

ALEXANDRE BROLEZE

**CRIAÇÃO AUXILIADA POR
TECNOLOGIA CASE DE AMBIENTE DE
DATA WAREHOUSE**

**Florianópolis - SC
2002**

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO**

ALEXANDRE BROLEZE

**CRIAÇÃO AUXILIADA POR
TECNOLOGIA CASE DE AMBIENTE DE
DATA WAREHOUSE**

Dissertação submetida à Universidade Federal de Santa Catarina
como parte dos requisitos para obtenção do grau de Mestre em
Ciência da Computação.

Orientador: Prof. Dr. Luiz Fernando Jacintho Maia

Florianópolis, fevereiro de 2002

**CRIAÇÃO AUXILIADA POR
TECNOLOGIA CASE DE AMBIENTE DE
DATA WAREHOUSE**

ALEXANDRE BROLEZE

Esta dissertação foi julgada adequada para a obtenção do título de **Mestre em Ciência da Computação** Área de Concentração **Sistemas de Computação** e aprovada em sua forma final pelo **Programa de Pós-Graduação em Ciência da Computação**.



Prof. Dr. Fernando O. Gauthier
Coordenador do Curso

BANCA EXAMINADORA



Prof. Dr. Luiz Fernando Jacintho Maia – Orientador



Prof.ª Dr.ª Cristiane Von Wangenheim



Prof. Dr. João Bosco da Mota Alves



Prof. Dr. Vitorio Bruno Mazzola

Em memória de meu pai, Valdomiro Antonio Broleze.

"Algo só é impossível até que alguém duvide e prove o contrário"
Albert Einstein.

AGRADECIMENTOS

À Universidade Federal de Santa Catarina;

Ao orientador Prof. Dr. Luiz Fernando Jacintho Maia pelo acompanhamento pontual e competente;

Ao Prof. Dr. João Bosco da Mota Alves, pelas idéias e incentivo;

Ao Prof. Dr. Paulo da Silva Borges, pelo apoio e exemplo crítico;

Aos demais professores e colaboradores do curso de Pós-Graduação em Ciências da Computação da UFSC;

À empresa Agrícola Fraiburgo S/A, nas pessoas de Dr^a Carla Castro Salomão e Sr. Adenir Vanz pelo incentivo.

À Newtech Informática, na pessoa do Sr. Daniel Stack, pelo material, ferramentas e conhecimentos disponibilizados.

Ao Sr. Alexandre Biazin, por sua amizade, colaboração e incentivo;

Ao Sr. Otto Lessing, pelo companheirismo e apoio;

Ao Sr. Vanderlei Tesa, que disponibilizou sua base de dados para a aplicação das ferramentas e metodologias deste;

A todos que de uma maneira ou outra contribuíram para a realização deste trabalho.

SUMÁRIO

LISTA DE SIGLAS	VII
LISTA DE FIGURAS	IX
LISTA DE TABELAS	X
RESUMO	XI
ABSTRACT	XII
1 INTRODUÇÃO	1
1.1 Origem	1
1.2 Justificativa	4
1.3 Objetivos	5
1.4 Organização	6
1.5 Metodologia	6
2 BUSINESS INTELLIGENCE	7
2.1 <i>Data Warehouse</i>	10
2.1.1 Características do <i>Data Warehouse</i>	15
2.1.2 Dados Primitivos e Derivados	20
2.1.3 Níveis do <i>Data Warehouse</i>	21
2.1.4 Dados Externos	21
2.1.5 Indicadores x Dimensões	22
2.1.6 Procedimentos de Carga	23
2.1.7 Modelo de Dados Segundo R.Kimball	24
2.1.7.1 Modelo Empresarial	24
2.1.7.2 Modelo Dimensional	25
2.1.7.2.1 O Modelo E-R e o Modelo Estrela	30
2.1.7.3 Modelo Físico	31
2.1.8 Modelo de Dados Segundo W.H.Inmon	34
2.1.8.1 Modelo de Dados de Alto Nível	34
2.1.8.2 Modelo de Dados de Nível Intermediário	35
2.1.8.3 Modelo de Dados de Baixo Nível	36
2.2 <i>Data Marts</i>	37
2.3 Metadados	38
2.3.1 Fontes dos Metadados	40
2.4 OLAP	41
2.4.1 Características OLAP	42

3	FERRAMENTAS CASE.....	45
3.1	Tipos de CASE.....	48
3.2	Benefícios da Tecnologia.....	49
3.3	Características Desejáveis.....	50
4	CRIAÇÃO DE AMBIENTE DE DW UTILIZANDO CASE.....	53
4.1	Adaptação da Metodologia.....	53
4.2	Plano de Migração.....	58
4.3	Desenvolvimento de Sistemas Baseados em Dados.....	61
4.3.1	Fase Operacional.....	62
4.3.2	Construção do <i>Data Warehouse</i>	67
4.3.3	Utilização Iterativa do <i>Data Warehouse</i>	70
4.4	Checklist Para Revisão do Projeto.....	72
5	ESTUDO DE CASO.....	75
5.1	Escolha de Uma Base de Dados Para o Trabalho.....	75
5.1.1	Descrição da Base de Dados Operacional.....	76
5.2	Escolha da Ferramenta.....	77
5.2.1	Critérios de Escolha.....	77
5.2.2	Ferramenta Escolhida.....	78
5.3	Plano de Migração.....	79
5.4	Desenvolvimento dos Sistemas.....	83
5.5	Revisão do Projeto.....	86
6	CONCLUSÕES E SUGESTÕES PARA NOVOS TRABALHOS.....	88
7	BIBLIOGRAFIA.....	90
	Anexo I – Data View Generator.....	93
	Anexo II – Ambiente de desenvolvimento Genexus.....	94
	Anexo III – Gxplorer Metadata Manager.....	95
	Anexo IV – Gxplorer Olap Manager.....	96
	Anexo V – Pivô Table Excel 2000.....	97

LISTA DE SIGLAS

BI	– <i>Business Intelligence</i> – Inteligência de Negócios.
BD	– Banco de Dados.
DBF	– <i>Data Base File</i> – Arquivo de Banco de Dados.
CAD	– <i>Computer Aided Design</i> – Projeto Auxiliado por Computador
CASE	– <i>Computer Aided Software Engineer</i> – Engenharia de Sistemas Auxiliada por Computador.
DASD	– <i>Direct Access Storage Device</i> – Dispositivo de Armazenamento de Acesso Direto.
DFD	– Diagrama de Fluxo de Dados.
DIS	– <i>Data Item Sets</i> – Conjunto de Itens de Dados.
DM	– <i>Data Mining</i> – Mineração de Dados.
DSS	– <i>Decision Support System</i> – Sistema de Suporte à Decisão.
DW	– Data Warehouse – Armazém de Dados.
EIS	– <i>Executive Information Systems</i> – Sistema de Informações executivas.
ER	– Entidade Relacionamento.
ERP	– <i>Enterprise Resource Planning</i> – Ferramenta de Planejamento Empresarial.
JAD	– <i>Joint Application Design</i> – Comitê de desenvolvimento de aplicações.
OLAP	– <i>On-line Analytical Processing</i> – Processo analítico em tempo real
OLTP	– <i>On-line Transactional Processing</i> – Processo transacional em tempo real.

- SDLC – *System Development Life Cycle* – Ciclo de vida de desenvolvimento de sistemas.
- SGBD – Sistema Gerenciador de Banco de Dados.

LISTA DE FIGURAS

- Figura 1 – Componentes do Data Warehouse.
- Figura 2 – Necessidade de padronização na integração de aplicações.
- Figura 3 – Nível de granularidade.
- Figura 4 – Cubo de Informações.
- Figura 5 – Modelo Dimensional Tipo Estrela.
- Figura 6 – Modelo dimensional do tipo Floco de Neve.
- Figura 7 – Dependências transitivas entre tabelas.
- Figura 8 – Representação da modelagem de alto nível.
- Figura 9 – Os elementos do modelo de dados de nível intermediário.
- Figura 10 – Data Warehouse e Data Mart.
- Figura 11 – Migração para o ambiente projetado.
- Figura 12 – Desenvolvimento de Sistemas de Nível Operacional.
- Figura 13 – Fases do Desenvolvimento do DW
- Figura 14 – Construção de Relatórios/Consultas ao DW
- Figura 15 – Criação do Objeto Transação DW para a Dimensão Faixa Etária.
- Figura 16 – Criação do Objeto Transação DW para indicadores diversos.

LISTA DE TABELAS

Tabela 1 – Comparação entre Banco de Dados Operacionais e Data Warehouse.

Tabela 2 – Características da qualidade dos dados.

Tabela 3 – Dados Primitivos e Derivados.

Tabela 4 – Exemplos de Ferramentas CASE.

Tabela 5 – Fases do Desenvolvimento de um DW.

Tabela 6 - Indicadores e Dimensões da Área de Negócios: Vendas.

Tabela 7 - Indicadores e Dimensões da Área de Negócios: Finanças.

RESUMO

Atualmente, na maioria das organizações, são encontrados sistemas computacionais destinados a auxiliar e controlar as mais diversas atividades, gerando uma quantidade assustadora de dados.

Pode-se afirmar que bancos de dados e os sistemas que os manipulam se tornaram componentes essenciais no cotidiano da sociedade moderna. Contudo, analisar e obter informações úteis à administração e condução do negócio a partir destas bases de dados constitui-se em um dos grandes desafios atuais.

Verifica-se como solução alternativa ao problema acima citado a utilização de armazéns de dados focados sobre as necessidades do negócio, também conhecidos como ambientes de *data warehouse*. Porém, a implementação destes ambientes, devido às restrições existentes, ainda é considerada, em muitos casos, utópica.

A presente dissertação estuda como alternativa a utilização de tecnologia CASE para auxiliar e disciplinar o processo de construção de ambientes de *data warehouse*. Apresenta conceitos relacionados ao desenvolvimento de sistemas de apoio a decisão e metodologias existentes. Verifica, ainda, através de estudo de caso, quais os passos para a aplicação prática da tecnologia.

ABSTRACT

Nowadays, in most organizations, there are computer systems designed to aid and control the several activities, generating an enormous amount of data.

It can be affirmed that databases and the systems that manipulate them became essential components in modern society. However, to analyze and to obtain useful information to manage and conduct the business from these databases is one of the great current challenges.

As an alternative solution to the problem mentioned above, we observe the use of data banks focused on the business needs, also known as "*data warehouse*". Even so, the construction of these environments it is still considered, in many cases, utopian, due to the restrictions described in this work.

The CASE technology is studied in this work to aid and to discipline the process of data warehouse construction. It presents concepts related to the development of support decision systems, existing methodologies, and studies which steps for practical application of the technology.

1 INTRODUÇÃO

Este trabalho procura estudar a aplicação da tecnologia CASE na construção de ambientes de *Data Warehouse* e analisar através de estudo de caso os passos necessários à estruturação de um sistema de apoio à decisão.

1.1 Origem

A necessidade de controlar de forma racional o grandioso volume de informações trabalhadas hoje pelo homem constitui-se em um grande desafio; segundo (WURMAN, 1991), uma edição do *The New York Times* em um dia da semana, contém mais informações do que um mortal comum poderia receber durante toda sua vida na Inglaterra no século XVII. E é com o objetivo de controlar, organizar e processar estas informações, que são construídos computadores e sistemas cada vez mais avançados.

A grande maioria dos sistemas utiliza bancos de dados, que se tornaram componentes essenciais no cotidiano da sociedade moderna, sendo difícil imaginar nosso dia-a-dia sem interação com aplicações que manipulam esses bancos. O grande desafio está em interagir e extrair conclusões sobre eles.

Muitas empresas criaram departamentos de tecnologia de informação como área responsável por sistemas, métodos e tecnologias utilizadas,

buscando aumentar seu poder competitivo e melhorar a qualidade da informação.

Atualmente, na maioria das organizações podem ser encontradas estruturas de tecnologia de informação, e muitas destas têm investido, nos últimos anos, em sistemas de gestão, o que desencadeou uma avalanche tecnológica, originada inicialmente pelos ERPs (*Enterprise Resource Planning*), que conseguiram integrar os dados dos diversos departamentos da empresa e reduzir custos operacionais (BISPO,1998).

Com eles vieram de carona uma série de soluções que se tornaram o grande diferencial das empresas, como, entre outras tecnologias, os CRMs (*Customer Relationship Management*), que representam soluções integradas para acompanhamento de clientes e os SADs (Sistemas de Apoio a Decisão), que, através de tecnologias de banco de dados e respectivas ferramentas, fornecem diversas informações permitindo, por exemplo, simulações e visualização de cenários futuros (REZENDE,1999).

Atualmente, o principal foco da área de tecnologia da informação está em trabalhar os sistemas para extrair da melhor forma possível informações que possam facilitar e apoiar decisões. Contudo, os sistemas atuais não foram projetados para suportar a extração desse tipo de informações; segundo (INMON,1997) a arquitetura de desenvolvimento espontâneo, utilizada hoje pela maioria das organizações, simplesmente não basta para atender as necessidades do futuro. O que se faz necessário, é uma mudança de arquitetura, sendo sugerido o ambiente projetado de *data warehouse*.

Pode-se afirmar que *data warehouse*, ou simplesmente DW, é resumidamente, um banco de dados contendo registros extraídos do ambiente operacional ou ambiente de produção da empresa, sendo que estes dados foram selecionados, depurados e otimizados para o processamento de consultas, levando em consideração informações essenciais do ramo de atividades da organização. Também fazem parte da tecnologia DW os programas construídos para a extração de dados do ambiente operacional e os sistemas que fornecem estes dados aos seus usuários (BISPO,1998).

DW consiste na união de tecnologias capazes de criar grandes *containers* de dados estruturados, organizados por assunto, detalhados, não voláteis, históricos, integrados e focados sobre os propósitos da empresa (INMON, 1997).

Segundo (HACKATHORN, 1997), o objetivo de um *data warehouse* é fornecer uma "imagem única da realidade do negócio", proporcionando uma visão dos pontos relevantes aos administradores e estrategistas da organização, bem como a evolução destes pontos nas escalas desejadas.

O DW, com o auxílio de métodos, técnicas e ferramentas, como tecnologia de *Data Mining*, permite a extração de informações capazes de apoiar decisões, com o que os administradores têm liberdade para interagir com a informação montando seus relatórios, projeções, pesquisas e avaliações da forma que lhes é conveniente, encontrando limites apenas na sua imaginação e conhecimento.

Existem exemplos de implantação e utilização de ambientes de DW; entretanto, dificuldades como altos custos, falta de pessoal capacitado, deficiência nas ferramentas e metodologias, entre outros fatores, tornam arriscadas as tentativas de aplicação. Na década de 90, muitos projetos de *data warehouse* fracassaram.

Como as organizações necessitam justamente do resultado objetivo da tecnologia de DW, considera-se válido o estudo de alternativas capazes de minimizar os desafios da sua concepção (¹).

Uma destas alternativas é utilizar ferramentas de auxílio ao desenvolvimento, também conhecidas como ferramentas CASE, que tornem o trabalho de construção do DW mais fácil e intuitivo.

Segundo (REZENDE, 1999) CASE (*Computer Aided/Assisted Software Engineering*), sigla traduzida para Engenharia/Análise de sistemas apoiada/assistida por computador, é uma ferramenta ou conjunto de técnicas facilitadoras de desenvolvimento de software moderno.

¹ Investigar estas alternativas é um dos objetivos deste trabalho.

Existem vários tipos de ferramentas CASE, algumas são capazes de apoiar certas fases do ciclo de vida do software, absorvendo informações diversas referentes ao software, facilitando a criação e manutenção deste. Normalmente o desenvolvimento é estruturado em uma base de conhecimento, também conhecida como repositório, descrita pelo analista na ferramenta. A partir da base de conhecimento podem ser gerados os programas e estruturas de bancos de dados necessários ao novo sistema, bem como as novas instalações e reorganizações decorrentes de eventuais manutenções. Também é possível levantar estatísticas e formar pontos de auditoria ao longo do processo.

Este projeto busca unir estas tecnologias (CASE e DW), possibilitando a construção facilitada e disciplinada de ambientes de *data warehouse* através do uso dos princípios de engenharia e qualidade de software e ferramentas CASE, gerando ainda um modelo de implementação para auxiliar empresas que pretendam aumentar seu poder de decisão com a utilização de ambientes DW.

1.2 Justificativa

Possuir um grande volume de informações não significa necessariamente que a organização tenha um poder estratégico ou de decisão equivalente, a qualidade, a organização e a inteligência na manipulação e no acesso a estas informações é o que faz diferença.

Com o amadurecimento e propagação dos sistemas computacionais encontram-se hoje, na maioria das organizações, sistemas informatizados controlando os mais diversos departamentos. Contudo a extração de informações de caráter estratégico destes sistemas constitui um dos grandes desafios impostos aos profissionais de Tecnologia de Informação.

Verifica-se como opção para levantar este tipo de informação a adoção da tecnologia de *data warehouse*, que ainda possui limitações como,

entre outras, alto custo, falta de profissionais capacitados e deficiência nas ferramentas, validando esforços em busca de métodos capazes de possibilitar sua implantação, de disciplinar o processo de construção, diminuindo os altos custos financeiros e humanos necessários atualmente.

1.3 Objetivos

Os objetivos do presente trabalho são:

Objetivo geral: Investigar conceitos relacionados à construção de ambiente de *data warehouse* utilizando tecnologia CASE;

Objetivos específicos:

- Levantar conceitos sobre:
 - Business Intelligence;
 - *Data Warehouse*;
 - CASE;
- Investigar tecnologias disponíveis em CASE;
- Verificar os métodos tradicionais de criação de ambiente de *data warehouse*;
- Discutir métodos alternativos de criação;
- Desenvolver protótipo de um ambiente de *data warehouse* utilizando CASE;
- Verificar a validade de utilização de ferramentas CASE para a construção de ambientes de DW.

1.4 Organização

Para melhor entendimento e apresentação, o presente trabalho está organizado nas seguintes partes:

Capítulo 1 – **Introdução**: Neste tópico apresenta-se a origem, os objetivos, organização e a metodologia do presente trabalho.

Capítulo 2 – **Business Intelligence**: Contextualização: conceitos básicos das tecnologias envolvidas e relativas à inteligência de negócio.

Capítulo 3 – **CASE**: Definição de ferramentas CASE, características, ambientes e temas relacionados a ferramentas de apoio ao desenvolvimento.

Capítulo 4 – **Criação de ambiente DW utilizando CASE**: Descreve-se neste capítulo os passos necessários à criação auxiliada por CASE de ambiente de DW.

Capítulo 5 – **Estudo de Caso**: Aplica-se a metodologia e os passos descritos no capítulo 4 a um caso real, verificando assim sua validade.

Capítulo 6 – **Conclusões**: Finalmente apresentam-se as conclusões e algumas sugestões para futuros trabalhos que possam aperfeiçoar e complementar o estudo do tema proposto.

1.5 Metodologia

O presente trabalho consiste no estudo sobre a construção de ambiente de *data warehouse* e temas afins fundamentado através de levantamento bibliográfico e estudo de caso.

