, 2004).

A utilização dessa arquitetura necessita que os analistas passem a conhecer as áreas da empresa de forma incremental. As duas abordagens têm boas características, mas a escolha de uma delas depende do que as empresas estão necessitando, por isso os modelos da arquitetura devem ser definidos no início do projeto. A seguir serão mostrados alguns conceitos sobre a modelagem dimensional de um DW.

2.4.2. Modelagem Dimensional

Através da modelagem dimensional ou multidimensional é possível observar o banco de dados de vários ângulos, usando a abordagem no formato de um cubo, que pode conter duas, três ou quantas dimensões for possível. Nos ambientes operacionais construídos atualmente, a técnica de modelagem Entidade Relacionamento (ER) tem sido a mais utilizada nos desenvolvimentos dos projetos. (DILL, 2002) Com o surgimento dos sistemas e processos de DW necessitou-se de uma nova técnica de modelagem que se adequasse melhor à forma de implementação o do novo ambiente.

A modelagem dimensional é mais simples de ser modelada, mais expressiva e facilita o entendimento, mas por ser uma modelagem nova, ainda não possui técnicas detalhadas de desenvolvimento firmes como já tem o modelo ER. (Machado, 2004).

Conforme (Dill, 2002), a modelagem dimensional é uma técnica para conceitualização e visualização de um conjunto de medidas existentes num negócio. É muito útil na construção de dados sumarizados e reorganização das informações, como é o caso dos DW's.

Apesar dessa forma de modelagem ser nova, é muito utilizado na modelagem de dados para construção de DW's, por facilitar a organização e visualização das informações que irão conter no mesmo. Essa fase do processo é de extrema importância, pois segundo Soares (1998), pode definir o sucesso ou o fracasso no desenvolvimento de um DW, pois sendo bem feita trará os

resultados desejados, mas se for mal elaborado o modelo não trará resultados satisfatórios.

O modelo dimensional é formado por dois elementos básicos (MACHADO, 2004):

- Tabela de Fatos;
- Tabela de Dimensões;

2.4.2.1. Tabela Fato

A tabela de fatos, conforme os estudos de Soares (1998) representam as informações que serão analisadas, sendo formada normalmente por valores numéricos que representam dados de medidas. Mas nem sempre essas tabelas possuem totais ou valores numéricos, passando a ser considerada uma tabela que fará o mapeamento dos eventos ocorridos.

Soares (1998) afirma que a tabela de fatos geralmente possui uma grande quantidade de informações. Segundo Machado (2004) cada fato representa um item, uma transação ou um evento do negócio. Esses dados mostram os acontecimentos diários de uma empresa, o que auxiliará no processo de análise da empresa.

A tabela de fatos é formada normalmente por uma chave primária composta pelas chaves primárias das tabelas dimensões.

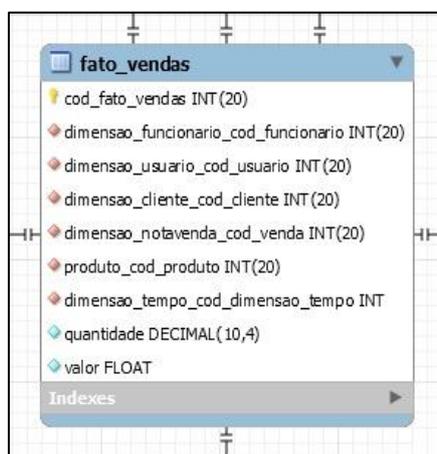


Figura 12 – Exemplo de Tabela Fato.

Na Figura 11 é mostrado um exemplo de tabela fato onde possui uma chave primaria e duas métricas sendo quantidade e valor, as demais colunas sendo chaves estrangeiras que vem das tabelas dimensionais.

2.4.2.2. Tabela De Dimensão

Segundo Machado (2004) as tabelas de dimensões são os elementos que fazem parte da tabela de fato. Essas tabelas armazenam os dados das dimensões da empresa e normalmente são menores do que a tabela de fatos e não possuem atributos numéricos de somatórios.

As tabelas de dimensões podem ser utilizadas por mais de uma tabela de fato, pois descrevem e classificam seus elementos, podendo estar inseridas em mais de um assunto no DW.

Conforme Dill (2002) uma das principais funções da criação de uma tabela de dimensões é de servir como fonte de informações para uma consulta efetuada no DW ou como cabeçalhos de linha nas respostas oferecidas aos usuários finais.

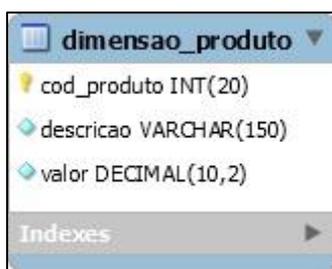


Figura 13 – Exemplo de Tabela Dimensional.

A Figura 12 mostra um exemplo de tabela dimensional, logo se vê que diferente da tabela relacional, não é necessário que tenha todas as colunas que a tabela relacional possui, pois é necessário apenas dados principais.

A seguir serão definidos os modelos dimensionais que podem ser utilizados para implementar a arquitetura física de um DW.

2.4.2.3. Modelo *Star Schema*

O modelo estrela é o mais utilizado na modelagem de DW por sua rapidez nas consultas. Esse modelo é composto por uma entidade central chamada de fato e um conjunto de entidades menores ligadas a ela, chamadas de dimensões (MACHADO, 2004). A Figura 13 mostra um modelo de representação do modelo estrela.

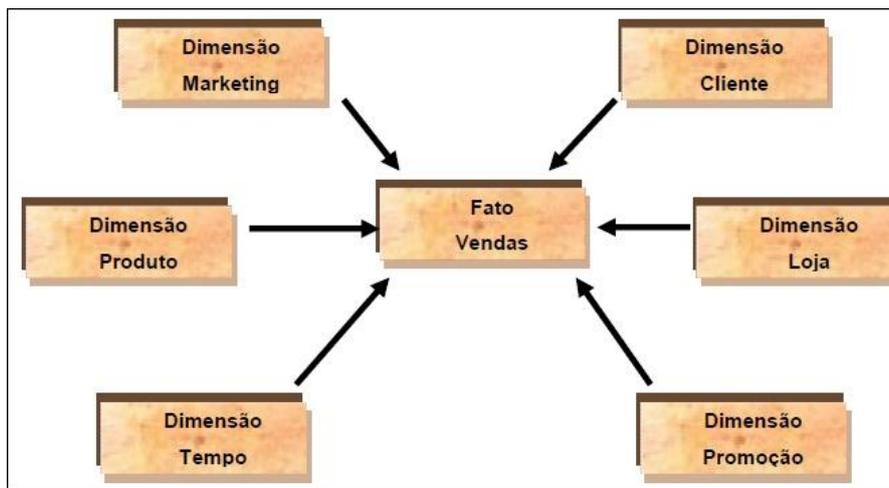


Figura 14 – Exemplo de modelo estrela (MOREIRA, 2006).

Segundo Singh (2001) uma característica importante desse modelo é fato de suas dimensões serem desnormalizadas. A desnormalização é o fato de ter dados sendo repetidos na base de dados, para conferir simplicidade e desempenho nas consultas. Portanto, dados podem ser repetidos em dimensões, para garantir um maior desempenho.

O modelo estrela oferece algumas vantagens importantes, segundo Singh (2001):

- Permite criar uma estrutura multidimensional com um modelo de dados simples;
- Facilita a definição de relacionamentos hierárquicos dentro de cada dimensão e facilita as consultas entre muitas tabelas;

- Reduz a quantidade de consultas feitas entre tabelas fisicamente, melhorando o desempenho;
- Ao simplificar a visualização do modelo de dados, diminui a chance de os usuários fazerem consultas erradas que consomem a máquina e trazem resultados duvidosos;
- Permite a expansão e a evolução do DW com pouca manutenção;

A simplicidade de entendimento dos dados mostrados a partir desse modelo e fácil manutenção do banco de dados influenciam em muitos aspectos na análise e escolha do modelo que a empresa irá utilizar.

2.4.2.4. Modelo *Snowflake*

Segundo Machado (2004):

O modelo floco de neve é o resultado da decomposição de uma ou mais dimensões que possuem hierarquias entre seus membros.

Conforme Singh (2001):

Normalizando os dados das tabelas dimensionais de um modelo estrela transforma o mesmo em um modelo floco de neve.

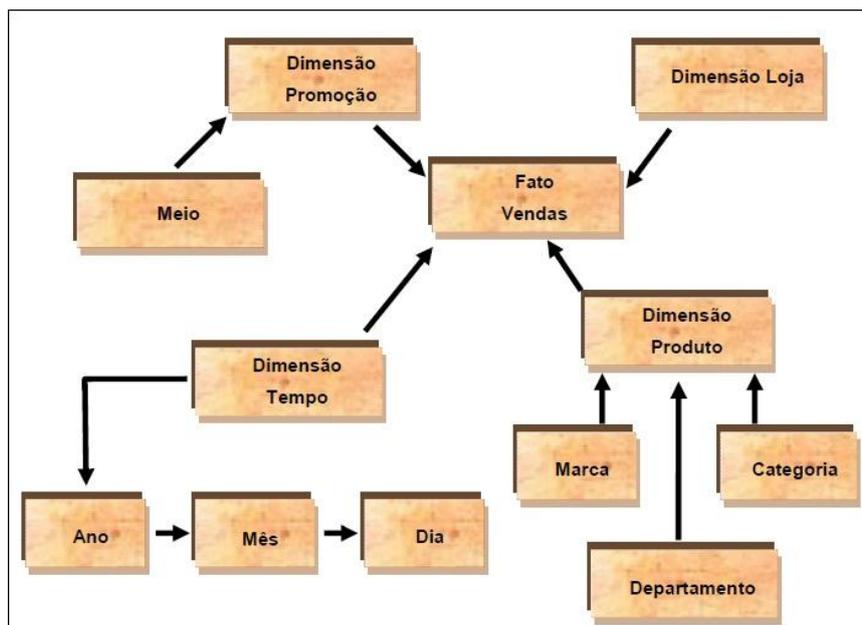


Figura 15 – Exemplo do modelo floco de neve (MOREIRA, 2006).

Através desse modelo podem ser definidos relacionamentos entre tabelas de dimensões, o que não pode ocorrer no modelo estrela, o que torna esse modelo de fácil entendimento por estar normalizado, mas torna difícil adquirir informações da sua base de dados. Esse modelo possui algumas desvantagens em relação ao modelo estrela (Singh, 2001):

- A estrutura dos dados normalizada acaba se tornando mais complexa;
- A navegação executada sobre o modelo se torna mais difícil, por ter que passar por muitas tabelas para chegar ao resultado;
- A carga e a manutenção dos dados no DW acabam sendo mais difícil de ser executado e dificulta a administração desse processo;

O modelo floco de neve é visualmente mais fácil de identificar as hierarquias, mas a perda de performance por causa do aumento das tabelas torna-o mais inviável do que o modelo estrela. Mas, existem casos em que a identificação dessas hierarquias se faz necessária, dependendo apenas do domínio em que está sendo aplicado.

2.5. *BUSINESS INTELLIGENCE*

O termo BI foi apresentado pelo Gartner Group em meados da década de 1990. Contudo, o conceito iniciou muito antes, com suas raízes nos sistemas de geração de relatórios de Sistema de Informação Gerencial (SIG) dos anos 1970. No início dos anos 80, surgiu o conceito de Sistemas de Informações Executivos (SIE). Esse conceito expandiu o suporte computadorizado aos gerentes e executivos de nível superior.

Através do constante desenvolvimento e evolução dos métodos e tecnologias empregadas no desenvolvimento de sistemas de informações, tornou-se possível oferecer maior capacidade e agilidade no processamento dos dados e apoio ao processo decisório INMON (1997). A criação de técnicas como de DW e de soluções como BI, permitiu a elaboração de sistemas que possibilitam um maior controle dos processos internos e a definição e acompanhamento de estratégias de gestão dos seus indicadores.

Diante deste contexto advindo através do entendimento de que o desempenho e a eficiência são significantes formas de competitividade, surge o conceito *Business Intelligence* cuja aplicação é justificada pela observação de ATRE e MOSS (2003):

Nos competitivos dias atuais e no aumento da incerteza mundial, a qualidade e a conveniência da organização de uma aplicação de "*Business Intelligence*" (BI) pode significar não só a diferença lucro e perda, mas também a diferença entre sobrevivência e falência. (ATRE e MOSS, 2003).

