

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE CIÊNCIAS DA COMPUTAÇÃO
(Bacharelado)

**PROTÓTIPO DE *DATA WAREHOUSE* APLICADO A
COMPANHIA DE SEGUROS DE AUTOMÓVEIS**

TRABALHO DE CONCLUSÃO DE CURSO SUBMETIDO À UNIVERSIDADE
REGIONAL DE BLUMENAU PARA A OBTENÇÃO DOS CRÉDITOS NA
DISCIPLINA COM NOME EQUIVALENTE NO CURSO DE CIÊNCIAS DA
COMPUTAÇÃO — BACHARELADO

KELVIN JACOB WARMELING

BLUMENAU, JUNHO/1999

1999/1-33

PROTÓTIPO DE *DATA WAREHOUSE* APLICADO A COMPANHIA DE SEGUROS DE AUTOMÓVEIS

KELVIN JACOB WARMELING

ESTE TRABALHO DE CONCLUSÃO DE CURSO, FOI JULGADO ADEQUADO
PARA OBTENÇÃO DOS CRÉDITOS NA DISCIPLINA DE TRABALHO DE
CONCLUSÃO DE CURSO OBRIGATÓRIA PARA OBTENÇÃO DO TÍTULO DE:

BACHAREL EM CIÊNCIAS DA COMPUTAÇÃO

Prof. Wilson Pedro Carli — Orientador na FURB

Prof. José Roque Voltolini da Silva — Coordenador do TCC

BANCA EXAMINADORA

Prof. Wilson Pedro Carli

Prof. Oscar Dalfovo

Prof. Everaldo A. Grahl

DEDICATÓRIA

Dedico este trabalho,

A meus pais (Jacob Warmeling e Martha Dimon Warmeling), familiares, parentes e amigos, principalmente àqueles que com seu carinho e otimismo, em momento algum nos negam uma palavra de conforto e solidariedade. E especialmente às pessoas que souberam me compreender nestes anos de escalada, até a minha conclusão do Curso. Também ao meu cunhado e minha irmã, que me emprestaram o computador para que eu concluísse o trabalho da melhor forma possível. E minha namorada que soube compreender tudo de forma coerente durante a conclusão do trabalho.

AGRADECIMENTOS

Agradeço a todas as pessoas que colaboraram para a conclusão deste Trabalho de Conclusão de Curso – TCC, onde muitos contribuíram direta ou indiretamente, durante as várias fases deste percurso, em especial ao professor Wilson Pedro Carli, meu orientador.

O mesmo foi fator determinante para o aumento dos conhecimentos, tanto no âmbito pessoal como no âmbito acadêmico.

No âmbito pessoal, a grande contribuição de cada pessoa que em seu cotidiano, permitiu que fossem extraídas muitas lições de vida, importantes para o meu crescimento humano. No aspecto acadêmico, o exemplo mais recente, sem dúvida, foi o início, o desenvolvimento e a conclusão deste trabalho.

A um “SER SUPERIOR”, presente em cada momento de minha vida.

“Para a pessoa que não sabe o seu caminho, qualquer caminho serve”.

Autor: Dr. Lair Ribeiro

SUMÁRIO

DEDICATÓRIA	III
AGRADECIMENTOS	IV
SUMÁRIO.....	V
LISTA DE FIGURAS	VII
LISTA DE TABELAS.....	VII
RESUMO	IX
ABSTRACT	X
1 INTRODUÇÃO.....	1
1.1 OBJETIVOS.....	2
1.1.1 OBJETIVOS ESPECÍFICOS.....	2
1.2 ORGANIZAÇÃO DO TEXTO.....	3
2 HISTÓRICO DO SEGURO.....	4
2.1 AS COMPANHIAS SEGURADOS E OS SEGUROS.....	6
2.2 ÁREAS DE ATUAÇÃO NO MERCADO SEGURADOR.....	6
2.2.1 RAMOS ELEMENTARES.....	6
2.2.2 RAMO BENEFÍCIOS.....	7
2.3 SEGURO DE AUTOMÓVEIS.....	8
3 DATA WAREHOUSE	11
3.1 CARACTERÍSTICAS DE <i>DATA WAREHOUSE</i>	13
3.2 DADOS PRIMITIVOS E DADOS DERIVADOS.....	15
3.3 AMBIENTE DE <i>DATA WAREHOUSE</i>	16
3.4 ARQUITETURA GENÉRICA DE <i>DATA WAREHOUSE</i>	19
3.5 GRANULARIDADE.....	21
3.6 PARTICIONAMENTO.....	24
3.7 PARTICIONAMENTO DE DADOS.....	25
3.8 METADADOS.....	27
3.9 DIMENSIONALIDADE.....	30
3.10 PROCESSAMENTO ANALÍTICO ONLINE (OLAP).....	30
3.11 ESQUEMA FUNCIONAL DO <i>DATA WAREHOUSE</i>	32
3.12 <i>DATA WAREHOUSE</i> X <i>DATA MART</i>	33

4	MIGRAÇÃO DE DADOS E SUA IMPORTÂNCIA	34
4.1	PROCESSOS ASSOCIADOS	35
4.1.1	PROCESSO DE EXTRAÇÃO.....	35
4.1.2	PROCESSO DE LIMPEZA	36
4.1.3	PROCESSO DE TRANSFORMAÇÃO.....	36
4.1.4	PROCESSO DE TRANSPORTE (MIGRAÇÃO)	37
4.2	COMPLEXIDADES DO PROCESSO MIGRATÓRIO.....	38
4.3	CONSIDERAÇÕES SOBRE O DESENVOLVIMENTO DE UM DATA WAREHOUSE	39
4.3.1	ALTERNATIVAS DE IMPLANTAÇÃO	40
4.3.2	ASPECTOS DE MODELAGEM.....	41
4.4	FERRAMENTAS PARA MIGRAÇÃO	42
5	FERRAMENTAS	44
5.1	PROTOTIPAÇÃO	44
5.1.1	VANTAGENS DA PROTOTIPAÇÃO	44
5.1.2	FASES DA PROTOTIPAÇÃO.....	45
5.1.2.1	EXAME DE VIABILIDADE DO PROJETO	45
5.1.2.2	IDENTIFICAÇÃO DE NECESSIDADES E REQUISITOS	46
5.1.2.3	DESENVOLVIMENTO DO MODELO VIVO DE TRABALHO.....	46
5.1.2.4	DEMONSTRAÇÃO E USO DO MODELO	46
5.1.2.5	REVISÃO E MELHORAMENTOS	46
5.1.2.6	UTILIZAÇÃO DO PROTÓTIPO	47
5.2	DELPHI	47
5.2.1	CARACTERÍSTICAS	47
5.2.2	RECURSOS	48
5.2.3	AMBIENTE DE DESENVOLVIMENTO.....	48
5.3	SYSTEM ARCHITECT	50
5.4	ROTEIRO PARA PROJETO DE <i>DATA WAREHOUSE</i> SEGUNDO W.H. INMON	50
5.4.1	JUSTIFICATIVA DE CUSTOS	50
5.4.2	BUSCA PELOS DADOS OPERACIONAIS	51
5.4.3	Complexidade de transformação e integração.....	53
6	DESENVOLVIMENTO DO PROTÓTIPO	57
6.1	VIABILIDADE.....	57
6.2	MIGRAÇÃO DE DADOS PARA <i>DATA WAREHOUSE</i> CONFORME CAPÍTULO 4.....	57
6.3	NECESSIDADES E REQUISITOS	58
6.4	DESENVOLVIMENTO DO MODELO VIVO	62
7	CONCLUSÃO.....	66
	REFERÊNCIAS BIBLIOGRÁFICAS.....	68

LISTA DE FIGURAS

FIGURA 1 - COMPONENTE DE UM <i>DATA WAREHOUSE</i>	14
FIGURA 2 - NÍVEIS DA ARQUITETURA DE DADO NUM AMBIENTE <i>DATA WAREHOUSE</i>	16
FIGURA 3 - UM EXEMPLO DE DADOS BASEADOS EM ASSUNTOS/NEGÓCIOS	17
FIGURA 4 - A QUESTÃO DA INTEGRAÇÃO	17
FIGURA 5 - A QUESTÃO DA NÃO-VOLATILIDADE	18
FIGURA 6 - A QUESTÃO DA VARIAÇÃO EM RELAÇÃO AO TEMPO	19
FIGURA 7 - ARQUITETURA GENÉRICA DE <i>DATA WAREHOUSE</i>	20
FIGURA 8 - NÍVEIS DE GRANULARIDADE	22
FIGURA 9 - PARTIÇÕES DE DADOS GERENCIADAS INDEPENDENTEMENTE PODE SER ENVIADAS A DIFERENTES COMPLEXOS DE PROCESSAMENTO SEM ACARRETAR CONSEQÜÊNCIA PARA OS SISTEMAS	25
FIGURA 10 - METADADOS PARA DADOS NÃO ESTRUTURADOS E DADOS DE FONTES EXTERNAS.	29
FIGURA 11 - MODELO GENÉRICO DA METODOLOGIA DE PROTOTIPAÇÃO	45
FIGURA 12 - TELA PRINCIPAL DO DELPHI 3.	49
FIGURA 13 - TELA DE DEFINIÇÃO DAS TABELAS.	49
FIGURA 14 – PASSAGEM DOS DADOS DOS SISTEMAS TRANSACIONAIS PARA O <i>DATA WAREHOUSE</i> [INM97]	51
FIGURA 15 – DIAGRAMA DE CONTEXTO <i>DATA WAREHOUSE</i>	59
FIGURA 16 - DIAGRAMA DE FLUXO DE DADOS (DFD) <i>DATA WAREHOUSE</i> - NÍVEL 0	59
FIGURA 17 - DETALHA-SE O MODELO ENTIDADE X RELACIONAMENTO DO <i>DATA WAREHOUSE</i>	61
FIGURA 18 - TELA PRINCIPAL DO SISTEMA DE <i>DATA WAREHOUSE</i>	62
FIGURA 19 - TELA DE CADASTRO CORRETOR	62
FIGURA 20 - TELA DE CADASTRO DE APÓLICE	63
FIGURA 21 - TELA DE CADASTRO DE SINISTROS	64
FIGURA 22 - TELA DE PARAMETRIZAÇÃO E RESULTADO DA PESQUISA	65

LISTA DE TABELAS

TABELA 1 - COMPARAÇÃO ENTRE BANCO DE DADOS OPERACIONAIS E <i>DATA WAREHOUSE</i>	13
TABELA 2 - COMPARAÇÃO ENTRE DADOS PRIMITIVOS E <i>DADOS DERIVADOS</i>	15
TABELA 3 - NÍVEIS DUPLOS DE GRANULARIDADE PARA DADOS RESUMIDOS.	24
TABELA 4 - COMPARATIVO OLTP E OLAP	31
TABELA 5 - TABELA DE FERRAMENTAS COMERCIAIS PARA AUXILIO À MIGRAÇÃO.	43
TABELA 6 – DICIONÁRIO DE DADOS DO SISTEMA DE <i>DATA WAREHOUSE</i>	61

RESUMO

Este trabalho tem por objetivo, demonstrar a filosofia de *Data Mart*, que pode ser utilizada por uma companhia Seguradora, auxiliando na tomada de decisão sobre o preço dos seguros de automóveis. Pela abrangência da tecnologia, será focado seus principais conceitos e componentes, uma visão da aquisição dos dados existentes e identificação dos metadados. Paralelamente será desenvolvido um protótipo para visualização e transformação dos metadados em informação para auxiliar a comercialização de Seguro de Automóveis. A filosofia *Data Warehouse*, será apresentada com suas divisões e os conceitos que fazem necessários para implantação da tecnologia, além, de um estudo das informações que são consideradas estratégicas nas companhias seguradoras para análise e tomada de preços para seguro de automóveis.

ABSTRACT

This work has for objective, to demonstrate Data Mart philosophy, that it can be used by a company Insurance company, aiding in the taking of decision on the price of the insurances of automobiles. For the inclusion of the technology, it will be focused its main concepts and components, a vision of the acquisition of the existent data and identification of the metadados. Parallely a prototype will be developed for visualization and transformation of the metadados in information to aid the commercialization of Insurance of Automobiles. The philosophy Dates Warehouse, it will be presented with its divisions and the concepts that do necessary for implantação of the technology, beyond, of a study of the information that you/they are considered strategic in the companies insurance companies for analysis and taking of prices for insurance of automobiles.

1 INTRODUÇÃO

De acordo com [DAL98], a evolução da Tecnologia de Informação e o crescimento do uso de computadores interconectados, praticamente todas as empresas de médio e grande porte estão utilizando sistemas informatizados para realizar seus processos mais importantes, o que com o passar do tempo acaba gerando uma enorme quantidade de dados relacionados aos negócios, mas não relacionados entre si. Estes dados armazenados nos sistemas operacionais da empresa são um recurso, mas de modo geral, raramente servem como recurso estratégico no seu estado original. Os sistemas convencionais de informática não são projetados para gerar e armazenar as informações estratégicas, o que torna os dados vagos e sem valor para o apoio ao processo de tomada de decisões das organizações.

Em termos simples, um *Data Warehouse*, ou em português, Armazém de Dados, pode ser definido como um banco de dados especializado, o qual integra e gerencia o fluxo de informações a partir dos bancos de dados corporativos e fontes de dados externas à empresa. Um *Data Warehouse* é construído para que tais dados possam ser armazenados e acessados de forma que não sejam limitados por tabelas e linhas estritamente relacionais. A função do *Data Warehouse* é tornar as informações corporativas acessíveis para o seu entendimento, gerenciamento e uso.

Um *Data Warehouse* oferece os fundamentos e os recursos necessários para um Sistema de Apoio a Decisão (SAD) eficiente, fornecendo dados integrados e históricos que servem desde a alta direção, que necessita de informações mais resumidas, até as gerências de baixo nível, onde os dados detalhados ajudam a observar aspectos mais táticos da empresa. Nele, os executivos podem obter de modo imediato respostas para perguntas que normalmente não possuem respostas em seus sistemas operacionais e, com isso, tomar decisões com base em fatos, não com intuições ou especulações.

Estudar e conhecer a tecnologia de *Data Warehouse* pode ajudar os empresários a descobrir novas formas de competir, sendo que estamos convivendo com a globalização, onde percebemos a concorrência acirrada entre as grandes corporações em todo o mundo. Diante deste fato, necessitamos das informações que estão armazenadas nos bancos de dados, para que diante desta informações os executivos possam tomar decisões importantes, procurando outros nichos de mercado ainda não explorados pelas empresas. Desta maneira,

disponibilizam melhores produtos ou serviços para o mercado, mais rápido do que os concorrentes, sem aumentar o custo do produto ou do serviço. Não existe ainda metodologia formal para implementação de um *Data Warehouse*, elas devem ser adaptadas às características e às expectativas de cada empresa, mas o principal objetivo em todas elas é o de descobrir maneiras diferentes de atuar no mercado e quais as mudanças internas que devem ocorrer para atender as novas realidades.

Em princípio, um *Data Warehouse* provê um banco de dados especializado que gerencia o fluxo de informações a partir dos bancos de dados corporativos e fonte de dados externa à empresa, para aplicações do usuário final. A função do *Data Warehouse* é tornar as informações corporativas acessíveis para entendimento, gerência e uso [CAR97].

Neste contexto, este trabalho, visa o desenvolvimento de um protótipo para implantação em Seguradora no ramo de automóveis, que possibilitará uma maior verificação dos sinistros pagos, quais os tipos de veículos (somente serão avaliados os veículos da marca Volkswagen) que ocorrem mais sinistros, as pessoas que neles estão dirigindo e os locais de ocorrência. Desta forma, pode-se verificar individualmente os veículos e os motoristas (idade e sexo) com maior incidência de Sinistro/Acidentes.

A implementação do protótipo foi utilizado o carregamento dos dados no Banco de Dados *Paradox do Delphi 3*, com dados reais, para simulação da tecnologia na decisão de tomada de preço para seguro de automóveis nas seguintes abordagens: Modelo do veículo, ano do veículo, motorista (idade, sexo, tempo de carteira).

1.1 OBJETIVOS

O principal objetivo deste trabalho é:

- a) demonstrar a filosofia de *Data Warehouse* e como pode ser utilizada em uma Companhia Seguradora;

1.1.1 OBJETIVOS ESPECÍFICOS

Os objetivos específicos são:

- a) identificação das maneiras de comercialização e estratégias de mercado de uma companhia seguradora;

- b) estudo da tecnologia *Data Warehouse*;
- c) utilizar a técnica, para desenvolvimento de um plano para a aquisição dos dados (informações necessárias para tomada de preços);
- d) facilitar a identificação dos sinistros pagos pela seguradora, e proporcionar maior lucratividade na comercialização de seguros de automóveis.
- e) estudo de alternativas para migração de dados;
- f) apresentação das metodologias utilizadas para o desenvolvimento do protótipo;
- g) demonstração da tecnologia através da construção de um protótipo utilizando o banco de dados Paradox do Delphi 3;
- h) teste e validação do protótipo.

1.2 ORGANIZAÇÃO DO TEXTO

O texto será organizado em 7 (sete) capítulos, distribuídos conforme segue:

Primeiramente será apresentado as objetivos e organização do trabalho.

O segundo capítulo apresenta os principais ramos de seguros oferecidos no mercado, a organização, seu faturamento, resultado operacional.

O terceiro capítulo apresenta um abordagem teórica do assunto de *Data Warehouse*, discorrendo sobre os principais conceitos e componentes associados ao tema.

O quarto capítulo será abordado aspectos relacionados com a migração de dados, quais ferramentas e a interface com o usuário do *Data Warehouse*.

O quinto capítulo descreve a metodologia de prototipação genérica e a ferramenta Delphi 3, utilizadas para o desenvolvimento do protótipo.

O sexto será apresentada a metodologia utilizada no desenvolvimento do protótipo, e o desenvolvimento do mesmo.

O sétimo são expostas as conclusões obtidas pelo autor em relação ao tema pesquisado.

2 HISTÓRICO DO SEGURO

Segundo [BRA98], na Babilônia, há 23 séculos antes de Cristo, atravessar o deserto em caravanas, para vender seus camelos nas cidades vizinhas, representava perigo para os camaleiros. O medo, a incerteza e as ameaças à integridade de seu patrimônio os levaram a buscar uma forma de proteção. Assim, fizeram um acordo entre si: cada criador que perdesse um animal, por morte ou desaparecimento, teria a garantia de outro, pago pelos demais criadores.

Sem dúvida, este acordo era uma atitude solidária entre componentes de um mesmo grupo. Mas, visto de outro ângulo, já representava uma forma primária de seguro: sem despesas, o criador atingido pelo prejuízo da perda tinha essa despesa eliminada pelos demais componentes do grupo, através da reposição de outro animal, igual ao perdido.

Os hebreus e os fenícios também adotaram prática semelhante. Grandes navegadores que eram, seus barcos cruzavam constantemente os mares Egeu e Mediterrâneo. Os riscos eram sempre grandes e, mais uma vez, membros de um mesmo grupo procuraram garantir-se entre si contra eventuais prejuízos. E os navegadores fizeram um acordo que garantia, a quem perdesse um navio, a construção de outro, pago pelos demais participantes da mesma viagem.

Como se pode notar, até então as perdas cobertas pelos acordos eram feitas pela reposição do objeto perdido. Mas, uma nova modalidade de seguros surge no século XII (depois de Cristo) chamada Contrato de Dinheiro a Risco Marítimo, e formalizava, através de documento assinado, acordo entre duas pessoas. Uma delas, o financiador, emprestava dinheiro ao navegador, no valor do barco e das mercadorias transportadas. Se durante a viagem, o barco sofresse algum acidente, o dinheiro emprestado não era devolvido. Caso contrário, esse dinheiro voltava para o financiador, acrescido de juros.

A preocupação com o transporte marítimo era muito grande, e a causa desta preocupação era eminentemente econômica, pois o comércio exterior dos países se dava apenas por mar, e a concretização da venda das mercadorias, era a única garantia da economia desses países. Numa visão mais geral, pode-se verificar que essa preocupação de garantir a economia dos países através do seguro existe até hoje. Apenas a forma do seguro é que mudou e se aperfeiçoou cada vez mais.