O estudo de caso será realizado sobre uma base teste e a pesquisa bibliográfica será baseada em fontes escritas como livros, revistas, periódicos, jornais, artigos científicos, bem como em materiais disponibilizados em provedores na Internet.

2 BUSINESS INTELLIGENCE

As decisões necessariamente devem ser apoiadas em informações, segundo (GATES, 1999) a informação é algo que alguém deseja obter e está disposto a pagar por ela; não é tangível e nem mensurável, mas é um produto valioso no mundo contemporâneo, porque proporciona poder. A busca de informações é atualmente o alvo da maior parte dos governos e das empresas. É através da informação que é possível ter um melhor suporte ao processo decisório. É função das ferramentas, que vão dar o suporte a este processo, levantar as informações necessárias de uma forma rápida, confiável e exibi-las de uma forma facilmente compreensível.

A evolução ocorrida nos últimos anos, tanto com hardware como software, permitiu a criação de sistemas complexos e o controle dos mais diversos setores, gerando um excesso de informação na maior parte das organizações, dando origem ao fenômeno conhecido como "paradoxo da informação". Este fenômeno refere-se a impossibilidade de extrair certos tipos de informação de forma não automatizada nos atuais sistemas de informação que são muito eficientes na coleta e armazenagem de grandes volumes de dados, mas ineficientes para extração de informações de apoio a decisão.

Business Intelligence surgiu da necessidade de informação e das dificuldades existentes para sua obtenção na forma desejada pelos administradores. Segundo (CÔRTEZ, 2001) *Business Intelligence* (BI) é um conjunto de conceitos e metodologias que, fazendo uso de acontecimentos (fatos) e sistemas baseados nos mesmos, apóia a tomada de decisões em

negócios e tem como principais características: extrair e integrar dados de múltiplas fontes, fazer uso de experiência, analisar dados contextualizados, trabalhar com hipóteses, procurar relações de causa / efeito e transformar os registros obtidos em informação útil para o conhecimento empresarial.

São ferramentas de BI: Planilhas Eletrônicas, Geradores de queries e de relatórios, DSS (*Decision Support System*), EIS (*Executive Information System*), DW (*Data Warehouse*), *Data Mart*, OLAP (*On line analytical processing*) e DM (*Data Mining*).

De acordo com Glen Hillam, diretor para América Latina da Seagate Software, o BI é como uma pirâmide, composta por várias camadas de ferramentas. Na base estão os softwares para elaboração de relatórios. Nas faixas intermediárias encontram-se as ferramentas de OLAP, que agilizam os processos de consulta e análises, fornecendo os dados no formato multidimensional; as soluções EIS – sistemas de informações previamente preparadas para os executivos de alto nível; e as DSS que fornecem informações para agilizar a tomada de decisões. E no topo ficam as DM – técnicas para correlacionar fatos a partir de massa de dados, identificando tendências.

O ponto de partida para a aplicação das ferramentas de *Business Intelligence* são os bancos de dados das organizações, uma vez que os SGBDs (Sistemas Gerenciadores de Bancos de Dados) se tornaram componentes indispensáveis em quaisquer processos que utilizem registros de informações e que estes armazenam uma enorme quantidade de dados sobre a empresa, ainda que não totalmente relacionados (REZENDE, 1999).

Os SGBDs convencionais trabalham com processamento transacional e normalmente mantém a última posição dos dados, ou seja, sempre a versão atualizada. Este tipo de processo, o mais comum nos sistemas atuais, é conhecido como OLTP (*On line Transactional Processing*) e os sistemas que tratam deste tipo de processo em *Business Intelligence* são denominados sistemas de nível operacional ou apenas sistemas operacionais (INMON, 1997).

A partir dos sistemas de nível operacional é proposta a construção de bases de dados voltadas à análise e focadas sobre as necessidades estratégicas de informação do negócio. Estas bases de dados são conhecidas como armazéns de dados ou como ficaram popularizados: ambientes de "*Data Warehouse*".

O *Data Warehouse* é visto como solução para o problema do armazenamento e acesso aos dados para propósitos gerenciais e possui grandes diferenças estruturais em relação aos bancos de dados normais, principalmente por ser planejado e destinado para o processamento analítico de informações. Tal processamento é conhecido como OLAP (*On Line Analytical Processing*).

As definições de OLAP possibilitaram a criação de uma série de ferramentas para consulta do DW que permitem a manipulação das informações pelos gerentes, administradores e analistas, limitados apenas pela imaginação e conhecimento destes. As ferramentas OLAP são aplicações que os usuários finais têm acesso para extraírem os dados das bases, com os quais geram relatórios capazes de responder as suas questões gerenciais (BISPO, 1998). Elas surgiram juntamente com os sistemas de apoio a decisão para a extração e análise dos dados contidos no *Data Warehouse*.

Com o *Data Warehouse* surgiu ainda a tecnologia de mineração de dados (*Data Mining*), utilizada para pesquisa de informações não triviais sobre o negócio, onde a partir de suposições e pesquisas encontram-se informações obscuras ou escondidas nos bancos de dados e para identificação de tendências.

O termo *Data Mining* refere-se ao processo de extrair informação válida de máxima abrangência, ainda não conhecida, a partir de grandes bases de dados, com o objetivo de utilizá-las na tomada de decisões. Permite aos usuários explorar e inferir informação útil a partir dos dados, descobrindo relacionamentos escondidos no banco de dados (CÔRTEZ, 2001).

Data Mining ou simplesmente DM também pode ser considerado uma forma de descobrimento de conhecimento em bancos de dados (KDD -

Knowledge Discovery in Databases), área de pesquisa que envolve Inteligência Artificial e Banco de Dados.

2.1 Data Warehouse

Segundo (INMON, 1997), 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.

Para (HARJINDER, 1996), *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.

Segundo (BARQUINI, 1996), *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.

A utilização de ambientes de *data warehouse* esta diretamente relacionada à tomada de decisões nas organizações, segundo (BISPO, 1998) a palavra decisão é formada pelo prefixo 'de' (prefixo latino aqui com o significado de parar, extrair, interromper) que se antepõe à palavra 'caedere' (que significa cindir, cortar). Tomada ao pé da letra, a palavra decisão significa "parar de cortar" ou "deixar fluir". Uma decisão precisa ser tomada sempre que se esta diante de um problema que possui mais que uma alternativa para a sua solução.

Nas organizações as decisões normalmente são tomadas com base na experiência dos administradores, quando poderiam também ser baseadas em fatos históricos que foram armazenados pelos diversos sistemas de informação utilizados pelas empresas. Segundo (CÔRTEZ, 2001) o DW permite a geração de dados integrados e históricos auxiliando os executivos

das empresas a decidirem embasados em fatos e não em intuições ou especulações, o que reduz a probabilidade de erros aumentando a velocidade na hora da decisão. Cerca de 88% dos diretores admitem que dedicam quase 75% do tempo às tomadas de decisão apoiadas em análises subjetivas (Aspect International Consulting, 1997), menosprezando o fato de que por volta de 100% destes tem acesso a computadores.

Um *Data Warehouse* é construído para que os dados informacionais possam ser armazenados e acessados de forma não limitadas por tabelas e linhas estritamente relacionais. A função do *Data Warehouse* (DW) é tornar as informações corporativas acessíveis para o seu entendimento, gerenciamento e uso. Como o DW está separado dos bancos de dados operacionais, as consultas dos usuários não impactam nestes sistemas, que ficam resguardados de alterações indevidas ou perdas de dados.

Um DW 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 que normalmente não existem em seus sistemas operacionais e, com isso, tomar decisões com base em fatos, ao invés de especulações ou intuições (BISPO, 1998).

Na tecnologia DW são utilizados novos métodos de estruturação de dados e novas tecnologias, tanto para armazenamento, como para recuperação de informações. A necessidade destes novos métodos e tecnologias surgiu da constatação, primeiro de que existe uma carência de informação não atendida pelos aplicativos comerciais convencionais, que atuam a nível operacional do negócio, e segundo, pelo fato de que a tecnologia de armazenamento de dados utilizada nestes aplicativos não atende às necessidades detectadas. Graças aos avanços nos bancos de dados relacionais, no processamento paralelo e na tecnologia distribuída, é possível a elaboração de ambientes DW.

Como as empresas demoram vários anos para gerar e armazenar um volume considerável de informações, é normal que estes dados estejam espalhados por diversos locais e que tenham sido gerados por sistemas desenvolvidos em diferentes ambientes e linguagens. Um dos desafios da implantação de um DW é justamente a integração destes dados, eliminando as redundâncias e identificando informações iguais que possam estar representadas sob formatos diferentes em sistemas distintos.

Por definição um projeto de DW tem por objetivo organizar os dados operacionais de uma empresa, em um local onde o ambiente operacional não seja afetado, da melhor forma possível para que possam ser aplicadas técnicas de análise e extração de dados. Além disso, considera que os analistas precisam de informações para montarem seus planos táticos e estratégicos. Por exemplo, um sistema operacional, para fins de funcionamento, não precisa da taxa do dólar do ano passado, porém o analítico pode querer avaliar a evolução do dólar frente o volume de compras de seus produtos e portanto precisará (CÔRTEZ, 2001).

O DW consiste em uma evolução sobre o conceito tradicional de bancos de dados. Conforme (BATINI, 1986), "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". (BATINI, 1986) 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 DW 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".

Com base nos autores (INMON, 1996), (BARQUINI, 1996), KIMBALL (1996) e (ONEIL, 1997) algumas das principais diferenças entre bancos de

dados operacionais e ambientes de *data warehouse* são relacionadas na tabela 1.

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 – terrabytes
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*.

Na estrutura do *data warehouse* encontra-se em um extremo os sistemas operacionais, que são as aplicações normais projetadas para atender ao dia a dia do negócio da empresa, garantindo sua operação, também chamados de sistemas de produção, mais aos dados externos, que servem para o acompanhamento e medição do negócio e do mercado. E em outro extremo encontram-se as ferramentas de consulta ao *data warehouse* como ferramentas OLAP, ferramentas de consulta/relatórios e aplicativos avançados, como é demonstrado na figura 1.

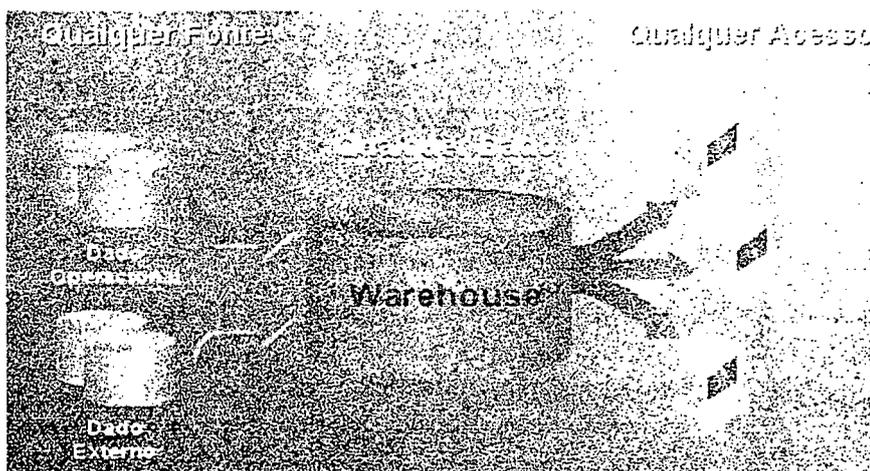


Figura 1 – Estrutura do Data Warehouse

O DW é, na verdade, uma espécie de ponte capaz de fazer a integração entre as aplicações de nível operacional e os gerentes, figurando não apenas como um grande banco de dados, mas como uma fonte poderosa de informações sobre o negócio.

Uma questão muito importante no planejamento e desenvolvimento de um *data warehouse* é a definição dos diversos modelos de dados que darão suporte e orientação ao trabalho do ambiente do DW. (KIMBALL, 1996) e (INMON, 1996) descrevem metodologias de modelagem de dados para *Data Warehouse*, ambas apresentam três níveis de modelagem para os dados: alto nível (empresarial); intermediário (dimensional); e baixo nível (físico).

Com relação aos modelos de dados do DW, deve-se definir algumas estratégias que orientarão o desenvolvimento do DW e pode-se efetivamente

utilizar as ferramentas de DW, as quais podem ser divididas em dois grupos: ferramentas de povoação (aquelas que extraem dados dos sistemas operacionais da empresa) e ferramentas de consulta (aquelas que realizam consultas sobre o DW).

O estudo da tecnologia de *data warehouse* pode auxiliar os administradores a descobrir formas alternativas de competir em uma economia globalizada, fortalecendo a condução do negócio e possibilitando o investimento em produtos ou serviços mais adequados às necessidades e tendências do mercado, tornando-se mais competitiva.

Ainda não são encontradas metodologias formais para construção de ambientes de *data warehouse*, sendo que a organização deve escolher a que mais se adapta às suas características e expectativas, 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.

2.1.1 Características do Data Warehouse

Na definição apresentada por (INMON, 1997), um *data warehouse* deve ser:

Orientado por assunto:

Ao contrário dos sistemas transacionais que estão voltados aos processos e aplicações específicas, nos ambientes de *data warehouse* toda a modelagem é voltada aos assuntos principais da empresa, levando em consideração um conjunto de informações relativas a determinadas áreas estratégicas da organização. Como exemplo pode-se citar em uma revenda de automóveis: vendas, serviços, consórcios, entre outras como áreas de negócio.

Integrado:

A integração é considerada ponto chave na construção de ambiente de DW, pois é responsável pela padronização e possibilita a criação de uma representação única para os dados de todos os sistemas. Grande parte dos esforços de desenvolvimento de um DW é empregada na análise dos sistemas, que normalmente se encontram em vários formatos e padrões. Vários autores citam o exemplo do atributo sexo encontrado em uma aplicação com os valores M ou F, em outra aplicação com os valores H ou M e em outra aplicação com os valores possíveis 0 ou 1, ambos para representar Masculino ou Feminino, fazendo-se necessária a padronização na integração destas aplicações para valores mais adequados, como é demonstrado na figura 2:

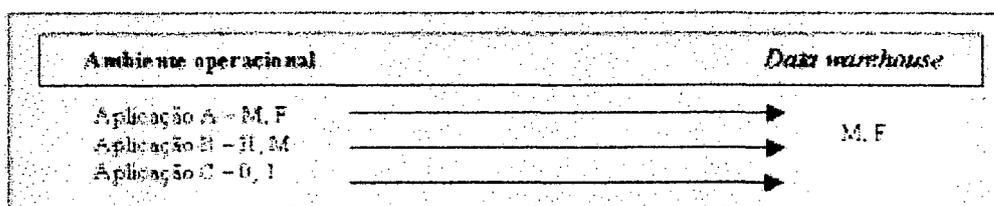


Figura 2 – Necessidade de padronização na integração de aplicações.

Variável em relação ao tempo:

Um DW geralmente mantém histórico dos dados durante um período de tempo muito superior ao dos sistemas transacionais, que possuem a finalidade de fornecer sempre a posição atual dos dados, por outro lado no DW é muito importante demonstrar a evolução ou comportamento dos indicadores do negócio na dimensão do tempo. A dimensão tempo estará sempre associada a qualquer fato. Os registros existentes no DW também são comparados a fotografias dos dados dos sistemas em determinado momento, denominados *snapshots*, e obrigatoriamente deverá conter, nestes registros, uma chave de tempo para expressar a data em que os dados foram extraídos.

Não volátil:

As únicas operações possíveis nos ambientes de *data warehouse* são a carga inicial e a consulta dos *front-ends* aos dados. A maneira como os dados são carregados e tratados no DW, normalmente com operações de

inserção e seleção é muito diferente dos sistemas transacionais que possuem vários controles e atualizações de seus registros. Por exemplo, num sistema de contabilidade pode-se fazer alterações nos registros. Já no DW, o que acontece é somente ler os dados na origem e gravá-los no destino.

Deve-se levar em consideração que nos procedimentos de carga os dados são trabalhados e normalmente passam por filtros e seleções antes de serem inseridos no DW, sendo que muitos dos registros nunca saem do ambiente transacional e outros são tão resumidos que não se encontram fora do DW. (INMON, 1996) afirma que do ponto de vista de integração, não são mais os mesmos dados do ambiente operacional, à luz destes fatores, a redundância de dados entre os dois ambientes raramente ocorre, resultando em menos de 1 por cento de duplicações.

São apresentadas ainda as seguintes características:

Localização:

Fisicamente os dados de um DW podem estar armazenados de três formas:

a) Centralizado: Modelo bastante utilizado onde o banco de dados forma um DW integrado, procurando maximizar o poder de processamento e agilizando a busca dos dados. Existe o inconveniente do investimento em hardware necessário para comportar a base muito volumosa, e a força de processamento elevada para atender satisfatoriamente as consultas simultâneas de muitos usuários.

b) Distribuído em *Data Marts*: Considerada uma alternativa viável para quem precisa de performance, pois não sobrecarrega os servidores atendendo as consultas satisfatoriamente uma vez que os dados são armazenados por áreas de interesse.

c) Armazenamento por níveis de detalhe: Pode-se armazenar dados altamente resumidos num servidor, dados resumidos noutra nível de detalhe intermediário no segundo servidor e os dados mais detalhados (atômicos), num terceiro servidor. Os servidores da primeira camada podem ser otimizados para

suportar um grande número de acessos e um baixo volume de dados, enquanto alguns servidores nas outras camadas podem ser adequados para processar grandes volumes de dados, mas baixo número de acesso.

Credibilidade dos Dados:

Sem dúvidas a credibilidade dos dados é de suma importância para qualquer projeto, não apenas em DW, mas em qualquer sistema. Uma vez que se faz necessária a extração de dados para suportar decisões estratégicas, distorções ou dados não dignos podem resultar em relatórios inúteis, sem importância e podem trazer altos riscos para a avaliação e condução do negócio. Por exemplo, algo aparentemente simples, como um CEP errado, pode não ter nenhum impacto em uma transação de compra e venda, mas pode influir nas informações referentes a cobertura geográfica.

"Não é apenas a escolha da ferramenta certa que influi na qualidade dos dados", afirma Richard Rist, vice-presidente Data Warehousing Institute. Segundo ele, conjuntos de coleções de dados, processos de entrada, metadados e informações sobre a origem dos dados, são importantíssimos. Outras questões como a manutenção e atualização dos dados e as diferenças entre dados para bancos transacionais e para uso em *data warehousing* também são cruciais para o sucesso dos projetos.

A Tabela 2 descreve um conjunto de características, normalmente utilizadas para verificar a qualidade dos dados, e indica algumas das maneiras de medir o nível de qualidade dos mesmos do DW. Nem todas as características precisam necessariamente ser levantadas, deve-se escolher as que representam maior fator de risco para o ambiente proposto e trabalhar em cima destas características.

Características da qualidade dos dados	Descrição	Exemplos de medida
Precisão	Grau de informações que estão corretas	Percentual de correção
Abrangência	Grau de dados requisitados e atendidos	Percentual atendimentos
Consistência	Consistência dos dados/Liberdade de contradição	Percentual de condições satisfeitas

Tabela 2 – Características da qualidade dos dados

Granularidade:

O volume de dados é afetado diretamente pela granularidade, que é o nível de detalhe ou de resumo dos dados contidos em um DW. Quanto maior for o nível de detalhe, menor será a granularidade. Em um nível alto o espaço em disco e o número de índices necessários é bem menor, contudo existe uma correspondente diminuição da possibilidade de utilização dos dados para atender a consultas com maior nível de detalhe, conforme apresentado na figura 3.