Business Intelligence não é simplesmente um produto ou tão pouco um sistema. Trata-se de uma arquitetura e ao mesmo tempo uma coleção de operações integradas, assim como aplicações de suporte à decisão e bancos de dados que forneçam aos tomadores de decisão fácil acesso às informações do negócio (ATRE e MOSS, 2003).

Barbieri (2001) sustenta *Business Intelligence* como um guarda-chuva conceitual que envolve Inteligência Competitiva (CI – Conjunto de informações da concorrência e do mercado), Gerência de Conhecimentos (KMS – Conjunto de informações de uma empresa criadas, agrupadas, recuperadas e organizadas), *Internet Business Intelligence*, pesquisa e análise de mercados. Barbieri ainda complementa que por meio da aplicação de BI, é possível obter um leque de informações de extrema importância para os membros diretivos de uma organização e que a utilização de tais ferramentas propicia diferenciação neste contexto competitivo no qual estão inseridas as empresas.

Segundo Oliveira (2002) o *Business Intelligence* é uma extensa categoria de programas de aplicação e tecnologias para acumular, armazenar, analisar e fornecer acesso a dados com o objetivo de ajudar os usuários a tomarem as melhores decisões de negócio. Aplicações de BI incluem atividades de suporte à decisão, perguntas e relatórios, processamento analítico on-line, análises estatísticas, estimativas e *Data Mining*.

BI é um meio estruturado de traduzir dados em conhecimento. Coleta-se informações de bases operacionais, armazena-se isso de forma modelada e, depois, realiza-se consultas através de ferramentas para prover informações que se traduzam em vantagem competitiva, onde diversas ferramentas e

metodologias são empregadas. BI vai desde a definição dos objetivos do negócio a serem contemplados, a preparação de dados existentes ou externos, a modelagem e construção de repositórios, a escolha de *Front Ends* para os usuários, a tradução de informação em decisões e ações, até a monitoração dos resultados de negócio obtidos por todo este processo (OLIVEIRA, 2002).

A arquitetura de uma solução de BI é formada por três principais componentes:

O primeiro é responsável por extrair, limpar, integrar e carregar no *Data Warehouse* dados de diferentes fontes. Existem no mercado, softwares, que duplicam cadastros de clientes, automaticamente. São chamados ferramentas para limpeza de dados.

O segundo componente é aquele que armazena os dados para fim de análise. Pode ser um DW ou um DM que nada mais é do que um DW desenvolvido para um departamento específico.

O último e terceiro componente é aquele que torna disponível aos usuários finais o acesso aos dados através de uma variedade de ferramentas OLAP e *Data Mining*. Elas possibilitam aos usuários manipular informações de forma flexível, utilizando uma abordagem de análise multidimensional, nas quais indicadores podem ser visualizados por diferentes ângulos.

Esses três componentes aparecem em qualquer solução de BI, o que pode variar são as tecnologias empregadas na solução e no objetivo do projeto.

O BI é uma tecnologia que oferece recursos para estruturar, acessar e explorar um grande volume de dados, para desenvolver padrões e conhecimentos que permitam identificar padrões que auxiliam na tomadas de decisões (KIMBALL, 2000).

Conforme Barbieri (2001), essas informações são guardadas normalmente em *Data Warehouse* (DW) ou *Data Mart* (DM), com o objetivo de auxiliar a transformação dessas informações em conhecimento estratégico para a empresa.

Após essas informações serem transformadas e organizadas, necessita-se que sejam trabalhadas, fazendo combinações, compilações e distribuições, com o objetivo de transformá-las em conhecimento.

Os benefícios de ter um sistema de BI segundo Sousa (2003) são:

- Antecipar mudanças no mercado;
- Antecipar ações dos competidores;
- Descobrir novos ou potenciais competidores;
- Aprender com os sucessos e falhas das outras empresas;
- Conhecer melhor os seus possíveis parceiros ou aquisições;
- Conhecer novas tecnologias, produtos ou processos que tenham impacto no seu negócio;
- Entrar em novos negócios;
- Rever suas próprias práticas do negócio;
- Auxiliar em novas implementações de ferramentas gerenciais.

A utilização das tecnologias de BI faz com que os gerentes e administradores analisem com mais detalhes suas empresas e os seus concorrentes, obtendo resultados que auxiliaram nas mudanças internas que se poderão ser necessárias, nas decisões táticas e estratégicas a serem implantadas para que a empresa cresça.

Para melhor compreensão segue imagem ilustrativa de um processo para de realização necessária para obter um ambiente com BI.

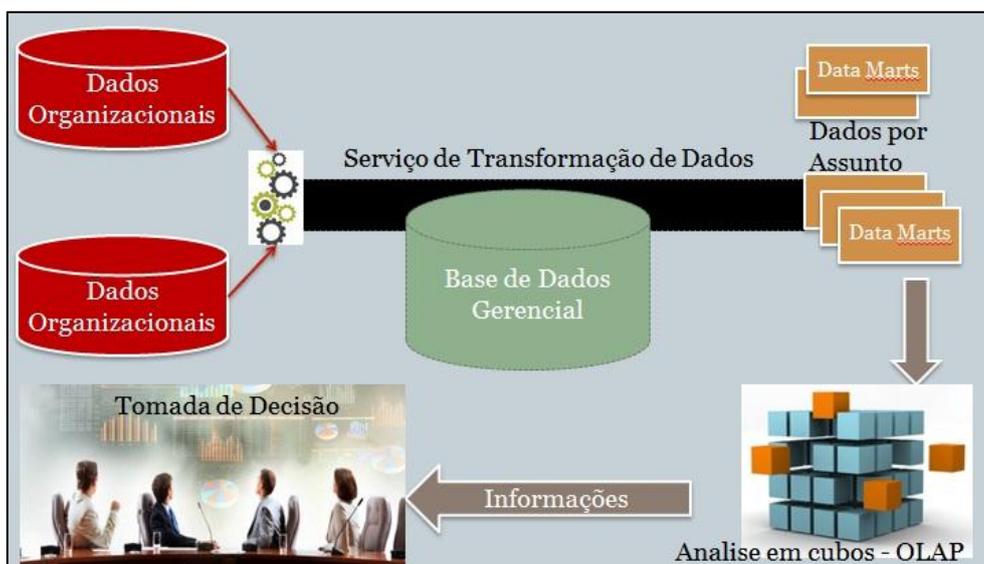


Figura 16 – Processo de Business Intelligence (autoria própria).

2.6. BANCO DE DADOS ORACLE

2.6.1. Evolução Histórica da Oracle

O Oracle é um sistema gerenciador de banco de dados (SGBD) desenvolvido no final dos anos 70 por Lawrence J. Ellison ao descobrir uma oportunidade não abrangida por outras companhias, por um protótipo funcional de um banco de dados relacional.

A Oracle Corporation foi instituída em 1977 e desde então é líder de mercado. A linguagem criada para o processamento de transações é chamado de PL/SQL, e atualmente está na versão 12C.

Com o passar dos anos, a Oracle desenvolve varias versões para o banco de dados, as quais demonstradas por meio da tabela abaixo:

Versão	Ano de criação	Características
Oracle 2	1979	<ul style="list-style-type: none"> • Primeiro Lançamento Público • Funcionalidades básicas SQL (<i>queries and joins</i>)
Oracle 3	1981	<ul style="list-style-type: none"> • Execução automática de instruções SQL • Transações • Consultas em bloqueios
Oracle 4	1984	<ul style="list-style-type: none"> • Estabilidade
Oracle 5.0	1986	<ul style="list-style-type: none"> • Cliente-Servidor
Oracle 5.1		<ul style="list-style-type: none"> • Consultas distribuídas
Oracle 6	Julho 1988	<ul style="list-style-type: none"> • Bloqueio linha-nível • Backup de dados em linha • PL/SQL no banco de dados
Oracle 6.1		<ul style="list-style-type: none"> • Servidor Paralelo

Tabela 4 - Versões de Banco de Dados Oracle - ORACLE 2 a ORACLE 6.1 (RIGO, 2012).

Versão	Ano de criação	Características
Oracle 7.0	Junho 1992	<ul style="list-style-type: none"> • Constraints • Armazenamento de funções e padrões • Triggers • Visualizar compilação • Manual de funções SQL definidas • Segurança • Limite de recursos – perfis • Auditoria reforçada • Replicação de dados
Oracle 7.1		<ul style="list-style-type: none"> • Replicação de dados simétricos • Recuperação paralela • SQL dinâmico DBMS_SQL • Consulta e criação de índices, carregamento de dados
Oracle 7.2		<ul style="list-style-type: none"> • Arquivos de dados redimensionável • Subconsulta na cláusula FROM • Criação de tabelas paralelas
Oracle 7.3		<ul style="list-style-type: none"> • Índices bitmap • Leitura assíncrona das tabelas • Opção contexto • Introdução Db_verify • Trigger compilação, depuração • Limitações de cláusulas de extensões • Histogramas • Dependenciais • Oracle Trace

Tabela 5 - Versões de Banco de Dados ORACLE 7.0 a Oracle 7.3 (RIGO, 2012).

Versão	Ano de criação	Características
Oracle 8.0	Junho 1997	<ul style="list-style-type: none"> Banco de dados objeto relacional Padrão SQL 3 Recuperação de partições individuais Security Server introduzidas pela administração central do usuário. Termo de senha, senha de perfis, permitem esquema de senha personalizada. Links de banco de dados privilegiado (sem necessidade de senha a ser armazenada) START otimizado de consultas Organização de tabelas por índice Verificação de restrição de integridade
		<ul style="list-style-type: none"> Dados de tinta introduzida na base de dados Novo formato ROWID DML paralela Fusão/partições de equilíbrio Pool de conexão para suportar mais usuários simultâneos Muitos melhoramentos de desempenho para o SQL/PSQL/OCI fazendo uso mais eficiente da CPU/Memória. V7 limites alargados
Oracle 8i (8.1.5)		<ul style="list-style-type: none"> A recuperação rápida Iniciar – Checkpoint taxa de auto ajustada por atender roll Reorganização de índices/tabelas de índices Log Miner introduzidos – Permite online ou redo logs arquivados a ser visualizada através do SQL Manual de Segurança Gestão de propriedade de recursos Estatísticas do otimizador Procedimentos armazenados Java (Oracle Java VM) Banco de dados virtual privado Analisar as tabelas em paralelo Dados de espera – auto transporte e aplicação de redo logs Tablespaces transportáveis entre bancos de dados Drop da coluna na tabela Índices Funcional – NLS, maiúsculas e minúsculas
Oracle 8i (8.1.6)		<ul style="list-style-type: none"> DBA Studio introduzida XML Parser para Java Novas funções SQL Comando ALTER FREELISTS Checksums sempre na tablespace SYSTEM Novo PLSQL criptografar/descriptografar Usuários e esquemas separados Muitos aprimoramentos de desempenho
Oracle 8i (8.1.7)		<ul style="list-style-type: none"> Static Servidor HTTP incluído Java Server Pages (JSP) motor OIS – Oracle Integration Server introduzidas JVM Accelerator para melhorar a performance de código Java PLSQL Gateway introduziu para implementação de PL/SQL soluções com base na Web Enterprise Manager Enhancements – incluindo novos relatórios baseados em HTML e avançada funcionalidade de replicação incluído
		<ul style="list-style-type: none"> New Character Database Set utilitário ed migração incluído

Tabela 6 - Versões de Banco de Dados Oracle ORACLE 8.0 a ORACLE 8i 8.1.7 (RIGO, 2012).

Versão	Ano de criação	Características
Oracle 9i Release 1 (9.0.1)	Junho 2001	<ul style="list-style-type: none"> • Segmentos de reversão tradicional (RBS) • Oracle Parallel Server's (OPS) de escalabilidade • Consulta Flashback (dbms_flashback.enable), este recurso permitirá que os usuários para corrigir as operações sem chamar o DBA • Oracle Ultra Search para pesquisar bancos de dados, sistemas de arquivos • VI (Virtual Interface) suporte ao protocolo, uma alternativa para o TCP/IP disponíveis para o uso com o Oracle Net (SQL*Net) • Mineração de dados • Apoio de cursor de rolagem, permite a busca para trás de um conjunto de resultados • Gerenciamento de memória dinâmica • Reorganização online do índice • Build em XML Developers Kit (XKD), novos tipos de dados para XML (XMLType) • Cost Based Optimizer agora considera também a memória e CPU, Não só custos de acesso do disco • Programas PL/SQL nativa pode ser compilado para binários • Proteção de dados Deep – segurança grão fino e de auditoria • Separação – lista de particionamento de uma lista de valores • Oracle Nameserver
Oracle 9i Release 2 (9.2.0)		<ul style="list-style-type: none"> • Gerenciado localmente tablespaces SYSTEM • Sistema de arquivos de cluster para Windows e Linux • Compressão do segmento de dados (comprimir as chaves nas tabelas – apenas quando os dados de carga) • Criar bancos de dados standby lógicos com dados da Guarda • Melhorias na segurança – Default Install contas bloqueadas, VPD em sinônimos, AES, Usuários migrar para o diretório

Tabela 7 – Versões de Banco de Dados Oracle ORACLE 9i Release 1 (9.0.1) a ORACLE 9i Release 2 (9.2.0) - (RIGO, 2012).