Verificou-se que a importância do comércio marítimo continuou a deixar a sua marca na história do seguro. Certamente não foi por acaso que o primeiro contrato de seguro, nos moldes atuais, foi firmado na cidade de Gênova, na Itália. Corria o ano de 1347, quando foi firmado um contrato de seguro pela primeira vez com emissão de apólice. E, mais uma vez, era um contrato de seguro de transporte marítimo.

Posteriormente, foram teorias e desenvolvimentos científicos que vieram impulsionar o seguro. Assim é que a Teoria das Probabilidades, desenvolvida por Pascal, associada à estatística, foi uma grande responsável pelo impulso que teve o seguro, pois a partir de então, os valores dos riscos e dos prêmios de seguros puderam ser calculados de forma mais justa. Nota-se que esta influência foi tão grande, que tais critérios permanecem até hoje.

Com o advento da máquina e da Era Industrial, no século XIX, surgiram e se desenvolveram outras modalidades de seguro, como o de incêndio, para proteção das máquinas, o de transportes terrestres, garantindo a movimentação dos bens produzidos, e o de vida visando a garantia de um pecúlio familiar.

Desde a Revolução Industrial, o seguro tem feito frente aos riscos criados pelo avanço tecnológico e pela complexidade geral da nova sociedade. Há novos meios de transporte, indústrias muito complicadas, profissões perigosas, usa-se o automóvel massivamente, as pessoas têm cada vez uma consciência mais apurada de seus direitos de reclamar os danos sofridos pela atuação de outras pessoas, etc.

A imaginação do homem é que concebeu a idéia do seguro. Este se tornou o meio encontrado pelo homem para sua proteção, prevenindo-o de prejuízos econômicos ocasionados pela destruição de seus bens. Firmou-se, assim, um conceito universal aceito: é mais fácil suportar coletivamente as conseqüências de eventos individuais, do que deixar o indivíduo só e isolado, exposto a essas conseqüências.

Os tempos modernos assistem ao aparecimento de entidades de seguros que administram volumes extraordinariamente elevadas de prêmios, com influência econômica ou social crescente, e que empregam dezenas de milhares de pessoas que se chamam de Companhias Seguradoras.

2.1 AS COMPANHIAS SEGURADOS E OS SEGUROS

Conforme [BRA98], companhias de seguros são instituições que operam a política traçada pelo Conselho Nacional de Seguros Privados, a elas, cabem administrar bem os seguros que lhes são confiados, que são vendidos através de profissionais chamados de corretores de seguros.

Seguro é a operação pela qual o segurado, mediante o pagamento de um prêmio e observância de cláusulas de um contrato, obriga o segurador a responder perante ele, por prejuízos ocorridos no objeto do seguro.

O objeto de seguro são coisas, bens, pessoas, responsabilidades, obrigações, direitos e garantias que se quer garantir contra as conseqüências danosas previstas no contrato.

2.2 ÁREAS DE ATUAÇÃO NO MERCADO SEGURADOR

Os seguro dividem-se em dois ramos, conforme[BRA98]:

- a) ramos elementares - caracteriza-se por seguros de bens e de responsabilidade civil, como por exemplo o seguro de automóvel, incêndio, transporte, etc.
- b) ramo benefícios - caracteriza-se por seguros de pessoas, como por exemplo o seguro de vida, saúde, acidentes pessoais coletivo, etc.

2.2.1 RAMOS ELEMENTARES

Nos Ramos Elementares, que compreendem os seguros de Automóveis, Incêndio, Transportes, Cascos Marítimos, Aeronáuticos e Responsabilidade Civil, entre outros, a Bradesco Seguros, juntamente com as demais seguradoras do Grupo, tem ocupado posição de liderança no mercado brasileiro. Nestes ramos, foram registradas vendas que chegaram a R\$ 1,281 bilhões em 1997, o que correspondeu a cerca de 45% do total de prêmios de seguro arrecadados [BRA98].

Os seguros de residências dobraram em 1997. A liderança no ramo de Transportes foi mantida.

O lançamento do Seguro Bradesco Náutico durante o XV Salão Náutico Internacional, no Rio de Janeiro, em abril, multiplicou por dez o total de prêmios neste ramo somente

durante o primeiro semestre de 1997. O sucesso alcançado por essa modalidade de seguro, inovadora no mercado, levou a Bradesco Seguros a receber o prêmio *Marketing Best 97*.

Conforme [BRA98], as privatizações e concessões de serviços público contribuíram para ampliar o mercado de seguros. Antigas estatais, hoje sob comando privado, tornaram-se clientes, estabelecendo parcerias que se estenderam para outros tipos de operações de seguros. A Bradesco Seguros acrescentou à sua carteira, em 1997, novos seguros para plataformas de petróleo, cascos marítimos, aeronáuticos, siderurgia, mineração, portos, rodovias, ferrovias, energia, telecomunicações, indústrias têxteis, de papel e celulose e de alumínio.

As coberturas para os Ramos Elementares estão abaixo discriminadas:

- a) incêndio - define que a cobertura básica do seguro Incêndio é dada contra os seguintes riscos básicos: incêndio, queda de raio e explosão de gás de aparelhos de uso doméstico ou usado em iluminação.
- b) automóvel - o seguro de automóvel garante ao segurado o reembolso de prejuízos materiais que este venha a sofrer no veículo segurado em consequência dos seguintes riscos: colisão, incêndio e roubo.
- c) Transportes – Verificou-se, que o ramo de transportes divide-se em três modalidades e para cada uma das modalidades abaixo, existem condições particulares:
 - marítimo, fluvial e lacustre;
 - terrestre, ferroviário e rodoviário;
 - aéreo.

2.2.2 RAMO BENEFÍCIOS

No Ramo Benefícios, que compreendem os seguros de Saúde, Vida e Acidentes Pessoais, a Bradesco Seguros, faturou em 1997, o montante de R\$ 1,566 bilhão, com crescimento de 25,58% em relação a 1996. Este ramo representa 55% do total da carteira de seguros da Bradesco.

O Ramo de seguro de vida, de acordo com [BRA98], divide-se em três tipos de seguros:

- a) vida individual - se destina a pessoa física, garantindo o pagamento de indenização aos beneficiários, ou ao próprio segurado, no caso de sua sobrevivência, após um prazo determinado;
- b) vida em grupo - se destina a garantir o pagamento de uma indenização ao beneficiário do segurado, no caso de sua morte;
- c) acidentes pessoais coletivo - se destina a garantir o pagamento de uma indenização ao beneficiário do segurado, no caso de morte por acidente, ou ao próprio segurado no caso de invalidez por acidente, cabendo também o reembolso das despesas médico-hospitalares também decorrente de acidente.

O Ramo de seguro saúde, mais conhecido com “Seguro Saúde”, divide-se em dois tipos de seguros:

- a) o seguro saúde Individual; e
- b) seguro saúde empresa.

O objetivo deste seguro é ressarcir as despesas com cirurgia, desde o material empregado até honorários médicos, estadias em hospitais, tratamentos e consultas que o segurado venha a necessitar, já regularizado conforme normas governamentais.

2.3 SEGURO DE AUTOMÓVEIS

Na área de seguro de Automóveis, retomou-se uma política de vendas mais agressiva, o que possibilitou que, em 1997, a participação deste ramo aumentasse de 19% para 25,57% no total da carteira de seguros, com um volume de prêmios de R\$728 milhões.

A comercialização da cobertura opcional de Assistência Auto Dia e Noite foi intensificada no exercício, com cerca de 80% dos veículos segurados passando a dispor deste serviço que, durante o ano de 1997, foi utilizado por aproximadamente 100 mil segurados.

Lançou-se, ainda, a cobertura de Auto Reserva, com grande aceitação entre os clientes, que fornece, em caso de perda total, um veículo reserva por até 30 dias e, nos casos de colisão, por até 15 dias.

Atualmente a Bradesco Seguros S/A, tem os seguros comercializados através do profissional chamado, corretor de seguros e através das agências do Banco Bradesco em todo o Brasil.

O objetivo do seguro de automóvel é garantir ao segurado a reparação dos danos causados ao seu veículo do segurado e a terceiros, e reposição de outro bem do mesmo modelo no caso de perda total ou roubo do veículo. O seguro de automóveis poderá ser contratado para os mais diversos modelos de veículos: automóveis de passeio, automóveis esportivos, caminhões, picapes, taxi, rebocador e semi-reboque.

As coberturas para os veículos são as seguintes:

- a) colisão, incêndio, roubo ou furto (total ou parcial do veículo);
- b) raio ou explosão;
- c) submersão em caso de enchentes, granizo, furacão e terremoto;
- d) colisão ou capotagem;
- e) queda em precipícios ou de pontes;
- f) queda sobre o veículo de qualquer objeto externo.

Os riscos excluídos no seguro de automóveis são:

- a) perturbações de ordem pública (guerra, greve, rebelião, tumulto, etc);
- b) convulsões da natureza;
- c) veículos em trânsito por estradas ou caminhos impedidos ao tráfego ou areias fofas;
- d) má conservação;
- e) danos ocorridos durante a participação do veículo em gincanas, apostas, corridas, etc;
- f) quando o veículo estiver sendo dirigido por motorista embriagado ou drogado;
- g) motorista não habilitado;

Os veículos segurados também poderão contratar as cobertura de Danos Materiais e Danos Pessoais, causados pelo veículo a terceiros, assistência Dia e Noite (24 horas), seguro de Acidentes Pessoais e Passageiros, para os ocupantes do veículo, carro reserva, diárias de paralisação ou Perda de Renda, cobertura para acessórios e equipamentos. Os veículos tem cobertura em todo o território nacional inclusive nos países do Mercosul.

Atualmente, a Bradesco Seguros S/A, tem o seu plano de vendas de seguro de automóveis, voltada ao tipo de veículo que cada segurado possui. Isto significa que não depende do perfil do segurado (idade, tempo de carteira, estado civil, etc). O prêmio do seu seguro (prêmio – valor que o segurado paga), muitas vezes não corresponde com o perfil do motorista do veículo. Os prêmios do seguro tem variação pelo tipo de veículo de cada segurado (esportivo, passeio, utilitários, transportes, etc;). Uma das questões que estão em pauta hoje é, porque um segurado que sai com o seu carro uma vez por semana, paga o mesmo valor de outro que viaja 5 (cinco) dias por semana. A probabilidade de sinistro de ambos os segurados é totalmente oposta. Desta forma, o trabalho tem o objetivo de auxiliar na verificação dos segurados e/ou motorista que mais tem sinistros, verificando a idade, sexo, estado civil e tempo de carteira. Possibilitando desta forma uma melhor adequação do perfil do segurado e/ou motorista com o valor do seguro a ser pago.

3 DATA WAREHOUSE

Conforme a atual bibliografia podem ser encontrados muitos conceitos sobre *DATA WAREHOUSE*, como os apresentados a seguir:

- a) segundo [GON97], *Data Warehouse* é uma nova *buzzword* (conceito) reiterando uma velha promessa. Esse conceito, embora surgido recentemente, baseia-se na aplicação de antigas idéias que somente agora puderam ser viabilizadas pela conjunção de diferentes tecnologias, tais como: banco de dados relacionais e multidimensionais, interface gráfica, PC, sistemas operacionais em rede, discos rígidos de grande capacidade de armazenamento e velocidade de acesso, OLAP, entre outros;
- b) segundo [INM97], que é tido como o pai do conceito, *Data Warehouse* é uma coleção de dados orientados por assuntos, integrados, variáveis com o tempo e não voláteis, para dar suporte ao processo gerencial de tomada de decisão;
- c) *Data Warehouse* é um processo em andamento que aglutina dados de fontes heterogêneas, incluindo dados históricos e dados externos para atender à necessidade de consultas estruturadas e *ad-hoc*, relatórios analíticos e de suporte a decisão, conforme Harjinder [HAR96];
- d) segundo [BAR96], *Data Warehouse* é uma coleção de técnicas e tecnologias que juntas disponibilizam um enfoque pragmático e sistemático para tratar com o problema do usuário final de acessar informações que estão distribuídas em vários sistemas da organização.

Para entender o que é um *Data Warehouse*, é importante fazer uma comparação com o conceito tradicional de banco de dados. Conforme [BAT86], "Um banco de dados é uma coleção de dados operacionais armazenados e utilizados pelo sistema de aplicações de uma empresa específica". Os dados mantidos por uma empresa são chamados de "operacionais" ou "primitivos". Refere-se aos dados no banco de dados como "dados operacionais", distinguindo-se de dados de entrada, dados de saída e outros tipos de dados.

Levando em consideração esta definição sobre dados operacionais, pode-se dizer que um *Data Warehouse* é, na verdade, uma coleção de dados derivados dos dados operacionais para sistemas de suporte à decisão. Estes dados derivados são, muitas vezes, referidos como dados "gerenciais", "informacionais" ou "analíticos" [INM97].

Os bancos de dados operacionais armazenam as informações necessárias para as operações diárias da empresa, são utilizados por todos os funcionários para registrar e executar operações pré-definidas, por isso seus dados podem sofrer constantes mudanças conforme as necessidades atuais da empresa. Por não ocorrer redundância nos dados e as informações históricas não ficarem armazenadas por muito tempo, este tipo de banco de dados não exige grande capacidade de armazenamento [DAL98].

Já um *Data Warehouse* armazena dados analíticos, destinados às necessidades da gerência no processo de tomada de decisões. Isto pode envolver consultas complexas que necessitam acessar um grande número de registros, por isso é importante a existência de muitos índices criados para acessar as informações da maneira mais rápida possível. Um *Data Warehouse* armazena informações históricas de muitos anos e por isso deve ter uma grande capacidade de processamento e armazenamento dos dados que se encontram de duas maneiras, detalhados e/ou resumidos [DAL98], na tabela 1 estão relacionadas algumas diferenças entre bancos de dados operacionais e *Data Warehouse*, bem como as diferenças dos dados que eles manipulam.

Com base nestes conceitos pode-se concluir que o *Data Warehouse* não é um fim, mas sim um meio que as empresas dispõem para analisar informações históricas podendo utilizá-las para a melhoria dos processos atuais e futuros. *Data Warehouse* são resumos de dados retirados de múltiplos sistemas de computação normalmente utilizados a vários anos e que continuam em operação. *Data Warehouse* são construídos para que tais dados possam ser armazenados e acessados de forma que não sejam limitados por tabelas e linhas estritamente relacionais. Os dados de um *Data Warehouse* podem ser compostos por um ou mais sistemas distintos e sempre estarão separados de qualquer outro sistema transacional, ou seja, deve existir um local físico onde os dados desses sistemas serão armazenados, conforme [DAL98], a Figura 1 ilustra os principais componentes de um *Data Warehouse*, mostrando que entre as fontes de dados e os acessos a estes dados está o *Data Warehouse*. Antes deste deslocamento, sempre haverá a aplicação de técnicas de filtragem, agrupamento e/ou refinamento dos dados.

Características	Bancos de dados Operacionais	<i>Data Warehouse</i>
Objetivo	Operações diárias do negócio	Analisar o negócio
Uso	Operacional	Informativo
Tipo de processamento	OLTP	OLAP
Unidade de trabalho	Inclusão, alteração, exclusão	Carga e consulta
Número de usuários	Milhares	Centenas
Tipo de usuário	Operadores	Comunidade gerencial
Interação do usuário	Somente pré-definida	Pré-definida e <i>ad-hoc</i>
Condições dos dados	Dados operacionais	Dados Analíticos
Volume	Megabytes – gigabytes	Gigabytes – terabytes
Histórico	60 a 90 dias	5 a 10 anos
Granularidade	Detalhados	Detalhados e resumidos
Redundância	Não ocorre	Ocorre
Características	BD operacionais	<i>Data Warehouse</i>
Estrutura	Estática	Variável
Manutenção desejada	Mínima	Constante
Acesso a registros	Dezenas	Milhares
Atualização	Contínua (tempo real)	Periódica (em <i>batch</i>)
Integridade	Transação	A cada atualização
Número de índices	Poucos/simples	Muitos/complexos
Intenção dos índices	Localizar um registro	Aperfeiçoar consultas

Tabela 1 - Comparação entre Banco de Dados Operacionais e *Data Warehouse*.
 Fonte: Dal'Alba, Adriano. Um Estudo Sobre *Data Warehouse*, 1998.

3.1 CARACTERÍSTICAS DE *DATA WAREHOUSE*

Conforme definição de alguns autores o *Data Warehouse* é um banco de dados contendo dados extraídos do ambiente de produção da empresa, que foram selecionados e depurados, tendo sido otimizados para processamento de consulta e não para processamento de transações. Em geral, um *Data Warehouse* requer a consolidação de outros recursos de dados além dos armazenados em Banco de Dados relacionais, incluindo informações provenientes de planilhas eletrônicas, documentos textuais, etc [CAM98].

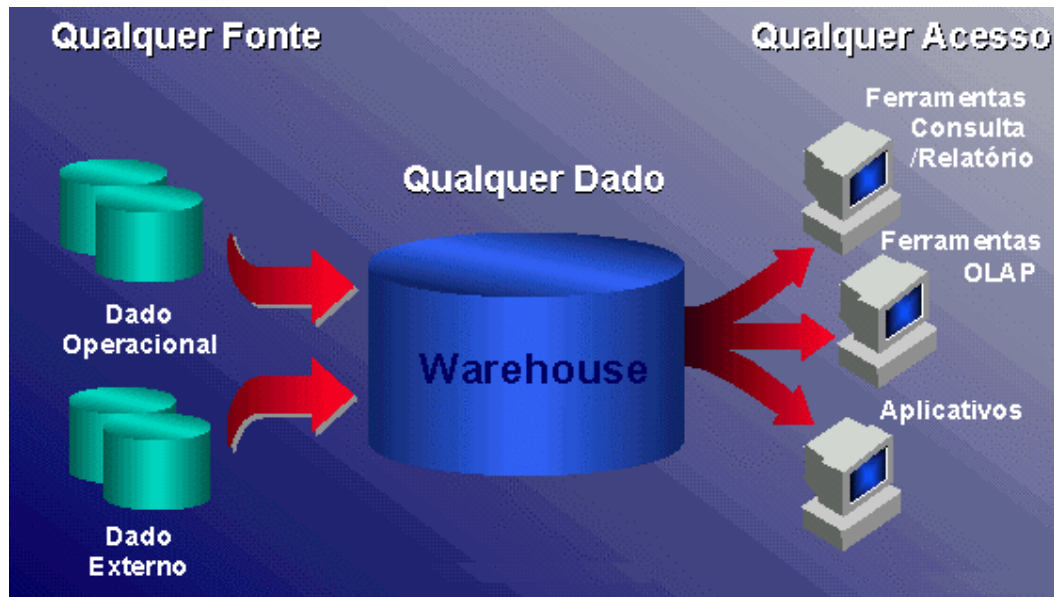


Figura 1 - Componente de um *Data Warehouse*.
 Fonte: Dal'Alba, Adriano. Um Estudo Sobre *Data Warehouse*, 1998.

De acordo com Richard Hackathorn (outro pioneiro no tema), o objetivo de um *Data Warehouse* é fornecer uma "imagem única da realidade do negócio". De uma forma geral, sistemas de *Data Warehouse* compreendem um conjunto de programas que extraem dados do ambiente de dados operacionais da empresa, um banco de dados que os mantém, e sistemas que fornecem estes dados aos seus usuários.

Nas últimas décadas as empresas tem se esforçado para aumentar seus sistemas de processamento de transações on-line (OLTP). O subproduto desses esforços é o vasto depósito de descrição de dados, virtualmente cada aspecto do desempenho operacional de negócios. As informações que não podem ser coletadas internamente são fornecidas por meio de fontes independentes em resposta a uma necessidade sempre crescente de inteligência de mercado [HAR98].

Os *Data Warehouse* são geralmente criados para permitir que os usuários tenham acesso aos dados e sejam compatíveis com o OLAP necessários à tomada de decisão. Uma classe específica de *Data Warehouse*, referida como repositório de dados operacionais, "armazena" realmente os dados, removendo os critérios não-voláteis aplicados a um *Data Warehouse* para apoio a decisões gerenciais. Nesse aspecto, os repositórios de dados operacionais estão mais próximos de um banco de dados OLTP.

3.2 DADOS PRIMITIVOS E DADOS DERIVADOS

Segundo [IMN97], há fundamentalmente duas espécies de dados - dados primitivos e dados derivados - e não há razão plausível para estas duas espécies conviverem no mesmo ambiente de banco de dados.