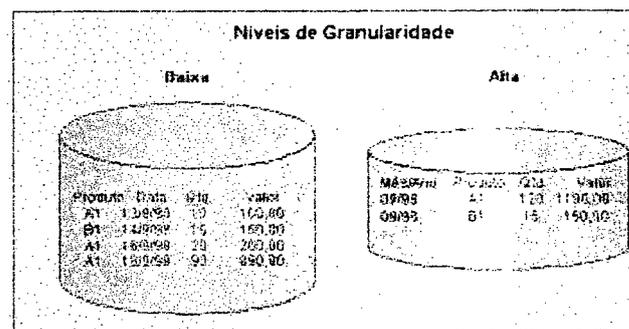


Figura 3 – Nível de granularidade.

Se for implementado um nível de granularidade muito baixo, é possível responder a praticamente todas as consultas, em compensação uma grande quantidade de recursos computacionais se faz necessária. O balanceamento do nível de granularidade é um dos aspectos mais críticos no planejamento de um DW, 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 DW, faz sentido pensar em dois ou mais níveis de granularidade, na parte detalhada dos dados. A adoção de mais de um nível de granularidade pode ser considerada como ideal para quase todas as empresas.

2.1.2 Dados Primitivos e Derivados

No ambiente projetado de *data warehouse* existe basicamente duas espécies de dados: dados primitivos e dados derivados.

Segundo (INMON, 1997) os dados primitivos são dados detalhados utilizados na condução das operações cotidianas da empresa e dados derivados são dados resumidos ou calculados de forma a atender as necessidades da gerência da empresa.

Existem grandes diferenças entre os dados primitivos e os derivados, como pode ser observado na tabela 3, tornando necessário um tratamento diferenciado.

Dados Operacionais/ Primitivos	Dados Derivados de DW
Orientados a aplicação	Orientado a assunto
Detalhados	Resumido ou refinado
Preciso no momento do acesso	Representa valores over time, snapshots
Serve a comunidade clerical	Serve a comunidade gerencial
Pode ser atualizado	Não é atualizado
Executa repetitivamente	Executa Heuristicamente
Requer processamento entendido a priori	Não requer processamento entendido
Compatível com ciclo de vida	Completamente diferente do ciclo de vida
Performance sensitive	Performance relaxada
Acesso de uma unidade a cada vez	Acesso de um conjunto a cada vez
Orientado a transações	Orientado a análise
Controle de atualização crítico	Controle de atualização não é crítico
Disponibilidade alta	Disponibilidade relaxada
Gerenciado em sua totalidade	Gerenciamento por subconjuntos
Não redundante	Redundante
Estrutura estática, conteúdo variável	Estrutura flexível
Poucos dados usados em um processo	Muitos dados usados em um processo
Suporta operações do dia-a-dia	Suporte necessidades gerenciais
Probabilidade alta de acesso	Probabilidade baixa de acesso

Tabela 3 – Dados Primitivos e Derivados

Resumidamente, faz-se uso dos dados primitivos que são aqueles que estão no ambiente operacional da empresa para originar os dados derivados, que serão utilizados para o processamento analítico no ambiente do DW.

É importante ressaltar que as consultas/relatórios baseadas nos dados primitivos devem estar presentes nos sistemas operacionais da

empresa, enquanto que no ambiente de DW estarão as informações destinadas a consultas/relatórios analíticos ou gerenciais.

2.1.3 Níveis do *Data Warehouse*

Os níveis da arquitetura do *data warehouse* em um ambiente projetado apresentam as seguintes características:

- Nível operacional: detalhado, cotidiano, valores atuais, alta probabilidade de acesso, baseado em aplicações;
- Nível atômico / *data warehouse*: mais granular, variável no tempo, integrado, baseado em negócios, algum nível de resumo;
- Nível departamental: paroquial, alguns dados derivados, alguns primitivos, típico de departamentos como contabilidade, engenharia, produção, marketing, etc;
- Nível individual: temporário, *ad-hoc*, heurístico, não-repetitivo, baseado em PC's ou estações de trabalho.

2.1.4 Dados Externos

Para responder a perguntas como "Qual a situação de nossa empresa no mercado x?", necessita-se a entrada de dados referentes à medição do negócio, como pesquisas de mercado, dados dos concorrentes, compradores, enfim, o máximo de informações que não estão presentes nos sistemas de nível operacional e são utilizados pela gerência.

Também se faz necessária a obtenção de dados relativos à padronização do DW, como por exemplo cotação do dólar, ouro e outros indicadores necessários as comparações.

Na maioria dos casos é preciso construir um sistema para a coleta destas informações. Isto facilita, padroniza e organiza a entrada de informações.

2.1.5 Indicadores x Dimensões

Um dos principais desafios do analista de negócios é levantar quais os indicadores e as dimensões que serão implementadas no DW, e uma vez definidos, obter-se-á a estrutura dos dados no ambiente de DW.

Os indicadores dizem respeito aos acumuladores, ou seja, os dados que são necessários na comparação ou medição do negócio: "O que se quer comparar". Como exemplos de indicadores pode-se citar:

- Volume total de compras;
- Custo médio;
- Comissões pagas;
- Impostos recolhidos.

As dimensões são as escalas de comparação: "Quando ou como comparar". Como exemplos de dimensões pode-se citar:

- Ano;
- Trimestre;
- Semana;
- Mês;
- Ramo de Atividades;
- Produto.

A modelagem dimensional surgiu com a tecnologia de DW, como aperfeiçoamento ao conceito de banco de dados possibilitando a observação da informação no formato de um cubo, contendo duas, três ou mais dimensões, que representam as visões de negócio.

A figura 4 representa a modelagem de um banco de dados com três dimensões: Produto, tempo e cliente, sendo que em cada uma das células estarão armazenados os indicadores, possibilitando com o uso da tecnologia OLAP visualizar a evolução da organização.

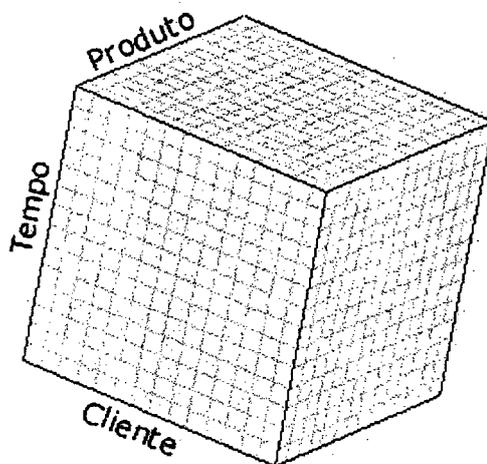


Figura 4 - Cubo de Informações

Com as informações já modeladas o usuário irá fazer suas solicitações que são trabalhadas da seguinte forma:

Procura pelas ocorrências do objeto pedido pelo usuário na respectiva coluna. Depois de toda a varredura é apresentado ao usuário o seu pedido.

Após a análise acima descrita, as informações obtidas serão cruzadas com as de outras dimensões para uma consulta mais completa.

2.1.6 Procedimentos de Carga

Os procedimentos de carga são os programas responsáveis pela alimentação dos dados do DW. Estes procedimentos valem as bases de dados operacionais nos momentos estipulados, executando as funções necessárias ao levantando dos indicadores e construção dos registros no DW.

Dependo dos estudos dos analistas do negócio, certos procedimentos de carga podem ser executados diariamente, semanalmente, anualmente ou na escala definida como dimensão para análise dos indicadores.

Pode-se afirmar que a construção dos procedimentos de carga é uma das etapas mais trabalhosas na construção de um DW, uma vez que nestes procedimentos estão todas manipulações e filtros para a leitura dos sistemas operacionais e geração dos registros do DW.

2.1.7 Modelo de Dados Segundo R.Kimball

O processo de criação do modelo de dados é determinante para o sucesso ou fracasso de projetos de ambientes de DW, uma vez que permite a compreensão dos dados, das regras de negócio e de suas fontes.

(KIMBALL, 1996) sugere a criação de três modelos de dados que são:

- Modelo das regras de negócio ou empresarial;
- Modelo dimensional;
- Modelo físico.

2.1.7.1 Modelo Empresarial

Na fase inicial é realizada a análise do modelo de dados, focalizada sobre as estruturas de informação como atributos e relações entre eles, permitindo a criação de um modelo entidade-relacionamento normalizado das regras de negócio.

A construção deste modelo é facilitada com a utilização de perguntas capazes de auxiliar na descoberta de dados relevantes para o DW e os dados que irão resultar em estruturas dimensionais, por exemplo, se o DW que estiver

sendo construído for sobre vendas em *marketing*, o tipo de perguntas que se deve procurar responder são:

- Quem compra o produto? (os clientes e sua estrutura);
- Quem vende o produto? (organização das vendas, os canais de distribuição e assim por diante);
- O que é vendido? (estrutura de produto);
- Quando é vendido? (estrutura de tempo);
- Como é vendido? (contrato, telefone e assim por diante);
- Quais são as características da venda? (promoção, venda normal, etc.).

É bastante comum iniciar um processo questionando os usuários quanto o que eles querem ver. Contudo muitas vezes serão elaboradas perguntas cujas respostas podem ter respostas de difícil implementação, previsão ou armazenamento.

O segundo passo para o desenvolvimento do modelo empresarial é a normalização do modelo, uma vez definidas todas as entidades e relações é possível normalizar o modelo. Pode-se utilizar a 3ª Forma Normal que é alcançada quando os atributos dependem exclusivamente da sua chave.

O terceiro passo, para desenvolver um modelo entidade-relacionamento normalizado das regras de negócio, é a definição das restrições ou regras de integridade. As restrições de integridade ajudam a garantir a consistência dos resultados das consultas.

2.1.7.2 Modelo Dimensional

O modelo dimensional consiste nos ângulos em que a informação será analisada, isto é, as perspectivas de evolução dos indicadores escolhidos. Por exemplo, um gerente de um supermercado que pretende melhorar suas vendas ou saber se suas promoções estão trazendo bons resultados precisa examinar os dados sobre as vendas disponíveis na empresa. Uma avaliação deste tipo

requer uma visão histórica do volume de vendas sob múltiplas perspectivas, como volume de vendas por produto, volume de vendas por marca, volume de vendas por filial, volume de vendas por período de tempo, etc.

As dimensões são as perspectivas envolvidas, no caso do supermercado: produto, marca, filial e mês. Eles normalmente correspondem a campos não numéricos em um banco de dados. As medidas, como vendas ou despesas com promoção, correspondem geralmente a campos numéricos em um banco de dados.

As agregações feitas em termos de dimensões e coordenadas para armazenar as medidas dão origem ao termo modelagem multidimensional, que corresponde a uma técnica de projeto lógico bastante usada para DW, com o objetivo é apresentar o dado em uma arquitetura padrão e intuitiva, que permita acessos de alta performance (KIMBALL, 1996). Cada eixo no espaço multidimensional corresponde a um campo ou coluna de uma tabela relacional e cada ponto um valor correspondente à interseção desses campos ou colunas.

Os dados multidimensionais podem ser armazenados e representados em estruturas relacionais, para isso é necessário utilizar formas específicas de modelagem como o modelo "Estrela" e o modelo "Floco de Neve" descritos a seguir.

Modelo Estrela

Segundo (VALENTE, 1996), modelos de bases de dados relacionais apresentam tabelas com relacionamentos complexos e com múltiplas uniões circulares entre dois pontos do modelo. Para a maioria dos usuários que utilizam ferramentas para compor suas consultas é necessário que o acesso à base de dados seja simples o suficiente para facilitar o acesso direto os seus dados. Para acomodar as necessidades de todos os usuários e facilitar a atualização do DW o projetista deve criar um modelo que o usuário final possa facilmente entender em termos do negócio.

No modelo dimensional estrela existe uma tabela dominante no centro, chamada tabela de fatos, com múltiplas junções conectando-a a outras tabelas, sendo estas chamadas de tabelas de dimensão. Cada uma das tabelas secundárias possui apenas uma junção com a tabela central.

Conforme demonstrado na figura 5, o modelo Estrela tem a vantagem de ser simples e intuitivo, mas também faz uso de novos enfoques de indexação e união de tabelas. A tabela de fatos pode conter milhares ou milhões de valores e medidas do negócio da empresa, como transações de vendas ou compras. Cada uma das medidas é retomada mediante a interseção de todas as dimensões.

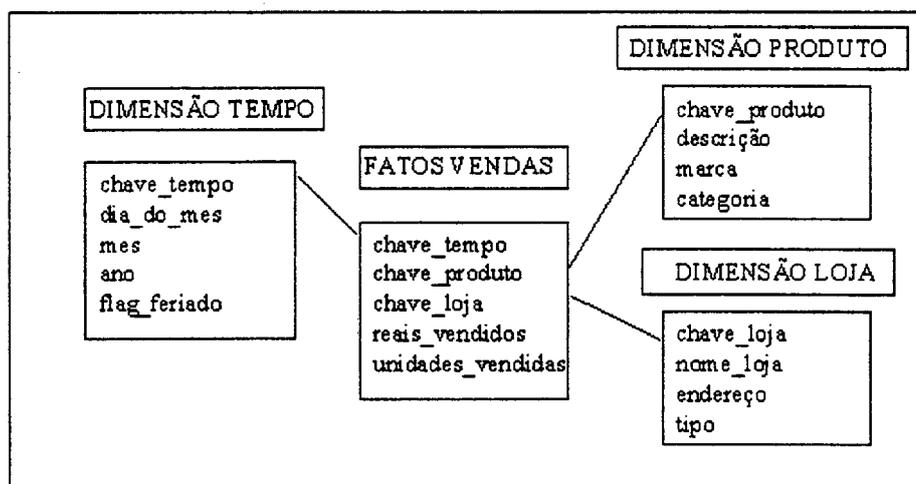


Figura 5 – Modelo dimensional do tipo Estrela.

As descrições textuais das dimensões do negócio são armazenadas nas tabelas de dimensão e ajudam a definir um componente da respectiva dimensão e servem como fonte para restrições em uma consulta ou como cabeçalhos de linha no conjunto de resposta do usuário.

Segundo (BARQUINI, 1996) as tabelas dimensões tendem a utilizar tipos caracteres ao invés de numéricos, de forma que suas linhas são muito mais longas, mas em pouca quantidade ocupando uma pequena percentagem de espaço em disco. As tabelas de fatos podem utilizar até 95% da área destinada ao DW.

Segundo (KIMBALL, 1996) a tabela de fatos na verdade representa os relacionamentos muitos-para-muitos entre as tabelas de dimensões, esta tem como chave primária uma chave composta de todas as chaves estrangeiras das tabelas de dimensão.

A tarefa de antecipar as consultas que serão realizadas com maior frequência é necessária na modelagem do DW, onde podem ser implementadas redundâncias seletivas e relacionamentos pré-estabelecidos para facilitar o acesso, objetivando um bom desempenho do modelo.

Modelo Floco de Neve

Uma variação do modelo estrela originou o modelo de dados *Snow Flake* (floco de neve), onde cada uma das “pontas da estrela” passa a ser o centro de outras estrelas, onde cada tabela de dimensão é normalizada “quebrando-se” a tabela original ao longo de hierarquias existentes em seus atributos.

É demonstrado um exemplo do modelo dimensional floco de neve na figura 6, onde a dimensão tempo inclui ano, que contém mês, que por sua vez possui dia do mês, sendo que cada um dos relacionamentos muitos para um geraria uma nova tabela.

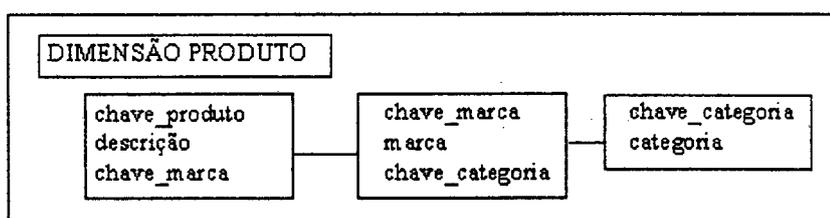


Figura 6 – Modelo dimensional do tipo Floco de Neve.

Segundo (KIMBALL, 1996) os projetistas devem resistir à tentação de transformar modelos Estrela em modelos Floco de Neve, devido ao impacto da complexidade deste tipo de estrutura sobre o usuário final, enquanto que o ganho em termos de espaço de armazenamento seria pouco relevante.

Segundo (KIMBALL, 1996) dentre as vantagens da utilização do modelo dimensional tipo estrela encontram-se:

- Modelo Estrela tem uma arquitetura padrão e previsível. As ferramentas de consulta e interfaces do usuário podem se valer disso para fazer suas interfaces mais amigáveis e fazer um processamento mais eficiente;
- Todas as dimensões do modelo são equivalentes, ou seja, podem ser vistas como pontos de entrada simétricos para a tabela de fatos. As interfaces do usuário são simétricas, as estratégias de consulta são simétricas, e o SQL gerado, baseado no modelo, é simétrico;
- O modelo dimensional é totalmente flexível para suportar a inclusão de novos elementos de dados, bem como mudanças que ocorram no projeto. Essa flexibilidade se expressa de várias formas, dentre as quais pode-se citar:
 - Todas as tabelas de fato e dimensão podem ser alteradas simplesmente acrescentando novas colunas a tabelas;
 - Nenhuma ferramenta de consulta ou relatório precisa ser alterada de forma a acomodar as mudanças;
 - Todas as aplicações que existiam antes das mudanças continuam rodando sem problemas;
 - Existe um conjunto de abordagens padrões para tratamento de situações comuns no mundo dos negócios. Cada uma destas tem um conjunto bem definido de alternativas que podem então ser especificamente programadas em geradores de relatórios, ferramentas de consulta e outras interfaces do usuário. Dentre estas situações pode citar:
 - Mudanças lentas das dimensões: ocorre quando uma determinada dimensão evolui de forma lenta e assíncrona;

- Produtos heterogêneos: quando um negócio, tal como um banco, precisa controlar diferentes linhas de negócio juntas, dentro de um conjunto comum de atributos e fatos, mas ao mesmo tempo esta precisa descrever e medir as linhas individuais de negócio usando medidas incompatíveis;
- Outra vantagem é o fato de um número cada vez maior de utilitários administrativos e processo de software serem capazes de gerenciar e usar agregados, que são de suma importância para a boa performance de respostas em um DW.

2.1.7.2.1 O Modelo E-R e o Modelo Estrela

A complexidade é uma das grandes diferenças entre o modelo E-R e o modelo Estrela, uma vez que por ser normalizado o E-R gera uma grande quantidade de tabelas conectadas entre si dificultando sua compreensão, o modelo multidimensional do tipo estrela, por sua vez, apresenta uma tabela central, a tabela de fatos, em uma estrutura mais simples, ligada a outras tabelas de dimensão figurando em um modo mais intuitivo e prático tanto para analistas como para os usuários.