Versão	Ano de criação	Características
Oracle 10g Release 1 (10.1.0)		<ul style="list-style-type: none"> • Grid Computing – uma extensão da funcionalidade de clustering • Melhorias de gerenciamento (self-tuning características) • Melhorias de desempenho e escalabilidade • Automated Storage Management (ASM) • Automatic Workload Repository (AWR) • Automatic Database Diagnostic Monitor (DAMS) • Capacidade de Undrop uma tabela a partir de uma lixeira • Flashback operações disponíveis em linha, transação, tabela ou banco de dados de nível • Capacidade de transporte de tabela em todos os tipos de máquina • Data Pump – mais rápido movimento de dados com expdp e impdp • Capacidade de mudar o nome de tabela • Declaração Nova base de dados 'drop' • Novo agendados de banco de dados – DBMS_SCHEDULER • Suporte para espaços de tabela de até 8 exabytes de tamanho
Oracle 10g Release 2 (10.2.0)	Setembro 2005	<ul style="list-style-type: none"> • As senhas para DB são criptografadas • Async • Asmcmd novo utilitário para o gerenciamento de armazenamento ASM
Oracle 10g Express Edition (Oracle XE)	Março 2006	<ul style="list-style-type: none"> • Liberdade para desenvolver e implementar aplicativos de muitas plataformas • Suporte para uma grande variedade de ambientes de desenvolvimento • Recursos de performance, confiabilidade e segurança • Permite que os desenvolvedores tirem total proveito do Oracle Application Express para rápido desenvolvimento e implementação de aplicativos baseados na Web

Tabela 8 - Versões de Banco de Dados Oracle ORACLE 7.0 a ORACLE 7.3 (RIGO, 2012).

Versão	Ano de criação	Características
Oracle 11g		<ul style="list-style-type: none"> • Alta capacidade de cluster de banco de dados • Aceleração da automação do data Center e o gerenciamento da carga de trabalho • Grids seguros, altamente disponíveis e escaláveis de servidores e armazenamento de baixo custo • XML binário para melhor desempenho de aplicativos • Oracle Real Application Testing – inovação no teste e gerenciamento de alterações em ambientes de TI
		<ul style="list-style-type: none"> • Oracle Data Guard – banco de dados em standby com alto desempenho, segurança e recuperação • Gerenciamento do armazenamento orientado por regras de negócios • É possível atingir índices de compactação de 2x a 3x ou até mais para todos os dados • Registro total de todas as alterações nos dados • Oracle Flashback transaction – facilita a reversão de uma transação efetuada com erro, bem como de qualquer transação dependente • Parallel Backup and Restore – ajuda a melhorar o desempenho do backup e restauração de bancos de dados grandes • Oracle Fast Files – armazena grandes objetos • Criptografia aparente • XML mais rápido • Data Recovery Advisor – ajuda os administradores a reduzir significativamente a parada para recuperação, o que permite automatizar investigação de falhas, determinar planos de recuperação e lidar com várias situações de crise • Novo Compilador Java Just-in-time para executar procedimentos Java no banco de dados mais rapidamente • Integração nativa com o Visual Studio 2005 para desenvolvimento de aplicativos .NET no Oracle; ferramentas de migração de Acess com Oracle Application Express; e um recurso para fácil criação de consultas do SQL Developer e rápida codificação de rotinas SQL e PL/SQL • Hot patching’ – alta disponibilidade do sistema ao permitir que correções sejam aplicadas sem a necessidade de interromper a operação do SGBD • Pool de conexões e cachês de resultados das consultas: produtos Query Result Caches e Database Resident Connection Pooling • Cubos OLAP incorporados

Tabela 9 - Versões de Banco de Dados Oracle ORACLE 11g (RIGO, 2012).

2.7. PENTAHO

Pentaho atualmente é a plataforma de inteligência de negócios (BI) que utiliza código aberto sob a licença pública da Pentaho (PPL) mais popular e poderosa no mundo. A sede da empresa está localizada em Orlando na Flórida, com escritórios em São Francisco, Califórnia e por toda a Europa.

Conforme descrito no site oficial, o Pentaho nasceu do desejo de se alcançar uma mudança positiva por megavendedores burocráticos que ofereciam produtos “pesados” baseados em plataformas antigas, direcionadas as suas aplicações já existentes e às custas de suas capacidades de inovação na área de BI. Sob esta expectativa, cinco veteranos da indústria (por isto o nome Penta) fundaram o Pentaho em 2004.

Atualmente a ferramenta conta com seis soluções/componentes disponíveis sendo elas:

- *Pentaho Data Integration (PDI)*: também conhecido como Kettle, é uma ferramenta de ETL (Extração, Transformação e Carga);
- *Pentaho Analysis Services (PSW)*: conhecido como Mondrian OLAP Server, é uma ferramenta para Análise de informação (OLAP);
- *Pentaho Reporting Designres (PRD)*: derivado do projeto JFreeReport
- *Pentaho Data Mining*: derivado do projeto Weka, um conjunto de ferramentas relacionadas com a Mineração de Dados (Data Mining);
- *Pentaho DashBoard*; para criação de *dashboard*, é utilizado a ferramenta de criação, edição e visualização conhecido como *Community Dashboard Editor (CDE)*.
- *Pentaho for Apache Hadoop*: também conhecido como *Pentaho BI Suite*, é a versão para Haddop, uma plataforma de software em Java de computação distribuída para clusters e processamento de grandes massas de dados.

Existem algumas vantagens ao utilizar o Pentaho sendo elas:

VANTAGENS	DESVANTAGENS
Licença não tem custo no Pentaho.	Documentação insuficiente.
Indicadores com acesso pela web não importando o SO (Windows, Mac, Linux).	Suporte limitado para versão gratuita.
Customização dos Indicadores de Negócio (Dashboards, Cubos e Relatórios).	Poucas referências na Internet.

Tabela 10 - Vantagens e Desvantagens do Pentaho.

A empresa Pentaho dispõe de duas versões uma que é gratuita e outra paga:

- Pentaho CE (*Community Edition*): é *Open Source* dispendo de relatórios básicos, análises simples, integração de dados, conectividade board.
- Pentaho EE (*Enterprise Edition*): dispõem de relatórios básicos, análise simples, integração de dados, conectividade *board*, instalar, implantar, admin, segurança, *dashboards*, *interactive report*, análise interativa, *data discovery* e *visualization*, *certified software* e qualidade assegurada, releases de manutenção, apoias, serviços profissionais.

A ferramenta Kettle (ETL) possui mais de 35 tipos de conexões com banco de dados como Oracle, MySQL, PostgreSQL e Microsoft SQL Server dentre outros.

Essa ferramenta também é conhecida como PDI (*Pentaho Data Integration*), e permiti ler e escrever em mais de 35 formatos de SGBD e realiza a importação de arquivos de texto, planilhas de Excel e base de dados.



Figura 17 - Exemplo do Processo de ETL.

Através do Mondrian é possível utilizar a ferramenta *Schema-Workbench* para criação de Cubos OLAP, que são compostos de arquivos XML e definem as

dimensões e conexões de dados. Para facilitar o uso da ferramenta, a edição e criação dos cubos são feitas por uma interface gráfica que gera os arquivos .xml que serão publicados para que usuários finais possam ter acesso as informações.

Usuários podem visualizar e editar os cubos já publicados através do JPivot que é uma biblioteca de identificadores (tags), responsável pelo desenvolvimento de tabelas OLAP, e utiliza o Mondrian como servidor.

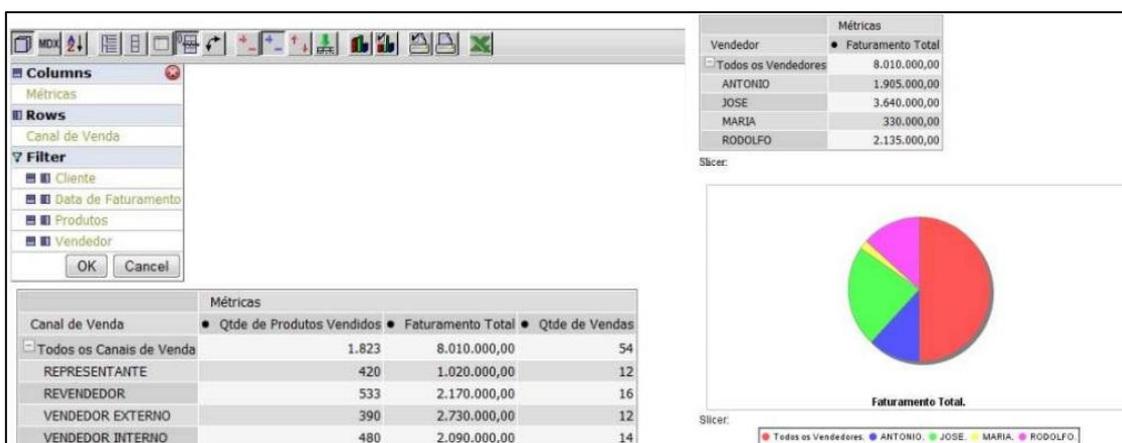


Figura 18 - Exemplo de Edição e Visualização do Cubo no JPivot.

3. PROJETO PARA CONSTRUÇÃO DE UM *DATA WAREHOUSE*

Conforme Mata (2005, p. 25), a utilização das tecnologias de BI têm como principal objetivo melhorar todos os produtos e serviços oferecidos pelas empresas, visando obter uma maior satisfação do usuário final e dos profissionais envolvidos no processo de desenvolvimento do projeto.

Para se construir um DW ou DM que são tecnologias de BI, devem ser seguidos alguns aspectos. Nesse trabalho será exemplificado o processo de criação proposto por Barbieri (2001, p. 68). Os passos para execução do projeto de construção de um DW consistem em:

- Planejamento de um *Data Warehouse*;
- Levantamento das necessidades da empresa;
- Modelagem Dimensional;
- Projeto físico do Banco de Dados;
- Projeto de ETL;
- Desenvolvimento das Aplicações;
- Validação e Testes;

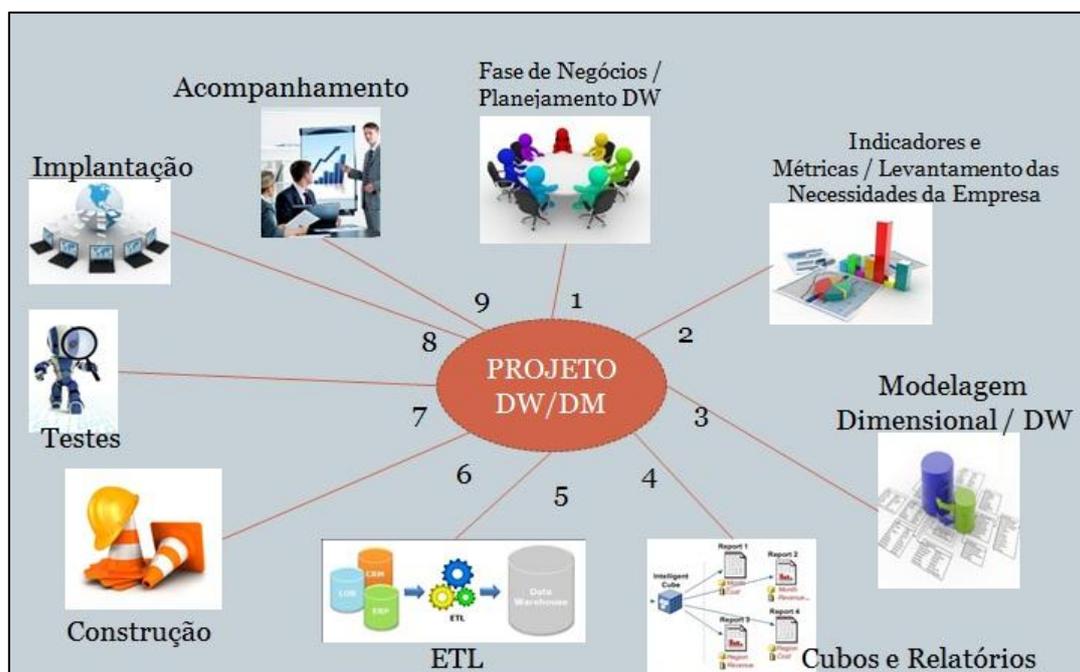


Figura 19 – Processo para construção de um *Data Warehouse* (autoria própria).