Os dados primitivos são essencialmente dados de origem operacional, envolvidos nas atividades rotineiras das empresas, ao passo que os dados derivados são subprodutos de compilações, cálculos, sintetizações, classificações e outras operações que o transformam em informação útil para o processo decisório.

A tabela 2, de conformidade com [INM97], apresenta as principais características próprias de cada espécie de dados:

Dados Primitivos	Dados Derivados
Baseados em aplicações	Baseados em assuntos ou negócios
Detalhados	Resumidos ou refinados
Representam valores correntes	Representa valores históricos
Atendem à comunidade funcional	Atendem à comunidade gerencial
Alta probabilidade de atualização	Baixa probabilidade de atualização
Processados repetitivamente	Processados de forma heurística
Requisitos conhecidos previamente	Requisitos desconhecidos previamente
Gerados no ciclo de vida convencional	Ciclo de vida próprio e diferenciado
Ênfase na performance	Performance atenuada
Acessados de forma unitária	Acessados por conjunto
Voltados para transações	Voltados para análise
Atualizações controladas pelo usuário	Atualizações controladas no processo
Alta disponibilidade	Disponibilidade atenuada
Gerenciados em sua totalidade	Gerenciados por subconjuntos
Não contempla a redundância	Redundância não pode ser ignorada
Estrutura fixa; conteúdos variáveis	Estrutura flexível
Processados em pequenos volumes	Processados em grandes volumes
Atendem às necessidades cotidianas	Voltados para necessidades gerenciais
Alta probabilidade de acesso	Remota probabilidade de acesso

Tabela 2 - Comparação entre Dados primitivos e *Dados Derivados*.

Fonte: [INM97]

Em decorrência da classificação dos dados em primitivos e derivados e dentro de um conceito moderno de projetar os sistemas de apoio à decisão, os dados assumem a

possibilidade de serem organizados de maneira a formar níveis ou camadas, no intuito de atender às diferentes necessidades de informações.

Um modelo desta arquitetura assim, como suas principais características e composição é apresentado na figura 2, de acordo com proposição de [INM97].

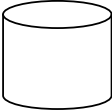
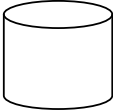
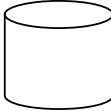
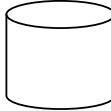
 Operacional	 Data Warehouse	 Departamental	 Individual
Detalhado Cotidiano Valores atuais Alta chance de acesso Baseado em sistemas aplicativos	Mais granular Variável no tempo Integrado Baseado em negócios Dados levemente resumidos	Paroquial Muitos dados derivados Poucos dados primitivos Característico de departamentos	Temporário Ocasional Heurísticos Não repetitivo Baseado em PC ou Estações de trabalho

Figura 2 - Níveis da arquitetura de dado num ambiente *Data Warehouse*.
Fonte: [INM97]

Interessante notar que o nível departamental pode receber outras conotações – *Data Mart*, *OLAP*, Bancos Multidimensionais – sem prejuízo, todavia, de sua característica básica.

3.3 AMBIENTE DE DATA WAREHOUSE

No coração do ambiente projetado encontra-se o *Data Warehouse*. O *Data Warehouse* é o alicerce do processamento dos Sistemas de Apoio a Decisão (SAD). Em virtude de haver uma fonte única de dados integrados no *Data Warehouse*, e uma vez que os dados apresentam condições de acesso, a tarefa do analista de SAD no ambiente de *Data Warehouse* é incomensuravelmente mais fácil do que no ambiente clássico. Descreve-se abaixo alguns dos aspectos mais importantes do *Data Warehouse* [INM97]:

Um *Data Warehouse* é um conjunto de dados baseado em assuntos, integrado, não-volátil, e variável em relação ao tempo, de apoio às tomadas de decisões gerenciais [INM97].

- a) orientados por assunto - O fato de o *Data Warehouse* ser baseado em assuntos/negócios é demonstrado na Figura 3. Os sistemas operacionais clássicos são organizados em torno das aplicações da empresa. No caso de uma companhia

de seguros, as aplicações podem ser automóvel, saúde, vida e perdas. Os principais assuntos ou negócios da empresa podem ser cliente, apólice, prêmio e indenização;

Dados baseados em assuntos/negócios

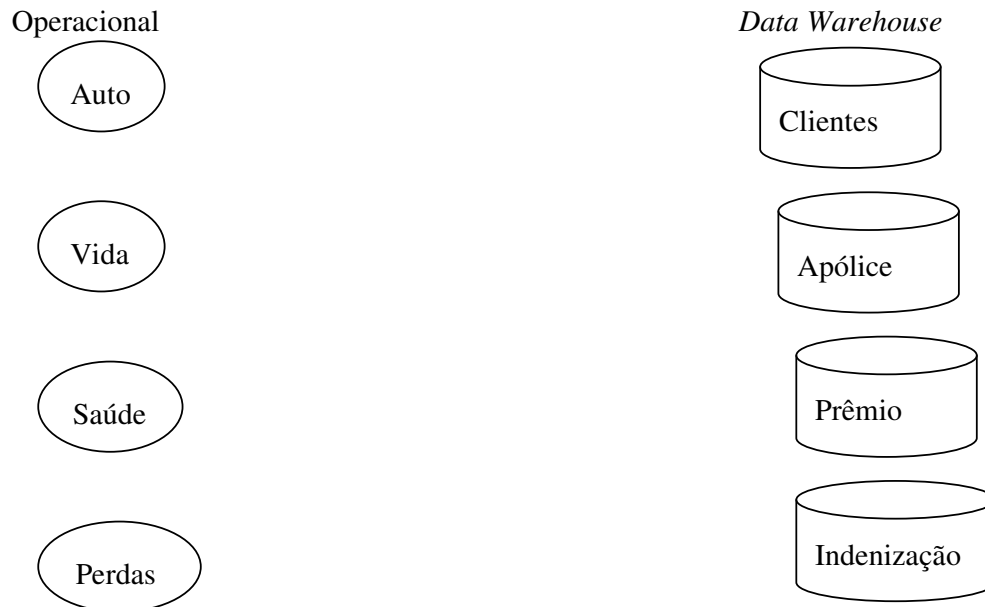


Figura 3 - Um exemplo de dados baseados em assuntos/negócios

Fonte: adaptado de [IMN97]

- b) integrado - A segunda característica marcante da *Data Warehouse* é o fato de que ele é *integrado*. De todos os aspectos do *Data Warehouse*, esse é o mais importante. A figura 4 ilustra a integração que ocorre quando os dados passam do ambiente operacional baseado em aplicações para o *Data Warehouse*. O processo de introdução dos dados no *Data Warehouse* é conduzido de forma que as muitas inconsistências das aplicações seja desfeitas. No que concerne ao código de gênero, pouco importa se os dados existentes no *Data Warehouse* são codificados como m/f ou 1/0. O que realmente importa é que além de a codificação para o *Data Warehouse* ser feita, ela deve, ainda, ser feita de forma consistente e independente da aplicação de origem;

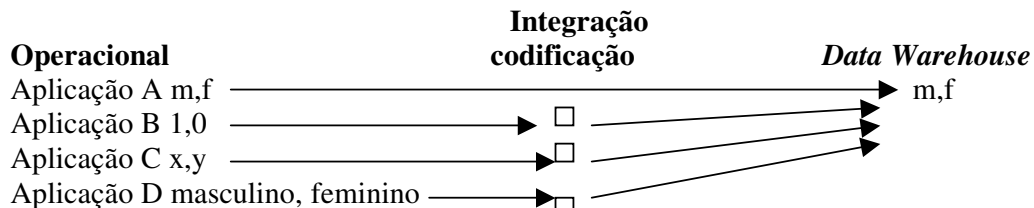


Figura 4 - A questão da integração

Fonte [IMN97]

c) não-volátil - A terceira característica importante de um *Data Warehouse* consiste em que ele é não-volátil. A figura 5 ilustra a não volatilidade de dados, e mostra que os dados operacionais são regularmente acessados e tratados um registro por vez. No ambiente operacional, os dados sofrem atualizações. Contudo, os dados existentes no *Data Warehouse* apresentam um conjunto de características muito diferentes. Os dados do *Data Warehouse* são carregados (normalmente em grandes quantidades) e acessados. Mas a atualização dos dados (geralmente) não ocorre no ambiente de *Data Warehouse* [INM97];

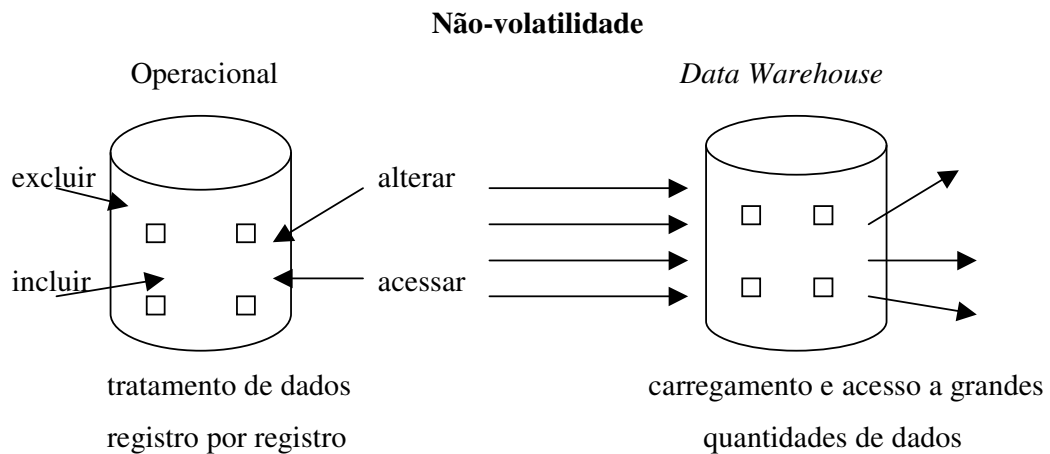


Figura 5 - A questão da não-volatilidade

d) variável em relação ao tempo - A última característica significativa do *Data Warehouse* diz respeito ao fato de ele ser variável em relação ao tempo. A figura 6 ilustra os diversos modos pelos quais a variação ao tempo se manifesta [INM97]:

- o horizonte de tempo válido para o *Data Warehouse* é significativamente maior do que o dos sistemas operacionais. Um horizonte de tempo de 5 a 10 anos de dados é normal para o *Data Warehouse*.
- bancos de dados operacionais contem dados de “valor corrente” – dados cuja exatidão é válida para o momento de acesso. Assim sendo, dados de valor corrente podem ser atualizados. Dados existentes no *Data Warehouse* não passam de uma série sofisticada de instantâneos, capturados num determinado momento.

- a estrutura de chave dos dados operacionais pode conter, ou não, elementos de tempo como ano, mês, dia etc. A estrutura de chave do *Data Warehouse* sempre contém algum elemento de tempo, pois trata-se de banco de dados históricos.

Variação em relação ao tempo

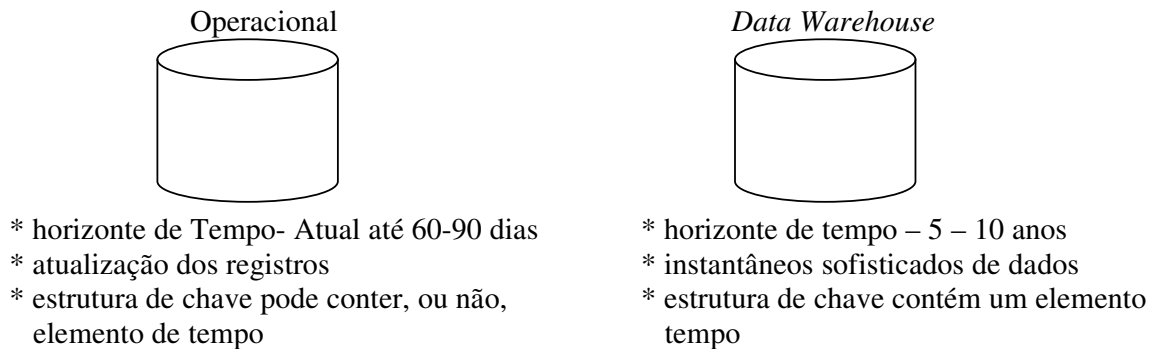


Figura 6 - A questão da variação em relação ao tempo

3.4 ARQUITETURA GENÉRICA DE DATA WAREHOUSE

A seguir é descrita uma arquitetura genérica proposta por [ORR96] e ilustrada na Figura 7. Esta descrição genérica procura apenas sistematizar papéis no ambiente de *Data Warehouse*, permitindo que as diferentes abordagens encontradas no mercado atualmente possam ser adaptadas a ela, deve-se considerar que esta arquitetura tem o objetivo de representar a funcionalidade de um *Data Warehouse* sendo que várias camadas propostas podem ser atendidas por um único componente de *software*.

Esta arquitetura é composta pela camada dos dados operacionais e outras fontes de dados que são acessados pela camada de acesso aos dados. As camadas de gerenciamento de processos, transporte e *Data Warehouse* formam o centro da arquitetura e são elas as responsáveis por manter e distribuir os dados. A camada de acesso à informação é formada por ferramentas que possibilitam os usuários extrair informações do *Data Warehouse*. Todas as camadas desta arquitetura interagem com o dicionário de dados (metadados) e com o gerenciador de processos [DAL98].

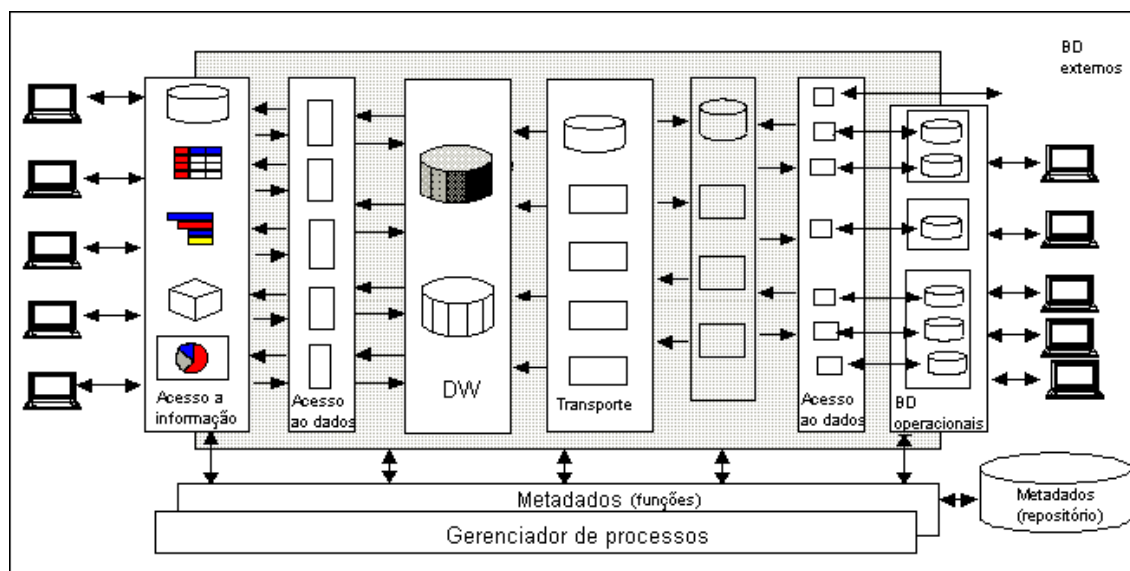


Figura 7 - Arquitetura genérica de *Data Warehouse*.

Fonte: [DAL98]

- a) camadas de bancos de dados operacionais e fontes externas: É composto pelos dados dos sistemas operacionais das empresas e informações provenientes de fontes externas que serão integradas para compor o *Data Warehouse*;
- b) camada de acesso a informação: Envolve o *hardware* e o *software* utilizado para obtenção de relatórios, planilhas, gráficos e consultas. É nesta camada que os usuários finais interagem com o *Data Warehouse*, utilizando ferramentas de manipulação, análise e apresentação dos dados, incluindo-se as ferramentas de *Data-Mining* e visualização;
- c) camada de acesso aos dados: Esta camada faz a ligação entre as ferramentas de acesso à informação e os bancos de dados operacionais. Esta camada se comunica com diferentes sistemas de bancos de dados, sistemas de arquivos e fontes sob diferentes protocolos de comunicação, o que se chama *acesso universal de dados*;
- d) camada de metadados(Dicionário de dados): Metadados são as informações que descrevem os dados utilizados pela empresa, isto envolve informações como descrições de registros, comandos de criação de tabelas, diagramas Entidade/Relacionamentos (E-R), dados de um dicionário de dados, etc. É necessário que exista uma grande variedade de metadados no ambiente de *Data Warehouse* para que ele mantenha sua funcionalidade e os usuários não precisem se preocupar onde residem os dados ou a forma com que estão armazenados;

- e) camada de gerenciamento de processos: É a camada responsável pelo gerenciamento dos processos que contribuem para manter o *Data Warehouse* atualizado e consistente. Está envolvida com o controle das várias tarefas que devem ser realizadas para construir e manter as informações do dicionário de dados e do *Data Warehouse*;
- f) camada de transporte: Esta camada gerência o transporte de informações pelo ambiente de rede. Inclui a coleta de mensagens e transações e se encarrega de entregá-las em locais e tempos determinados. Também é usada para isolar aplicações operacionais ou informacionais, do formato real dos dados nas duas extremidades;
- g) camada do *Data Warehouse*: É o *Data Warehouse* propriamente dito, corresponde aos dados utilizados para obter informações. As vezes o *Data Warehouse* pode ser simplesmente uma visão lógica ou virtual dos dados, podendo não envolver o armazenamento dos mesmos ou armazenar dados operacionais e externos para facilitar seu acesso e manuseio.

3.5 GRANULARIDADE

O mais importante aspecto do projeto de um *Data Warehouse* é a questão da granularidade. A granularidade diz respeito ao nível de detalhe ou de resumo contido nas unidades de dados existentes no *Data Warehouse*. Quanto mais detalhe, mais baixo o nível de granularidade. Quanto menos detalhe, mais alto o nível de granularidade [INM97].

A granularidade de dados tem se mantido como uma questão delicada de projeto. Nos primeiros sistemas operacionais que foram criados, a granularidade era tida como certa. Quando os dados detalhados eram atualizados, era quase certo que eles seriam armazenados no nível mais baixo de granularidade. No entanto, no ambiente de *Data Warehouse*, a granularidade não é um pressuposto.

A razão pela qual a granularidade é a principal questão de projeto consiste no fato de que ela afeta profundamente o volume de dados que residem no *Data Warehouse* e, ao mesmo tempo, afeta o tipo da consulta que pode ser atendida. O volume de dados contidos no *Data Warehouse* é balanceado de acordo com o nível de detalhe de uma consulta.

Conforme [DAL98], o nível de granularidade afeta diretamente o volume de dados armazenado no *Data Warehouse* e ao mesmo tempo o tipo de consulta que pode ser respondida.

Quando se tem um nível de granularidade muito alto o espaço em disco e o número de índices necessários se tornam bem menores, porém há uma correspondente diminuição da possibilidade de utilização dos dados para atender a consultas detalhadas.

A Figura 8 exemplifica o conceito acima utilizando os dados históricos das vendas de um produto, um nível de granularidade muito baixo pode ser caracterizado pelo armazenamento de cada uma das vendas ocorridas para este produto e um nível muito alto de granularidade seria o armazenamento do somatório das vendas ocorridas por mês.