(KIMBALL, 1995) afirma que no modelo dimensional, a escolha das dimensões e atributos de fatos é baseada na opinião do usuário, o que faz com que o modelo reflita as suas necessidades.

No processo de mapeamento do modelo E-R para o modelo Estrela pode-se constatar que:

- Os dados da empresa devem ser planejados visando sua integração, para que o DW possa ser construído de forma consistente e simplificada.

- Os modelos E-R representam todos os possíveis processos de negócio da empresa, enquanto cada modelo dimensional é focalizado em único assunto de interesse, sendo assim, é possível que de um modelo E-R surjam vários modelos dimensionais que normalmente estarão interligados.

Segundo (KIMBALL, 1997) os passos para o mapeamento do E-R para o modelo Estrela são:

1. Separar o diagrama E-R nos diferentes processos de negócio e modelar cada um separadamente;
2. Selecionar os relacionamentos muitos-para-muitos que contenham fatos aditivos e numéricos, no modelo E-R, para transformá-los em tabelas de fato;
3. Por fim, deve-se desnormalizar todas as tabelas remanescentes em tabelas com chaves simples, que se conectam diretamente as tabelas de fato. Estas tabelas se tomam as tabelas de dimensão.

2.1.7.3 Modelo Físico

A construção do modelo físico tem a finalidade de zelar pelo desempenho, pois, apesar de toda a evolução ocorrida, os computadores ainda têm restrições em velocidade de processamento e recursos de armazenamento.

O modelo físico também é dependente do SGBD e da configuração de hardware. As diretrizes encontradas na definição de (KIMBALL, 1996) procuram não focar nenhum SGBD ou hardware específico.

O processamento paralelo pode trazer bons resultados para as aplicações de DW. Baseia-se na tarefa de decompor uma única operação de consulta em fluxos paralelos com execução simultânea, desde que o SGBD

utilizado permita estruturar fisicamente os discos de forma que o processamento não fique limitado a sucessão física das linhas das tabelas.

Segundo (MCGUFF, 1998) existem três tipos diferentes de processamento paralelo:

- SMP (*Symmetric Parallel Processing*): processo paralelo simétrico, os processadores compartilham uma memória comum, mais fácil de administrar as máquinas;
- MPP (*Massively Parallel Processing*): processo maciçamente paralelo, os processadores não compartilham nada, melhor desempenho global;
- *Clustering*: grupos de máquinas unidas por uma rede de alta velocidade de conexão.

(MCGUFF, 1998) descreve algumas considerações básicas que devem ser levadas em consideração pelos arquitetos de dados quando estiverem desenvolvendo o modelo físico:

- As Chaves alfanuméricas devem ser substituídas por chaves numéricas. O impacto pode ser significativo no espaço requerido para o armazenamento dos dados e índices. Mas o espaço em disco não é tão crítico quanto o tempo necessário para processar estas chaves. Tempos de procura para índices compostos por números inteiros são muito menores que os tempos de pesquisa por chaves textuais.
- Os atributos em uma relação são transitivamente dependentes de todas as entidades mais fracas. Por exemplo, o atributo *descrição_do_grupo* da tabela *grupo_de_produtos* é transitivamente dependente na relação com a tabela *subgrupo_de_produtos*. O processo de normalização removeu o campo *descrição_do_grupo* da entidade mais fraca (*subgrupo_de_produtos*) e o colocou somente na entidade mais forte (*grupo_de_produtos*). Neste caso pode ser benéfica a desnormalização deste relacionamento colocando o

campo *descrição_do_grupo* também da tabela *subgrupo_de_produtos*, isto melhorará o desempenho das consultas que envolvam as duas tabelas e a descrição do grupo, conforme ilustra a Figura 7.

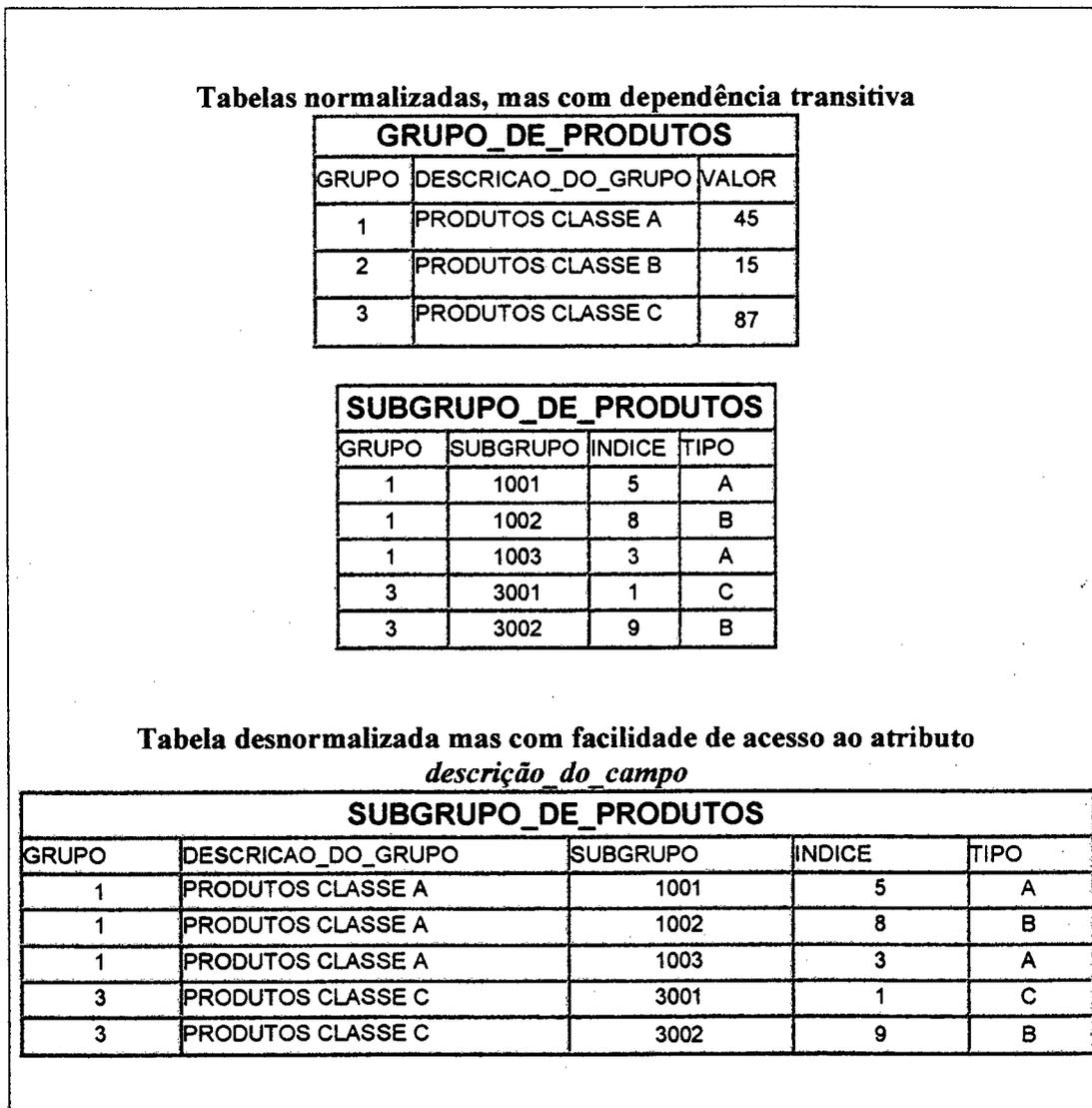


Figura 7 – Dependências transitivas entre tabelas.

2.1.8 Modelo de Dados Segundo W.H.Inmon

(INMON, 1996), também sugere um modelo de dados com três níveis de modelagem: Alto nível, nível intermediário e a modelagem de baixo nível.

2.1.8.1 Modelo de Dados de Alto Nível

Este nível apresenta as entidades e seus relacionamentos, que se encontram em seu mais alto nível de abstração, conforme ilustrado na Figura 8.

Para determinar quais entidades participam deste nível é necessário estabelecer o "escopo de integração" que é quem define as fronteiras do modelo de dados e deve ser definido antes do início do processo de modelagem. O escopo de integração pode ser definido pelos analistas, pela gerência e pelo usuário final e deve ser redigido em não mais do que cinco páginas e em uma linguagem de fácil entendimento para as pessoas de negócios.

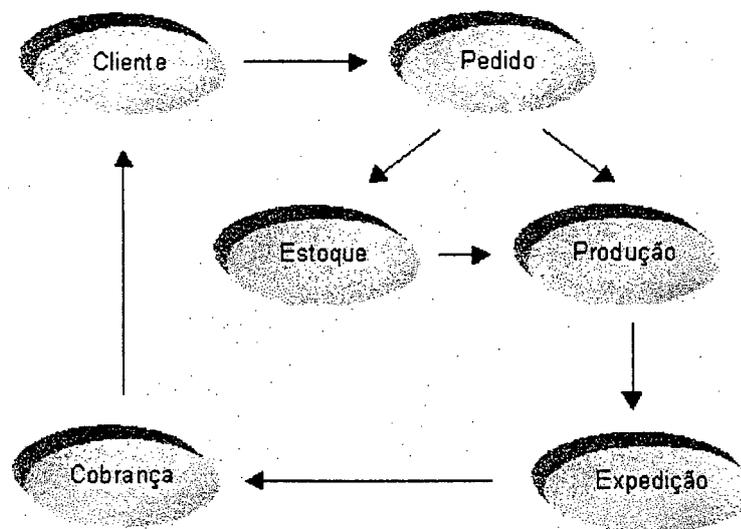


Figura 8 – Representação da modelagem de alto nível.

2.1.8.2 Modelo de Dados de Nível Intermediário

O modelo de dados de nível intermediário é criado a partir das áreas de interesse ou entidades identificadas no alto nível de modelagem, para cada uma destas áreas ou entidades é desenvolvido um nível intermediário próprio.

Este nível intermediário é composto por quatro elementos apresentados na Figura 9.

- 1 Um agrupamento primário de dados, o qual é composto pelos atributos que aparecem uma única vez em cada área de interesse. Como todos os agrupamentos de dados o agrupamento primário contem atributos e chaves;
- 2 Um agrupamento secundário de dados que engloba os atributos que podem aparecer mais de uma vez na mesma área de interesse;
- 3 Um conector que representa os relacionamentos dos dados entre as áreas de interesse;
- 4 O tipo dos dados.

Esses quatro elementos de modelagem são usados para identificar os atributos de dados de um modelo e os relacionamentos entre tais atributos.

Normalmente cada agrupamento de dados existente no modelo de dados resulta na definição de uma tabela durante o processo de projeto do banco de dados.

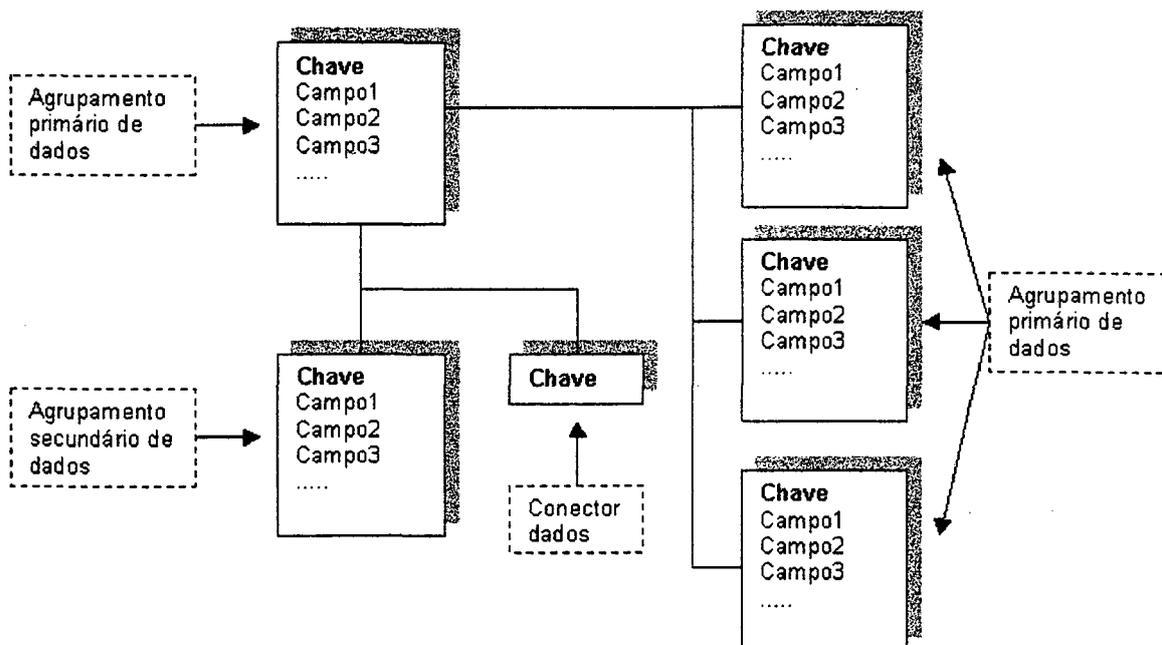


Figura 9 – Os elementos do modelo de dados de nível intermediário (INMON, 1997).

2.1.8.3 Modelo de Dados de Baixo Nível

O modelo de baixo nível, também chamado de modelo físico, de dados é criado a partir do modelo de nível intermediário simplesmente expandindo este de forma que ele passe a apresentar chaves e características físicas. Neste ponto o modelo físico de dados necessita ser alterado para receber as características de performance.

No caso do DW, o primeiro passo para a inclusão dos fatores de performance consiste em decidir sobre a granularidade e o particionamento dos dados, além da alteração da estrutura da chave para a inclusão do elemento de tempo. Depois que a granularidade e o particionamento tiverem sido incluídos, várias outras atividades de projeto físico são embutidas no projeto.

Uma das principais atividades no projeto físico se refere a otimização das operações de E/S (entrada/saída) física. E/S física é a atividade que

introduz os dados no computador a partir do meio de armazenamento ou envia os dados do computador para o meio de armazenamento.

A tarefa do projetista do DW consiste em organizar os dados fisicamente de forma que durante a execução de uma E/S seja transferida uma massa de dados que apresente alta probabilidade de ser acessada.

Por exemplo, se um programador precisar acessar e recuperar cinco registros e estes registros estiverem dispostos em blocos de dados diferentes, serão necessárias cinco E/S. Mas se o projetista tiver condições de prever a necessidade de agrupar os registros e coloca-los no mesmo bloco físico, somente uma E/S será necessária, fazendo com que o programa seja executado com mais eficiência.

2.2 Data Marts

Inicialmente as implementações de *data warehouse* baseavam-se em uma arquitetura centralizada, exigindo uma metodologia rigorosa e uma completa compreensão do negócio.

O *data mart* ou *warehouse* departamental, que é uma abordagem descentralizada, onde as informações são divididas nas diversas áreas de interesse ou departamentos, onde o todo irá formar o DW conforme demonstrado na figura 10, e passou a ser uma das opções de arquitetura *data warehouse*.

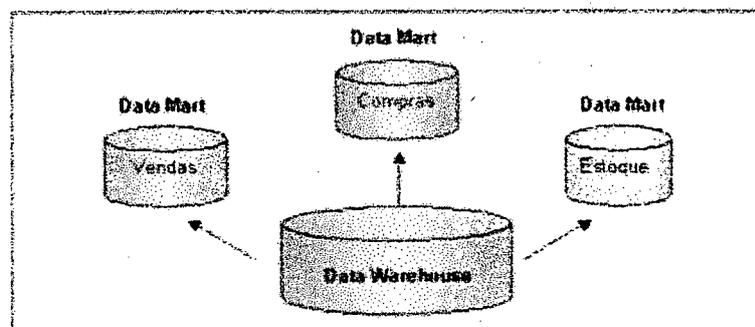


Figura 10 – Data Warehouse e Data Mart

Os *data marts* podem surgir de duas maneiras. A primeira é *top-down* e a outra é a *botton-up*.

Top-down: quando a empresa cria um DW e depois parte para a segmentação, ou seja, divide o DW em áreas menores gerando assim pequenos bancos orientados por assuntos departamentalizados.

Botton-up: quando a empresa por desconhecer a tecnologia, prefere primeiro criar um banco de dados para somente uma área. Com isso os custos são bem inferiores de um projeto de DW completo. A partir da visualização dos primeiros resultados parte para outra área e assim sucessivamente até resultar num *Data Warehouse*.

No ambiente de DW e no *Data Mart* verifica-se a aplicação da mesma tecnologia com variações como o volume de dados e a complexidade de carga. A principal diferença é a de que os *Data Marts* são voltados somente para uma determinada área, já o DW é voltado para os assuntos da empresa toda.

2.3 Metadados

São dados que fazem referência a outros dados, como por exemplo atributos das tabelas agregadas ou utilizadas, cálculos necessários, descrições, periodicidade das cargas, histórico de mudanças etc., que são gerados em praticamente todas as fases de um projeto de *data warehouse*.

Segundo (CHEROBIN, 2000), geralmente os metadados em DW podem ser apresentados em três camadas diferentes:

Metadados operacionais: definem a estrutura dos dados mantidos pelos bancos operacionais, usados pelas aplicações de produção da empresa;

Metadados centrais do DW: 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;

Metadados do nível do usuário: organizam os metadados do DW para conceitos que sejam familiares e adequados aos usuários finais.

Podem ainda ser classificados conforme a classe de seus componentes:

Mapeamento: descrevem como os dados de sistemas operacionais são transformados antes de entrarem no DW. Exemplos dessa classe de metadados podem ser os que identificam campos fontes, mapeamentos entre atributos, conversões, codificações, padrões, etc.;

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;

Miscelânea: esta classe define diversos tipos de metadados, informações da situação de desenvolvimento de partes do DW, informações sobre volume dos dados para estimativas de tempo e recursos, etc.;

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;

Padrões de acesso: mantêm informações sobre frequência e tipo de acesso aos dados.

Encontram-se hoje ferramentas capazes de armazenar e gerenciar metadados como Sybase Warehouse Control Center e Prism Warehouse Directory, mas ainda não existe um padrão para manipulação deste tipo de informação.

Segundo (INMON, 1997), os metadados mantêm informações sobre "o que está e onde" no DW. Tipicamente os aspectos que sobre os quais os metadados mantêm informações são:

- A estrutura dos dados, segundo a visão do programador;
- A estrutura dos dados, segundo a visão dos analistas de SAD;
- A fonte de dados que alimenta o DW;

- A transformação sofrida pelos dados no momento de sua migração para o DW;
- O modelo de dados;
- O relacionamento entre o modelo de dados e o DW;
- O histórico das extrações de dados.