3.1. PLANEJAMENTO DO *DATA WAREHOUSE*

Esta é a primeira etapa para a construção de um DW, no qual é definido o escopo do projeto que deverá atender as necessidades gerenciais da empresa.

Segundo Barbieri (2001) essa etapa pode ser definida em quatro etapas básicas, sendo elas:

- Definição do foco do negócio;
- Definição da abordagem do sistema;
- Planejamento para a integração dos dados;
- Definição da arquitetura tecnológica;

3.2. LEVANTAMENTO DAS NECESSIDADES DA EMPRESA

Nessa etapa devem ser analisados dois modelos: modelo dimensional e modelo fonte de dados.

O primeiro modelo representa os dados necessários para ter no final do projeto um sistema que dê suporte nas decisões da empresa, devendo-se fazer uma análise detalhada desses dados.

O segundo modelo representa os dados das bases originais e/ou as fontes externas de informações, sendo guardadas as suas descrições, formas de armazenamento e de uso desses dados. O modelo de fonte de dados será a base para a construção do modelo dimensional (BARBIERI, 2001).

3.3. MODELAGEM DIMENSIONAL

A modelagem dimensional é uma fase crítica para o projeto, podendo causar o fracasso do projeto se feito de forma errada. Os dados devem ser analisados de forma diferente da modelagem relacional, em que os dados serão

armazenados de forma compacta, dependendo do grau de granularidade escolhido no projeto (BARBIERI, 2001).

3.4. PROJETO FÍSICO DE BANCO DE DADOS

Nessa etapa deverá ser criado o banco de dados do DW no SGBD escolhido. Criando as tabelas de dimensões e fatos com seus atributos, assim como fazendo suas indexações, relacionamentos e implantações das regras (BARBIERI, 2001).

Com a conclusão desta etapa, têm-se o banco de dados com todas as suas tabelas de fatos e dimensões criadas e com as ligações entre as chaves primárias e estrangeiras.

3.5. PROJETO DE ETL (*EXTRACT, TRANSFORM AND LOAD*)

Nesta etapa deve ser feita a transferência das informações das fontes de dados originais para o banco de dados do DW. Essa fase é subdividida em cinco etapas, conforme Barbieri (2001), que serão descritas a seguir.

3.5.1. Filtro de Dados

Esse processo é utilizado para evitar que dados indesejados possam ser carregados para o DW. Um exemplo seria no caso de uma consulta em que se necessita de informações das vendas efetuadas no valor acima de R\$ 100,00. Somente serão selecionadas, na base de origem as informações que se encaixam na condição citada.

3.5.2. Integração de Dados

Esse processo define a forma que deve ser feita a padronização de dados que são correlacionados, mas que são guardados de forma diferentes em bases diferentes, ou mesmo em fontes diferentes. É de essencial importância que essa integração ou padronização seja feita antes de ser carregada as informações para o DW. Um exemplo seria o sexo de uma pessoa, em que numa base ele é guardado como sendo “F” ou “M” e em outra é guardada como sendo “0” ou “1”. Antes de transferir essas informações para o DW, as mesmas devem ser padronizadas.

3.5.3. Condensação de Dados

É a forma de reduzir a quantidade de registros que serão guardadas no DW. Esse processo será utilizado dependendo do tipo de granularidade escolhida para a construção do DW. Um exemplo seria colocar as informações no DW por trimestre ou por ano. Assim, reduziria a quantidade de registros a serem guardados, mas também impossibilitaria uma pesquisa que necessitasse de mais detalhes.

3.5.4. Conversão de Dados

É a definição das formas em que serão transformados os dados de unidades, formatos ou dimensões diferentes. Um exemplo seria quando em uma base original os produtos são cadastrados como caixa e no DW eles devem ser cadastrados como unidade.

3.5.5. Derivação de Dados

É a utilização de formulas para adquirir dados virtuais, a partir de dados existentes nas bases de origem. Sendo utilizada para diminuir os processamentos em tempo de execução de uma consulta no DW. Um exemplo seria acrescentar um campo para o somatório de todas as vendas de um mês numa tabela de fatos do DW.

3.6. DESENVOLVIMENTO DE APLICAÇÕES

Nessa etapa devem ser construídas aplicações que busquem as informações na base de dados do DW, para mostrar as consultas ao usuário final. Essas ferramentas podem ser desenvolvidas tanto para desktop quanto para a Web. Dependendo das necessidades e condições da empresa, podem ser utilizadas ferramentas de *Data Mining*, que irão proporcionar informações de busca de padrões.

Cada empresa define a melhor forma de disponibilizar essas informações para os usuários do sistema, mas a forma mais usada atualmente é a internet, por trazer uma comodidade maior para as empresas.

As preocupações com a interface do sistema devem ser mantidas em todo o processo de desenvolvimento do mesmo, para evitar telas muito carregadas de informações e relatórios sem sentido.

3.7. VALIDAÇÃO E TESTES

Nessa fase o sistema deve ser testado de várias formas, considerando as consultas que mais exigem processamento e trabalhando com um volume maior de informações. Após os testes realizados, o sistema deve ser liberado para algumas pessoas usarem e verificarem se possui alguma falha ou não, como por exemplo, se está faltando alguma informação ou se tem alguma consulta retornando resultados errados.

Após a conclusão dessa fase o sistema pode ser liberado para todos os usuários e deverá ser ministrado um treinamento de como utilizar o sistema. Assim, pode ser considerado que o desenvolvimento do sistema de DW foi concluído, restando somente a fase de manutenção do sistema ou de atualização, dependendo das necessidades da empresa.

4. ESTUDO DE CASO

Uma vez percorrido sobre os conceitos gerais de DW, BD, BI, bem como sobre Oracle e Pentaho, no presente capítulo cabe tão somente desenvolver as propostas para criação do *Data Warehouse* e integrar o BI ao BD usando o Pentaho. Também serão mostrados as modelagens e trechos de códigos para criação de cada componente para complementar o trabalho.

Iniciando-se a implementação do estudo de caso, parte-se dos downloads necessários quanto aos aplicativos para o desenvolvimento das técnicas de DW e BI, com as descrições de cada um dos utilizados, bem como a localização de sua disponibilidade, sendo eles:

- Banco de dados ORACLE DATABASE EXPRESS Edition 11g Release 2. Disponível em: <http://www.oracle.com/technetwork/database/database-technologies/express-edition/downloads/index.html>
- SQL Developer (Versão utilizada 4.1.1), Disponível em: <http://www.oracle.com/technetwork/developer-tools/sql-developer/downloads/index.html>
- Banco de Dados ORACLE DATABASE ENTERPRISE MANAGER 11g, Disponível em: <http://www.oracle.com/technetwork/oem/enterprise-manager/downloads/index.html>
- Pentaho Community Edition
Versões utilizadas:
 - Biserver-ce 5.2.0.0
 - Marketplace 5.2.0.0
 - pdi-ce 5.2.0.0
 - psw-ce 3.10.0.1Disponível em: <http://community.pentaho.com/>
<http://sourceforge.net/projects/pentaho/files/?source=navbar>
- Java SE Development Kit (Versão utilizada 7u79), Disponível em: <http://www.oracle.com/technetwork/pt/java/javase/downloads/jdk7-downloads-1880260.html>

- JDBC Oracle. Disponível em:
<http://www.java2s.com/Code/Jar/j/Downloadjdbcoraclejar.htm>

4.1. CRIAÇÃO DA MODELAGEM DO BANCO RELACIONAL

Para elaboração da modelagem foi levado em conta processos de uma empresa de materiais para construção no qual as principais atividades seriam compras e vendas de produtos, sem ter a necessidade de gerar dados de banco e caixa que fazer parte do financeiro e não gerando também dados para contabilidade da empresa, vendo que esse processo é feito por terceiros.

Após a análise sobre as necessidades básicas de controle foi criado a modelagem do banco relacional das tabelas que possam atender um sistema básico e gerar dados para o DW armazenar em suas dimensões e gerando resultados nas tabelas fato. Assim é possível que a ferramenta de integração (PDI) do Pentaho possa gerar resultados rápidos e consistentes. A figura abaixo mostra a modelagem relacional do banco de dados.

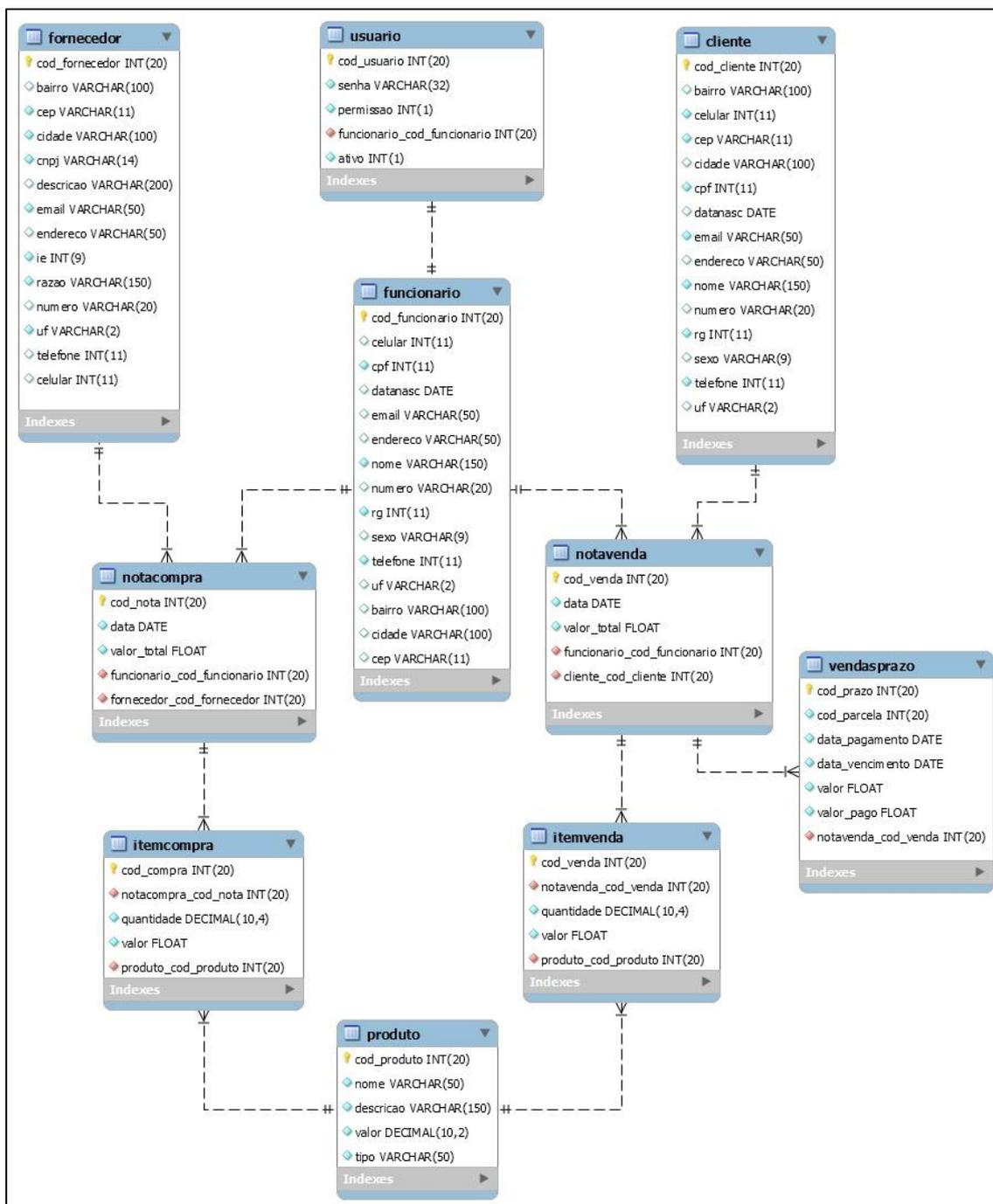


Figura 20 - Modelagem do banco de dados relacional.