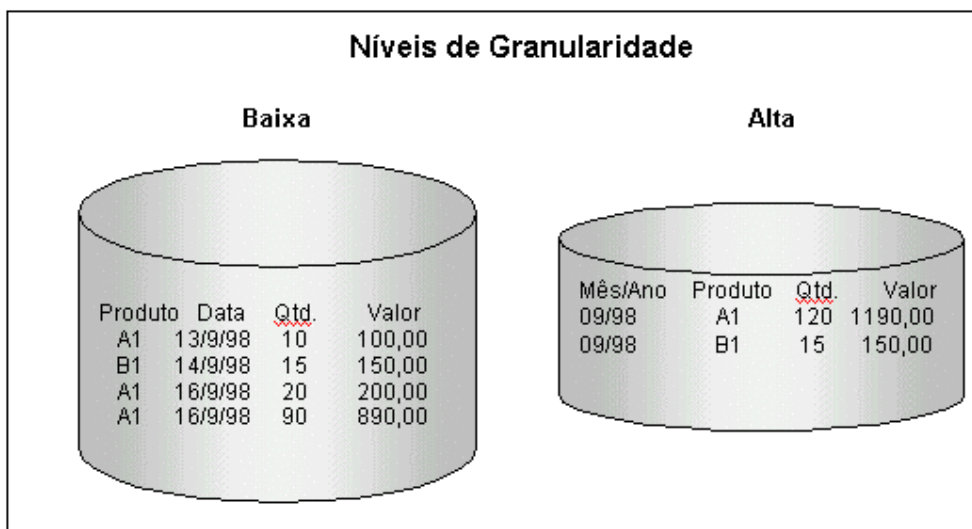


Figura 8 - Níveis de granularidade.
Fonte: [DAL98]

Com um nível de granularidade muito baixo, é possível responder a praticamente qualquer consulta, mas uma grande quantidade de recursos computacionais é necessária para responder perguntas muito específicas. No entanto, no ambiente de *Data warehouse*, dificilmente um evento isolado é examinado, é mais comum ocorrer a utilização de uma visão de conjunto dos dados.

Os dados levemente resumidos compreendem um nível intermediário na estrutura do *Data Warehouse*, são derivados do detalhe de baixo nível encontrado nos dados detalhados atuais. Este nível do *Data Warehouse* é quase sempre armazenado em disco. Na passagem

para este nível os dados sofrem modificações, por exemplo, se as informações nos dados detalhados atuais são armazenadas por dia, nos dados levemente resumidos estas informações podem estar armazenadas por semanas. Neste nível o horizonte de tempo de armazenamento normalmente fica em cinco anos e após este tempo os dados sofrem um processo de envelhecimento e podem passar para um meio de armazenamento alternativo.

Os dados altamente resumidos são compactos e devem ser de fácil acesso, pois fornecem informações estatísticas valiosas para os Sistemas de Informações Executivas, enquanto que nos níveis anteriores ficam as informações destinadas aos Sistemas de Apoio a Decisão (SAD) que trabalham com dados mais analíticos procurando analisar as informações de forma mais ampla.

O balanceamento do nível de granularidade é um dos aspectos mais críticos no planejamento de uma *Data Warehouse*, pois na maior parte do tempo, há uma grande demanda por eficiência no armazenamento e no acesso aos dados, bem como pela possibilidade de analisar dados em maior nível de detalhes. Quando uma organização possui grandes quantidades de dados no *Data Warehouse*, faz sentido pensar em dois ou mais níveis de granularidade na parte detalhada dos dados. Na realidade, a necessidade de existência de mais de um nível de granularidade é tão grande que a opção de projeto que consiste em duplos níveis de granularidade deveria ser o padrão para quase todas as empresas.

O chamado *nível duplo* de granularidade, ilustrado na Tabela 3, se enquadra nos requisitos da maioria das empresas. Na primeira camada de dados ficam os dados que fluem do armazenamento operacional e são resumidos na forma de campos apropriados para a utilização de analistas e gerentes. Na segunda camada, ou nível de dados históricos, ficam todos os detalhes vindos do ambiente operacional, como há uma verdadeira montanha de dados neste nível, faz sentido armazenar os dados em um meio alternativo como fitas magnéticas.

Com a criação de dois níveis de granularidade no nível detalhado do *Data Warehouse*, é possível atender a todos os tipos de consultas, pois a maior parte do processamento analítico dirige-se aos dados levemente resumidos que são compactos e de fácil acesso e para ocasiões em que um maior nível de detalhe deve ser investigado existe o nível de dados históricos. O

acesso aos dados do nível histórico de granularidade é caro, incômodo e complexo, mas caso haja necessidade de alcançar esse nível de detalhe, lá estará ele.

Primeira Camada	Segunda Camada
Dados resumidos por produto	Dados detalhados por produto
Produto A1 – maio/1998	Produto A1
Valor total: R\$ 1.270,00	02/5/1998- Valor R\$ 100,00 – Quantidade 20
Quantidade total: 254	09/5/1998- Valor R\$ 50,00 – Quantidade 10
Valor médio: R\$ 5,00	12/5/1998- Valor R\$ 125,00 – Quantidade 25
	20/5/1998- Valor R\$ 350,00 – Quantidade 70
	22/5/1998- Valor R\$ 110,00 – Quantidade 22
	29/5/1998- Valor R\$ 320,00 – Quantidade 64

Tabela 3 - Níveis duplos de granularidade para dados resumidos.

Fonte: [DAL98]

3.6 PARTICIONAMENTO

O particionamento de dados diz respeito à repartição dos dados em unidades físicas separadas que podem ser tratadas independentemente. No *Data Warehouse*, as questões referentes ao particionamento de dados não enfocam a necessidade de o particionamento ser feito ou não, mas como ele deve ser feito [INM97].

Freqüentemente é mencionado que se tanto a granularidade quanto o particionamento forem apropriadamente executados, então quase todos os outros aspectos de projeto e implementação do *Data Warehouse* brotarão naturalmente. No entanto, se a granularidade não for conduzida de forma apropriada e de o particionamento não for cuidadosamente projetado e implementado, então nenhum outro aspecto de projeto terá importância.

Particionamento se refere a divisão de dados em unidades físicas separadas que podem ser manipuladas independentemente. Acesso mais flexível aos dados é possível com unidades físicas menores. Uma unidade de dado é única para cada partição. Particionamento é acompanhado da aplicação dos seguintes critérios: data, linha de negócios, geografia, unidade organizacional e todos os anteriores [OLI98].

3.7 PARTICIONAMENTO DE DADOS

No ambiente de *Data Warehouse*, a questão não é se os dados de detalhe corrente serão particionados e sim como eles serão particionados. A figura 9 ilustra o particionamento [INM97].

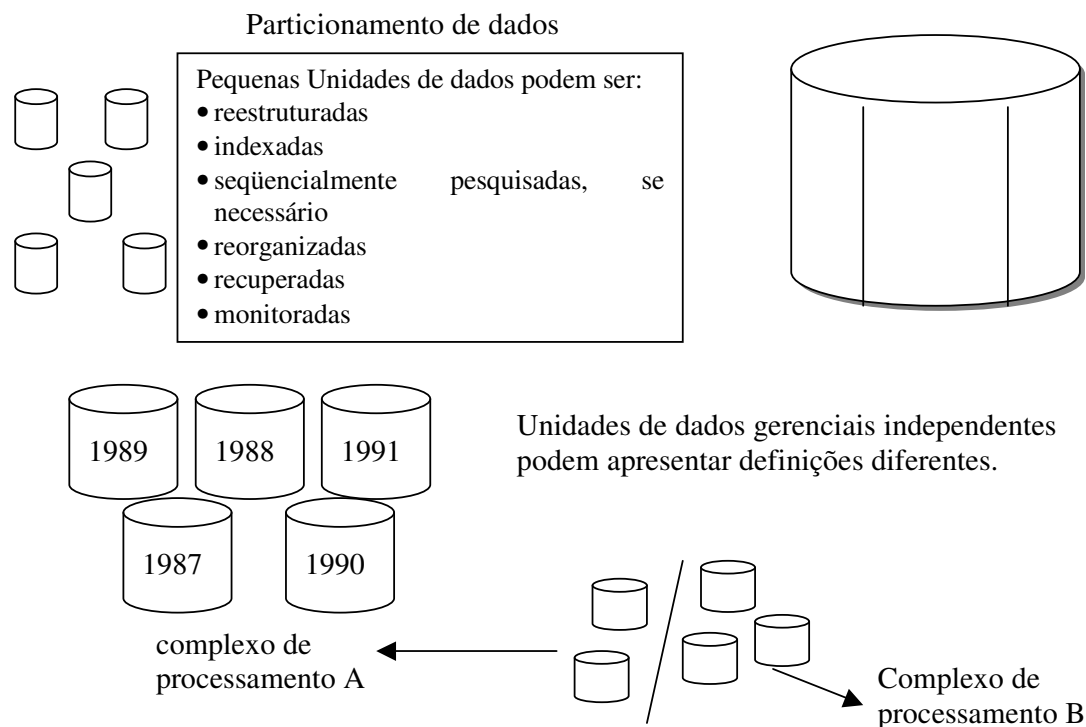


Figura 9 - Partições de dados gerenciadas independentemente pode ser enviadas a diferentes complexos de processamento sem acarretar consequência para os sistemas.

Fonte: [IM97]

De acordo com [INM97], o objetivo do particionamento dos dados de detalhe corrente consiste em repartir os dados em unidade físicas menores. Por que é tão importante? Unidades físicas menores proporcionam ao pessoal de operação e ao projetista muito mais flexibilidade no gerenciamento dos dados do que é proporcionado pelas unidades físicas maiores.

Quando os dados residem em unidades físicas de tamanho maior, entre outras coisas, eles não podem ser:

- a) reestruturados facilmente;
- b) indexados livremente;
- c) pesquisados seqüencialmente, se necessário;
- d) reorganizados facilmente;

- e) recuperados facilmente;
- f) monitorados facilmente.

O que realmente significa particionar os dados? Segundo [INM97], o particionamento de dados ocorre quando dados de uma mesma estrutura são divididos em mais de uma unidade física de dados. Além disso, toda unidade de dados pertence a uma e somente uma partição.

As opções referentes ao particionamento são tarefas da competência exclusiva do desenvolvedor. Contudo, no ambiente de *Data Warehouse* é quase obrigatório que um dos critérios de particionamento seja o critério de data. Como exemplo dos critérios que uma empresa de seguros pode escolher para o particionamento de seus dados, examine as seguintes unidades físicas de dados:

- a) indenizações por doença em 1988;
- b) indenizações por doença em 1989;
- c) indenizações por doença em 1990;
- d) indenizações por morte em 1987;
- e) indenizações por morte em 1988;
- f) indenizações por morte em 1989;
- g) indenizações por morte em 1990;
- h) indenizações por perdas em 1988;
- i) indenizações por perdas em 1989;
- j) indenizações por perdas em 1990;

Para particionar os dados, a empresa de seguros utilizou o critério de data – ou seja, ano – e de tipo de indenização.

Uma das questões mais desafiadoras para o desenvolvedor do *Data Warehouse* é a dúvida entre efetuar o particionamento no nível de sistema ou no nível de aplicação. O particionamento no nível de sistema é, até certo ponto, função do SGBD e do sistema operacional. O particionamento no nível de aplicação é feito pelo código da aplicação e é, única e exclusivamente, controlado pelo desenvolvedor e pelo programador. Quando o particionamento de dados é feito no nível de aplicação, o SGBD e o sistema operacional não tomam conhecimento de qualquer relação existente entre uma partição e outra.

Como regra, faz sentido particionar os dados do *Data Warehouse* no nível de aplicação. Há algumas razões importantes para esse fato. A mais importante de toda é que, no nível de aplicação, é possível existir uma definição de dados diferente para cada ano. Pode existir a definição dos dados de 1988 e a definição dos dados de 1989 e elas podem ser iguais ou não. Os dados existentes no *Data Warehouse* são caracterizados como um conjunto de dados que abrange um longo período de tempo.

Quando o particionamento é feito no nível de sistema, o SGBD inevitavelmente pressupõe uma definição de dados única. Considerando que o *Data Warehouse* mantém dados referentes a um longo período de tempo – até 10 anos – e considerando que a definição, normalmente, não faz sentido permitir que o SGBD ou o sistema operacional imponham a condição de existência de uma única definição de dados.

Um outro importante fator decorrente da permissão para que o particionamento de dados seja gerenciado no nível de aplicação consiste no fato de que ela pode ser transferida de um complexo de processamento para outro, impunemente. Quando a carga de trabalho e o volume de dados se transformam em um verdadeiro fardo para o ambiente de *Data Warehouse*, esse fator pode ser percebido como uma vantagem concreta.

A prova de fogo do particionamento de dados ocorre quando é feita a pergunta: “É possível acrescentar um índice a uma partição sem que qualquer interrupção seja sentida pelas outras operações?” Se houver a possibilidade de acrescentar um índice livremente, significa que o particionamento está satisfatório. Caso não seja fácil acrescentar um índice, a partição deve ser dividida de maneira mais adequada.

3.8 METADADOS

Metadados são normalmente definidos como "dados sobre os dados". Podem ser definidos também como um abstração dos dados, ou dados de mais alto nível que descrevem dados de um nível inferior. Os metadados têm um papel muito importante na administração de dados, mas no *Data Warehouse* podem ser considerados de suma importância pois é a partir deles que as informações serão processadas, atualizadas e consultadas [DAL98].

Como os usuários de *Data Warehouse* procuram por fatos não usuais e relações não conhecidas previamente eles precisam examinar os dados e para isso necessitam conhecer a

estrutura e o significado dos dados do *Data Warehouse*, o que não ocorre em um ambiente operacional onde os usuários trabalham com aplicações que contém as definições de dados embutidas e simplesmente interagem com as telas do sistema sem precisar conhecer como os dados são mantidos pelo banco de dados [DAL98].

Geralmente os metadados em um *Data Warehouse* podem ser apresentados em três camadas diferentes:

- a) Metadados operacionais: Definem a estrutura dos dados mantidos pelos bancos operacionais, usados pelas aplicações de produção da empresa;
- b) Metadados centrais do *Data Warehouse*: São orientados por assunto e definem como os dados transformados devem ser interpretados, incluem definições de agregação e campos calculados, assim como visões sobre cruzamentos de assuntos;
- c) Metadados do nível do usuário: Organizam os metadados do *Data Warehouse* para conceitos que sejam familiares e adequados aos usuários finais.

Os metadados podem ser classificados conforme a classe de seus componentes:

- a) Mapeamento: Descrevem como os dados de sistemas operacionais são transformados antes de entrarem no *Data Warehouse*. Exemplos desta classe de metadados podem ser os que identificam campos fontes, mapeamentos entre atributos, conversões, codificações, padrões, etc.;
- b) Histórico: Com a evolução dos sistemas operacionais as regras de negócio da empresa podem mudar, cabe a estes metadados manter o histórico de mudanças destas regras, pois as regras certas devem ser aplicadas aos dados certos;
- c) Miscelânea: Esta classe define diversos tipos de metadados, informações da situação de desenvolvimento de partes do *Data Warehouse*, informações sobre volume dos dados para estimativas de tempo e recursos, etc.;
- d) Algoritmos de sumarização: Mostram a relação entre os diferentes níveis de detalhes dos dados, indicando inclusive que nível de sumarização é mais adequado para um dado objetivo;
- e) Padrões de acesso: Mantém informações sobre frequência e tipo de acesso aos dados.

Conforme visto acima os dados sobre desempenho e monitoramento também são qualificados com metadados, eles podem ser criados por processos que monitoram atividades

como extração, carga e uso dos dados. Dados que identificam questões relativas a qualidade dos dados também devem estar disponíveis para os usuários, afim de que estes identifiquem a acuracidade de suas análises.

Segundo [INM97] os metadados englobam o *Data Warehouse* e mantém informações sobre "o que está aonde" no *Data Warehouse*. Tipicamente os aspectos sobre os quais os metadados mantêm informações são:

- a) a estrutura dos dados segundo a visão do programador;
- b) a estrutura dos dados segundo a visão dos analista de SAD;
- c) a fonte de dados que alimenta o *Data Warehouse*;
- d) a transformação sofrida pelos dados no momento de sua migração para o *Data Warehouse*;
- e) o modelo de dados;
- f) o relacionamento entre o modelo de dados e o *Data Warehouse*;
- g) o histórico das extrações de dados.

Segundo [INM97], o grau de importância dos metadados para dados externos e não estruturados é revelado pelo tipo de informação, conforme demonstrado na figura 10.

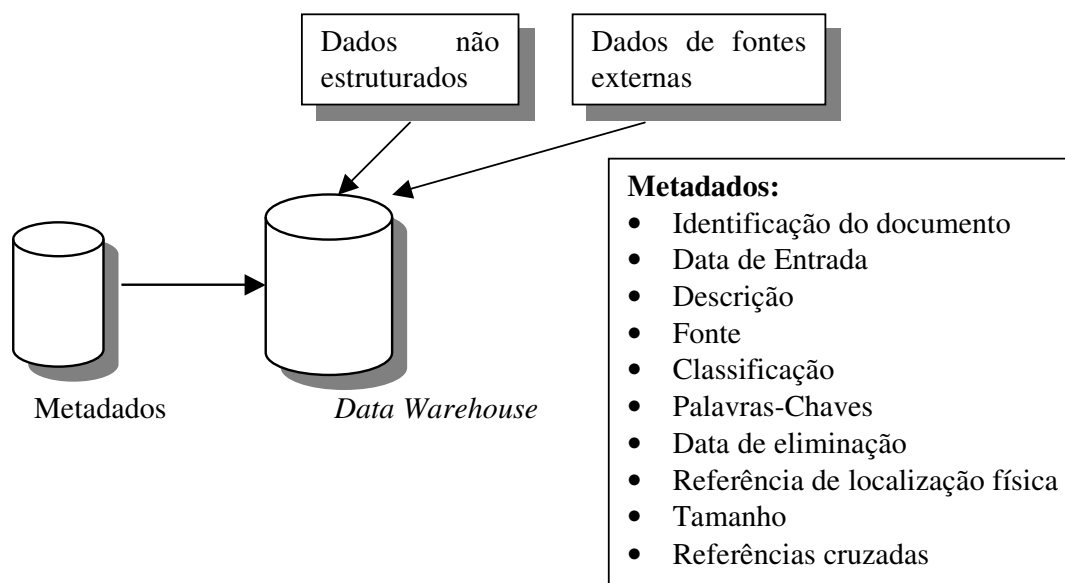


Figura 10 - Metadados para dados não estruturados e dados de fontes externas.

3.9 DIMENSIONALIDADE

Obter respostas para questões típicas de análise de negócios, com intuito de formular estratégias efetivas, mensurar tendências, identificar desvios e auxiliar na tomada de decisões, requer visões de dados sob diferentes perspectivas e, por vezes implica na necessidade de utilizar abordagens específicas para viabilizar o atendimento destes tipos de consultas [BAP98].

Segundo [CAM97], as perspectivas sob as quais um dado pode ser analisado são denominados de “dimensões. São exemplos de dimensões, modelo, loja, marca, fabricante, local, produto, tempo e outras instâncias que fornecem contexto para a análise.

Os indicadores quantitativos para cada dimensão denominadas de “medidas” e geralmente são expressos em valores monetários ou volumes físicos, dados tipicamente numéricos. São exemplos de medidas, valor das vendas, quantidade de itens, total de peças, valor do custo, número de alunos, etc.

A forma como são armazenadas as agregações resultantes das combinações entre dimensões e medidas, recebe a denominação de “multidimensional”.

Segundo [KIM96], os dados no *Data Warehouse* devem ser armazenados segundo a abordagem de modelagem multidimensional ou simplesmente MMD, com o objetivo de estruturá-los de forma a permitir análises mais complexas e eficientes e mais próximas da realidade como e empresa vê e pensa o seu negócio.

De acordo com [WEL95], na teoria todos os dados podem ser considerados multidimensionais. No entanto, na prática, esta generalização não ocorre, sendo reservado somente para dados que representam objetos ou eventos que possam ser classificados por dois ou mais de seus atributos.

3.10 PROCESSAMENTO ANALÍTICO ONLINE (OLAP)

A sigla *OLAP* originada de *On Line Analytical Processing*, refere-se ao tipo de processamento e ferramentas voltadas para a análise de dados típica do suporte a decisão, onde os dados são apresentados através do modelo de visão multidimensional. As visões independem da forma como os dados estão armazenados [INM97].

Segundo [LAU96], OLAP é uma tecnologia projetada para permitir acesso e análises multidimensionais sobre os diversos níveis de negócios da empresa.

Do ponto de vista prático, conforme [LAU96], OLAP sempre envolve consultas interativas aos dados, percorrendo um caminho de análises através de múltiplas camadas, podendo aprofundar-se até o nível mais baixo de uma informação específica.

Se comparada com o tipo de processamento OLTP, a modalidade OLAP apresenta características completamente distintas, conforme pode ser verificado na tabela 4.