2.3.1 Fontes dos Metadados

As fontes dos metadados são diversas, muitas informações de metadados precisam ser colhidas e refinadas durante o processo de construção do DW dos próprios analistas de SAD e futuros usuários e outras podem ser encontradas em fontes como:

Repositórios de ferramentas CASE: Normalmente as ferramentas CASE mantêm uma base de conhecimento sobre as aplicações geradas com dados estruturados que facilitam a integração entre a origem dos metadados e o repositório do ambiente de DW. Estes repositórios permitem a extração de informações sobre: origem, fluxo e formato dos dados e definições de negócios;

Documentação do desenvolvimento dos sistemas operacionais: Quanto melhor a documentação dos sistemas existentes mais fácil será o trabalho de determinação dos metadados, pois nas especificações dos sistemas podem ser encontradas diversas informações relevantes como estruturas, origens e fluxos dos dados;

Código fonte dos sistemas de nível operacional: Uma boa codificação implica na inserção de informações relevantes a procedimentos e rotinas no próprio código fonte dos sistemas e sua análise pode revelar informações muito importantes. Às vezes se torna necessária a procura por regras através da análise do fluxo de execução destes sistemas, tornando a tarefa de obtenção dos metadados demorada e cara. Outra opção é simplesmente executar e

utilizá-los como forma de esclarecer dúvidas que a documentação nem sempre contempla;

Entrevistas: O processo de construção de ambientes de *data warehouse* envolve uma série de profissionais, como analistas, gerentes, programadores, usuários, que normalmente possuem informações de vital importância sobre os sistemas operacionais, definição das necessidades de informação e de negócio e fontes de dados. Através das entrevistas podem ser obtidas informações não explícitas na documentação ou nos sistemas como requisitos para testes dos dados e indicadores de qualidade dos dados;

Ambiente de DW: São muitas as informações que podem ser extraídas do próprio ambiente de DW, entre estas verificam-se: frequência de acessos, nível de agregação, tempo de resposta, entre outras, que contribuirão para a criação de estruturas de metadados e para outros propósitos.

2.4 OLAP

Não basta possuir informações, é necessário o poder de consultar e interagir com estas informações. OLAP (*On-Line Analytic Processing*) é um nome genérico dado a um conjunto de conceitos e ferramentas relativas à consulta analítica e interação on-line com dados armazenados nos sistemas de informação da organização, destacando-se o *data warehouse*, respondendo às possíveis correntes de perguntas dos analistas, gerentes e executivos.

Antes do surgimento do conceito OLAP, as consultas eram realizadas sobre relatórios montados nos sistemas operacionais da empresa, gerando dificuldades na interação com as informações contidas nos bancos de dados. Entre os problemas que motivaram a busca de novas formas de consulta e produção de relatórios pode-se citar:

- Construção delegada normalmente a área de informática que nem sempre tinha o conhecimento das reais necessidades de informação;

- Demora na construção de novos relatórios;
- Relatórios que quase sempre se apresentavam de forma estática;
- Acúmulo de diferentes tipos de relatórios num mesmo sistema gerando problemas de manutenção.

OLAP é implementado normalmente em modo cliente/servidor e procura oferecer respostas rápidas as consultas, possibilitando inclusive a criação de um microcubo, sendo que sua funcionalidade é caracterizada pela análise multidimensional e dinâmica dos dados, apoiando o usuário final em suas atividades.

2.4.1 Características OLAP

Segundo (FUJIWARA, 2001) as características da análise OLAP são:

Drill Across

O Drill Across ocorre quando o usuário pula um nível intermediário dentro de uma mesma dimensão. Por exemplo: a dimensão tempo é composta por ano, semestre, trimestre, mês e dia. O usuário estará executando um Drill Across quando ele passar de ano direto para semestre ou mês.

Drill Down

O Drill Down ocorre quando o usuário aumenta o nível de detalhe da informação, diminuindo o grau de granularidade.

Drill Up

O Drill Up é o contrário do Drill Down, ele ocorre quando o usuário aumenta o grau de granularidade, diminuindo o nível de detalhamento da informação.

Drill Throught

O Drill Throught ocorre quando o usuário passa de uma informação contida em uma dimensão para uma outra. Por exemplo: Estou na dimensão de tempo e no próximo passo começo a analisar a informação por região.

Slice And Dice

O Slice and Dice é uma das principais características de uma ferramenta OLAP. Como a ferramenta OLAP recupera o microcubo, surgiu a necessidade de criar um módulo que se convencionou de Slice and Dice para ficar responsável por trabalhar esta informação. Ele serve para modificar a posição de uma informação, alterar linhas por colunas de maneira a facilitar a compreensão dos usuários e girar o cubo sempre que tiver necessidade.

Alertas

Os Alertas são utilizados para indicar situações de destaque em elementos dos relatórios, baseados em condições envolvendo objetos e variáveis. Servem para indicar valores mediante condições, mas não para isolar dados pelas mesmas.

Ranking

A opção de ranking permite agrupar resultados por ordem de maiores ou menores, baseado em objetos numéricos (*Measures*). Esta opção impacta somente uma tabela direcionada (relatório) não afetando a pesquisa (Query).

Filtros

Os dados selecionados por uma Query podem ser submetidos a condições para a leitura na fonte de dados. Os dados já recuperados pelo Usuário podem ser novamente "filtrados" para facilitar análises diretamente no documento.

Sorts

Os sorts servem para ordenar uma informação. Esta ordenação pode ser customizada, crescente ou decrescente.

Breaks

Os Breaks servem para separar o relatório em grupos de informações (blocos). Por exemplo: O usuário tem a necessidade de visualizar a informação por cidades, então ele deve solicitar um Break. Após esta ação ter sido executada, automaticamente o relatório será agrupado por cidades, somando os valor mensuráveis por cidades.

3 FERRAMENTAS CASE

Para (REZENDE, 1999) CASE (*Computer Assisted/Aided Software Engineering*), representa uma ferramenta ou conjunto de técnicas facilitadoras de desenvolvimento de software moderno, que faz uso de técnicas estruturadas para facilitar o trabalho do Engenheiro de Software.

Segundo (PRESSMAN, 1995) as tecnologias CASE compreendem uma ampla variedade de tópicos que abrangem métodos de engenharia de software e procedimentos de gerenciamento de projetos.

Estes aplicativos foram criados inicialmente para gerar interfaces, mais tarde para converter linguagens e gerar as estruturas de dados, evoluíram e passaram a absorver informações sobre o projeto e a auxiliar atividades ao longo do processo de software, popularizaram-se e passaram a figurar como utilitários muito eficientes no desenvolvimento e manutenção de sistemas.

As ferramentas de tecnologia CASE possuem facilidades gráficas para o planejamento de sistemas. A bancada de trabalho coleta informações de projeto para acionar o gerador de códigos (REZENDE, 1999).

A simples capacidade gráfica não é suficiente para caracterizar uma ferramenta como CASE, os objetos diagramados devem ser colocados em um banco de dados do projeto, que pode também conter detalhes dos elementos dos dados e a lógica do processo. O modelo lógico do sistema construído no banco de dados do projeto pode ser testado para se verificar sua complexidade

e consistência, antes de ser impresso para formar uma especificação de sistema.

Um produto CASE deve construir dentro de si próprio um banco de dados do projeto, a um nível mais alto do que comandos de linguagens de programação ou definição de elementos de dados. Este banco de dados do projeto, chamado de Repositório, deve manter informações sobre os dados que serão armazenados no sistema, sobre a lógica comercial dos processos a serem implementados, sobre a diagramação das telas e relatórios e outras informações relativas aos requisitos dos sistemas e do projeto.

Segundo (FISCHER, 1991), os produtos de CASE são um subgrupo especial de auxiliares de desenvolvimento e manutenção, que contêm geradores de código e ferramentas de engenharia reversa que extraem a lógica em nível de especificação do código. Nesta definição os auxiliares de programação, tais como exercitadores de código, depuradores e geradores de dados de teste não são classificados como ferramentas CASE, pois eles não constroem um banco de dados do projeto. Segundo (GANE, 1990), tampouco são produtos de CASE as ferramentas de reestruturação que simplesmente traduzem um grupo de comandos em linguagem fonte para um outro grupo (mais legível ou melhor modificável) de comandos em linguagem fonte.

Estas ferramentas normalmente reduzem certos problemas do desenvolvimento, principalmente erro humano, pois por meio da geração automática de código fonte, estruturas das bases de dados, controle da nomenclatura de variáveis e atributos (dicionário de dados) e auto-teste de diversos pontos do projeto e do software é possível reduzir a possibilidade de inconsistências.

Os pacotes de análise de requisitos e de especificação do projeto são exemplos de ferramentas CASE que permitem ao projetista exibir e editar graficamente o esquema do software. A edição gráfica interativa consome muito menos tempo, sem falar que é mais agradável, que a restauração de projetos em papel. Além disso, essas ferramentas de especificação de projetos oferecem recursos de verificação de consistência que averiguam se todos os

módulos do software comunicam-se adequadamente e se todas as estruturas de dados estão totalmente especificadas.

Existem ferramentas CASE mais enfocadas que as de especificação de projeto. Um exemplo de ferramenta enfocada é o gerador de interface com o usuário. Pode-se citar nesta classe de ferramentas CASE os pacotes de geração e de projeto de formulários interativos, que permitem ao projetista de software desenvolver e especificar uma interface com o usuário baseada em formulários para promover aplicativos. A operação dessas ferramentas geralmente é do tipo "o que se vê é o que se obtém".

A tecnologia CASE também é valiosa quanto existe falta de programadores, o que é comum na maioria dos departamentos de processamento de dados. As filas de espera (*backlogs*) dos programas acontecem por causa do tempo gasto na implementação de um aplicativo, bem como da dificuldade de recrutamento de programadores qualificados e talentosos. A tecnologia CASE ajuda a reduzir a fila de espera, tornando mais produtiva cada hora do programador.

Conforme (FISCHER, 1991) as ferramentas CASE não eliminam a criatividade; pelo contrário, oferecem ferramentas para demonstrá-la melhor.

As ferramentas CASE consistem em um equipamento potente para complementar nossos métodos manuais. Os aplicativos CASE aprimoram a criatividade, pois possibilitam a construção de projetos mais elaborados em menos tempo. A ferramenta de diagramação de fluxo de dados elimina o trabalho enfadonho de desenhar, oferecendo mais liberdade ao analista para pesquisar arquiteturas alternativas. Os programas de layout geram uma interface com o usuário permitindo que o técnico se concentre nos fatores humanos do projeto da interface, deixando para a ferramenta a tarefa de voltar a gerar uma nova implementação de trabalho a cada ciclo de feedback do usuário.

Um dos principais ganhos no desenvolvimento utilizando CASE é ter um maior tempo e esforço para destinar a análise dos sistemas pretendidos, uma vez que as ferramentas devem e podem automatizar uma série de

atividades, normalmente desgastantes, relativas a tradução das especificações para o software propriamente dito.

3.1 Tipos de CASE

As ferramentas CASE normalmente são separadas em dois grupos principais:

- a) CASE horizontal: apóia todo o processo de software.
- b) CASE vertical: apóia fases específicas do processo de software

No que se refere a CASE vertical (PRESSMAN, 1995) citando (QED, 1989) afirma que estas ferramentas podem ser classificadas por função, por seus papéis como instrumentos para os gerentes e para o pessoal técnico, pelo uso que elas têm nas várias etapas do processo de engenharia de software, pela arquitetura de ambiente (hardware e software) que as suporta ou até mesmo pela origem ou custo delas. Contudo sugere utilizar a função como critério primordial de classificação elegendo, por exemplo, os seguintes aspectos:

- a) Planejamento de sistemas comerciais;
- b) Gerenciamento de projetos;
- c) Suporte;
- d) Análise e projeto;
- e) Programação;
- f) Integração e testes;
- g) Prototipação;
- h) Manutenção;
- i) Estrutura.

Na tabela 4 são relacionados alguns exemplos de ferramentas CASE disponíveis.

Classificação CASE	Ferramentas
Planejamento	Foundation, Interactive Engineerig Workbench; Information Engineering Facility;
Gerenciamento	MS Project, Super Project, MacProjetc II, ESTIMATES;
Especificação de requisitos	CORE, RMS/PC, R-Trace;
Especificação formal	CADIZ, OBJ;
Documentação	Interleaf, Page Maker;
Comunicação	MS mail, BBS
Controle de Qualidade	Q/Auditor; Synop, CHIMES
Gerenciamento de versão / configuração	SCCS (Unix), PVCS;
Análise e Projeto de Software	Rational Rose, System Architect, PC Case, HOOD ...
Projeto e desenvolvimento de Interfaces	TeleUse, Interviews, PV-Wave, UIMS/X, Inventor Toolkit, Trident, Xcessory, AVS/Express, XVT, SL-GMS, Vision
Programação	Delphi, Visual Basic, Visual C++

Tabela 4 – Exemplos de Ferramentas CASE

3.2 Benefícios da Tecnologia.

Para muitas organizações de desenvolvimento de software, as vantagens qualitativas das ferramentas CASE têm um peso maior que as

quantitativas. O tempo gasto no desenvolvimento será quase sempre menor com o auxílio das ferramentas CASE, mas talvez seu maior benefício tenha a forma de garantia, ou consciência tranqüila, de que a tarefa está sendo executada devidamente, como foi programada e segundo as especificações do usuário.

As ferramentas CASE são vantajosas porque revelam inúmeras exigências (e surpresas) antes que se inicie a implementação. Oferecem dispositivos para manterem as especificações do projeto atualizadas em face de manutenção contínua do software, reconstrução do projeto, aprimoramentos e evolução do código.

Os aplicativos CASE, de alta qualidade, são mais atraentes de se usar do que escrever o código à mão, ajudam a projetar, a documentar, mantêm a sincronização com o código-fonte e estruturas de dados, reduzem os riscos de falhas e surpresas e normalmente reduzem o tempo total do desenvolvimento.

3.3 Características Desejáveis

Segundo (FISCHER, 1991) as ferramentas CASE devem satisfazer alguns critérios básicos para fazerem parte da caixa de ferramentas dos utilizadores de software, isto é, devem apresentar algumas características fundamentais para que possam ser aceitas pelos métodos de desenvolvimento-padrão das organizações, dentre estas características pode-se citar:

Fracionamento da complexidade:

Uma das metas principais da tecnologia CASE é decompor os requisitos e os projetos em componentes manejáveis. Sua função é simplificar, explicar e reduzir.

Adequação a um público diversificado:

Para as fases de requisitos e de projetos do ciclo de software, as ferramentas CASE devem servir diversos mestres. Por um lado, sua saída deve ser inteligível para os usuários finais e para as organizações contratantes, que pagam pelo desenvolvimento do software. Por outro lado, devem oferecer uma ajuda valiosa aos próprios utilizadores; caso contrário, é perda de tempo utilizá-las.

Menor custo de desenvolvimento:

A utilização de uma ferramenta CASE deve custar menos e ser mais eficaz ao longo prazo do que a montagem de um sistema de software pelos métodos tradicionais. As ferramentas CASE devem reduzir substancialmente o empenho despendido em implementação e manutenção, oferecendo especificações e projetos de qualidade superior.

Quantitativas e verificáveis:

As especificações e projetos gerados pelas ferramentas CASE devem articular meticolosa e concisamente as características e os componentes do software a ser construído. Cada exigência da implementação tem que ser verificável, e poder ser encontrada no documento dos requisitos. Os critérios de desempenho, as limitações e as condições de erro devem estar especificadas no projeto.

De fácil manutenção:

As especificações e projetos produzidos ou montados por uma ferramenta CASE devem ser adaptáveis às modificações nas metas dos requisitos e dos projetos. Quando um documento do projeto perde a sincronização com o código subordinado, torna-se inútil, e pode até causar perda de tempo aos utilizadores em futuros aperfeiçoamentos do software.

Orientação gráfica:

As boas ferramentas CASE apresentam informações visuais de especificações e projetos. São para a engenharia de software o que os programas CAD (*Computer Aided Design*) são para o projeto esquemático e o

layout. Tanto usuários finais como para os utilizadores, é muito mais fácil compreender uma ilustração gráfica do que ler inúmeras páginas do texto descrito.

4 CRIAÇÃO DE AMBIENTE DE DW UTILIZANDO CASE

Neste capítulo apresentam-se os passos necessários a construção de ambientes de *data warehouse* utilizando tecnologia CASE.

A construção de sistemas de apoio à decisão possui uma série de etapas definidas, comentadas adiante, que devem ser cumpridas também com a utilização de ferramentas de suporte ao desenvolvimento.

As ferramentas têm como principal função acelerar os processos necessários e facilitar a integração e implantação da tecnologia DW.

4.1 Adaptação da Metodologia.

A metodologia apresentada no presente trabalho é baseada nas propostas de (INMON, 1997) e (KIMBALL, 1996).

O primeiro passo para o desenvolvimento do projeto é a designação de uma equipe que agregue diferentes tipos de conhecimento sobre as tecnologias disponíveis e sobre as necessidades de informação da empresa. Os responsáveis pela análise e projeto do DW recebem o nome de Analistas de SAD ou Analistas de Negócio e além de bons conhecimentos em análise, programação e em bancos de dados, necessitam de visão administrativa / gerencial. Esta é uma das razões dos elevados custos de implantação desta tecnologia, uma vez que é percebida a falta de pessoal capacitado. Também

devem ser envolvidos outros profissionais com qualificações diversas, como gerentes, usuários, administradores de dados e bancos de dados, programadores, pessoal de suporte, etc., pois nem toda a informação necessária à medição do negócio está nos sistemas ou na documentação.

O passo seguinte consiste em planejar a migração para o ambiente projetado de *data warehouse*. Este planejamento “*global*” é originado pela necessidade da divisão do projeto em etapas, normalmente relacionadas às áreas de interesse geral da empresa. Assim como na antiga máxima: “*Dividir para conquistar*”, a criação do ambiente de DW deve ser uma atividade conduzida passo a passo, mediante uma parcela finita de informações por vez, ou seja, deve seguir um plano de atividades focadas a uma determinada porção de informações, uma vez implementada, deve partir para outra porção e assim por diante.

O planejamento global citado no parágrafo anterior, após identificar as principais áreas de interesse da empresa, deve gerar um diagrama representando a divisão e a seqüência de trabalho do projeto de DW. Este diagrama pode sofrer alterações no decorrer da implantação uma vez que as áreas de interesse normalmente estão inter-relacionadas e diversos fatores só serão conhecidos no momento da análise e detalhamento dos sistemas operacionais.

Nesta etapa do projeto, se faz necessária à observação dos seguintes itens:

- Criação de um plano de migração.
- Revisão da metodologia.
- Formulação de um checklist para a revisão do projeto.

Estas atividades devem ser repetidas para cada uma das divisões do projeto formando uma espécie de espiral de tarefas com início, mas sem fim determinado, uma vez que o após a conclusão de todas as etapas previstas no planejamento global haverá inclusão de novas linhas de análise e refinamentos.

É importante levar em consideração que a construção de ambientes de apoio à decisão esta intimamente ligada aos sistemas existentes na organização, na medida que novos aplicativos venham a ser implantados haverá necessidade de levantar quais informações podem ser interessantes ao DW.

Observando as fases de desenvolvimento de sistemas desde a construção dos programas de nível operacional, passando pela implementação do DW, até a consulta as informações do *data warehouse* pode-se afirmar que as ferramentas CASE hoje disponíveis, são capazes de auxiliar em diversas tarefas reduzindo consideravelmente o esforço de desenvolvimento, passando conseqüentemente ao analista de negócios um trabalho mais voltado ao levantamento das necessidades de informação estratégica e gerencial da organização. Estas ferramentas também podem forçar a execução do projeto de DW de maneira lógica, cumprindo os passos necessários ao desenvolvimento de um SAD de comprovada qualidade.