4.2. CRIAÇÃO DO MODELO FÍSICO

Após a modelagem estar completa foi feita a implementação dos códigos para criação das tabelas do bando de dados relacional, os scripts necessários para criação baseados na modelagem, podem ser vistos no Apêndice deste trabalho.

Para gerar dados nas tabelas foi feito a conexão com um sistema simples feito em PHP para que as inserções fossem feitas mais rapidamente utilizando formulários simples, segue o código de conexão com a linguagem PHP, pois como essa informação não se encontra de forma fácil na internet, ela servirá de complemento para o trabalho.

```
1  <?php
2  $ora_host      = "127.0.0.1";
3  $ora_user      = "SYSTEM";
4  $ora_senha     = "*****";
5  $port         = "1521";
6  $server_name  = "XE";
7  $charset      = "WE8MSWIN1252";
8  // $dominio    = "http://$_SERVER[SERVER_NAME]/";
9
10 $banco="(DESCRIPTION= (ADDRESS_LIST=
11 (ADDRESS=(PROTOCOL=TCP)(HOST=$ora_host)(PORT=$port)))
12 (CONNECT_DATA=(SERVICE_NAME= $server_name )))";
13 $c = oci_connect($ora_user, $ora_senha, $banco);
14 if (!$c) {
15     $e = oci_error();
16     echo "Erro ao conectar ao Banco Oracle" . $err['message'];
17 }
18 ?>
```

Figura 21 - Script para conexão da linguagem PHP com banco de dados Oracle (Jefferson Quintiliano, 2015).

4.3. CRIAÇÃO DA MODELAGEM DO DW VENDAS

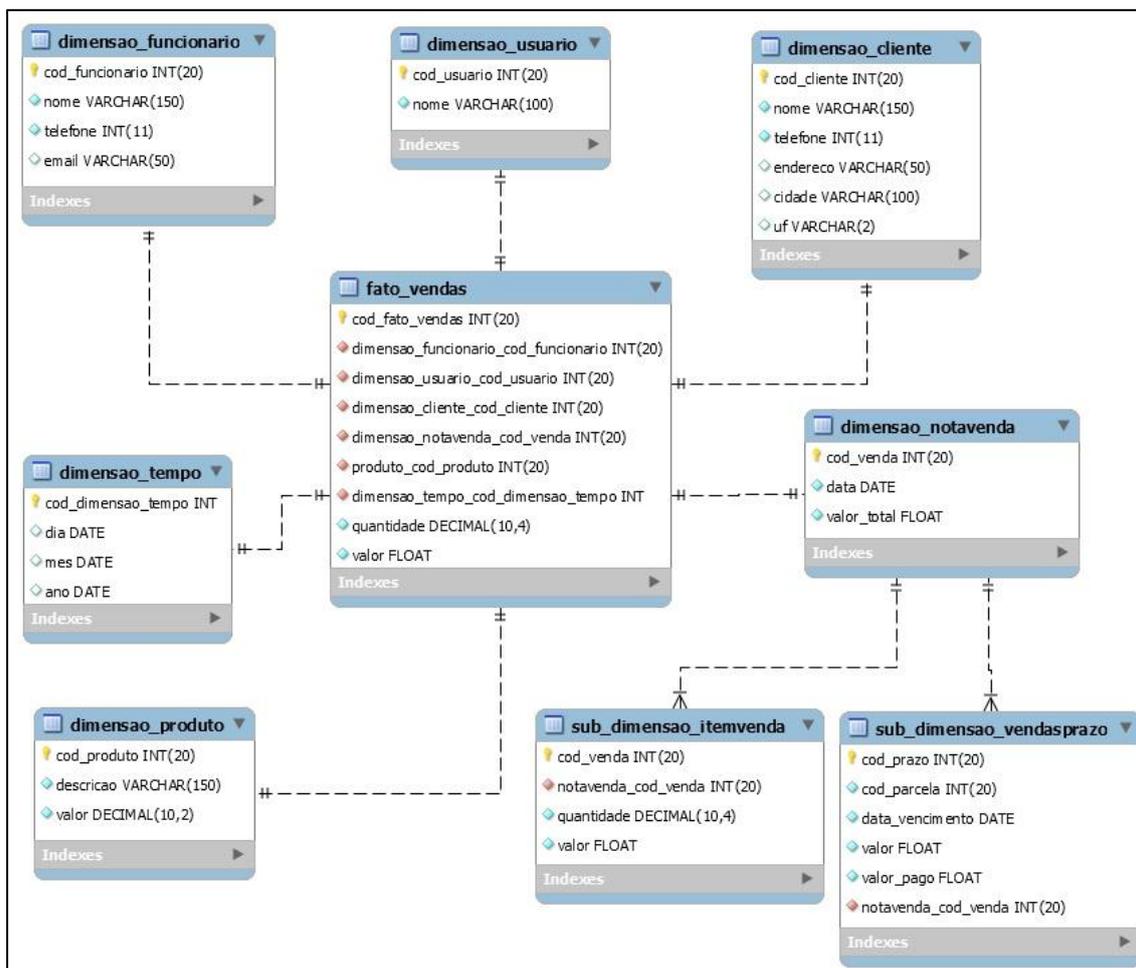


Figura 22 - Modelagem da tabela Fato_Vendas do DW.

Na Figura 26 é possível ver que a tabela fato “vendas” é composta por uma chave primária de identificação e duas métricas sendo elas Quantidade e Valor, as demais colunas são chaves estrangeiras das tabelas dimensionais que contem os dados.

Com esse relacionamento entre as dimensões e a tabela fato é possível gerar resultados através de consultas no banco.

4.4. CRIAÇÃO DA MODELAGEM DO DW COMPRAS

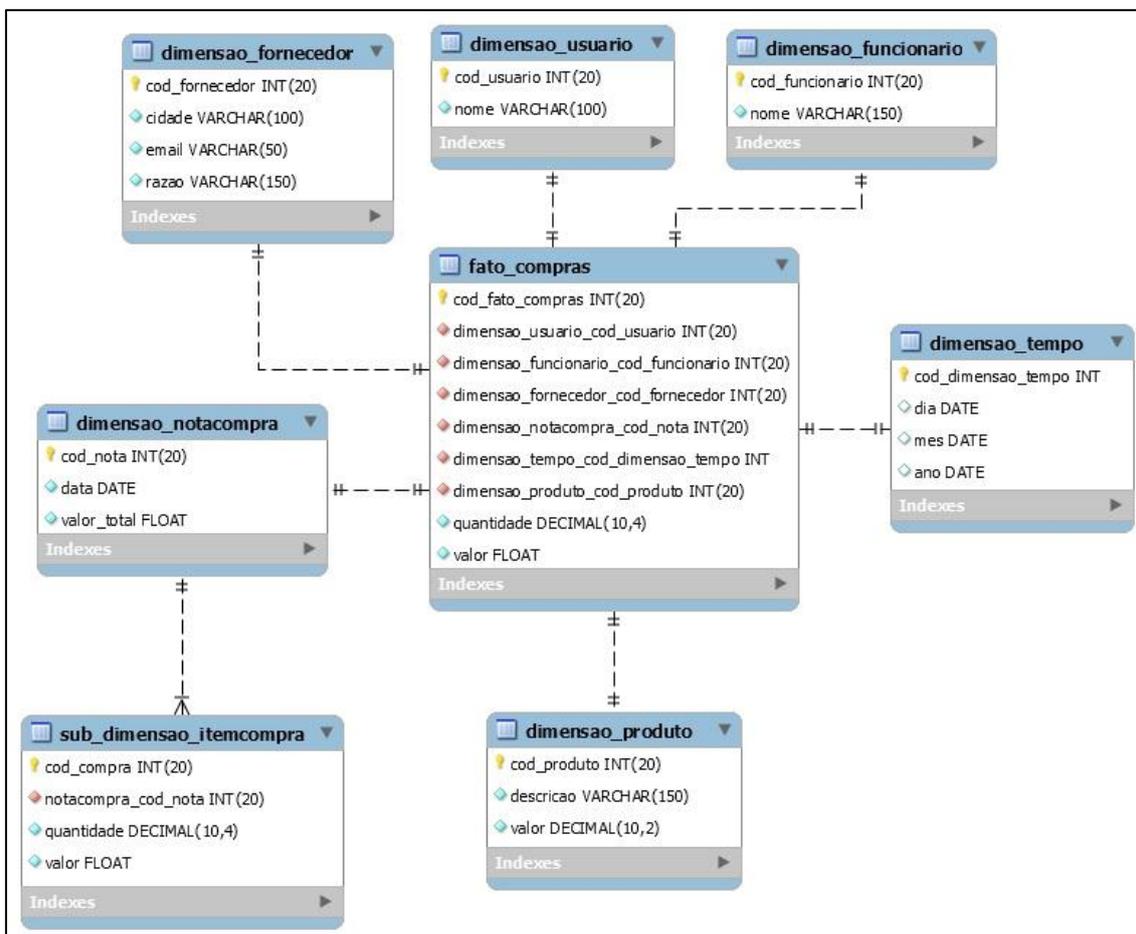


Figura 23 - Modelagem da tabela Fato_Compras do DW.

Na Figura 27 mostra o relacionamento entre a tabela fato compras que é composta por uma chave primária de identificação e duas métricas sendo elas Quantidade e Valor, as demais colunas são chaves estrangeiras das tabelas dimensionais que contem os dados.

Após a criação das duas modelagem o DW pode ser criado de duas formas sendo a primeira:

- 1) Gerar Manualmente: implementando scripts para criação dos modelos dimensionais;
- 2) Gerar Automático: através da ferramenta do *Pentaho Data Integration* (PDI) é possível criar as tabelas de acordo com o banco relacional e estruturado pelo ETL.

Analisando as duas possibilidades e pela quantidade de tabelas, será utilizada o PDI para elaboração do ETL e criação das tabelas dimensionais.

4.5. CRIAÇÃO DO ETL (EXTRACT, TRANSFORM AND LOAD)

Para o processo de ETL, utiliza-se a ferramenta *Spoon* do *Pentaho Data Integration* (PDI), ela permite estruturar o processo de extração, transformação e carga dos dados no DW, e ainda permite detalhar os passos de forma prática.

Após a modelagem do DW estar completa, cria-se primeiro a conexão com o banco de dados Oracle XE (banco relacional) e Oracle ORCL (*Data Warehouse*), e depois é feito o processo de ETL que também serve para criação das tabelas do DW.

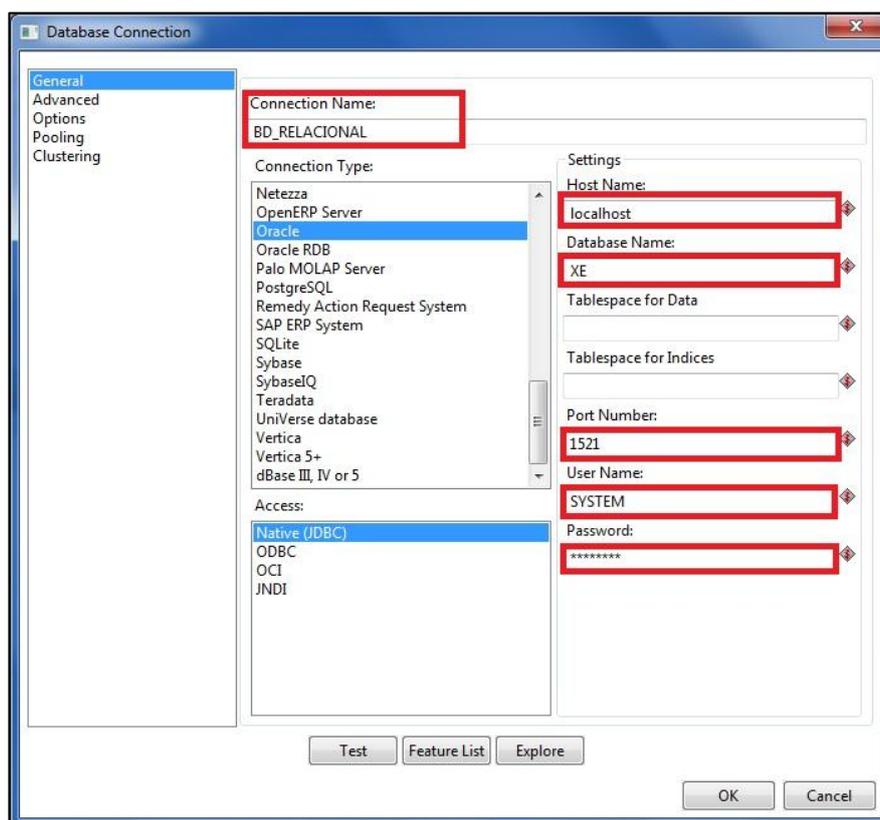


Figura 24 - Conexão PDI Pentaho com Banco Relacional Oracle XE.