OLTP	OLAP
Modelo Relacional	Modelo Multidimensional
Dados Individualizados	Dados sumarizados
Valor Presente	Valor histórico
Acessa um registro por vez	Acessa muitos registros por vez
Orientação para o processo	Orientação para o negócio

Tabela 4 - Comparativo OLTP e OLAP

Para suportar as características da tecnologia OLAP, dois tipos de processos são bastante comuns: “*slice and dice*” e “*drill down*”.

O tipo “*slice and dice*” corresponde à técnica de alternar a ordem das dimensões, mudando assim a orientação segundo a qual os dados são visualizados, permitindo ao usuário investigar os diferentes inter-relacionamento existentes entre eles.

Já o tipo “*drill down*” consiste em permitir que o usuário se aprofunde pelas hierarquias das dimensões, buscando obter níveis de detalhamento maiores para explicar uma informação oriunda de um nível resumido.

Segundo [WEL95], Codd e Date formularem uma lista de 12 regras para avaliar a eficácia de uma ferramenta de OLAP com relação às necessidades deste tipo de processamento:

- a) visão conceitual multidimensional;
- b) transparência de localização e forma dos dados;
- c) facilidade de acesso a fontes de dados homogêneas e heterogêneas;
- d) desempenho de consultas consistente;
- e) operação em arquitetura Cliente/Servidor;
- f) dimensionalidade genérica;

- g) manipulação dinâmica de matrizes esparsas;
- h) suporte multi-usuário;
- i) operações com dimensões sem restrições;
- j) manipulação de dados intuitiva;
- k) relatórios e consultas flexíveis;
- l) níveis de agregação e dimensões ilimitados;

3.11 ESQUEMA FUNCIONAL DO DATA WAREHOUSE

Do ponto de vista da funcionalidade, de acordo com [CIO96], um *Data Warehouse* é constituído por três componentes funcionais diferentes, sendo que cada um dos quais precisa ser adequadamente customizado para atender às necessidade de uma empresa.

O primeiro componente enfoca a questão da aquisição dos dados nos sistemas de origem e fontes externas. Os dados são identificados, copiados, formatados e preparados para serem carregados no *Data Warehouse*. O processo de aquisição ocorre basicamente em três etapas:

- a) a catalogação de todos os dados, criando um inventário de sua localização e significado;
- b) extração dos dados dos sistemas de origem e correspondente limpeza e transformação de acordo com suas finalidades;
- c) transporte dos dados preparados para o ambiente *Data Warehouse*.

Segundo componente compreende as técnicas de armazenamento de dados, as quais são dominadas por gerenciadores de banco de dados relacionais, que se fundamentam em tecnologias de última geração em hardware, como processadores SMP, para processamento simétrico ou MPP para processamento paralelo, além de explorar novos recursos técnicos de software. A principal característica deste componente é eficácia no uso dos dados pelos usuário, seja através de sistemas de *Data Mining*, informação executiva ou suporte à decisão.

O terceiro e último componente deste esquema refere-se ao acesso. Com ele, muitos usuários finais, baseados em PCs ou *Workstations*, tem acesso aos dados do *Data Warehouse* com a ajuda de produtos para análise multidimensional, redes neurais, ferramentas de recuperação de dados ou de análise de dados. A maioria desses produtos de acesso aos dados

são muito poderosos e inteligentes e formam a base de viabilização de um *Data Warehouse*. Existem diversas categorias de produtos que atendem aos requisitos deste componente, entre elas:

- a) Ferramentas que simulam agentes inteligentes;
- b) Ferramentas para processamento OLAP;
- c) Ferramentas para recuperação de dados;
- d) Ferramentas para análises estatísticas;
- e) Ferramentas para consultas em ambientes gerenciáveis;
- f) Ferramentas para visualização de dados.

3.12 DATA WAREHOUSE X DATA MART

Conforme [OLI98], um *Data Mart* é um *Data Warehouse* de menor porte construído para armazenar dados ligados a um determinado aspecto do negócio da empresa. Aproximadamente 70% a 80% de todos os *Data Warehouse* correntemente em produção são, na verdade, *Data Marts*.

Veja as diferenças entre *Data Warehouse* e *Data Mart*:

- a) volume de informações: um *Data Warehouse* é construído para oferecer informações necessárias a toda a empresa. Já um *Data Mart* é construído para uma unidade ou função específica de negócios.
- b) objetivo: um *Data Warehouse* tem como objetivo principal otimizar a integração e gerenciamento dos dados originados de outras fontes de dados. Já os *Data Mart* são projetados para otimizar a entrega de informações para tomada de decisão.
- c) tratamento da informação: um *Data Warehouse* gerencia grandes quantidades de informação. Já um *Data Mart* está focado primariamente no sumário ou exemplos de informações.
- d) gerenciamento: o *Data Warehouse*, por suas características, pertence e é gerenciado pelos executivos de IS (Information Systems). Já um *Data Mart* pode pertencer e ser gerenciado por gerentes de linha.

4 MIGRAÇÃO DE DADOS E SUA IMPORTÂNCIA

Uma das etapas de fundamental importância na construção de uma *Data Warehouse* e com certeza a que mais consome recursos, é o carregamento dos dados no banco de dados do *Data Warehouse*.

A maioria das tarefas requeridas nesta etapa, - como extração, limpeza, transformação e transporte de dados dos sistemas em operação para o *Data Warehouse* – são de natureza extremamente crítica para o sucesso do projeto.

Segundo [MOR96], a grande questão que se apresenta neste caso não é propriamente técnica e sim gerencial. Muitos dos processos envolvidos – mapeamento, integração e avaliação de qualidade – ocorrem de fato durante a fase de análise e projeto do *Data Warehouse*.

Há um consenso entre os especialistas de que 80% (oitenta por cento) do tempo total de um projeto de *Data Warehouse* é consumido nas atividades de pesquisar e identificar fontes de dados, estabelecer regras de transformação e detectar e resolver problemas de integração e qualidade dos dados.

A grande qualidade e variedade de fatores associados ao problema influem fortemente nas estimativas de tempo e recursos para o desenvolvimento destas tarefas.

O volume de fonte de dados e a qualidade dos metadados correlacionados, assim como a origem das regras de negócios associadas a cada fonte, são grandes causadores de preocupação, especialmente quando estão memorizadas apenas nas cabeças de analistas e usuários ou encontram-se na forma de código fonte.

Segundo [INM97], a maioria dos desenvolvedores superestima o tempo necessário para outras atividades e subestima o tempo necessário para projetar e construir a interface entre o sistema e o *Data Warehouse*.

Os termos carregar ou carregamento, povoar ou povoamento também são usados para identificar esta fase, uma vez que tem o sentido de inserir registros de dados em um banco de dados.

4.1 PROCESSOS ASSOCIADOS

Segundo [INM97], o verdadeiro valor de um *Data Warehouse* vai além de sua capacidade de armazenar dados. Ele deve incorporar também o papel de gerenciador dos grandes fluxos de dados, que constituem a base de sustentação deste processo.

O primeiro deste fluxos se refere ao fluxo de entrada, responsável pela consolidação dos dados de base de dados operacionais do próprio meio ou de fontes externas e sua carga no *Data Warehouse*.

Por estas afirmações é aceitável deduzir que um *Data Warehouse* sempre é alimentado a partir de dados “importados”(próprios ou externos), portanto migrar dados de um ambiente para outro é um fato consumado neste conceito e um segundo ponto, é que sempre vai haver algum tipo de tratamento destes dados.

O tratamento adequado dos dados fonte está associado à execução de uma série de processos que objetivam assegurar a sua qualidade e eficácia.

4.1.1 PROCESSO DE EXTRAÇÃO

Literalmente um processo de extração pode ser definido como uma atividade que visa selecionar e copiar dados. Na prática vai mais além, passando por questões como a especificação dos dados a serem extraídos e o acesso às bases de dados físicas.

Segundo [INM97], apesar de ser teoricamente possível, tentar construir um ambiente *Data Warehouse* sem um modelo de dados, é como tentar navegar sem uma carta de navegação.

Por conseguinte o modelo de dados é parte integrante do projeto e fonte de referência para a especificação e mapeamento dos dados a serem extraídos.

No caso do acesso às bases de dados, é importante levar em consideração alternativas que permitam balancear desempenho, tempo e espaço em disco. As firmas de incrementar soluções disponíveis no ambiente operacional, sendo possível:

- a) Incrementar consultas online diretamente nas bases operacionais, com possibilidade de causar impacto no normal.

- b) Fazer cópias dos dados fontes, com risco de elevar o consumo de espaço em disco.
- c) Incluir processos específicos para extração em pontos estratégicos do ciclo de processamento operacional.

4.1.2 PROCESSO DE LIMPEZA

De acordo com [ORL96], a limpeza de dados é um processo que tem por finalidade validar os dados segundo critérios de aderência à padrões, consistência interna, integridade referencial e domínio de valores, promovendo a substituição do dado incorreto.

Diversas técnicas podem ser usadas nesta fase, como por exemplo consulta a listas oficiais ou padrões – cadastro de CEP, listas de endereços, tabelas de codificação de países, estados, municípios - checagem de arranjos de domínio de valores, checagem sobre tabelas de dados predefinidas, análise de padrões de exceções, frequência de distribuição e outras.

Segundo [BOH97], geralmente as rotinas associadas com a migração ocorrem dentro do ambiente dos sistemas de origem, uma vez que os tipos de condicionamento a que são submetidos os dados tem mais respaldo no ambiente que os criou.

Este fato pode induzir o projetista a implementar as alterações no ambiente existente, numa relação de causa e efeito. [INM97] adverte: “Se você esperar até que os sistemas existentes passem por uma limpeza, jamais construirá um *Data Warehouse*”.

Tem que haver absoluta independência entre os problemas e tarefas do ambiente existente e o *Data Warehouse*.

Entretanto, segundo [BOH97], em muitos casos, a correção dos dados pode envolver muitas questões de julgamento de valor para o projeto, sendo recomendado que, sempre que possível, a equipe de desenvolvimento trabalhe junto ao usuário no sentido de corrigir os dados na origem.

4.1.3 PROCESSO DE TRANSFORMAÇÃO

O processo de transformação de dados consiste na aplicação de um conjunto de regras que convertem valores de dados das fontes de origem para valores ajustados do ambiente global e integrado do *Data Warehouse*.

Segundo [ORL96], a transformação de dados é uma abstração deste mesmos dados e ocorre quando os mesmos passam por combinações, sumarizações, representação computacional, agregações e outras ações destinadas a mudar o valor do dado.

Segundo [BOH97], sustenta a idéia de uma área de trabalho comum, para onde os dados fonte seriam movidos, após serem convertidos para esquemas intermediários. A intenção é otimizar a reutilização do código programática, especialmente aquele das rotinas de conversão – limpeza, transformação e integração – por serem comuns a todos os dados de origem.

4.1.4 PROCESSO DE TRANSPORTE (MIGRAÇÃO)

O processo de transporte contempla as ações relacionadas com o movimento dos dados extraídos, limpos e transformados para o servidor do *Data Warehouse* e sua carga de banco de dados.

Segundo [ORL96], esta operação por ser executada utilizando a modalidade *batch* de processamento de dados ou através de transações individuais, comandadas por algum programa aplicativo com SQL (*Structured Query Language*) incorporado. Os conceitos deste dois métodos são inteiramente diferentes, tanto assim que muitas ferramentas apresentam componentes separados para manipular cada um destes métodos.

A método *batch* possibilita que os dados preparados possam ser movidos para dentro do *Data Warehouse*, em grandes lotes ou massa, função conhecida como *bulk load*, o que implica em executar processos de restauração do banco de dados periodicamente, podendo incorrer em consumo elevado de tempo e redução do índice de disponibilidade do *Data Warehouse*.

A outra alternativa consiste em usar o método de restauração seletiva em que são atualizados somente os registros alterados, o que proporciona maior facilidade de gerenciamento e garantia de integridade, podendo porém disputar recursos com os sistemas operacionais.

Conforme [BOH97], recomenda a utilização da integridade referencial durante o processo de *bulk load* como forma de garantir a integridade dos dados entre as tabelas de fatos e as tabelas de dimensões associadas.

4.2 COMPLEXIDADES DO PROCESSO MIGRATÓRIO

Migrar: passar de um local para outro. Parece muito fácil mas por trás desta simples movimentação de dados esconde-se uma grande e complexa tarefa, recheada de inúmeros e graves problemas.

Conforme [INM97], relaciona uma série de fatores críticos encontrados na passagem dos dados do ambiente operacional herdado para o ambiente *Data Warehouse* projetado.

- a) a extração dos dados do ambiente operacional para o ambiente *Data Warehouse* normalmente demanda uma mudança na tecnologia;
- b) a seleção dos dados no ambiente operacional pode ser muito complexa, implicando em diversas pesquisas, leituras indexadas, lógica de conexão e outras;
- c) as chaves de entradas operacionais geralmente precisam ser reestruturadas antes de serem gravadas;
- d) os dados precisam ser reformatados;
- e) os dados passam por uma limpeza;
- f) existem múltiplas fontes de dados, aumentando a complexidade lógica;
- g) quando há vários arquivos de entrada, a escolha das chaves deve ser feita antes que os arquivos seja intercalados;
- h) onde houver vários arquivos de entrada, a seqüência de arquivos pode não ser a mesma e até se incompatível;
- i) pode haver vários resultados;
- j) valores *default* devem ser fornecidos;
- k) a eficiência na escolha dos dados de entrada para a extração torna-se freqüentemente uma questão importante;
- l) freqüentemente é necessário produzir dados resumidos;
- m) a alteração de nomes de elementos de dados durante a passagem do ambiente operacional para ambiente *Data Warehouse* deve ser registrada;

- n) os registros de entrada que precisam ser lidos apresentam formatos exóticos ou não padronizados;
- o) a conversão de dados precisa ser feita;
- p) os relacionamentos de dados que foram embutidos na lógica dos antigos programas herdados precisam ser compreendidos e reestruturados antes que esses arquivos possam ser usados como entrada;
- q) a conversão de formato dos dados precisa ser feita;
- r) volumes massivos de dados de entrada devem ser levados em consideração;
- s) o projeto do *Data Warehouse* deve adequar-se ao modelo de dados corporativo;
- t) o *Data Warehouse* espelha as informações históricas necessárias, ao passo que o ambiente operacional focaliza as informações correntes;
- u) o *Data Warehouse* destina-se a suprir as carências informacionais da empresa ao passo que o ambiente operacional serve ao atendimento das necessidades funcionais imediatas da empresa;
- v) a transmissão do arquivo de saída recém criado que passará para o *Data Warehouse* deve ser cuidadosamente considerada.

4.3 CONSIDERAÇÕES SOBRE O DESENVOLVIMENTO DE UM DATA WAREHOUSE

Segundo [WEL95], o sucesso do desenvolvimento de uma *Data Warehouse* depende fundamentalmente de uma escolha correta da estratégia a ser adotada, de forma que seja adequada às características e necessidades específicas do ambiente a ser implementado. Esta escolha deve ser fundamentada pelo menos em três amplitudes: escopo do *Data Warehouse* (departamental, corporativo, etc.), o grau de redundância dos dados e os padrões de uso.

O escopo de um *Data Warehouse* pode ser tão amplo, abrangendo o conjunto de informações de toda a empresa ou tão restrito, abrangendo um único gerente. Quanto maior o escopo mais valor o *Data Warehouse* tem para a empresa e mais cara e trabalhosa a sua construção e manutenção. Por estas razões muitas empresas tendem a começar com um ambiente departamental e só após obter um retorno de seus usuários, expandir seu escopo.

Há essencialmente três concepções de redundância de dados:

- a) O *Data Warehouse* virtual, que consiste simplesmente em prover os usuários finais com facilidades adequadas para extração das informações diretamente dos bancos de dados de produção, não incorrendo em redundância, mas podendo sobrecarregar o ambiente operacional.
- b) O *Data Warehouse* centralizado, constituído por um único banco de dados físico incorporando grandes volumes de dados oriundos de diversos bancos operacionais, devendo ser carregado e mantido com intervalos regulares.
- c) O *Data Warehouse* distribuído, que mantém seus componentes distribuídos por diferentes bancos de dados físicos, normalmente possuindo um alto índice de redundância e exigindo procedimentos de carga e manutenção mais complexos.

Quanto aos padrões de uso de um *Data Warehouse*, estes também assumem um papel importante na escolha de alternativas para o ambiente. Relatórios e consultas pré-estruturadas podem satisfazer o usuário final e geram pouca demanda sobre o ambiente. Análises complexas, típicas de sistemas de suporte à decisão, exigem mais ambiente.

4.3.1 ALTERNATIVAS DE IMPLANTAÇÃO

Um *Data Warehouse*, em geral, é projetado e carregado passo a passo, seguindo uma abordagem evolutiva. Os altos custos envolvidos e o impacto no ambiente operacional justificam essa medida [BAP98].

Muitas empresas iniciam seu *Data Warehouse* a partir de uma área específica, normalmente sequiosa por informação e relevante para os negócios da empresa. Este enfoque recebe o nome de *Data Mart* ou *Data Warehouse* departamental, que vai crescendo aos poucos até atingir todas as áreas da empresa, evoluindo segundo uma estratégia *botton up* [BAP98].

De acordo com [CAM97], uma outra alternativa consiste em selecionar um grupo de usuários, dotá-los de ferramentas adequadas, construir um protótipo do *Data Warehouse* e deixa-los praticar com pequenas amostras de dados. Somente após a concordância do grupo quanto aos requisitos e funcionamento, o *Data Warehouse* seria efetivado.

Data Marts também gerados como partes de um *Data Warehouse* maior, proporcionando maior autonomia, melhor desempenho e simplicidade de uso.

4.3.2 ASPECTOS DE MODELAGEM

Segundo [BAP98], a modelagem de uma *Data Warehouse* é totalmente diferente da modelagem tradicional de sistemas. De fato há muitas diferenças entre o modelo tradicional de dados e o modelo de dados de *Data Warehouse*. Uma pequena mostra destas diferenças inclui:

- a) Modelar para eficiência analítica e não para eficiência operacional;
- b) Introduzir controles de redundância e desnormalização de dados, ao invés de procurar eliminá-los;
- c) Modelar o banco de dados visando facilitar a compreensão e o acesso e não para otimizar a utilização;
- d) Abranger dados históricos e não ignorá-los ou eliminá-los;
- e) Modelar para administrar o conteúdo e o contexto dos dados (metadados);
- f) Modelar para múltiplas comunidades de usuários e não para uma única;

Segundo [CAM97], o objetivo de um *Data Warehouse* é prover dados com qualidade. Os requisitos são indeterminados e dependem das necessidades de informações individuais de seus usuários e ao mesmo tempo são instáveis, dependem das variações de informações específicas mudem com frequência, os dados associados não mudam. Em suas atividades diárias as empresas lidam com um grupo de processos relativamente constante, e gravitando em redor mesmo existe um número finito de objetos e eventos. Por esta razão um modelo de dados constitui uma base sólida para identificar os requisitos de um *Data Warehouse*

De qualquer forma, [INM97] afirma que é um erro pensar que técnicas de projeto que servem para sistemas convencionais serão adequadas para a construção de *Data Warehouse*. Os requisitos de um *Data Warehouse* não podem ser conhecidos até que esteja parcialmente carregado e já em uso.

4.4 FERRAMENTAS PARA MIGRAÇÃO

Existem muitas ferramentas no mercado que facilita as tarefas de extração, limpeza e transformação de dados, bem como a análise e a garantia de qualidade dos dados. Segundo [WIL97], estas ferramentas contribuem com uma grande percentagem nos gastos com o carregamento e armazenamento de dados em projetos *Data Warehouse*.

Contudo, o campo vasto e o ambiente caracterizado pela derivação e replicação de informações, não oferece condições seguras para a atuação de uma ferramenta única, embora algumas ofereçam uma combinação destes serviços [BAP98].

O problema é que uma ferramenta que oferece todos os serviços precisa ser avaliada do ponto de vista da maximização do orçamento, o que torna importante entender as categorias de ferramentas e o que cada uma tem a oferecer [BAP98].

Conforme [ORL96] e [WIL97], classificam as ferramentas em três categorias: as voltadas para análise, as voltadas para qualidade e limpeza de dados e as voltadas para extração e transporte.

A seleção de ferramentas geralmente é uma tarefa demorada e frustrante. Ao buscar uma ferramenta, o projetista deve ter o conhecimento claro do que precisa ser feito e que fase do projeto esta sendo atendida [BAP98].