Existem diversas ferramentas CASE capazes de auxiliar o processo de desenvolvimento do DW, e cabe a equipe de analistas de negócio, o estudo das opções disponíveis e a decisão sobre qual ferramenta é mais adequada a cada uma das etapas passíveis de auxílio.

A seqüência de passos necessários à construção de ambientes de apoio a decisão pode variar dependendo das ferramentas escolhidas, mas de uma maneira geral, a grande maioria disponibiliza uma interface onde será desenhado o modelo de dados. À medida que este trabalho é executado, o aplicativo deve permitir também a definição do sistema de registro e os processos para a transformação dos dados operacionais em registros do DW. Com este conhecimento, passado a ferramenta CASE, se torna possível a geração das diversas estruturas do DW, dos procedimentos de carga e em alguns casos até relatórios padronizados de consulta as informações do armazém de dados.

Em muitas ferramentas o trabalho desempenhado na fase de construção do plano de migração é equivalente a fase de desenvolvimento dos

sistemas de *data warehouse*, contudo o analista de negócios deve levar em consideração as diversas fases do desenvolvimento principalmente para possibilitar a adequação às restrições impostas por hardware, software, rede e outros recursos.

Na tabela 5 estão listadas, de modo resumido, as fases do desenvolvimento de ambientes de DW. Dependendo da ferramenta utilizada o trabalho principal se resume ao plano de migração onde são inseridas as informações necessárias na base de conhecimento da ferramenta CASE, o que permitirá a geração automática dos sistemas e estruturas de dados.

1 Plano de Migração	
1.1 – Criação do Modelo de Dados	<u>Modelo Empresarial:</u> Diagrama de entidade-relacionamento normalizado das regras de negócio; <u>Modelo Dimensional:</u> Estruturas relacionais apresentadas através de modelos como o Estrela; <u>Modelo Físico:</u> Adequação do projeto as restrições impostas por hardware e software como velocidade, recursos de armazenamento e outras limitações.
1.2 – Definição do Sistema de Registro.	Identificação de quais os melhores dados para atender aos requisitos do modelo de dados.
1.3 – Adequação do Modelo de Dados ao Sistema de Registro	Adequação do modelo de dados ao sistema de registro, definindo os processos necessários para que os dados do sistema operacional se transformem nos registros do DW.
2 Desenvolvimento dos Sistemas	
2.1 – Sistemas Operacionais	Desenvolvimento dos sistemas convencionais destinados ao processamento transacional.
2.2 – <i>Data Warehouse</i>	Desenvolvimento dos armazéns de dados destinados ao processamento analítico.
2.3 – Sistemas de Consulta	Processo de construção dos relatórios e sistemas de consulta ao armazém de dados.
3 Checklist para Revisão do Projeto	Verificar a consistência do projeto e responder as mais diversas questões relacionadas ao sistema de apoio à decisão desenvolvido.

Tabela 5 – Fases do Desenvolvimento de um DW.

O plano de migração pode ser modelado inteiramente em uma ferramenta CASE, como exemplo cita-se o ERWIN que possibilita o desenho do modelo dimensional tipo estrela facilitando a criação da estrutura do DW, bem como a inserção dos metadados necessários, gerando posteriormente código SQL capaz de criar tabelas, índices e relacionamentos. Para isto é necessário indicar, na própria ferramenta o sistema de registro e adequá-lo ao modelo de dados desenhado. Outro exemplo é a ferramenta Gxplorer onde o modelo de dados é descrito através de tabelas com indicadores e dimensões, e com o conhecimento passado a ferramenta CASE na criação dos sistemas operacionais é possível utilizar os metadados presentes na base de conhecimento da ferramenta facilitando a criação do sistema de registro, também gera automaticamente a base de dados e auxilia na criação dos procedimentos de carga.

Na fase de desenvolvimento de sistemas também são encontradas boas aplicações para as ferramentas CASE, sendo que no ambiente operacional verifica-se a existência de uma grande variedade de ferramentas disponíveis como: Sculptor, Miro, GasPro, System Architect, Genexus, Clarion, Dr.Case, entre outros, que auxiliam nas diversas fases do ciclo de vidas do software.

Nas fases de construção de DW encontram-se ferramentas como: Data Auditing, WizRule, Migration Architect, QDB/Analyze, Business Objects, Maestro, Gxplorer, PowerPlay, entre outros e finalmente para consulta encontram-se Business Objects, Gxolap, entre outras. Na fase de consulta as informações do DW encontram-se: Data Prism, Crystal Reports, Impromptu, Discoverer/2000, Pilot Designer, etc.

À medida que a tecnologia de DW vem se tornando parte integrante da computação corporativa, os fornecedores de ERPs, softwares de gestão empresarial, começam a integrar a seus produtos ferramentas de análise e extração de dados, entre eles encontram-se nomes de empresas como: Oracle; Datasul, PeopleSoft, SAP, JD Edwards, Logocenter, Baan, SSA, NCR.

No checklist de revisão do projeto encontram-se ferramentas capazes de medir desempenho, tempos de resposta, fazer simulações, etc., mas de uma maneira geral esta é uma etapa bastante subjetiva e a maioria dos trabalhos normalmente desenvolvidos não contam com apoio de ferramentas automatizadas.

Independente das ferramentas selecionadas para auxiliar no desenvolvimento do projeto, o analista de negócios deve ter em mente as seguintes etapas: Plano de Migração; Desenvolvimento dos Sistemas; e Checklist para revisão do projeto.

4.2 Plano de Migração

Inicialmente deve ser construído o modelo de dados representando as necessidades de informação, sem levar em consideração a tecnologia ou aquilo que a empresa dispõe.

(INMON, 1997) afirma que o modelo de dados precisa identificar no mínimo o seguinte:

- Os principais assuntos, ou áreas de interesse da empresa;
- Os relacionamentos entre os principais assuntos;
- Os agrupamentos de chaves e atributos que representa mais plenamente os assuntos como:
 - Os atributos dos assuntos principais
 - As chaves dos assuntos principais
 - Os grupos repetitivos de chaves e atributos
 - Conectores entre as principais áreas de interesse
 - Relacionamentos por subtipos.

O passo seguinte à criação do modelo de dados é a definição do sistema de registro, que é a identificação de quais os melhores dados para atender aos requisitos identificados no modelo de dados. Para definir qual a melhor fonte de dados deve-se levar em consideração os dados mais

completos, mais adequados ao momento presente, mais exatos, mais adequados à fonte de entrada e os que mais se ajustam à estrutura do modelo de dados.

O próximo passo consiste no projeto do *data warehouse*, que é na verdade uma adequação do modelo de dados e do sistema de registro. Na figura 11, migração para o ambiente projetado, estão representados os três passos necessários:

- 1º - Criação do Modelo de Dados;
- 2º - Definição do Sistema de Registro;
- 3º - Adequação do Modelo de Dados e Sistema.

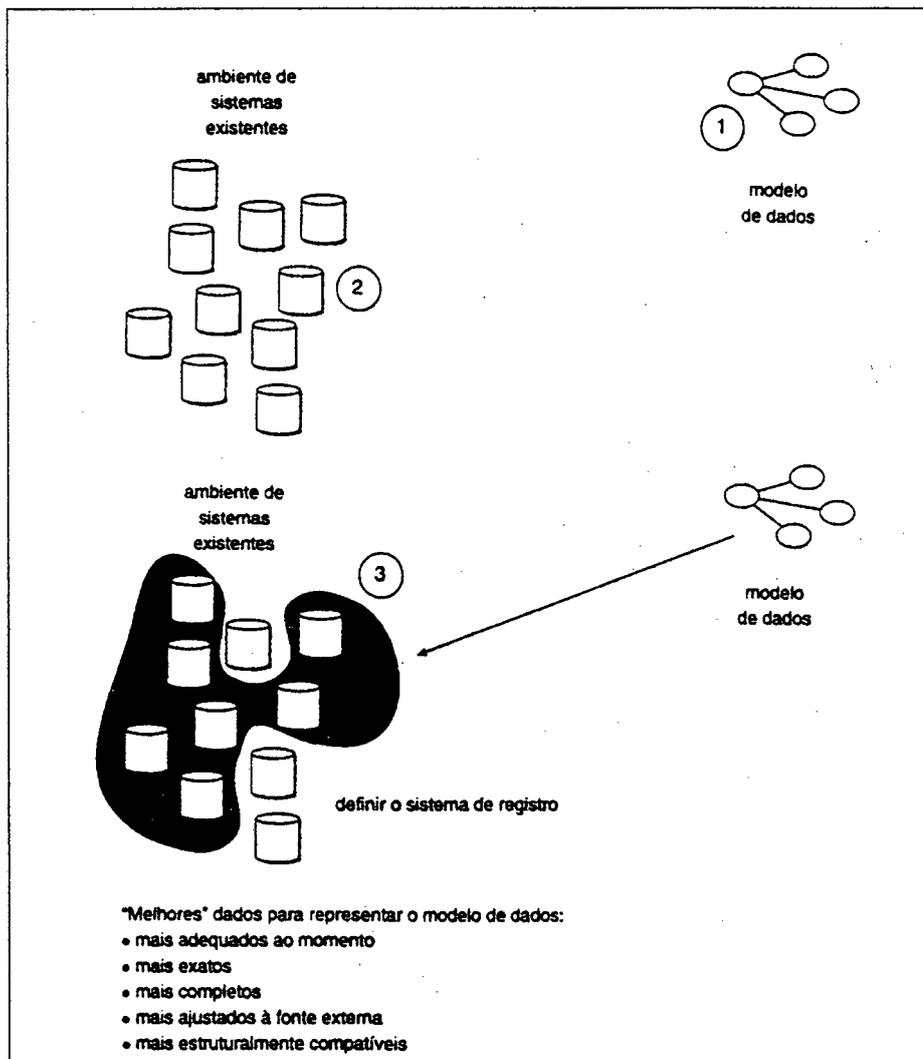


Figura 11 - Migração para o ambiente projetado

(INMON, 1997) afirma que o projeto do *data warehouse* basicamente deve ter o seguinte:

- Um elemento de tempo precisa ser acrescentado à estrutura da chave caso algum já não esteja presente;
- Todos os dados meramente operacionais precisam ser eliminados;
- Relacionamentos por integridade referencial precisam ser transformados em artefatos;
- Dados derivados que freqüentemente são necessários são acrescentados ao projeto;
- A estrutura dos dados precisa ser alterada, quando apropriado, para:
 - Incluir arrays de dados
 - Incluir dados redundantes
 - Separar dados em condições adequadas
 - Intercalar tabelas quando apropriado;
- A análise de estabilidade dos dados precisa ser feita.

O DW deve ser organizado em áreas de interesse ou assuntos. Normalmente encontram-se áreas de interesse como Vendas, Compras, Clientes, Fornecedores, Serviços, etc. Estas áreas de interesse serão normalmente armazenadas em tabelas separadas com conexão através de uma chave comum.

O passo seguinte consiste em projetar e construir as interfaces entre o sistema de registro, no ambiente operacional e os armazéns de dados. Estas interfaces também são conhecidas como procedimentos de carga e são responsáveis pela captura e refinamento periódico dos dados dos sistemas de nível operacional para povoamento do *data warehouse*. Este é um dos pontos mais críticos do processo e exige grande esforço de desenvolvimento, chegando a ocupar até 80% dos esforços necessários a construção de um DW.

A seguir é definida a primeira área para povoamento do DW e feedback. Uma vez que normalmente são necessários ajustes não é

aconselhada a inclusão de grande quantidade de dados por diminuir a flexibilidade do DW e dificultar a análise. Nesta etapa o usuário final tem a tarefa de examinar e opinar sobre a estrutura criada, fornecendo um feedback ao arquiteto de dados que irá efetuar as alterações necessárias. Este ciclo será realizado até a correção das possíveis inconsistências.

Segundo (INMON, 1997) o *data warehouse* deve atender as necessidades do SAD da empresa, não as necessidades operacionais, aconselhando uma frequência de no mínimo 24 horas para a renovação (carga) do DW e grande interação entre o arquiteto de dados e o usuário do DW para equilibrar as necessidades e possibilidades do ambiente.

4.3 Desenvolvimento de Sistemas Baseados em Dados

A adoção de uma metodologia significa escolher um caminho já percorrido, testado e normalmente aprovado para acompanhar o desenvolvimento evitando surpresas.

Na construção de sistemas de nível operacional podem-se observar metodologias que se baseiam no ciclo de vidas do software, começando com requisitos e terminando em código, contudo no desenvolvimento de sistemas de apoio à decisão é verificado o inverso, isto é, inicia com dados e encerra com requisitos.

O exposto no parágrafo anterior é a base para afirmações de que as ferramentas CASE não são aconselhadas na construção de ambientes de *data warehouse*, uma vez que estas fazem uso dos requisitos definidos para a criação das estruturas, base de dados e programas, mas se a ferramenta CASE for projetada para suportar a estrutura dos SADs pode-se modelar os indicadores e as dimensões com grande economia de recursos na construção de ambientes de DW, invalidando tais afirmações.

Na metodologia baseada em dados observam-se três fases distintas que são:

4.3.1 Fase Operacional

Na fase operacional, a construção dos diversos sistemas se inicia com a decisão de se implementar algo e irá seguir o curso do desenvolvimento tradicional de software, fazendo uso dos diversos modelos de ciclo de vidas conhecidos, conforme a necessidade e características do projeto.

Dentre as alternativas encontra-se a modelagem tradicional. As atividades encontradas nesta metodologia de desenvolvimento de sistemas operacionais são:

1 Atividades Iniciais do Projeto

A partir da decisão de construir um sistema operacional são realizadas atividades para o levantamento de informações necessárias a obtenção dos requisitos brutos do sistema, como, entre outras: entrevistas, coleta de dados, sessões JAD, elaboração do plano estratégico.

2 Uso de Código/ Dados Existentes

Análise aos sistemas existentes para o levantamento das estruturas, códigos e dados existentes visando a reutilização e integração do ambiente.

3 Dimensionamento/Divisão em fases

Identificação das fases de desenvolvimento gerando uma formalização dos requisitos e análise de capacidade

4 Formalização do Requisitos

Depois da reunião dos requisitos, dimensionados e divididos em fases é necessário formaliza-los gerando a especificação formal dos requisitos.

5 Análise da Capacidade

É necessário avaliar os recursos totais a serem consumidos pelo projeto. São avaliados recursos com: consumo de DASD, Requisitos de software, CPU, E/S, memória, rede/canal.

6 Definição do Ambiente Técnico

Deve ser definido o ambiente técnico estabelecendo no mínimo: Plataformas de hardware e sistema operacional a serem utilizados; SGBD(s) e linguagens de desenvolvimento; Suporte e rede.

7 Diagrama de Entidade Relacionamento – DER

A construção destes diagramas resultará na identificação das principais áreas de interesse e fornecerá documentação necessária à condução e análise rápida do sistema.

8 Conjuntos de Itens de Dados: Data Item Sets - DIS

Detalhamento das áreas de interesse identificados na atividade 7. O DIS contém atributos de dados, o agrupamento dos atributos e chaves, bem como os tipos de dados e os conectores, que são as representações dos relacionamentos, e os agrupamentos secundários de dados.

9 Análise de Performance

Necessária caso o processamento e/ou quantidade de dados sejam elevados para a identificação de possíveis picos ou gargalos no sistema e seu respectivo tratamento, seja em termos de maior suporte físico como na criação de estruturas, índices, arrays, separação de dados, entre outras técnicas utilizadas para garantir a performance.

10 Projeto Físico de Banco de Dados

Consiste em projetar fisicamente o banco de dados, onde estarão presentes: as tabelas, índices, atributos físicos de dados, estratégias de particionamento, designação de chaves, integridade referencial, tratamento de nulos e Clusterização/Intercalação.

11 Decomposição Funcional

Divisão do documento de requisitos em uma série de funções sucessivamente menores, descrevendo, de um nível alto até um nível mais baixo, as diferentes atividades a serem executadas.

12 Contexto Nível 0

Representação gráfica do nível mais alto de abstração das principais atividades a serem desenvolvidas, correspondendo, na especificação de processos, ao DER na modelagem de dados.

13 Contexto Nível 1-n

Representação gráfica mais detalhada das atividades, desdobras até um nível primitivo, correspondendo, em termos de projeto de processos, ao data item set (DIS).

14 Diagrama de Fluxo de Dados (DFD)

Desenho indicando as entradas, saídas, depósitos de dados e os fluxos de informação, podendo haver desdobramento em níveis para melhor descrição dos processos.

15 Especificação de Algoritmos, Análise de Performance

Especificação detalhada de algoritmos, delineando passo a passo o processo que deve ocorrer.

16 Pseudocódigo

Refinamento das especificações de algoritmos em forma de pseudocódigo, identificando todas as variáveis usadas em cálculos, transformações, atualizações, entre outras informações.

17 Codificação

Conversão do pseudocódigo em código-fonte.

18 Walkthrough – Revisão em Grupo

Explicação verbal do código na presença de pares para identificar erros ou falhas antes do teste.

19 Compilação

Tradução do código-fonte para linguagem de máquina pelo compilador originando os programas prontos para serem executados.

20 Teste de Sistema, Unidade e Stress

Verificar se as funções estão de acordo com os objetivos previamente delineados. Existem vários níveis possíveis de teste de unidade destinados a examinar desde o funcionamento geral, passando pela integração com outros sistemas, interface até a adequação as necessidades.

21 Implementação

Entre as atividade de implementação encontram-se: Treinamento, carga inicial de dados e programas, redação da documentação, monitoramento das atividades, etc.

Na figura 12 podem ser observadas as etapas do desenvolvimento de sistemas de nível operacional, onde estão marcados com o símbolo  os itens: 2,7,8,10, 12,13,14,16,17 e 19, que são as etapas onde as ferramentas CASE podem auxiliar total ou parcialmente no processo.

O ponto de partida para que a ferramenta CASE gere as estruturas, a base de dados e os programas é o dicionário de dados que deve ser criado com a utilização de padrões nos nomes dos atributos e domínios de tipos e tamanho de variáveis para que as necessidades de integração como os sistemas existentes e reutilização descritas no etapa 2 (Usar código e dados existentes) sejam saciadas.

Como o dicionário de dados só será obtido na etapa 10 (Projeto físico do banco de dados), a ferramenta CASE deve possibilitar ou auxiliar no desenho dos diagramas de entidade e relacionamentos, fase 7 (DER), e conjunto de itens de dados, fase 8 (DIS).

As ferramentas CASE por definição devem auxiliar as fases de desenvolvimento de sistemas, facilitando, mas não eliminando as respectivas etapas necessárias ao ciclo de vida do software.