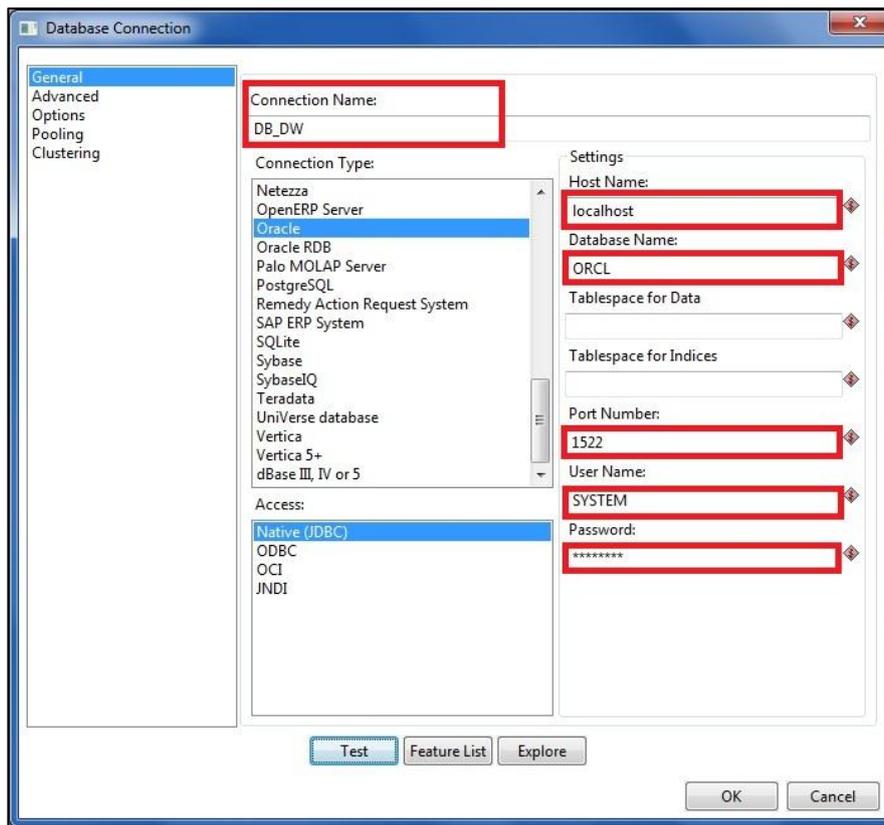


Figura 25 - Conexão PDI Pentaho com Banco Relacional Oracle ORCL.



Figura 26 - ETL Funcionário e criação da tabela DIM_FUNCIONARIO.

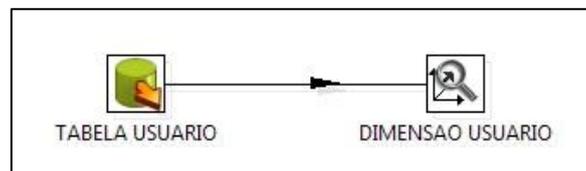


Figura 27 - ETL Usuário e criação da tabela DIM_USUARIO.

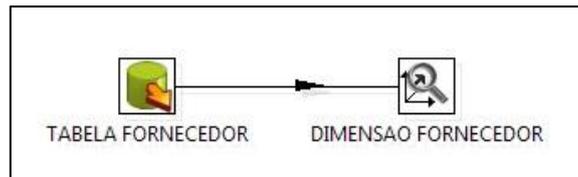


Figura 28 - ETL Fornecedor e criação da tabela DIM_FORNECEDOR.



Figura 29 - ETL Vendas a Praza e criação da tabela DIM_VENDASPRAZO.

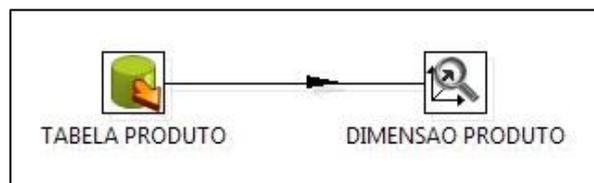


Figura 30 - ETL Produto e criação da tabela DIM_PRODUTO.

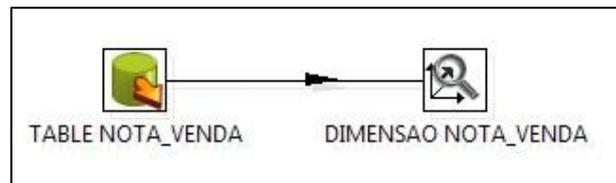


Figura 31 - ETL Nota Venda e criação da tabela DIM_NOTAVENDA.



Figura 32 - ETL Item Venda e criação da tabela DIM_ITEMVENDA.



Figura 33 - ETL Nota Compra e criação da tabela DIM_NOTACOMPRA.

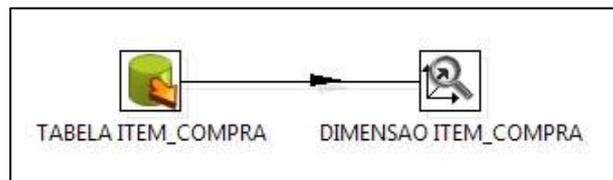


Figura 34 - ETL Item Compra e criação da tabela DIM_ITEMCOMPRA.

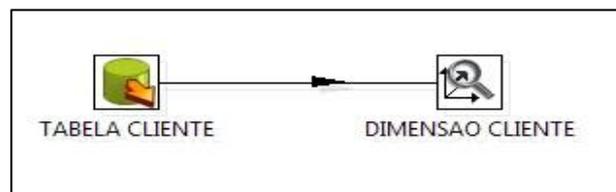


Figura 35 - ETL Cliente e criação da tabela DIM_CLIENTE

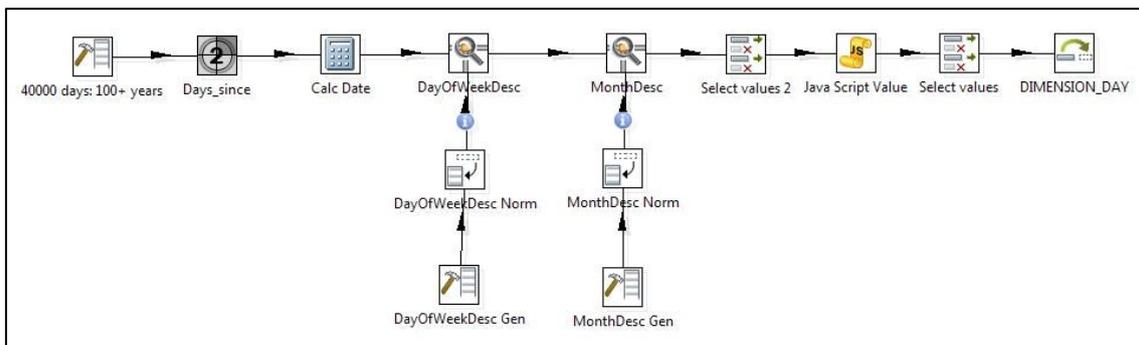


Figura 36 - ETL Data e criação da tabela DIM_DATA.

Na Figura 40, mostra e ETL e a criação da dimensão data foi utilizado um componente do PDI que dispõe de uma serie de datas que vai de 1950 a 2048, isso ajuda no momento de fazer as tabelas fato, pois a comparação por data fica mais simples de ser feita.

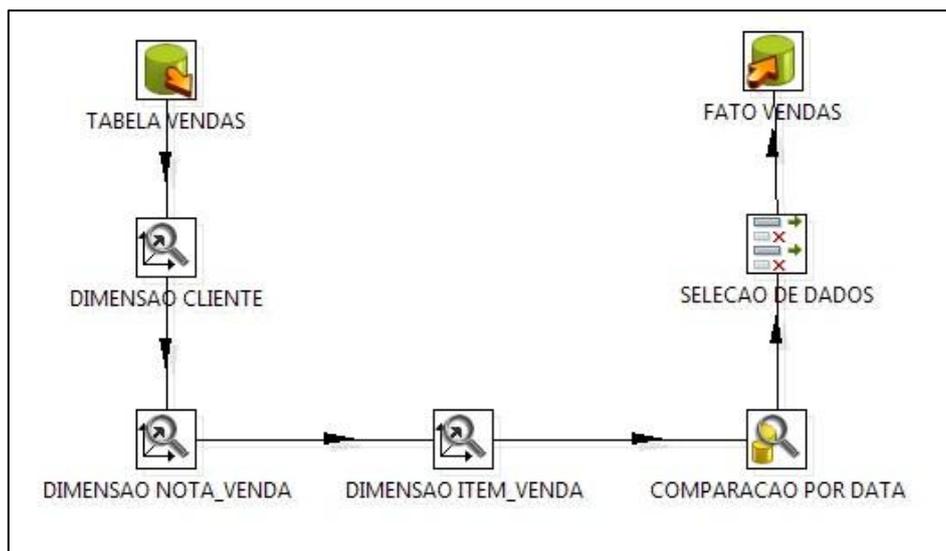


Figura 37 - ETL e criação da tabela FATO_VENDAS.

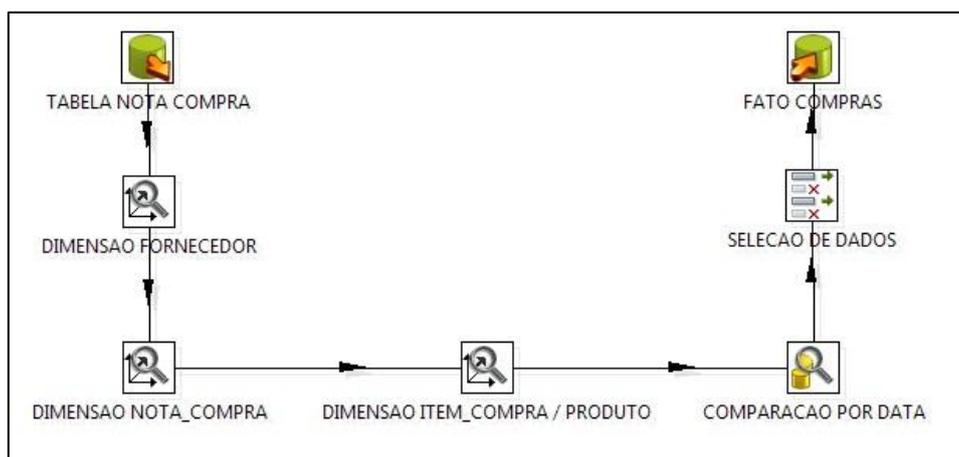


Figura 38 - ETL e criação da tabela FATO_COMPRAS.

Nos processos de ETL das tabelas Fato foi necessário criar uma SELECT para passar as colunas necessárias e fazer alguns INNER JOIN para que as não somente as dimensões, mas principalmente a SELEÇÃO DE DADOS possa ver os campos que serão atribuídos às tabelas fato.

Após a montar a estrutura do processo de ETL é a criação das tabelas dimensionais e fato, então é feito o carregamento dos dados no DW usando a ferramenta Spoon.

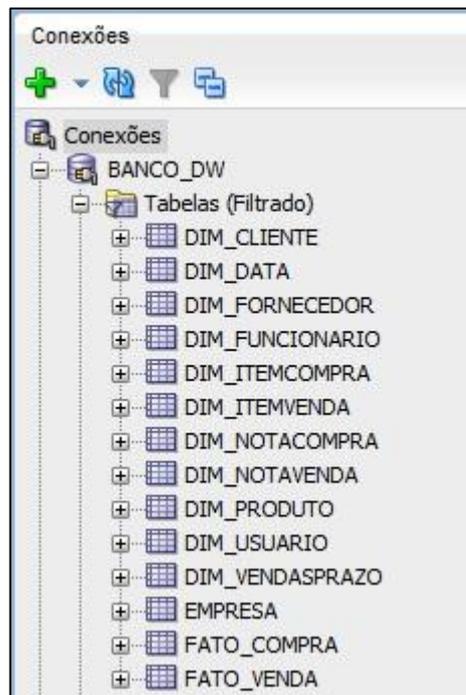


Figura 39 - Tabelas Dimensionais no Banco de Dados Oracle.