A tabela 5 apresenta as principais ferramentas para auxiliar na migração de dados em ambientes *Data Warehouse*.

Categoria	Ferramenta	Funcionalidade
Engenharia reversa de dados baseada em metadados	Bacman, LogicWorks ERWIN/ERX Embarcadero ER/1 Kismed KisMeta	Processa metadados para documentar sistemas e abstrair regras de negócio e relacionamentos
Engenharia reversa de dados baseada no conteúdo dos dados	Vality Integrity QDB Analyze Data Star WisRule	Processa o conteúdo do dado junto com o metadado para abstrair regras de negócio e relacionamentos automaticamente
Extração e transporte em batch; Geradores de código para extração baseados em parâmetros	Carleton Paspoer ETI Extract Prism Warehouse Mananger	Extração controlada de forma centralizada com geração automática dos programas; oferece conversão, transporte e gerenciamento de replicação
Extração e transporte em batch; O usuário escreve o código	3GL/4GL (COBOL et ali) Platinum InfoRefiner Platinum InfoPump Praxis OmniReplicator	Repositório para código de extração e conversão; interfaces com BD; fraca função de replicação
Replicação	IBM Data propagator Sybase Replication Server	Específico para replicação; inclui funções de extração e conversão limitada
<i>Middleware</i> – extração e transporte online interativo	CA Enterprise Access Platinum InfoHub Praxis Omni, Replicator Sybase Enterprise, Connect IBM DataJoiner Intersolv Sequelink	Mesma conceito das ferramentas batch, com consultas online e interface automatizado entre as diversas fontes e ferramentas de consulta
Qualidade do conteúdo dos dados baseada em filtros	Apertus Trillium	Operam na exportação de dados; suportam filtragens baseadas em parâmetros; gerenciam relacionamentos e transformações; mantém a integridade das chaves de dados, evitando problemas de interpretação durante as consultas
Qualidade do conteúdo dos dados baseada em relacionamentos	Vality DB Star WizRule IDI	Ajuda descobrir padrões, regras e relacionamentos e auxilia na localização de áreas com problemas de qualidade
Qualidade de dados para objetivos específicos	PostalSoft ACE Group 1 Nadis SSA	Aplicada para correção de nomes, endereços, etc.
Qualidade de dados – suporte à entrada de dados em áreas específicas	PostalSoft Library Mailers +4	Edição automática de endereços durante a entrada de dados; utilizam bibliotecas de classe
Tradução de dados	Data Junction Cambio	Auxilia na tradução de formatos de dados; atua junto com outras ferramentas

Tabela 5 - Tabela de ferramentas comerciais para auxílio à migração.

5 FERRAMENTAS

Para a construção do protótipo de *Data Warehouse*, utilizaram-se algumas técnicas e ferramentas de prototipação genérica, as quais são descritas neste capítulo. A implementação deu-se no ambiente visual Delphi 3, por ter maior familiaridade, bem como o banco de dados paradox.

Na elaboração do Diagrama de contexto, Diagrama de Fluxo de Dados foi utilizado o *System Architect*, pela facilidade na definição destes itens e com muita dificuldade na elaboração do MER (Modelo Entidade Relacionamento) e dicionário de dados do protótipo.

Para a elaboração e construção do *Data Warehouse* será utilizado o roteiro de desenvolvimento segundo W. H. INMON.

5.1 PROTOTIPAÇÃO

Segundo [MEL90], a prototipação é o conjunto de técnicas e ferramentas de software para o desenvolvimento de modelos de sistemas, buscando uma aproximação da realidade do que vai ser o sistema. A área de computação é uma das poucas que não produzia modelos de testes do produto final a ser colocado em operação. Isto mudou, em vista da necessidade de antecipar um modelo do sistema, para avaliação de sua funcionalidade.

A importância da prototipação não está associado apenas ao fator econômico, isto é, obter um *feedback* antes de investir no produto. Em projetos relacionados a saúde, por exemplo, pode ser um caso de vida ou morte.

5.1.1 VANTAGENS DA PROTOTIPAÇÃO

Segundo [MEL90], a construção de sistemas de apoio à decisão, mediante a utilização das técnicas e ferramentas da prototipação, tem grandes vantagens sobre as metodologias tradicionais e as estruturadas, tendo em vista as seguintes características:

- a) busca a participação direta do usuário final, colaborando para a elaboração do modelo de sistema;
- b) os métodos, técnicas e ferramentas utilizados possuem recursos que viabilizam a participação (conhecimento) do usuário a respeito do sistema e o do analista em

transportar as necessidades lógicas e físicas de informação para um conjunto de restrições impostas pelos recursos do computador;

- c) facilita e agiliza a etapa de identificação e especificação de requisitos de informação, permitindo que o usuário modifique sua concepção no decorrer do desenvolvimento do modelo do sistema.

5.1.2 FASES DA PROTOTIPAÇÃO

“Uma metodologia qualquer sempre será utilizada no desenvolvimento de sistemas” [MEL90]. Para tanto, seis fases que compreendem o desenvolvimento de um sistema, pela metodologia de prototipação genérica, são apresentadas na Figura 11:

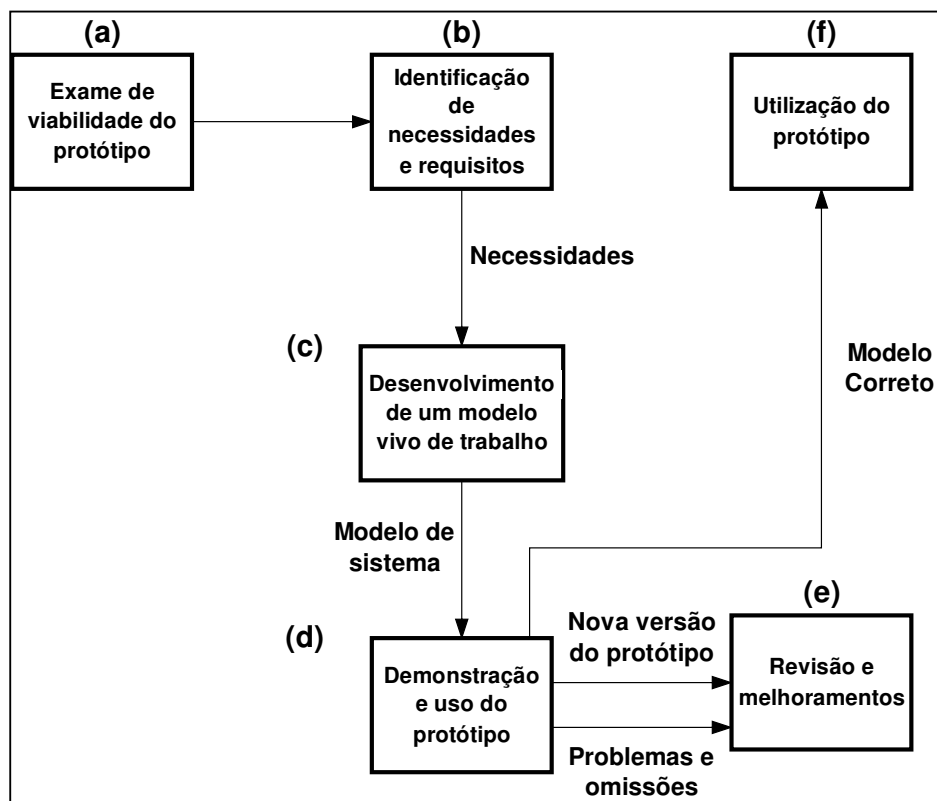


Figura 11 - Modelo genérico da metodologia de prototipação
 Fonte: Melendez Filho, Rubens. Prototipação de sistemas de informação, 1990.

5.1.2.1 EXAME DE VIABILIDADE DO PROJETO

Antes de usufruir do uso de uma metodologia de prototipação, seja qual for, torna-se essencial a análise e definição de três requisitos que compreendem o exame de viabilidade:

- a) recursos e requisitos para a viabilização do projeto;

- b) fatos geradores do desenvolvimento do projeto;
- c) políticas e diretrizes da área de informática.

5.1.2.2 IDENTIFICAÇÃO DE NECESSIDADES E REQUISITOS

Esta etapa compreende a identificação e definição dos objetos do sistema a ser prototipado, servindo de base para a elaboração do modelo preliminar do protótipo. Técnicas para elaboração de diagramas, estrutura de dados, funções, modelo lógico e outras são aplicadas. Porém, segundo Boar (apud Melendez), é importante lembrar que o sucesso desta etapa dependerá, sobretudo, da participação e do perfil do usuário.

5.1.2.3 DESENVOLVIMENTO DO MODELO VIVO DE TRABALHO

Nesta etapa, em curto espaço de tempo, deve ser construído o modelo de operacionalização do protótipo. Quatro são as técnicas utilizadas na construção do modelo vivo:

- a) técnica de construção de diálogos/interatividade;
- b) técnica de elaboração e geração de telas;
- c) técnica de elaboração e geração de relatórios;
- d) técnica de geração de estrutura e de relacionamento de dados.

A capacitação e experiência do analista é fator protuberante no sucesso desta etapa determinando o tempo de construção do protótipo.

5.1.2.4 DEMONSTRAÇÃO E USO DO MODELO

A demonstração é um recurso eficaz para expor o projeto e as idéias. Contribui decisivamente para a aprovação ou rejeição de uma solução qualquer. Todos os usuários que diretamente estão ligados com o sistema devem participar para que se identifique erros, omissões de dados, esquemas de segurança e se avalie a funcionalidade, a potencialidade e a técnica de diálogo aplicada no protótipo.

5.1.2.5 REVISÃO E MELHORAMENTOS

Em determinadas circunstâncias os problemas identificados são tantos que se torna viável o abandono de um programa ao invés de corrigi-lo ou acrescentar novos laços de

rotinas não previstas anteriormente. Quando as modificações são de pequeno volume e de fácil manutenção, é aconselhável que sejam realizadas na frente do usuário.

5.1.2.6 UTILIZAÇÃO DO PROTÓTIPO

Alguns autores, entre eles [MEL90], afirmam que não se deve implementar um protótipo. Tal afirmação parte do conceito do protótipo descartável, que serve somente para identificar os requisitos e depois é abandonado; ao contrário do protótipo evolutivo que geralmente é criado na própria linguagem que será a definitiva, sempre agregando novas estruturas (códigos, funções, bancos de dados, etc.) ao original.

Mas apesar do ponto de vista de [MEL90], acredita-se que ainda é uma das metodologias mais rápidas e simples, mesmo antes da versão final do sistema, para ter-se uma noção do funcionamento do mesmo.

A abordagem de metodologia de prototipação genérica pode ser classificada como sendo de prototipação básica ou fundamental. O protótipo, uma vez pronto, se presta tão somente como base a construção do sistema definitivo.

Este trabalho utiliza-se de algumas fases/etapas da metodologia de prototipação genérica proposta por [MEL90].

5.2 DELPHI

A programação, há tempos atrás, era uma atividade restrita a usuários “experts” em computação, pois exigia não apenas lógica, mas um profundo conhecimento dos recursos e das limitações das ferramentas disponíveis. A nova geração de ferramentas de programação visual está surgindo para possibilitar usuários, com algum conhecimento em desenvolvimento de aplicações, desenvolverem seus próprios programas, tornando menos abstrato a relação homem/máquina [RUB95].

5.2.1 CARACTERÍSTICAS

Algumas características do Delphi:

- a) descendente do Turbo Pascal;
- b) programação orientada a objetos;

- c) programação baseada em eventos;
- d) linguagem compilada e não interpretada;
- e) desenvolvimento rápido de aplicações (RAD);
- f) padrão SQL em banco de dados;
- g) conectividade através de ODBC.

5.2.2 RECURSOS

O Delphi contém um conjunto de recursos que vai desde seu criador de formulários (*Form Designer*) até o suporte transparente a todos os formatos mais comuns de banco de dados. Alguns dos principais recursos segundo [MAT96] e [OSI97]:

- a) componentes reutilizáveis e ampliáveis;
- b) suporte a OCX;
- c) modelos de aplicativos e de formulários;
- d) ambiente personalizado de desenvolvimento;
- e) programas compilados;
- f) recursos eficientes de acesso a dados;
- g) criação de bibliotecas de funções, as DLL's;
- h) assistente para criação de formulários.

Entre os recursos citados acima, destacam-se dois: os componentes reutilizáveis e ampliáveis e a criação de DLL, pois com o surgimento dos mesmos aprimoraram-se as formas de programação em módulos [MAT96].

5.2.3 AMBIENTE DE DESENVOLVIMENTO

O ambiente de desenvolvimento do Delphi 3 está de acordo com o padrão Windows 95. A figura 12 é constituída de quatro janelas, gerenciadas como um aplicativo MDI (*Multiple Document Interface*). A janela principal do Delphi controla várias janelas afins (*Object Inspector, Form e Code Editor*).

Neste trabalho será utilizado o bando de dados Paradox conforme figura 13, em se tratando de uma simulação da tecnologia, pela facilidade de implementação dentro do Delphi. Com o crescimento certo das informação no banco de dados, deveria ser pesquisado alguns

aspectos como confiabilidade, capacidade de gerenciamento e tempo de resposta para a utilização deste ou de outro sistema de banco de dados.

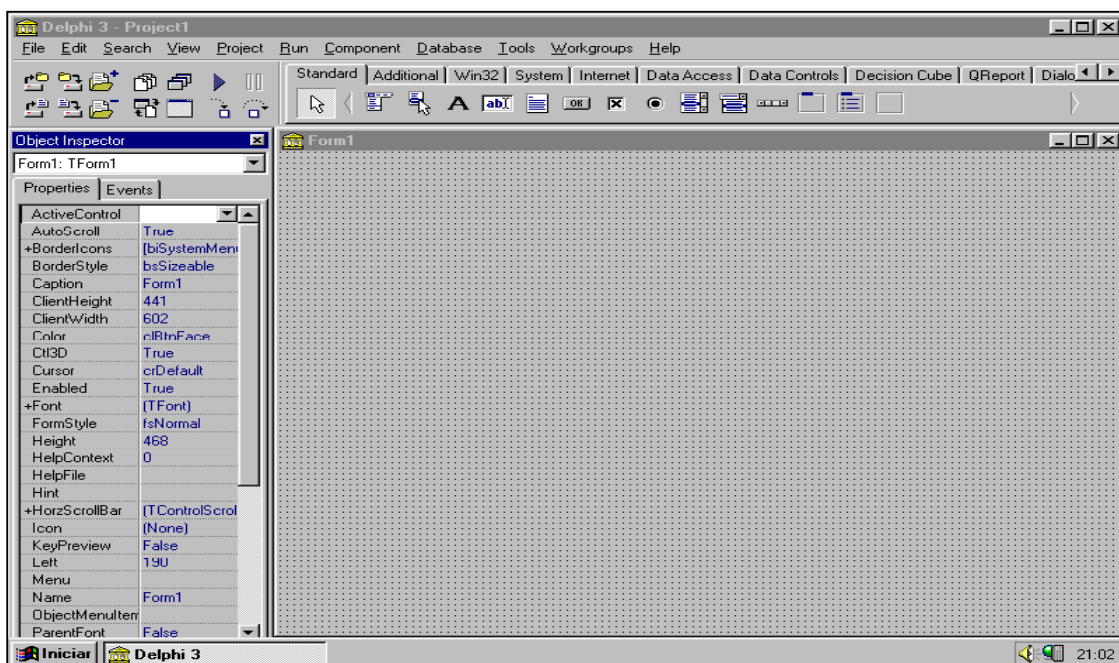


Figura 12 - Tela principal do Delphi 3.

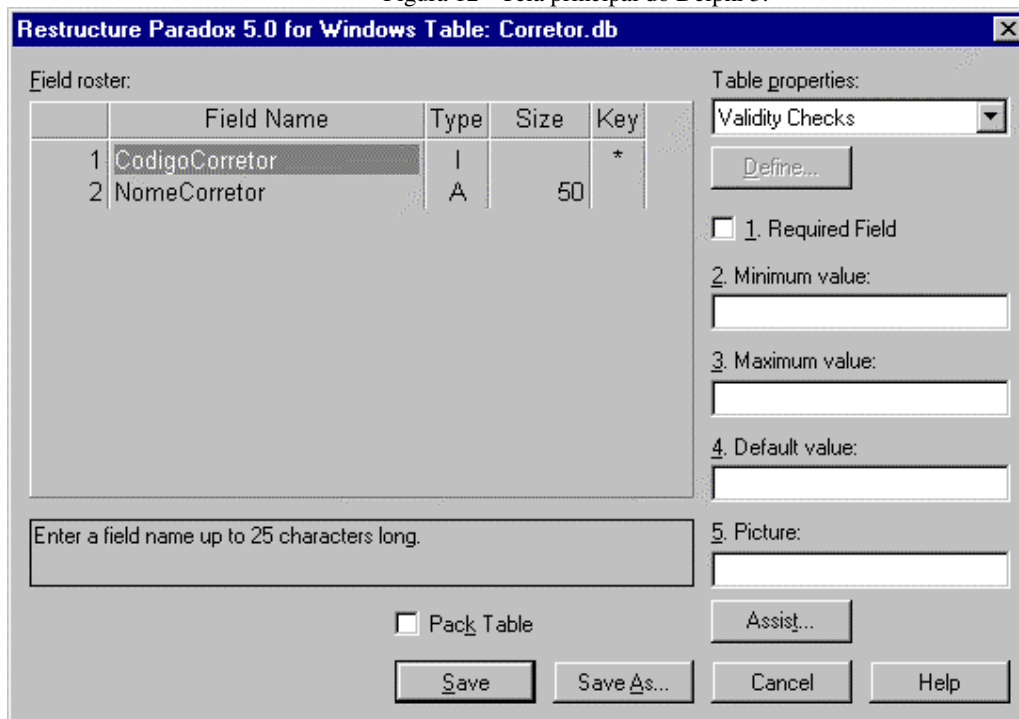


Figura 13 - Tela de definição das tabelas.

5.3 SYSTEM ARCHITECT

O *System Architect* é uma ferramenta *CASE* de boa flexibilidade e funcionalidade, com grande utilização no meio acadêmico e empresarial. Uma característica, é a compatibilidade com os sistemas operacionais Windows, facilitando muito o desenvolvimento dos sistemas.

O *System Architect* facilitou o desenvolvimento do Diagrama de Contexto e DFD (Diagrama de Fluxo de Dados). Facilidade no desenvolvimento porque possui um caixa de ferramentas com todos os itens como: processos, entidade externa, depósitos, etc. Possui facilidade na definição dos fluxos de dados, não permitindo nomes iguais aos fluxos. Para a criação do Diagrama de Nível 0, tem facilidade porque é somente necessário apertar o botão direito do *mouse* e automaticamente ele abre outra janela para as definições necessárias.

A dificuldade encontrada para a modelação do Protótipo foi a dificuldade na elaboração do MER (Modelo Entidade Relacionamento) e o dicionário de dados. Pontos e funções não encontrados no mesmo.

5.4 ROTEIRO PARA PROJETO DE *DATA WAREHOUSE* SEGUNDO W.H. INMON

Existem dois importantes aspectos vinculados à construção do *Data Warehouse*, o projeto da interface com os sistemas operacionais e projeto do *Data Warehouse* propriamente dito. De certa forma, projeto não é a descrição exata do que acontece durante a construção do *Data Warehouse*, uma vez que ele é construído de modo heurístico. Primeiro, o *Data Warehouse* é povoado com alguns dados. Tais dados são, então, usados e minuciosamente examinados pelo analista de SAD. Em seguida, com base no retorno proporcionado pelo usuário final, os dados são modificados e/ou outros dados são adicionados ao *Data Warehouse* [INM97].

5.4.1 JUSTIFICATIVA DE CUSTOS

Primeiramente, têm-se que aprovar o projeto de investimento junto aos diretores da empresa. Um dos aspectos interessantes do *Data Warehouse* é que a justificativa de custos do *Data Warehouse*, normalmente, não é feita mediante um critério previamente estabelecido de

retorno sobre o investimento. Para efetuar tal análise seria necessário que os benefícios do *Data Warehouse* fossem conhecidos antes da construção destes.