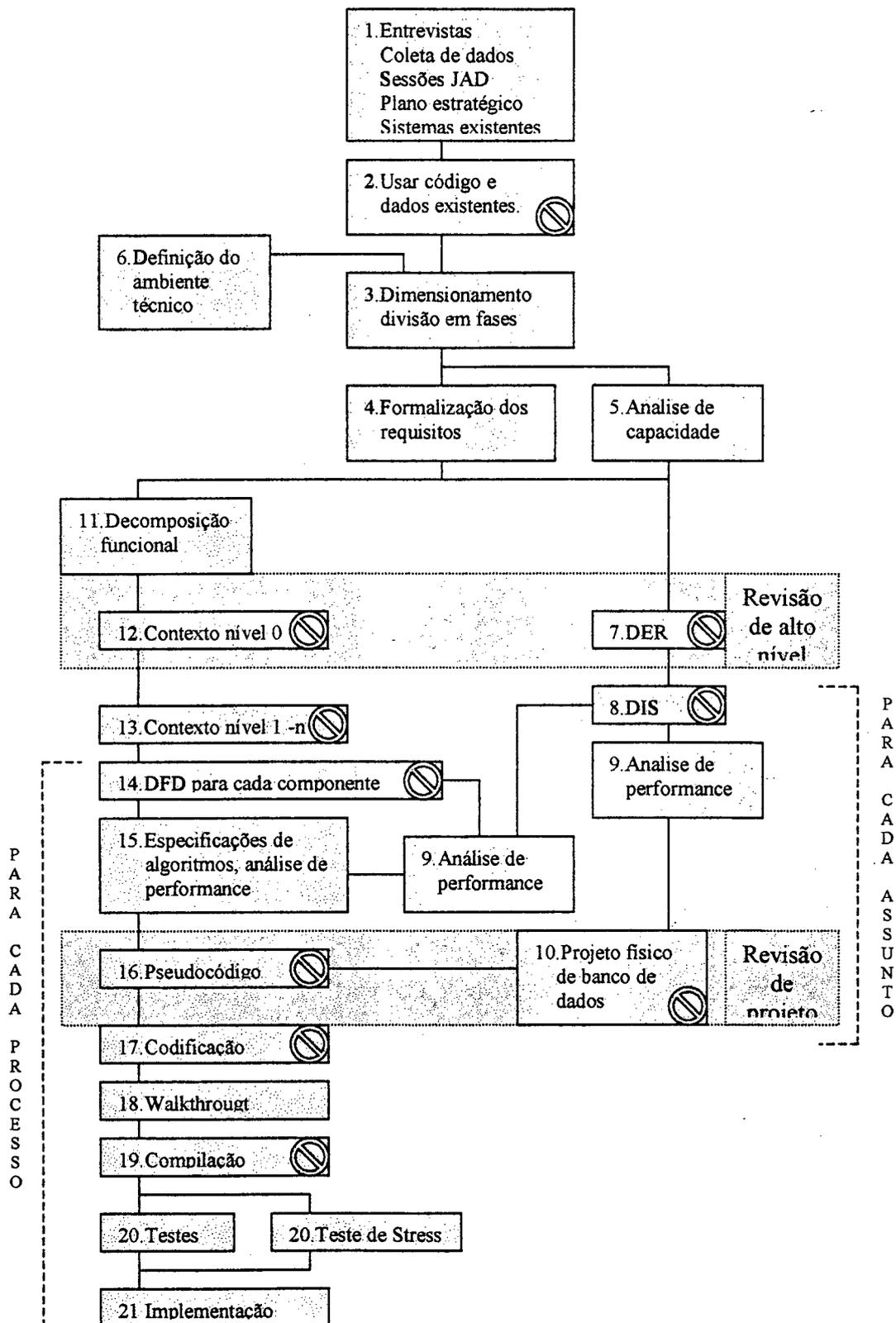


Figura 12. – Desenvolvimento de Sistemas de Nível Operacional.

4.3.2 Construção do *Data Warehouse*

Com a decisão de construir o ambiente de *data warehouse* deve ser implementado o modelo de dados já descrito no capítulo 4.2, fazendo uso das definições expostas nos capítulos 2.1.7 – Modelo de dados segundo R.Kimball, e capítulo 2.1.8 – Modelo de dados segundo W.H.Inmon.

Após a construção do modelo de dados às etapas para o desenvolvimento do DW são:

S1 Análise do Modelo de Dados

O modelo de dados traz um conjunto de informações de vital importância para o DW, e nesta fase deve haver a confirmação de que o modelo de dados atente aos critérios especificados. O modelo de dados deve contemplar os principais assuntos identificados, e cada assunto principal com sua própria definição de dados, separada e incluindo:

- Subtipos de dados;
- Atributos de dados;
- Definição clara dos relacionamentos entre dados;
- Definição dos agrupamentos de dados;
- Definição de chaves.

Cada agrupamento de dados apresentará ainda a descrição dos dados SAD e dos dados apenas operacionais.

S2 Dimensionamento

O dimensionamento é uma estimativa do ambiente SAD, ele projeta em termos brutos que quantidade de dados o DW vai conter, verificando assim a necessidade ou não de implementar vários níveis de granularidade.

S3 Avaliação Técnica

Diz respeito à satisfação pelo DW aos critérios:

- Capacidade de gerenciar grandes quantidades de dados;

- Capacidade de permitir que os dados sejam acessados de modo flexível;
- Capacidade de organizar dados de acordo com um modelo de dados;
- Capacidade tanto de receber quanto de enviar dados para uma ampla variedade de tecnologias;
- Capacidade de acessar dados um set por vez ou um registro por vez.

S4 Preparação do Ambiente Técnico

Após a definição da configuração arquitetônica do DW, se faz necessária à identificação técnica de como a configuração pode ser acomodada. Nesta etapa são verificados os pontos:

Rede; DASD; Sistema operacional que gerencia do DASD; Interface do e para o DW; Software usado para gerenciar o DW; O *data warehouse*.

S5 Análise das Áreas de Interesse

Nesta etapa deve ser selecionada a área de interesse para povoamento do DW, sendo que a primeira área a ser povoada deve ser grande o suficiente para ter sentido e pequena o suficiente para permitir sua implementação rápida.

S6 Projeto do *Data Warehouse*

Consiste no projeto físico de banco de dados para o DW, projeto este realizado com base no modelo de dados com algumas características que merecem consideração:

- Acomodação de vários níveis de granularidade;
- Orientação dos dados para os principais assuntos;
- Presença apenas de dados primitivos e dados derivados públicos;
- Ausência de dados não-SAD;
- Variação em relação ao tempo representada;

- Desnormalização física de dados onde houver exigência em relação à performance;
- Criação de artefatos de dados.

É comum preencher em um primeiro momento as principais estruturas do DW e posteriormente preencher os detalhes.

S7 Análise do Sistema-fonte

É a identificação do sistema de registro, ou seja, após definir qual o assunto a ser povoado é necessário verificar a existência e a validade das fontes de dados para os dados SAD, verificando questões como integração, chaves, unidades de medida, variação em relação ao tempo, estrutura, relacionamentos, etc. criando a interface do DW com os sistemas operacionais.

S8 Especificações

Consiste na formalização em especificações de programas da interface identificada no item 7.

S9 Programação

Atividade-padrão de programação como: Desenvolvimento de pseudocódigo; Codificação; Compilação; Walkthroughs; Testes.

S10 Povoamento

Trata questões além das cargas, como: Frequência de povoamento; Eliminação de dados povoados; Dados Obsoletos; Gerenciamento de granularidade; Renovação dos dados de amostra viva.

Na figura 13 estão representadas as fases de construção de ambientes de *data warehouse* assim como as fases que podem ser auxiliadas por ferramentas de apoio ao desenvolvimento, marcadas com o símbolo .

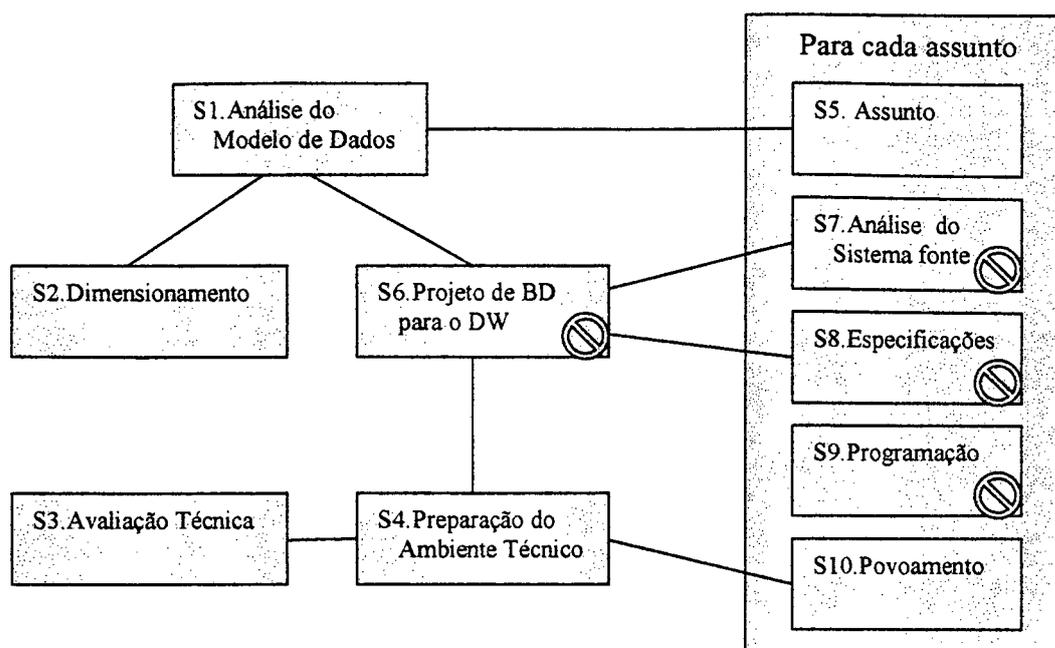


Figura 13 – Fases do Desenvolvimento do DW

4.3.3 Utilização Iterativa do *Data Warehouse*

Após a construção e povoamento do DW inicia-se sua utilização, onde o processamento é feito de modo muito iterativo e heurístico. As duas principais categorias de desenvolvimento de componentes de pesquisa no SAD são a “análise de ocorrência repetitiva”, também conhecida como análise departamental ou funcional, e o verdadeiro processamento heurístico, o nível individual.

Os passos necessários ao desenvolvimento de consultas heurísticas do SAD são:

A1 Delimitar Dados Necessários

Diante dos requisitos identificados para os relatórios alguns dados contidos no DW são selecionados para possível utilização, resultando na seleção de dados para análises posteriores.

A2 Programa para Extração de Dados

O passo seguinte é a construção dos programas para acessar e despojar os dados selecionados na etapa anterior. Estes programas devem ser suficientemente flexíveis, visto que o programa será executado, modificado e então executado novamente diversas vezes.

A3 Combinar, Intercalar, Analisar

Nesta etapa verifica-se a edição dos dados, combinação com outros e refinamento de dados, resultando em dados plenamente úteis para análise.

A4 Analisar Dados

Uma vez que os dados estejam prontos para análise se faz necessária a sua validação diante das necessidades especificadas pelo analista. Caso as expectativas não tenham sido satisfeitas deverá ocorrer outra iteração.

A5 Responder à Questão

Nesta fase se faz necessária à avaliação quanto à capacidade de responder as questões gerenciais e estratégicas da organização, onde o relatório é na maioria dos casos resultado de diversas iterações de processamento.

A6 Institucionalização

A necessidade de executar repetitivamente o relatório final deve ser avaliada nesta etapa, e uma vez comprovada, deve ser institucionalizada, ou seja, considerar o relatório como um conjunto de requisitos e recriá-lo como uma operação de ocorrência regular.

Na Figura 14 estão representados os passos necessários a construção de relatórios e ou consultas ao DW descritos nos passos A1 – A6.

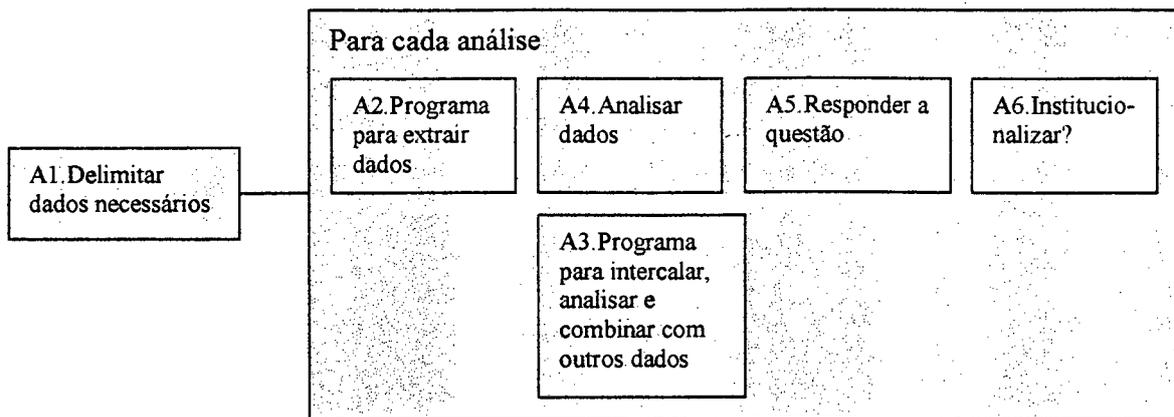


Figura 14 – Construção de Relatórios/Consultas ao DW

4.4 Checklist Para Revisão do Projeto

A construção de ambientes de *data warehouse* não segue o ciclo tradicional de desenvolvimento de sistemas conhecido como SDLC – *System Development Life Cycle*, principalmente por ser um processo iterativo, onde os requisitos são descobertos no próprio processo de desenvolvimento, contudo, como no desenvolvimento de sistemas de nível operacional, no projeto de DW também são inúmeros os benefícios proporcionados pela revisão do projeto, uma vez que podem ser encontrados diversos erros antes da codificação economizando recursos e aumentando a satisfação dos usuários.

O que o checklist deve ser baseado em questões como:

- Performance das transações;
- Adequação do espaço de carga / granularidade dos dados;
- Disponibilidade / performance do sistema;
- Capacidade;
- Grau de conclusão do projeto;
- Satisfação dos requisitos do usuário.

A revisão do projeto deve ser iniciada quando concluído o projeto de uma área de interesse, e deve envolver os administradores de dados e bancos de dados, programadores, analistas SAD, pessoal de operação, suporte, gerência e futuros usuários, sendo estes últimos juntamente com os analistas de SAD os mais importantes participantes.

O trabalho de revisão pode ser aplicado com maior ênfase a qualquer ponto que desperte algum tipo de receio ou preocupação tanto no desenvolvimento, gerenciamento ou mesmo na utilização.

Segundo (INMON, 1997) existem três resultados de uma revisão de projeto de *data warehouse* que são:

- Uma avaliação das questões dirigidas à gerência, bem como recomendações de futuras ações;
- Uma documentação da parte de projeto em que o sistema se encontra, no momento da revisão;
- Uma lista de itens de ação que estabelece atividades e objetivos específicos que resultam do processo de revisão.

A revisão normalmente é sustentada, entre tantas outras, por perguntas do tipo:

- Todas as pessoas envolvidas com o processo foram ouvidas?
- Alguma pessoa chave ficou de fora do processo?
- Os requisitos listados pelo usuário final foram atendidos?
- O DW atendeu todas as áreas de interesse?
- Os níveis de detalhe e de resumo estão adequados?
- A granularidade esta de acordo com as necessidade de informação?
- Existe mais alguma área de interesse que não foi contemplada? O que falta para sua inclusão?
- O volume de dados esta adequado?

- Os procedimentos de carga foram suficientemente testados?
- Podem existir gargalos?
- Qual será o suporte técnico necessário?
- Qual o tempo de instalação dos softwares?
- Como esta a interface?
- A estrutura de hardware esta preparada para suportar o volume de dados e consultas?
- O sistema de indexação do DW é suficiente?
- Os metadados estão de acordo com as expectativas?
- O sistema de alimentação de dados externos terá a alimentação e suporte necessário?
- A documentação esta pronta e adequada?

5 ESTUDO DE CASO

Neste capítulo é apresentado o estudo de caso realizado sobre a construção de parte de um ambiente de *data warehouse* apoiado por tecnologia CASE, seguindo o modelo proposto no capítulo anterior. O principal objetivo desta construção não é o produto final, mas sim a aplicação de uma metodologia alternativa e obtenção dos resultados para análise.

O presente estudo de caso tem como principais pontos envolvidos:

- Escolha de uma base de dados para o trabalho;
- Escolha das ferramentas CASE;
- Plano de Migração;
- Desenvolvimento dos Sistemas;
- Revisão do Projeto.

5.1 Escolha de Uma Base de Dados Para o Trabalho

É utilizada no presente estudo de caso a base de dados de uma empresa de pequeno porte, que atua na área de informática, com compra e venda de computadores, desenvolvimento de sistemas, prestação de serviços e consultoria.

Dentre as razões da escolha desta base de dados pode-se citar:

- Prontidão e abertura por parte da empresa de suas bases de dados e controles;
- Familiaridade com a área de atuação da empresa;
- Facilidade e acesso aos administradores para levantamento das necessidades do negócio;
- A necessidade de verificar a aplicabilidade da tecnologia de DW a empresas de pequeno porte;

5.1.1 Descrição da Base de Dados Operacional

Nos softwares de nível operacional da empresa encontram-se sistemas desenvolvidos basicamente nas linguagens de programação Visual Basic e Clipper, acessando bases de dados em Oracle, Access ou DBF.

Entre os módulos encontrados nos sistemas operacionais pode-se citar:

- Estoque;
- Contas a receber;
- Contas a pagar;
- Vendas / Faturamento;
- Financeiro;
- Contábil / Fiscal.

Algumas áreas da empresa ainda não têm controle informatizado de suas atividades e certas informações como cotação de outras moedas, oscilações do mercado e dados sobre a concorrência não se encontram registradas nos sistemas, tomando necessária à construção de um módulo operacional capaz de absorver informações especiais, de padronização ou informações não encontradas nos demais sistemas e que servirão para medir a evolução do negócio.

A documentação encontrada é bastante falha, fazendo-se necessária a análise geral da base de dados e sistemas para o levantamento de algumas informações sobre os dados, fluxos e processos.

5.2 Escolha da Ferramenta

A escolha de ferramentas para o suporte à criação de um ambiente de *data warehouse* é bastante penosa, visto que se faz necessária à avaliação de diversos pontos antes da aquisição e que normalmente as ferramentas disponíveis estão atreladas a uma solução específica atendendo parcialmente as necessidades do negócio, também é verificado que as empresas, consultorias e profissionais são bastante tendenciosos, destacando sempre as vantagens e defendendo a ferramenta que representam ou trabalham.

5.2.1 Critérios de Escolha

Dentre os critérios utilizados na escolha de uma ferramenta de suporte à criação do ambiente de *data warehouse* pode-se destacar:

- Recursos disponíveis;
- Preço;
- Facilidade de utilização;
- Facilidade de desenvolvimento;
- Disponibilidade de suporte;
- Opções de treinamento de pessoal;
- Consultoria e apoio ao desenvolvimento;
- Adequação aos bancos de dados;
- Plataformas de trabalho;
- Facilidades para visualização dos sistemas de nível operacional;
- Conhecimentos da equipe.