4.6. CUBOS OLAP

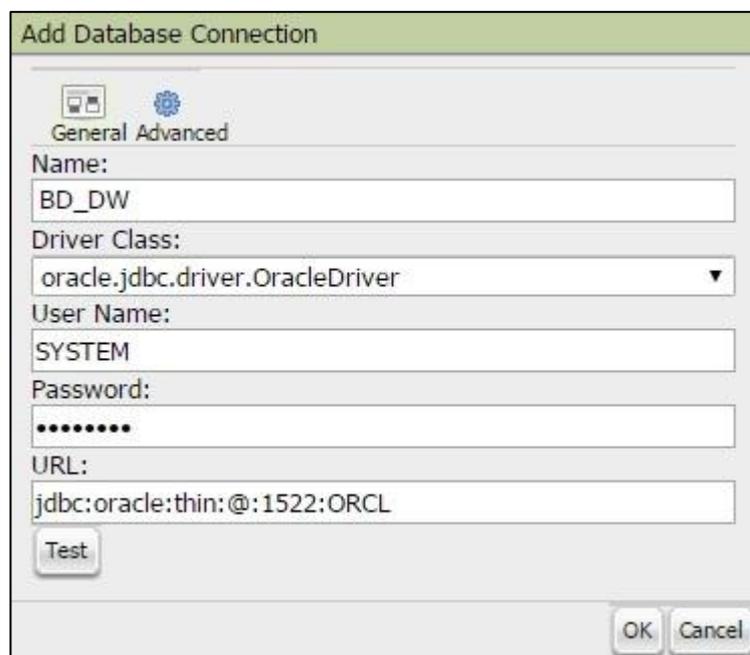


Figura 40 - Conexão do *Administration Console* com Oracle.

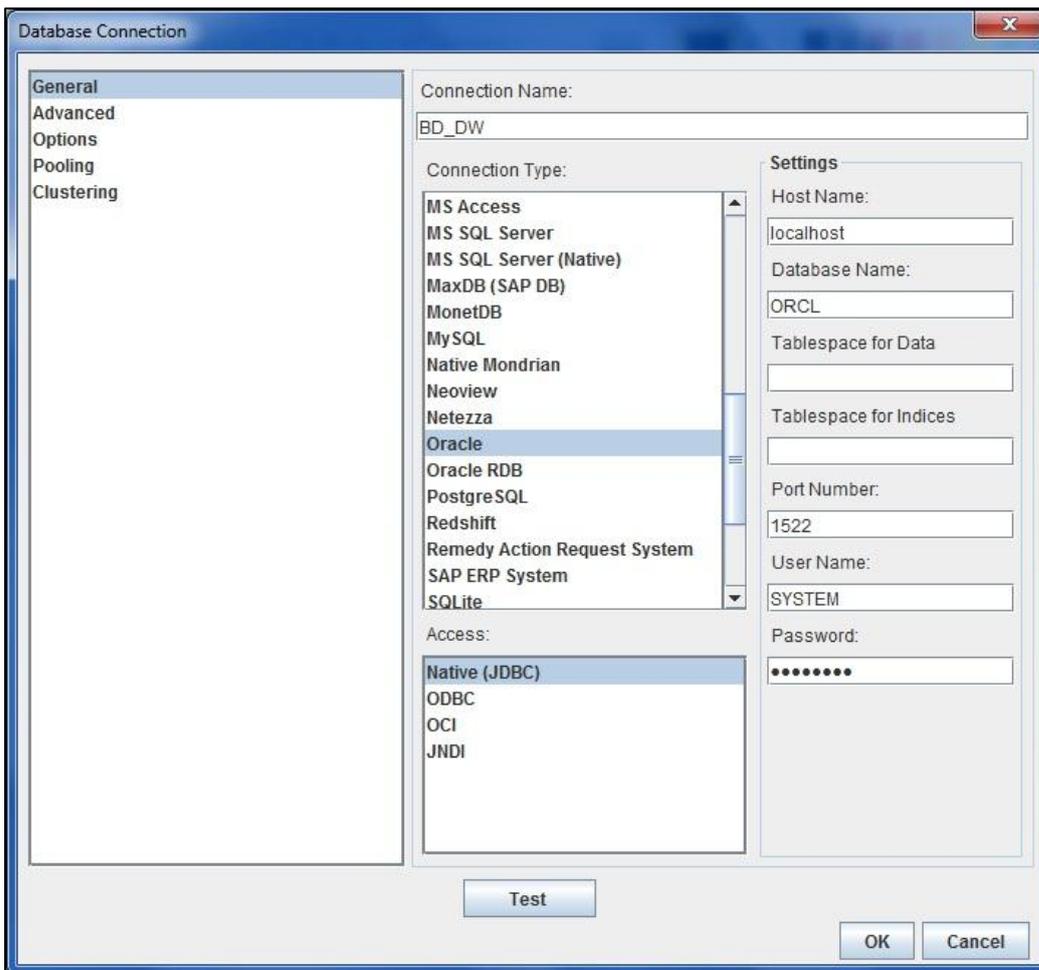


Figura 41 - Conexão do *Schema Workbench* com Oracle DW.

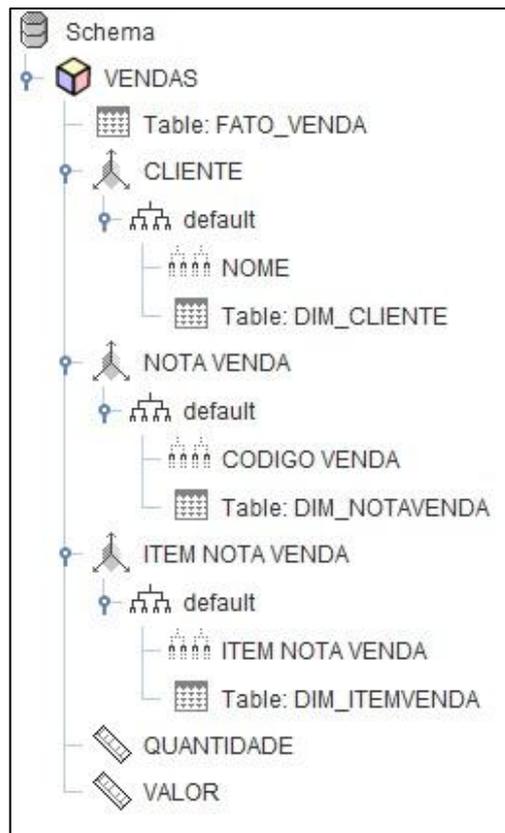


Figura 42 - Cubo Vendas criado no Schema Workbench.

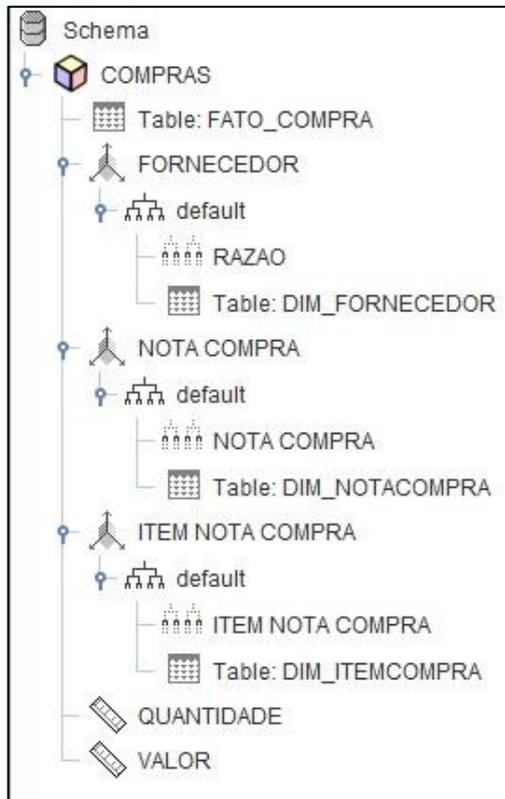


Figura 43 - Cubo Compras criado no Schema Workbench.

4.7. CUBO OLAP NO PENTAHO USER CONSOLE

Utilizando o *Pentaho User Console* (PUC) é possível visualizar os cubos gerados pelo *Schema Workbench* através do JPivot ou Saiku Analytics, nesse nível de acesso é possível visualizar todas as informações relacionadas ao cubo especificado, assim como editar as colunas e filtros de para melhor visualização.

O PUC também permite ver gráficos baseados no cubo analisado, e também permite a criação de *Dashboard* para que usuários finais possam visualizar e acompanhar informações sem tem que acessar JPivot.

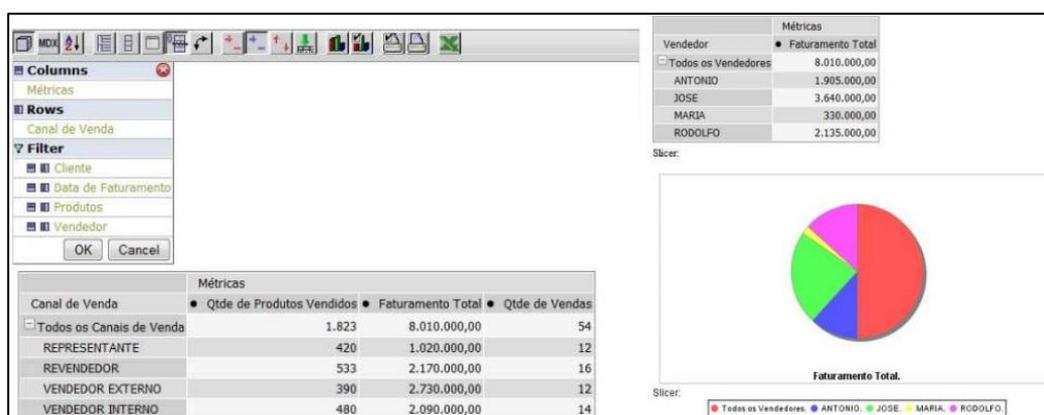


Figura 44 - Visualização do Cubo no JPivot dentro do PUC.

5. CONSIDERAÇÕES FINAIS

Essa pesquisa teve o intuito de apresentar uma das formas de integração de *Business Intelligence* com *Data Warehouse*, mostrando assim recursos para sistemas de apoio à tomada de decisão, onde a transformação das informações podem se tornar armas potentes a serem usadas pelas organizações.

Muitas empresas utilizam aplicações que a um determinado tempo vem realizando as necessidades e satisfazendo-as, mas nos tempos atuais apenas isso não basta para manter-se entre as melhores. Com esse pensamento hoje no mercado existem diversas empresas que já estão utilizando conceito de BI e algumas já estão implementando e instalando essas soluções, e vem crescendo o número de organizações que estão aderindo a essa “onda” de negócio.

Porém muitas também acabam se desestimulando e interferem na continuidade do processo de implantação do BI justamente pelas ferramentas necessitarem de inúmeras instalações e configurações de ambiente para que estejam aptas para serem usadas.

Mas não somente isso, fatores como infraestrutura, colaboradores aptos com conhecimento para criar um ou mais *Data Warehouse* e ainda definir o processo de ETL e Cubos, fora o acesso para usuários das informações através de *Dashboards*, relatórios e gráficos, tem sido outro fator que acaba atrasando a implantação do BI.

Através da pesquisa pode-se observar que conhecimento em estruturas de banco de dados, assim como conceitos, são partes fundamentais não somente para realizar tarefas para criação de DW, mas é de suma importância para o processo de ETL, fundamental para extração, transformação e carga dos dados, além de facilitar a criação de cubos de análise de informações.

A solução Pentaho é *Open Source* e possibilita edição e distribuição, após a utilização da ferramenta foi possível observar que se houver interesse na

edição do *Pentaho User Conselo* (PUC), é necessário possuir conhecimentos básicos em CSS3, Java Script, JSP, HTML, *Designe*.

Entretanto, o estudo não esgota o assunto, mas deixa para pesquisas futuras um caminho aberto, uma proposta lançada para novas implementações e análises sobre DW e BI utilizando Nuvem como base de dados para o DW e Banco de Dados *In-Memory* como base de dados relacional e comunicando-se com o *Business Intelligence*, a Figura a seguir demonstra essa proposta para melhor compreensão.