Na maioria dos casos, os verdadeiros benefícios do *Data Warehouse* não são conhecidos ou mesmo previstos no momento da construção porque o *Data Warehouse* é usado de forma inteiramente diferente de outros dados e sistemas. A atualização do *Data Warehouse* ocorre de um modo diferente do resto do processamento de informações. De fato, o analista de SAD não pode dizer quais as possibilidades e potencialidades do *Data Warehouse* até que a primeira iteração deste seja criada e esteja disponível. Então, uma vez que o analista de SAD ponha as mãos no *Data Warehouse*, ele poderá começar a liberar o potencial de processamento SAD.

5.4.2 BUSCA PELOS DADOS OPERACIONAIS

No início, há dados operacionais totalmente trancados no sistemas existentes. É tentador pensar que a criação do *Data Warehouse* consiste em apenas extrair dados operacionais e inseri-los no *Data Warehouse*. Nada poderia estar mais longe da verdade.

A figura 14 mostra uma representação simples da migração de dados do ambiente de sistemas já existentes para o *Data Warehouse*. Podemos verificar que várias aplicações irão contribuir para o *Data Warehouse*.

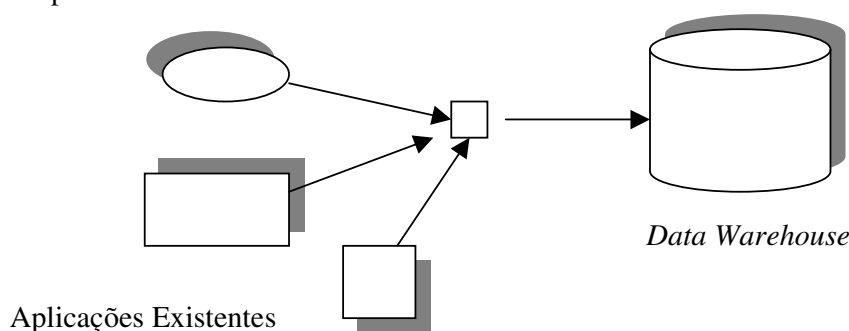


Figura 14 – Passagem dos dados dos sistemas transacionais para o *Data Warehouse* [INM97]

Por inúmeras razões, a figura 14 pode ser considerada exageradamente simplista. A idéia de que a construção de um *Data Warehouse* consiste meramente em um processo de extração tem como erro básico o fato de que os dados existentes no ambiente operacional não são integrados. A falta de integração é comum ao ambiente dos sistemas existentes, sendo que na época da criação das aplicações, a possibilidade de futura integração desta não era

considerada. Essa questão de falta de integração é o pesadelo dos programadores pelos programas de extração. Há mil e um detalhes a observar na programação apenas para extrair dados do ambiente operacional de forma correta.

Contudo, a integração (ou a falta de) dos sistemas existentes não é a única dificuldade na passagem de dados do ambiente dos sistemas operacionais existentes para o ambiente de *Data Warehouse*. Outro importante problema diz respeito ao acesso eficiente aos dados dos sistemas existentes.

Há três tipos de carga que podem ser feitos do ambiente operacional para o *Data Warehouse*:

- o carregamento de dados históricos;
- o carregamento de dados de valor corrente no ambiente operacional;
- o carregamento de alterações do *Data Warehouse* a partir de alterações (atualizações) que tenham ocorrido no ambiente operacional desde a última atualização do *Data Warehouse*.

Como regra, o carregamento de dados históricos representa um desafio menor uma vez que ele não é feito com frequência.

Da mesma forma, o carregamento de dados do ambiente operacional existente também constitui grande desafio porque precisa ser feito apenas uma vez. Em geral, o ambiente de sistemas operacionais existentes pode ser descarregado em um arquivo seqüencial e este pode ser descarregado no *Data Warehouse* sem acarretar danos ao ambiente online.

O carregamento de dados durante o processo normal – enquanto são efetuadas as alterações sobre o ambiente operacional – consiste no maior desafio ao arquiteto de dados. Não é fácil realizar o rastreamento eficiente e o tratamento dessas alterações. A varredura de arquivos existentes é, portanto, uma importante questão a ser enfrentada pelo arquiteto do *Data Warehouse*.

Há cinco técnicas comumente usadas para limitar a quantidade de dados pesquisados, conforme descrito abaixo:

- a primeira técnica consiste em pesquisar dados que apresentem marcas de tempo. Quando a aplicação registra um alteração, ou atualização de um registro;

- a segunda técnica de limitação de dados a serem pesquisados para uma extração para o *Data Warehouse* consiste em varrer um arquivo “delta”. Um arquivo desta é um arquivo criado por uma aplicação e que contém apenas as alterações efetuadas por esta.
- a terceira técnica consiste em varrer um arquivo de auditoria ou log. O arquivo de log ou de auditoria contém basicamente dados do mesmo tipo dos de um arquivo delta.
- a quarta técnica empregada no gerenciamento da quantidade de dados pesquisada durante a extração para o *Data Warehouse* consiste em modificar o código da aplicação.
- a última opção consiste em moldar um arquivo de imagem anterior e posterior. Segundo esta opção, um instantâneo de um banco de dados é tirado no momento da extração.

Outra importante observação a ser feita durante a passagem dos dados do ambiente de sistemas operacionais existentes para o ambiente de *Data Warehouse* é referente à necessidade de gerenciar o volume de dados. É preciso efetuar condensação dos dados; de outro modo, o volume de dados contidos no *Data Warehouse* logo ficará grande demais para ser controlado. A condensação dos dados deve ser iniciada no momento da extração.

5.4.3 COMPLEXIDADE DE TRANSFORMAÇÃO E INTEGRAÇÃO

À primeira vista, quando os dados são movidos do ambiente herdado para o ambiente de *Data Warehouse*, parece que nada além de simples extrações de dados de um local para o próximo está ocorrendo. Em virtude dessa enganosa simplicidade, muitas empresas começam a construir seus *Data Warehouse* manualmente.

Contudo, primeiras impressões podem ser muito enganadoras. O que em um primeiro momento parece ser nada mais do que a movimentação de dados de um local para outro transforma-se, em uma grande e complexa tarefa.

Precisamente, alguns dos tipos de funcionalidade necessário durante a passagem do ambiente operacional herdado para o ambiente de *Data Warehouse*:

- a extração de dados do ambiente operacional para o ambiente de *Data Warehouse* demanda uma mudança na tecnologia. Isso, normalmente, inclui a leitura segundo a tecnologia do SGBD operacional e a gravação dos dados por meio de uma tecnologia mais nova, de SGBD de *Data Warehouse*. A medida que os dados vão sendo transferidos, há necessidade de mudar a tecnologia.

- a seleção de dados do ambiente operacional pode ser muito complexa. Para qualificar um registro para o processamento de extração, diversas pesquisas a outros registros existentes em uma variedade de arquivos devem ser efetuadas, demandando leituras indexadas, lógica de conexão e assim por diante.
- as chaves de entrada operacionais geralmente precisam ser reestruturadas antes de serem gravadas. Muito raramente uma chave de entrada permanece inalterada ao ser lida no ambiente operacional e gravada no ambiente do *Data Warehouse*. Em casos simples, um elemento de tempo é acrescentado à estrutura de chave. Em casos complexos, toda a chave de entrada precisa passar por um novo processo de hashing, ou ser reestruturada.
- os dados são reformatados.
- os dados passam por uma limpeza. Em alguns casos, um algoritmo simples é aplicado aos dados de entrada para corrigi-los. Em situações complexas, recorre-se para sub-rotinas de inteligência artificial para limpar os dados de entrada deixando-os em uma forma de saída aceitável.
- existem várias fontes de dados. Segundo um conjunto de condições, a fonte de um elemento de dados de uma *Data Warehouse* consiste em um arquivo e, segundo um outro conjunto de condições, a fonte de dados para o *Data Warehouse* é outro arquivo. A lógica deve ser esclarecida para que a fonte de dados apropriada contribua com seus dados segundo o conjunto de condições correto.
- quando há vários arquivos de entrada, a escolha das chaves deve ser feita antes que os arquivos sejam intercalados. Isso significa que se diferentes estruturas de chaves são usadas nos diferentes arquivos de entrada, então a lógica que permite a escolha deve estar embutida no programa de intercalação.
- Onde houver vários arquivos de entrada, a seqüência de arquivos pode não ser a mesma e até se incompatível. Nesse caso, os arquivos de entrada terão de ter sua seqüência ajustada. Isso não é problema, a menos que haja muitos registros a resseqüenciar, o que quase sempre é o caso.
- pode haver vários resultados. Dados podem ser produzidos em diferentes níveis de resumo pelo mesmo programa de criação do *Data Warehouse*.

- Valores *default* devem ser fornecidos. Sob determinadas condições um valor de saída do *Data Warehouse* não terá fonte de dados. Nesse caso, o valor *default* a ser usado deve ser fornecido.
- a eficiência na escolha dos dados de entrada para a extração torna-se freqüentemente uma questão importante. Considere o caso em que no momento da renovação não há como diferenciar os dados operacionais que precisam ser extraídos daqueles que não precisam ser extraídos. Quando isso acontece, todo o arquivo operacional tem que ser lido. A leitura do arquivo inteiro é particularmente ineficiente porque só uma fração de registros é, de fato, necessária. Esse tipo de processamento faz com que o ambiente operacional esteja ativo, o que torna o ambiente online mais apertado para outros processos.
- Freqüentemente é necessário resumir dados. Você registros operacionais de entrada são “combinados” em um único registro de “perfil” do *Data Warehouse*. Para efetuar o resumo, os registros de entrada detalhados candidatos ao resumo devem ser dispostos em uma seqüência adequada. Caso diferentes tipo de registro contribuam para o registro resumido do *Data Warehouse*, a chegada destes deve ser coordenada de forma que um único registro seja produzido.
- A alteração de nomes de elementos de dados durante a passagem do ambiente operacional para o ambiente de *Data Warehouse* deve ser registrada. Quando um elemento de dados é motivo, seu nome, geralmente é alterado. A documentação dessa mudança precisa ser feita.
- Os registros de entrada que precisam ser lidos apresentam formatos “exóticos” ou não-padronizados. Há toda uma gama de tipos de entrada que devem ser lidos:
 - Registro de tamanho físico;
 - Registro de tamanho variável;
 - *Occurs depending on* (ocorrências condicionais);
 - Cláusula *occurs*.
- A conversão precisa ser feita. Contudo a lógica de conversão deve ser especificada e a mecânica de conversão (a aparências anterior e posterior) pode ser bem complexa. Em alguns casos, lógica de conversão se torna muito complicada

- Os relacionamentos entre dados que foram embutidos na lógica dos antigos programas herdados precisam ser compreendidos e desemaranhados antes que esses arquivos possam ser usadas como entrada. Esses relacionamentos são, com frequência, misteriosos, difíceis de alterar e não-documentados.
- A conversão do formato dos dados precisa ser feita. A conversão de EBCDIC para ASCII (ou vice-versa) precisa ocorrer.
- Volumes massivos de entrada devem ser levados em consideração. Quando há somente uma pequena quantidade de dados de entrada, diversas opções de projeto podem ser conciliadas. Mas quando há muitos registros de entrada, opções especiais de projeto (como cargas em paralelo ou leitura em paralelo) talvez tenham que ser empregadas.
- O projeto do *Data Warehouse* deve adequar-se ao modelo de dados corporativo. Dessa forma, há ordem e disciplina no projeto e estruturação do *Data Warehouse*. A entrada para o *Data Warehouse* ajusta-se às especificações de uma aplicação que foi escrita há muito tempo. As condições de negócios por trás da aplicação provavelmente já se alteraram umas dez vezes desde que a aplicação foi criada. Muita manutenção sem correspondente documentação deve ter sido feita sobre o código de aplicação. Além disso, a aplicação provavelmente não apresentará requisitos de integração para se acoplar a outras aplicações. Todas essas disparidades devem ser levadas em consideração durante o projeto e construção do *Data Warehouse*.
- O *Data Warehouse* espelha as informações históricas necessárias, ao passo que o ambiente operacional focaliza as informações correntes.
- O *Data Warehouse* destina-se a suprir as carências informacionais da empresa ao passo que o ambiente operacional serve ao atendimento das necessidades funcionais imediatas da empresa.
- A transmissão do arquivo de saída recém-criado que passará para o *Data Warehouse* deve ser cuidadosamente considerada. Em alguns casos, isso é muito fácil; em outros, definitivamente não é fácil, especialmente quando os sistemas operacionais são cruzados.

Esta lista é meramente o início das complexidades que vão desafiar o programador na largada para a carga do *Data Warehouse*.

6 DESENVOLVIMENTO DO PROTÓTIPO

Conforme [MEL90], descreve que a diferença entre as metodologias de pré-especificação e de prototipação não se restringem aos aspectos filosófico e estratégico de abordagem de problemas e soluções de sistemas. Assim sendo, este capítulo aborda o desenvolvimento específico deste trabalho, demonstrando a viabilidade deste protótipo, suas necessidades e requisitos, bem como sua implementação.

6.1 VIABILIDADE

Um dos principais fatos geradores, que foi levado em consideração para a realização deste trabalho foi a possibilidade da utilização desta tecnologia chamada de *Data Warehouse* em companhias seguradoras. Em função da crescente concorrência entre elas, e com o intuito de poder facilitar a melhor forma para que os valores cobrados dos segurados de automóveis estivesse mais próximo da sua realidade ou o seu perfil como condutor do veículo, e não o veículo que esta dirigindo. Utilizando desta maneiras todas as informações armazenadas no banco de dados operacional da seguradora.

O Delphi 3 também vem sendo largamente utilizado, pelo fato de ter sido adotado pela maioria dos programadores que já programavam com a linguagem Pascal. Tendo maior afinidade com esta linguagem de programação, inclusive a utilização do banco de dados Paradox.

6.2 MIGRAÇÃO DE DADOS PARA *DATA WAREHOUSE* CONFORME CAPÍTULO 4

Um processo importante na construção de um *Data Warehouse* refere-se a migração de dados do banco de dados operacionais para o banco de dados do *Data Warehouse*. Algumas tarefas nesta etapa são, extração, limpeza, transformação, etc. O processo de carregamento ou povoamento tem o sentido de inserir registros de dados em um banco de dados, neste caso em um banco de dados de *Data Warehouse*. Abaixo será citado os principais Processos Associados para o carregamento do *Data Warehouse*:

- a) processo de extração: este processo, foi efetuado através de informações geradas pelo Departamento de Sinistro, na Matriz da Bradesco Seguros no Rio de Janeiro,

através de um arquivo do tipo texto e posteriormente carregado na Sucursal Blumenau no sistema de *Data Warehouse*.

- b) processo de limpeza: algumas informações contidas no arquivo gerado, não serão utilizadas, como exemplo, a oficina prestadora dos serviços, data de entrega do processo, código do corretor, etc.
- c) processo de transformação: as informações geradas pela Matriz, possuem inconsistências que deverão ser adequadas as novas necessidades do *Data Warehouse*.
- d) processo de transporte (migração): este processo será executado de forma periódica, e as informações serão geradas semanalmente com todos os sinistro liquidados pela Sucursal. Este processo será feito através de método *batch*, possibilitando que os dados preparados possam ser movidos para o *Data Warehouse*, em grandes lotes.
- e) complexidades do processo migratório: em muitos fatores podemos destacar alguns, entre eles, a mudança de tecnologia, dados reformatados, limpeza, leitura no ambiente operacional complexa, etc.
- f) considerações sobre o desenvolvimento de um *Data Warehouse*: o *Data Warehouse* foi desenvolvido com o base nas informações contidas no banco de dados operacional somente no seguro de automóveis. Tendo em vista o alto custo para o desenvolvimento do sistema. Possibilitando selecionar um grupo de usuários, juntamente com o protótipo, praticando em quantidade de dados menores.

6.3 NECESSIDADES E REQUISITOS

Segundo [MEL90], trata-se da etapa que serve de base para a elaboração do modelo preliminar do protótipo.

Na figura 15, apresenta-se o Diagrama de Contexto, visando detalhar o modelo conceitual do Protótipo de *Data Warehouse*. O sistema foi desenvolvido com a ferramenta case *System Architect*, utilizando-se das definições e metodologias de Yourdon/DeMarco.

Os aperfeiçoamentos e complementações do Diagrama de Contexto são realizados através do Diagrama de Fluxo de Dados (DFD), em sucessivos refinamentos. [HUG97] define que o DFD é um modelo estático (pois representa uma única situação) que mostra como a informação é transformada pelo sistema. Na figura 16, é demonstrado o DFD - Nível 0.

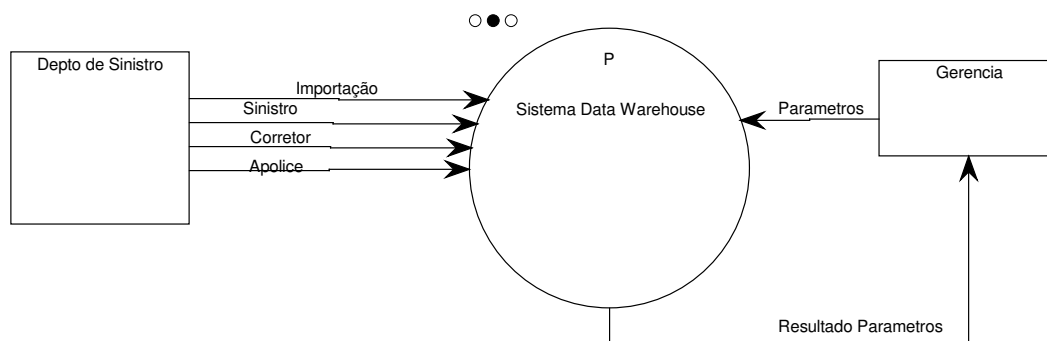


Figura 15 – Diagrama de Contexto *Data Warehouse*

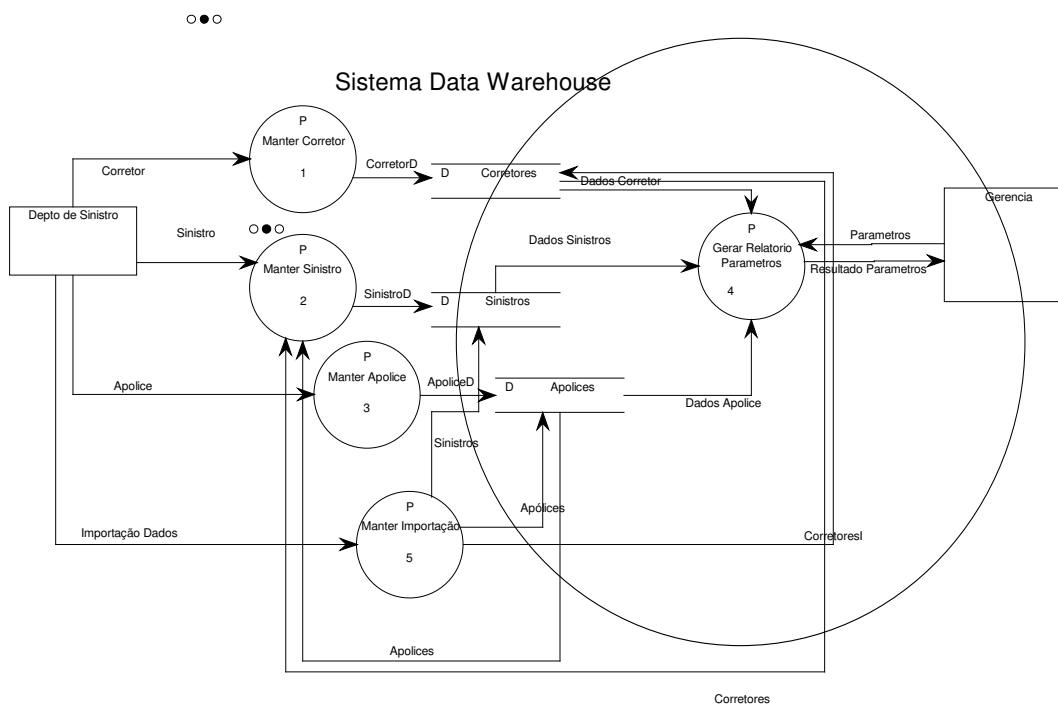


Figura 16 - Diagrama de Fluxo de Dados (DFD) *Data Warehouse* - Nível 0

De acordo com [HUG97], um processo é a transformação de fluxo de dados de informação em fluxo de dados de resultados, sendo que no DFD apresentado, existem quatro processos:

- processo 1: Atualizar Corretor - é o processo pelo qual são atualizados os corretores que possuem segurados com sinistro;
- processo 2: Atualiza Apólice - é o processo pelo qual são atualizados as apólice que possuem sinistros;
- processo 3: Atualizar Sinistro - é o processo pelo qual são atualizados e/ou cadastrados os sinistros ocorridos com os segurados;

- d) processo 4: Gerar Relatório/Parâmetros - é o processo pela qual, o sistema realizará o processamento com base nas informações selecionadas pelo usuário, conforme tela de parametrização, e a pesquisa em toda a base de dados do Sistema de *Data Warehouse*, percorrendo de forma seqüencial, e posterior visualização das informações na tela. ;
- e) processo 5: Manter Importação – é o processo pela qual, o sistema é carregado com as informações necessário ao banco de dados do *Data Warehouse*. As informações são geradas pela Matriz da Bradesco Seguros no Rio de Janeiro, através de um arquivo tipo texto. Este arquivo é acessado pelo sistema, fazendo desta maneira todos os processo necessários, entre eles: limpeza, extração, transporte, etc.