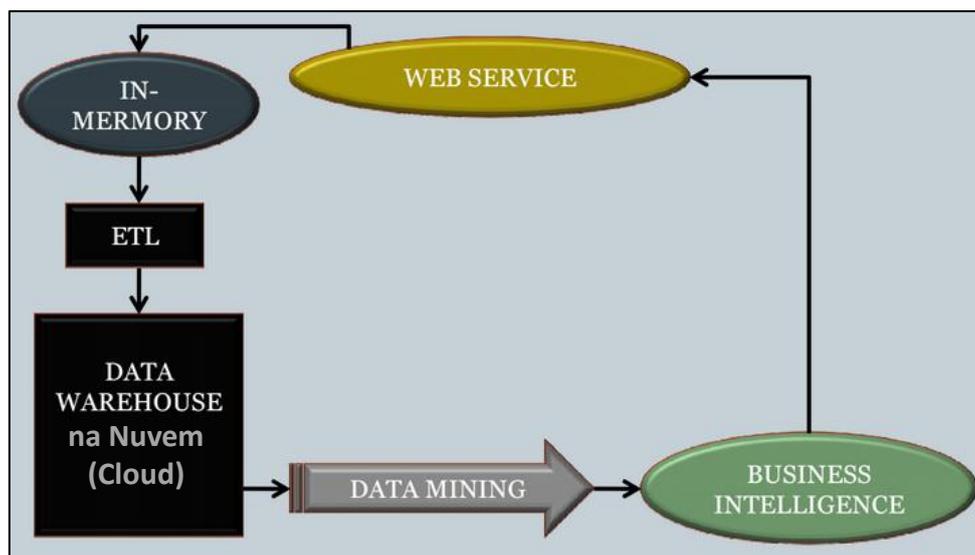


Figura 45 – Proposta de Projetos Futuros (autoria própria).

REFERENCIAS BIBLIOGRÁFICAS

BARBIERI, Carlos. BI – **Business Intelligence**: Modelagem e Tecnologia. Rio de Janeiro: Editora Axcel Books, 2001.

BARQUINI, Ramon 1996, 311 p. - **Planning and designing the Warehouse**, Prentice-Hall, New Jersey.

BERRY, Michael J.A., LINOFF, Gordon. 454 p. **Data Mining techniques: for marketing, sales, and customer support**. J. Wiley & Sons Inc, New York: 1997.

DILL, Sérgio Luis. **Uma Metodologia para Desenvolvimento de Data Warehouse e Estudo de Caso**. 2002. 131 p. Dissertação (Mestrado em Ciências da Computação) - Universidade Federal de Santa Catarina, Santa Catarina. Disponível em: <<http://150.162.90.250/teses/PGCC0318.pdf>>. Acesso em: 02/Fevereiro/2015.

HAND, David; MANNILA, Heikki; SMYTH, Padhraic; **Principles of Data Mining**, August 2001.

HARJINDER, G.; RAO, P. C. – **The Official Guide to Data Warehousing**, Que Corporation, 1996.

INMON, W. H. & Richard D. Hackathorn – **Como usar o Data Warehouse**, Infobook, 2 ed., Rio de Janeiro, 1997.

INMON, W. H. **Como construir o Data Warehouse**. Rio de Janeiro: Editora Campus, 1997.

INMON, W. H. **Building the Data Warehouse**. 3. ed. New York: John Wiley & Sons, 2003.

KIMBALL, Ralph; ROSS, Margy. **The Data Warehouse Lifecycle Toolkit: Expert Methods For Designing, Developing and Deploying Data Warehouses**. John Wiley & Sons Inc., New York: 1998.

KROENKE, David M. **Banco de Dados: Fundamentos, Projeto e Implementação**. 6. Ed. Rio de Janeiro: Editora LTC – Livros Técnicos e Científicos S.A, 1999.

MATA, Leonardo dos Santos da. **Guia para Construção de um Data Warehouse**. Trabalho de Prática de Sistemas de Informação I. Universidade Luterana de Palmas – Palmas, 2005. Disponível em:

<www.ulbrato.br/.../Leonardo%20S%20M%20-%20Estagio%20Supervisionado%20em%20Sistemas%20de%20Informacao.pdf> Acesso em: 20 setembro. 2006.

MACHADO, Felipe Nery Rodrigues. **Tecnologia e projeto de Data Warehouse**. São Paulo: Editora Érica, 2004.

MOREIRA, Eduardo. **Modelo Dimensional para Data Warehouse**.

IMASTERS, Vitória, 27 jan 2006. Disponível em:

<http://www.imasters.com.br/artigo/3836/bi/modelo_dimensional_para_data_warehouse/>. Acesso em: 19/janeiro/2015.

MOSS, L. T.; ATRE, S. **Business Intelligence roadmap: the complete project lifecycle for decision-support applications**. USA: Addison Wesley, 2003.

SCHENATZ, B. Nardelli, **Utilização de Data Mining em um sistema de informação gerencial para o diagnóstico da formação de Professores da Graduação**. Dissertação (Mestrado em Engenharia da Produção) 2005. Programa de Pós-Graduação em Engenharia de redução da Universidade Federal de Santa Catarina. Universidade Federal de Santa Catarina, Santa Catarina, 01/Março/2015.

SIDEMAR SERAIN, J. **Por que Business Intelligence?**. Disponível em: <http://imasters.uol.com.br/artigo/5415/bi/por_que_business_intelligence/>. Acesso em: 17/Outubro/ 2014.

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

SOUZA, Michel de. **Business Intelligence**. IMASTERS, Vitória, 01 outubro 2003. Disponível em:

<http://imasters.com.br/artigo/1409/bi/business_intelligence/>. Acesso em: 25/janeiro/2015.

WESTPHAL, Christopher; BLAXTON, Teresa. **Data mining solutions**. John Wiley & Sons Inc, Canadá, 1998.

OLIVEIRA, Wilson José de. **Data Warehouse**. Editora: Visual Books, 2002.

RIGO, Vagner Freitas. **Estudo da Metodologia de Tuning em Banco de Dados Oracle**. Trabalho de Conclusão do curso de Bacharelado em Ciência da Computação. Fundação Educacional do Município de Assis (FEMA), 2012.

APÊNDICE A – Scripts de Criação do Banco Relacional

```

1 CREATE TABLE CLIENTE (
2   cod_cliente INTEGER NOT NULL, bairro varchar2(100) DEFAULT NULL, celular INTEGER NOT NULL,
3   cep varchar2(11) NOT NULL, cidade varchar2(100) DEFAULT NULL, cpf INTEGER NOT NULL,
4   datanasc date DEFAULT NULL, email varchar2(50) NOT NULL, endereco varchar2(50) DEFAULT NULL,
5   nome varchar2(150) NOT NULL, numero varchar2(20) DEFAULT NULL, rg INTEGER NOT NULL,
6   sexo varchar2(9) DEFAULT NULL, telefone varchar2(50) NOT NULL, uf varchar2(2) DEFAULT NULL,
7   PRIMARY KEY (cod_cliente)
8 );

```

Script para criação da tabela Cliente.

```

1 CREATE TABLE FORNECEDOR (
2   cod_fornecedor INTEGER NOT NULL, bairro VARCHAR2(100) DEFAULT NULL, cep VARCHAR2(11) NOT NULL,
3   cidade varchar2(100) NOT NULL, cnpj varchar2(14) NOT NULL, descricao varchar2(200) DEFAULT NULL,
4   email varchar2(50) NOT NULL, endereco varchar2(50) DEFAULT NULL, ie INTEGER NOT NULL,
5   razao varchar2(150) NOT NULL, numero varchar2(20) DEFAULT NULL, uf varchar2(2) NOT NULL,
6   telefone INTEGER DEFAULT NULL, celular INTEGER DEFAULT NULL,
7   PRIMARY KEY (cod_fornecedor)
8 );

```

Script para criação da tabela Fornecedor.

```

1 CREATE TABLE FUNCIONARIO (
2   cod_funcionario INTEGER NOT NULL, celular INTEGER DEFAULT NULL, cpf INTEGER NOT NULL,
3   datanasc date DEFAULT NULL, email varchar2(50) DEFAULT NULL, endereco varchar2(50) DEFAULT NULL,
4   nome varchar2(150) NOT NULL, numero varchar2(20) DEFAULT NULL, rg INTEGER NOT NULL,
5   sexo varchar2(9) DEFAULT NULL, telefone INTEGER NOT NULL, uf varchar2(2) DEFAULT NULL,
6   bairro varchar2(100) DEFAULT NULL, cidade varchar2(100) DEFAULT NULL, cep varchar2(11) DEFAULT NULL,
7   PRIMARY KEY (cod_funcionario)
8 );

```

Script para criação da tabela Funcionário.

```

1 CREATE TABLE PRODUTO (
2   cod_produto INTEGER NOT NULL, nome varchar2(50) NOT NULL, descricao varchar2(150) NOT NULL,
3   valor NUMBER(10,2) NOT NULL, tipo varchar2(50) NOT NULL,
4   PRIMARY KEY (cod_produto)
5 );
6
7 CREATE TABLE USUARIO (
8   cod_usuario INTEGER NOT NULL, senha varchar2(32) NOT NULL, permissao INTEGER NOT NULL,
9   funcionario_cod_funcionario INTEGER DEFAULT NULL, nome varchar2(20) NOT NULL, ativo INTEGER NOT NULL,
10  PRIMARY KEY (cod_usuario),
11  CONSTRAINT fk_usuario_funcionario1 FOREIGN KEY (funcionario_cod_funcionario)
12  REFERENCES funcionario (cod_funcionario)
13 );

```

Script para criação da tabela Produto.

```

1 CREATE TABLE notacompra (
2   cod_nota INTEGER NOT NULL, data date NOT NULL, valor_total FLOAT NOT NULL,
3   funcionario_cod_funcionario INTEGER NOT NULL, fornecedor_cod_fornecedor INTEGER NOT NULL,
4   PRIMARY KEY (cod_nota),
5   CONSTRAINT fk_notacompra_fornecedor1 FOREIGN KEY (fornecedor_cod_fornecedor)
6   REFERENCES fornecedor (cod_fornecedor),
7   CONSTRAINT fk_notacompra_funcionario1 FOREIGN KEY (funcionario_cod_funcionario)
8   REFERENCES funcionario (cod_funcionario)
9 );
10
11 CREATE TABLE notavenda (
12   cod_venda INTEGER NOT NULL, data date NOT NULL, valor_total float NOT NULL,
13   funcionario_cod_funcionario INTEGER NOT NULL, cliente_cod_cliente INTEGER NOT NULL,
14   PRIMARY KEY (cod_venda),
15   CONSTRAINT fk_notavenda_cliente1 FOREIGN KEY (cliente_cod_cliente)
16   REFERENCES cliente (cod_cliente),
17   CONSTRAINT fk_notavenda_funcionario1 FOREIGN KEY (funcionario_cod_funcionario)
18   REFERENCES funcionario (cod_funcionario)
19 );

```

Script para criação da tabela Nota Compra e Nota Venda.

```

1 CREATE TABLE itemcompra (
2   cod_compra INTEGER NOT NULL, notacompra_cod_nota INTEGER NOT NULL,
3   quantidade NUMBER(10,4) NOT NULL, valor float NOT NULL, produto_cod_produto INTEGER NOT NULL,
4   PRIMARY KEY (cod_compra),
5   CONSTRAINT fk_itemcompra_notacompra1 FOREIGN KEY (notacompra_cod_nota)
6   REFERENCES notacompra (cod_nota),
7   CONSTRAINT fk_itemcompra_produto1 FOREIGN KEY (produto_cod_produto)
8   REFERENCES produto (cod_produto)
9 );
10
11 CREATE TABLE itemvenda (
12   cod_item_venda INTEGER NOT NULL, notavenda_cod_venda INTEGER NOT NULL,
13   quantidade NUMBER(10,4) NOT NULL, valor float NOT NULL, produto_cod_produto INTEGER NOT NULL,
14   PRIMARY KEY (cod_item_venda),
15   CONSTRAINT fk_itemvenda_notavenda1 FOREIGN KEY (notavenda_cod_venda)
16   REFERENCES notavenda (cod_venda),
17   CONSTRAINT fk_itemvenda_produto1 FOREIGN KEY (produto_cod_produto)
18   REFERENCES produto (cod_produto)
19 );

```

Script para criação da tabela Item Compra e Item Venda.

```

1 CREATE TABLE vendasprazo (
2   cod_prazo INTEGER NOT NULL, cod_parcela INTEGER NOT NULL, data_pagamento date NOT NULL,
3   data_vencimento date NOT NULL, valor float NOT NULL, valor_pago float NOT NULL,
4   notavenda_cod_venda INTEGER NOT NULL,
5   PRIMARY KEY (cod_prazo),
6   CONSTRAINT fk_vendasprazo_notavenda1 FOREIGN KEY (notavenda_cod_venda)
7   REFERENCES notavenda (cod_venda)
8 );

```

Script para criação da tabela Vendas a Prazo.