No DFD apresentado, constam três depósitos de dados, sendo:

- a) depósito corretor: onde serão armazenadas os informações sobre os corretores das apólices, conforme tabela 6;
- b) depósito apólice: onde serão armazenadas as informações das apólice que ocorreram sinistros, conforme tabela 6;
- c) depósito sinistros: onde serão armazenadas as informações sobre os sinistros ocorridos, conforme tabela 6;

Num projeto de protótipo, descreve [MEL90], o Modelo Lógico é mais importante do que o Modelo Conceitual. Neste trabalho será abordado o enfoque relacional, onde é adotado o conceito de Relações, onde cada Entidade Lógica é vista como uma relação ou tabela bidimensional, composta de linhas e colunas.

Segundo [HUG97], uma entidade é uma representação do mundo real, que participa do problema, e que é modelada de acordo com a relevância para o sistema. Dois critérios são essenciais para a definição do Modelo Entidade x Relacionamento:

- a) tem que ter atributos (características);
- b) tem que ter ocorrências.

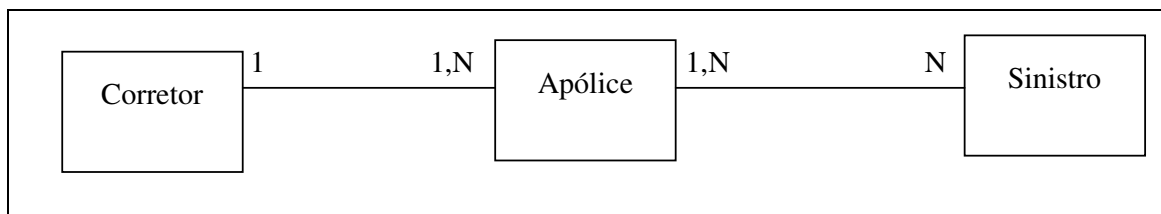


Figura 17 - Detalha-se o Modelo Entidade x Relacionamento do *Data Warehouse*

Conforme, [HUG97] define Dicionário de Dados como um repositório de dados sobre os dados do sistema, tratando-se de um banco de dados sobre os dados do sistema. Quanto ao Dicionário de Dados do sistema de *Data Warehouse*, o mesmo é descrito na tabela 6.

Tabela Corretor	Tipo	Tamanho
CodigoCorretor	Numérico	
NomeCorretor	Alfanumérico	50

Tabela Apolice	Tipo	Tamanho
NumeroApoliceItem	Alfanumérico	10
NomeSegurado	Alfanumérico	50
TipoPessoa	Numérico	
CPF CGC	Alfanumérico	11
TipoVeiculo	Numerico	
Chassi	Alfanumerico	20
TipoCobertura	Numerico	
VigenciaInicial	Data	
VigenciaFinal	Data	
Licença	Alfanumérico	7

Tabela Sinistro	Tipo	Tamanho
NumeroSinistro	Numérico	
DataSinistro	Data	
HoraSinistro	Tempo	
LocalSinistro	Numerico	
NomeCondutor	Alfanumérico	50
CPF Condutor	Alfanumérico	11
Sexo	Numerico	
Idade	Numérico	
DataHabilitação	Data	
EstadoCivil	Numérico	
TipoSinistro	Numérico	
TipoColisão	Numérico	
UtilizaçãoVeículo	Numerico	

Tabela 6 – Dicionário de Dados do Sistema de *Data Warehouse*

6.4 DESENVOLVIMENTO DO MODELO VIVO

Nesta fase elabora-se o modelo operacional do protótipo, com a apresentação das telas de cadastro dos dados e resultados obtidos. A figura 18, apresenta a tela principal do Sistema de *Data Warehouse*, com as opções de cadastro, relatório, sair e sobre.

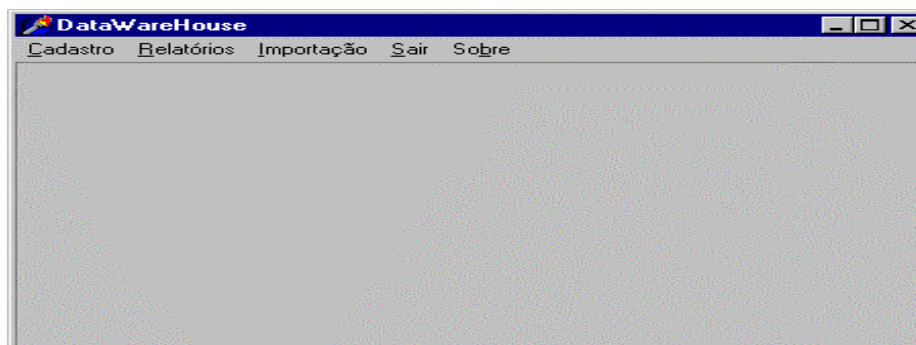


Figura 18 - Tela Principal do Sistema de *Data Warehouse*

Como já foi descrito anteriormente o processo de cadastro e/ou carregamento do Sistema de *Data Warehouse*, foi efetuado manualmente devido a indisponibilidade de acesso ao banco de dados da seguradora. As figuras 19, 20 e 21 apresentam as telas de cadastro de corretores, apólice e sinistros, onde foi utilizado o banco de dados Paradox do Delphi, e utilizando os componentes *Data Source* e *Data Set*, para acesso ao banco de dados.

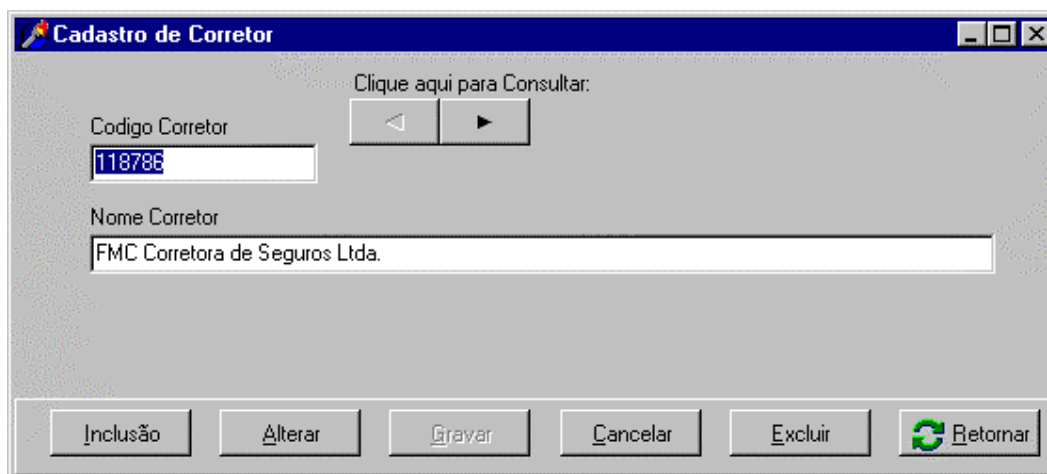


Figura 19 - Tela de Cadastro Corretor

Cadastro Apólice

Clique aqui para consultar

Numero Apólice - Item: 100548-0157

Chassi: 9bwzzz

Nome Segurado: Jose

Tipo Pessoa
 Fisica Juridica

CPF/CGC: 80049800906

Marca: VW

Tipo Veiculo
 Esportivo Passeio

Tipo Cobertura
 Compreensiva RCF

Vigencia Inicial: 10/10/99

Vigencia Final: 10/10/00

Licença: lyz1020

CodigoCorretor: FMC Corretora de Seguro:

Inclusão Alterar Gravar Cancelar Excluir Retornar

Figura 20 - Tela de Cadastro de Apólice

Utilizando-se desta tela foi construído a base de dados necessários para a simulação e análise, possibilitando desta forma uma tomada de preço conforme banco de dados. Sendo que o banco de dados tende a crescer constantemente onde as informações deverão ser acrescentadas mensalmente a base de dados.

O funcionamento do sistema de cadastro de corretor, apólice e sinistro, ocorrerá da seguintes maneira:

- a) os corretores serão cadastrados somente uma vez, tendo em vista que o código de corretor não muda, é chave primária;
- b) o cadastro das apólices somente serão efetuado quando ocorrer um sinistro, para possibilitar a identificação de alguns itens como: tipo de veículo, segurado, cobertura, vigência da apólice e corretora. O número da apólice é chave primária, e o corretor será chave estrangeira, possibilitando o relacionamento entre as tabelas;
- c) o cadastro dos sinistros será efetuado conforme informações obtidas dos processos em poder da seguradora, e serão identificados vários itens conforme figura 24.

- d) os processos acima citados poderão auxiliar quando da inclusão de algum sinistro não informado na base de dados operacional, visto que toda a base de dados do sistema de *Data Warehouse* será carregado semanalmente, através de arquivos gerados na Matriz da Bradesco Seguros, com todas as informações do sinistro liquidados pela companhia.

The image shows a software window titled "Cadastro Sinistro". It contains the following fields and controls:

- CPF Conductor: 800.498.009-06
- Sexo: Masculino Feminino
- Idade: 19
- Data da Habilitação: 10/09/98
- Estado Civil: Casado Solteiro Outros
- Tipo Sinistro: Colisão Furto
- Tipo Colisão: Frontal Traseira
- Utilização Veículo: Particular Vendas Outros
- Nome Conductor: Denize
- Local Sinistro: Secundarias Urbana Rodovia
- Numero Sinistro: 1
- Numero da Apolice Item: 1451010001
- Data Sinistro: 10/05/99
- Hora Sinistro: 10:30:00

Buttons at the bottom: Incluir, Alterar, Gravar, Cancelar, Excluir, Retornar.

Figura 21 - Tela de Cadastro de Sinistros

A figura 22, apresenta uma tela para o usuário do sistema de *Data Warehouse*, possa efetuar as suas pesquisas, diante de alguns parâmetros que deverá ser selecionado para a pesquisa, entre eles: idade, sexo, estado civil, tempo carteira, tipo veículo.

The screenshot shows a software window titled "Relatório de Parametrização". It contains a section for "Tipo Parametros" with several checkboxes and dropdown menus. The "Idade" checkbox is checked, and its range is set from 18 to 70. The "Sexo" checkbox is also checked, with "Masculino" selected. Other parameters like "Estado Civil", "Carteira", and "Tipo Veiculo" are not checked. Below this is a table with the following data:

Idade	Quant	Masculino	Feminino	Casado	Solteiro	Outros		
18	0	0						
19	1	0						
20	1	1						
21	0	0						
22	3	2						
23	0	0						

At the bottom of the window are three buttons: "Limpar Tela", "Retornar", and "Processar".

Figura 22 - Tela de Parametrização e resultado da pesquisa

Para ser realizada as pesquisas na base de dados, foram criado algumas rotinas de pesquisa seqüencial com o comando 'for', percorrendo desta forma todo o banco, e acumulando todas informações conforme parâmetros selecionados. Diante da possibilidade e necessidade de seleção de somente algumas idades, sexo, estado civil, etc., foi utilizado o evento 'OnFilterRecord', onde pode ser selecionado do banco de dados somente as idades, sexo, estado civil, etc., conforme parametrização.

Os itens pesquisados e selecionados são armazenados de forma temporária em vetores, até o final da pesquisa, e automaticamente são impressos na tela ou impressora conforme solicitação do usuário do Sistema de *Data Warehouse*.

Desta forma a seguradora poderá verificar não somente os valores dos sinistros pagos, mas também que tipo de motorista ocasionou o sinistro com o veículo segurado. Possibilitando uma análise mais detalhada na sua proposta de seguro. Tomando como base as informações dos segurados/motoristas dos veículos que tiveram sinistros.

7 CONCLUSÃO

Diante do estudo efetuado sobre *Data Warehouse*, constatou-se que as informações das grandes corporações, não são utilizadas de forma satisfatória. É nesse enfoque que os sistemas de *Data Warehouse* trabalham. Possibilitando avaliá-las e interpretá-las, como uma tecnologia meio, que facilita e auxilia na tomada de decisão.

Com a necessidade cada vez maior das decisões a serem tomadas pelas empresas, diante de um mercado competitivo, como é o mercado segurador, a tecnologia de *Data Warehouse*, torna-se instrumento importantíssimo a empresa, auxiliando para as tomada de decisões baseado em fatos históricos, e não em intuição. Utilizando de forma correta, todas as informações disponíveis nos bancos de dados, até então utilizados somente para consulta, alteração e impressão.

Com a sua utilização, da filosofia de *Data Warehouse*, juntamente com o roteiro de implementação de [INM97], podemos identificar quais as informações seriam essenciais para uma companhia seguradora. Utilizando de uma complexa transformação, limpeza e transporte dos dados, visto que a base operacional é de difícil e restrito acesso, e o volume de dados enorme.

Quanto ao seu desenvolvimento ter sido feito através da metodologia de prototipação estruturada, haja visto a mesma ser amplamente discutida por muitos autores, entre eles Melendez [MEL90], pareceu ser o instrumento ideal, em função do curto espaço de tempo, da carência de recursos e da disponibilidade de pessoal. Com a construção do protótipo, pode-se identificar as necessidades de uma sistema de *Data Warehouse*, a ser implementado em uma companhia seguradora.

Quanto as ferramentas utilizadas, o ambiente de programação visual Delphi 3, pode-se afirmar que ele possui recursos para a criação e/ou desenvolvimento dos mais completos sistemas, sendo ainda uma ferramenta em franca expansão. Prova disso é que já existe no mercado versões do Delphi 4. Já o banco de dados Paradox, apesar de ser de simples utilização, possui características necessárias para a simulação deste protótipo. Existe a necessidade de avaliação de qual o banco de dados seria mais apropriada para cada empresa, dependendo das suas necessidades. A maior dificuldade encontrada foi na elaboração e

utilização da ferramenta *CASE (System Architect)*, entre eles o *lay-out*, o MER e o Dicionário de Dados.

Através deste trabalho pode-se perceber que a filosofia de *Data Warehouse*, está em crescendo a cada dia. Visto que, se bem projetada, pesquisa, desenvolvida e utilizada por trazer uma relação custo/benefícios muito bom. Visto que em uma companhia seguro poderemos identificar quais os principais causadores de sinistro (prejuízo) a companhia seguradora. Sugere-se como extensão do trabalho o desenvolvimento de aplicativos para pesquisa em grandes base de dados, limpeza, transformação e transporte, para implantação em sistemas de *Data Warehouse*. E a apresentação destes conceitos nas disciplinas de banco de dados.

REFERÊNCIAS BIBLIOGRÁFICAS

- [BAP98] BAPTISTA, Evaristo – *Alternativas de Migração para Ambientes de Data Warehouse*, CPG da FURB, Blumenau, 1998 Monografia.
- [BAR96] BARQUINI, Ramon - *Planning and designing the Warehouse*, New Jersey, Prentice-Hall, 1996, 311 pg.
- [BAT86] BATINI, C. e LENZERINI, M. - *Comparative Analysis Of Methodologies For Database Schema Integration*, ACM Computing Surveys, New York, v.18, nº 4, pg.323-364, Dezembro 1986.
- [BOH97] BOHN, Kathy. **Convertendo dados para Data Warehouse**. Rio de Janeiro, DBMS, V.1, N.3, Julho 1997.
- [BRA98] BRADESCO SEGUROS S/A – Manual. *Mercado Segurador*, Rio de Janeiro 1998.
- [CAN96] CANTU, Marco. **Dominando o Delphi**. São Paulo : Makron Books, 1996.
- [CAM97] CAMPOS, Maria Luiza & ROCHA, Arnaldo V. Filho. *Data Warehouse*. **XVI Jornada de Atualização em Informática**, Brasília, p. 221-261, 1997.
- [CAM98] CAMPOS, Maria Luiza & ROCHA, Arnaldo V. Filho. *Data Warehouse*. **Notícias Sobre Data Warehouse**, 1998. Endereço Eletrônico : <http://www.ufrj.br>
- [CAR97] CARVALHO, Jackeline. *Data Warehouse: a Tecnologia que Revolucionaria Negócios*. **Computerworld**, Rio de Janeiro, v. 3, n 197, p 11-22, feb.1997.
- [CIO96] CIO/EUA. *Data Warehouse de A a Z*. Computerworld, V.3, N.158, P.9, Abril 1996.
- [DAL98] Dal’Alba, Adriano. Um Estudo Sobre *Data Warehouse*, Universidade de Caxias do Sul 1998. Endereço eletrônico : <http://www.geocities.com/SiliconValley/Port/5072/Sumario.htm>
- [DUN95] DUNTEMANN, Jeff. **Delphi: programming explorer**. Scottsdale: Coriolis Group Books, 1995.

- [GON97] GONÇALVEZ, Jorge Luiz Mendes. *Data Warehouse é Necessariamente um Megaprojeto?*. **Develpers' Magazine**, Rio de Janeiro, n. 6, p. 12-13, fev. 1997.
- [HAR96] HARJINDER, G. e RAO, P. C. – *The Officil Guide to Data Warehousing*, Que Corporation, 1996.
- [HAR98] HARRISON, H. Thomas **Intranet Data Warehouse**; São Paulo, Berkely Brasil, 1998.
- [HUG97] HUGO, Marcel. **Anotações de aula: Engenharia de software**. Blumenau: FURB, 1997.
- [INM97] INMON, William H. **Como Construir o Data Warehouse**; Rio de Janeiro: Campus, 1997.
- [KIM96] KIMBALL, Ralph. **Data Warehouse Toolkit**; São Paulo : Makron Books, 1998.
- [LAU96] LAUDON, K. C. e J.P. LAUDON. **Management Information Systems**. New Jersey, Prentice Hall, 1996.
- [MAT96] MATCHO, Jonathan...[et all]. **Usando o Delphi 2: o guia de referência mais completo**. Rio de Janeiro : Campus, 1996.
- [MEL90] MELENDEZ FILHO, Rubens. **Prototipação de sistemas de informação**. Rio de Janeiro : LTC, 1990.
- [MOR96] MORIARTY, T., Greenwood, R.P.. **Data's quest from source to query**. Database Programmin and design, V.9, N.10, Outubro 1996.
- [OLI98] OLIVEIRA, Adelize Generini. **Data Warehouse: Conceitos e Soluções**. Florianópolis : Advanced Editora, 1998.
- [ORL96] ORLI, Richard J. **Data Extraction, Transformation and Migration Tools**. USA, setembro, 1996, Endereço eletrônico : <http://www.kismeta.com/extract.html>

- [ORR96] ORR, Ken. – *Data Warehouse Technology*, The Ken Orr Institute – 1996, disponível na Internet via protocolo http, no endereço eletrônico : <http://www.kenorrinst.com/datawh.html>.
- [OSI97] OSIER, Dan. **Aprenda em 21 dias Delphi 2**. Rio de Janeiro : Campus, 1997.
- [RUB95] RUBENKING, Neil J. **Programação em Delphi para leigos**. São Paulo : Berkeley, 1995.
- [WEL95] WELDON, J. L., **Managing Multimentional data: Harnessing de Power** : Database Programming and Design, V.8, N.8, Agosto 1995.
- [WIL97] WILLIAMS, Joseph. **Ferramentas para a Transformação de dados** : Rio de Janeiro, DBMS, V.1, N.3, Julho 1997.