5.2.2 Ferramenta Escolhida

Dentre as soluções analisadas optou-se pelas ferramentas: Genexus Corporate, Gxplorer Metadata Manager e Gxplorer Olap Manager, desenvolvidas pela empresa uruguaia Artech.

Resumidamente as funções principais destas ferramentas são:

- Genexus Corporate: Ferramenta CASE utilizada para a criação da base de conhecimento, através da qual são geradas as bases de dados, os sistemas e as novas reorganizações. Permite a inserir conhecimento sobre as aplicações, modelar os indicadores e as dimensões do DW e auxilia na identificação do sistema de registro e adequação deste ao modelo de dados.
- Gxplorer Metadata Manager: Analisa a base de conhecimento e gera arquivo com análise de impacto e com a metadata dos indicadores e dimensões do DW possibilitando a análise das informações do armazém de dados;
- Gxplorer Olap Manager: Seleciona e analisa o arquivo de metadata e controla as permissões e acessos dos usuários as informações do DW permitindo a implementação do conceito de *data marts*.

O Gxplorer utiliza ainda uma planilha eletrônica nas consultas as informação do DW através do conceito de tabela pivô, onde são inseridas as dimensões e os indicadores de análise. Ao instalar as ferramentas do Gxplorer na estação de trabalho é acrescentada uma nova paleta de funções na planilha eletrônica, esta paleta de objetos permite a inserção e manipulação das informações do DW em uma planilha com recursos OLAP, permitindo as mais diversas consultas, efetuadas diretamente pelos administradores.

A ferramenta Gxplorer é, na verdade, dependente e complementar a ferramenta Genexus, uma vez que as dimensões, os indicadores e os

procedimentos de carga são todos descritos em uma base de conhecimento através do Genexus, ficando a cargo do Gxplorer a manipulação da base de conhecimento do DW e dos acessos/permisões. O usuário final vai, na verdade, utilizar uma tabela pivô, que pode ser, por exemplo, o Excel 2000. O usuário utiliza, na verdade, apenas a planilha eletrônica para a manipulação dos indicadores e dimensões do DW conforme suas permissões.

5.3 Plano de Migração

Primeiramente é necessário definir quais as áreas de negócio que serão atendidas, formando as principais divisões do DW. Como o objetivo deste estudo de caso é de testar a tecnologia, não se faz necessária a aplicação da tecnologia DW sobre todos os setores, sendo assim, foram selecionadas apenas as áreas de negócio Vendas e Financeiro.

Para cada uma das áreas de negócio deve-se construir o modelo de dados, nele estão relacionados os indicadores de medição do negócio e as suas dimensões de análise.

Neste trabalho foram selecionadas, para teste, as dimensões e os indicadores relacionados nas tabelas 6 e 7 para as áreas de negócios Vendas e Financeiro respectivamente.

Dimensões	Indicadores
Ano, Mês, Dia, Região, Categoria e Produto.	Qtde. Cotada Qtde. Vendida Valor das Vendas Qtde. de Devoluções Valor das Devoluções Comissões Pagas Gastos Gerais Venda Participação no Total de Vendas
Ano, Mês, Região, Categoria e Produto.	Resultado Líquido
Ano, Mês, Região, Categoria, Produto e Tributo.	Valor do Imposto Pago
Ano, Mês, Dia, Faixa Etária Cliente, Cliente, Categoria e Produto.	Qtde. Cotada Qtde. Vendida

Tabela 6 - Indicadores e Dimensões da Área de Negócios: Vendas.

Dimensões	Indicadores
Ano, Mês, Dia, Região e Loja.	Valor das Contas a Pagar Valor das Contas a Receber Valor em Caixa Valor em Bens Patrimoniais Posição no Mercado
Ano, Mês, Dia, Região, Loja, Categoria e Produto.	Valor dos Estoques Potencial de Vendas Custo de Venda

Tabela 7 - Indicadores e Dimensões da Área de Negócios: Finanças.

Uma vez definidos os indicadores e dimensões deve-se repassar este conhecimento para a ferramenta CASE. No caso do Genexus devem ser criados objetos do tipo "Transação DW", para cada uma das dimensões e quantos forem necessários para os indicadores.

Na figura 15 é apresentada a tela onde o conhecimento relativo à dimensão faixa etária é inserido. Para cada um dos atributos que formam esta dimensão é dado um apelido seguindo uma padronização, indicando nas três primeiras posições a tabela a que pertence, nas três posições posteriores é descrita a identificação do atributo e nas demais a qualificação do atributo. Ao salvar este objeto também são repassadas, à ferramenta, informações sobre o tipo e tamanho dos atributos.

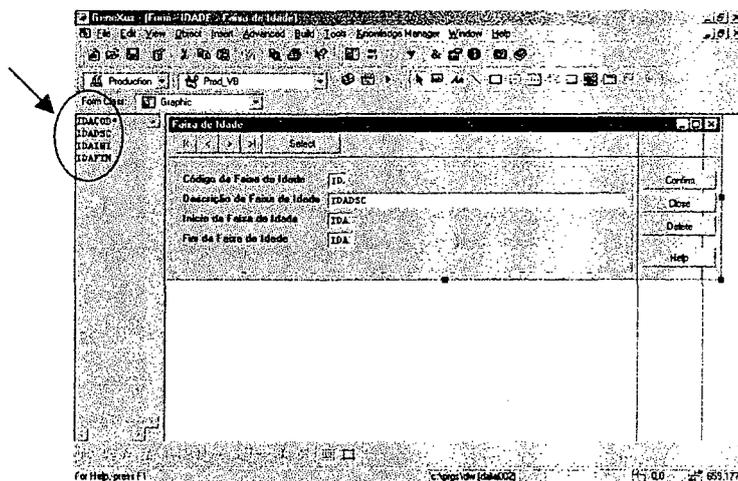


Figura 15 Criação do Objeto Transação DW para a Dimensão Faixa Etária.

Observando ainda a figura 15, após o atributo IDACOD, que representa o código da faixa etária, encontra-se o símbolo “ * ” indicando que este atributo é a chave da tabela de faixas etárias.

A construção de Transações DW para os indicadores segue o mesmo princípio, contudo os atributos que formam a chave da tabela onde serão armazenados os indicadores são os atributos que representam as dimensões de análise escolhidas, conforme pode ser observado na figura 16.

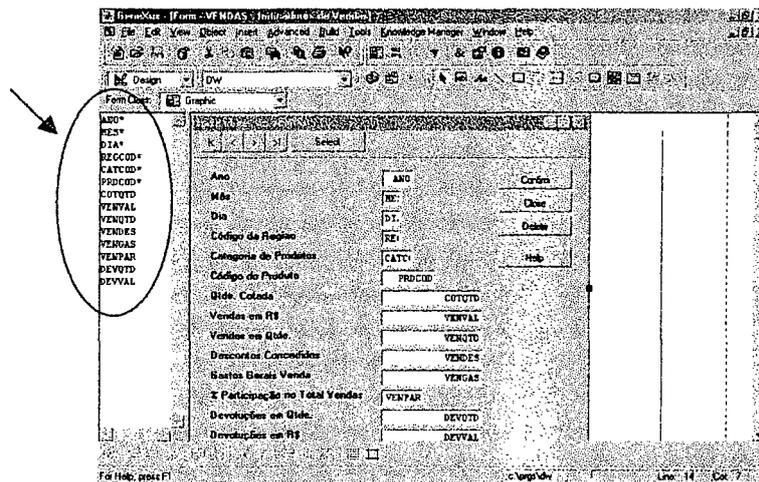


Figura 16 Criação do Objeto Transação DW para Indicadores Diversos.

Após a criação das Transações DW, onde são inseridas as informações que descrevem o modelo de dados, deve-se partir para a definição do sistema de registro. Esta atividade consiste em identificar quais as melhores fontes nos sistemas operacionais para alimentar o DW.

O trabalho de identificar a fonte de dados é facilitado quando o sistema que possui esta fonte tiver sido desenvolvido em uma ferramenta CASE, pois, neste caso, já existem dados padronizados (Metadados) descritos na base de conhecimento da ferramenta facilitando a localização das fontes necessárias.

Quando se utiliza CASE no desenvolvimento do DW, normalmente é criada uma nova base de conhecimento, separada das bases dos outros

sistemas, fazendo-se necessária à construção de visões para as fontes de dados. Estas visões irão permitir a manipulação dos dados para futura carga do DW. Na ferramenta Genexus esta atividade consiste em construir objetos denominados "Data Views", onde é identificado o banco de dados, o nome da tabela, dos índices, instância e usuário do BD, entre outras informações necessárias para chegar até aquela informação.

Foi utilizado neste trabalho o utilitário Data View Generator que gera conhecimento compatível com o Genexus, com as visões necessárias, a partir de informações sobre o banco de dados. Esta ferramenta acelera o processo de construção das visões de dados, pois cria automaticamente a estrutura da visão, a partir de informações sobre a fonte de dados.

Neste ponto o modelo de dados e o sistema de registro já estão descritos na ferramenta possibilitando a construção dos procedimentos de carga, que tem o objetivo de ler o sistema de registro onde estão identificadas as fontes de dados, efetuar os processos de sumarização, filtragem e adequação, para a gravação dos registros resultantes no DW. Esta é a etapa de adequação do modelo de dados ao sistema de registro.

Nesta operação é utilizado um objeto do tipo "procedimento" na ferramenta CASE, onde são inseridas as linhas de programação necessárias a transformação dos dados operacionais em dados do DW.

Nos procedimentos são inseridos os conhecimentos sobre onde serão realizadas as leituras no sistema de registro, através da referencia aos atributos dos sistemas operacionais ou atributos relacionados nos objetos de visão dos dados (Data Views), bem como as transformações necessárias para alimentar os indicadores e as dimensões definidas no modelo de dados.

Com o conhecimento repassado a ferramenta relativa ao plano de migração, composto pelo modelo de dados, sistema de registro e adequação do primeiro ao segundo, é possível gerar toda a estrutura do banco de dados do DW e os programas de carga automaticamente.

5.4 Desenvolvimento dos Sistemas

Encontra-se na fase do desenvolvimento dos sistemas três etapas distintas:

1º - Construção dos sistemas de nível operacional, que são os aplicativos tradicionais que controlam as atividades cotidianas da empresa com processamento OLTP;

2º - Construção do *data warehouse*, armazém de dados para o suporte a tomada de decisões;

3º - Construção dos sistemas de consulta, que fazem uso do ambiente DW para as consultas gerenciais, processamento OLAP.

Já é comprovado que as ferramentas CASE podem diminuir a menos da metade o esforço de desenvolvimento e manutenção dos sistemas de nível operacional, possibilitando uma maior concentração na análise e planejamento dos aplicativos. Estima-se que este ganho deve se manter também no desenvolvimento de ambientes DW e sistemas de consulta.

Na ferramenta Genexus, utilizada neste trabalho, o processo de construção de sistemas é iniciado com a criação de uma base de conhecimento, onde é possível inserir alguns objetos necessários ao desenho da aplicação pretendida. É possível trabalhar com uma base de conhecimento centralizada para as diversas aplicações, facilitando a integração, mas não há problemas em dividir as bases de conhecimento por aplicações distintas.

Existem três ambientes de trabalho no Genexus: Design, Protótipo e Produção. Os objetos da ferramenta são: Transações; Áreas de Trabalho; Procedimentos; Relatórios; Visões de Dados; Áreas Web; Menus. No design é possível trabalhar os requisitos nos objetos do tipo transação e modelar a aplicação com os demais objetos, mas não é possível executar a aplicação. Ao contrário, no protótipo e na produção os requisitos não podem ser alterados, mas é possível executar a aplicação. Existe apenas um ambiente desing por base de conhecimento e podem existir vários ambientes dos tipos protótipo ou

produção, destinados a geração dos programas em linguagens e bancos de dados diferentes conforme a necessidade e geradores instalados.

Na construção de sistemas os requisitos identificados para a aplicação são inseridos em objetos transação no ambiente design, e uma vez inseridos e documentados, os atributos poderão ser trabalhados nos demais objetos, bastando mencionar seu nome para a ferramenta encontrar sua referência com o nome da tabela a que pertence e ainda informações sobre integridade referencial e relacional.

Nas transações, uma vez que os requisitos sejam inseridos, a ferramenta gera uma tela com os atributos e as opções de consulta, inserção, alteração, deleção, navegação, etc., necessárias à manipulação direta dos dados. As manipulações mais sofisticadas dos atributos são realizadas através da modelagem de áreas de trabalho associadas a procedimentos. A criação de relatórios também é baseada nos atributos, que após serem referenciados permitem a identificação da tabela a que pertencem e das informações necessárias a sua integração com o restante da aplicação.

Depois de modelar a aplicação no ambiente design, isto é, criar os objetos transações, áreas de trabalho, relatórios e procedimentos, é possível criar o ambiente de protótipo ou de produção inserindo informações sobre a linguagem de programação e banco de dados utilizado. Na passagem do ambiente design para o ambiente produção é gerado um relatório chamado "análise de impacto", onde são descritas todas as mudanças necessárias na base de dados para suportar a aplicação e é criado ainda um programa chamado "rmenu.exe" capaz de fazer estas alterações automaticamente na base de dados.

No protótipo ou na produção, a partir da solicitação do analista, os objetos devem ser especificados, este processo consiste em gerar automaticamente o programa fonte das aplicações, com base nos objetos modelados anteriormente. Este processo de especificação também gera um relatório descrevendo os atributos, tabelas, relacionamentos e outras informações sobre o que o programa irá fazer e em quais tabelas.

Depois de especificar os objetos é possível compilar a aplicação, gerando um arquivo executável que pode ser disponibilizado ao usuário final. Assim está concluído a criação de um sistema de nível operacional.

Para a manutenção com mudança nos requisitos da aplicação o processo consiste em:

- Voltar ao ambiente de design;
- Fazer as alterações necessárias nas transações;
- Mudar para o ambiente de protótipo ou produção, gerando a análise de impacto e o programa de reorganização da base de dados;
- Alterar os demais objetos necessários à aplicação;
- Especificação dos objetos alterados;
- Compilação da nova versão da aplicação.

Para a construção dos sistemas necessários ao *data warehouse* utilizando a ferramenta Genexus o processo é parecido ao desenvolvimento dos sistemas de nível operacional, contudo são utilizados basicamente os objetos “transação DW” para a descrição dos indicadores e dimensões, “Data Views” para descrever o sistema de registro e “procedimentos” para a transformação dos dados operacionais em registros do DW. Estes objetos também são criados no ambiente design e especificados no ambiente protótipo ou produção, com a geração das estruturas do banco de dados, reorganizações e programa executável com os procedimentos de carga.

Tendo em vista que o modelo de dados, o sistema de registro e a adequação destes através dos procedimentos de carga, já estavam descritos na ferramenta, por ocasião da fase “Plano de Migração”, descrita no capítulo anterior, bastou repassar as informações do banco de dados e linguagem utilizada, gerar a análise de impacto, especificar os objetos para preparar a base de dados e criar os programas necessários.

Na fase de construção dos sistemas de consulta foram utilizadas as ferramentas: Gxplorer Metadata Manager, Gxplorer Olap Manager e Microsoft Excel 2000.

Na ferramenta Gxplorer Metadata Manager é criado um arquivo de metadados com base na análise ao modelo de produção gerado pela ferramenta Genexus. Após especificar onde esta gravado o modelo de produção é gerado um relatório de análise de impacto descrevendo os indicadores, as dimensões, as fórmulas e as consultas relativas ao *data warehouse*. Também pode ser configurados para futura consulta os indicadores, dimensões e fórmulas geradas a partir destes.

No Gxplorer Olap Manager são definidos os critérios de segurança como usuários e direitos de acesso, são visualizados os indicadores e dimensões válidos e definidas as informações relativas ao acesso a base de dados.

O usuário final vai utilizar uma planilha eletrônica para a consulta as informações do DW, no caso do Excel 2000, utilizado no presente estudo de caso é inserida uma nova paleta de objetos, permitindo a criação e manipulação de consultas ao DW baseado no arquivo de metadados gerado pelo Gxplorer Metadata Manager, sobre os acessos descritos no Gxplorer Olap Manager. Estas consultas possuem as características definidas no conceito OLAP, permitindo as mais diversas manipulações nas informações.

É possível criar uma série de consultas padrão por área de interesse e disponibilizar para consulta dos gerentes, que terão a opção de interagir com as informações manipulando-as como lhe for conveniente em uma planilha eletrônica, sistema que já é familiar em suas atividades.

5.5 Revisão do Projeto

O processo de revisão do projeto consiste em verificar a consistência e adequação dos sistemas desenvolvidos a realidade tanto das necessidades dos usuários como das restrições impostas por hardware e software.

A revisão do projeto é apoiada pela aplicação de uma série de perguntas, conforme descrito no capítulo 4.4. Também é necessária uma

análise crítica sobre o potencial estratégico e os horizontes proporcionados pela tecnologia de DW.

A revisão do projeto também pode ser executada durante a utilização do DW, a medida que algum ponto, seja de performance, seja de esgotamento de capacidade de armazenamento ou outro item que possa interferir na utilização da tecnologia formando assim um ciclo de monitoramento que pode ser executado periodicamente.

É aconselhada pelo menos uma revisão anual para levantar possíveis problemas e também verificar se existem necessidades novas ou não atendidas anteriormente e a possibilidade de implementação.

De uma maneira geral pode-se afirmar que o projeto desenvolvido no presente estudo de caso, devido a sua simplicidade, atendeu bem os requisitos e esta atendendo aos seus objetivos.

Nos anexos I, II, III, IV e V estão as telas dos aplicativos Data View Generator, Gxplorer Metadata Manager, Gxplorer OLAP Manager, Genexus e a tabela pivô Excel 2000.

6 CONCLUSÕES E SUGESTÕES PARA NOVOS TRABALHOS

Dentre as conclusões alcançadas com o presente trabalho pode-se citar:

Existem hoje boas opções de ferramentas destinadas a criação de ambientes de *data warehouse*, entre elas pode-se citar Genexus Explorer por seus recursos e adequação ao desenvolvimento de sistemas de apoio a decisão;

Um dos principais problemas encontrados atualmente na construção de ambientes de *data warehouse* é a falta de mão de obra capacitada, ou seja, de analistas de negócio, capazes de levantar os indicadores e suas dimensões para a criação de um modelo de medição de negócios destinado a auxiliar na tomada de decisões.

É aconselhada a separação dos atributos de medição de negócios dos atributos operacionais no momento da construção dos sistemas de nível operacional, em outras palavras, mesmo que não se construa um *data warehouse*, é importante dividir as informações segundo sua aplicação, isto poderá auxiliar muito em futuras implementações.

Pode-se afirmar ainda que a tecnologia de *data warehouse*, hoje, também pode ser aplicada a empresas de pequeno e médio porte auxiliando na condução do negócio e respondendo uma série de perguntas estratégico-administrativas, visto que os custos das ferramentas, dos bancos de dados, do desenvolvimento e da implantação descenderam a níveis aceitáveis e aplicáveis.

Observando as fases de desenvolvimento de sistemas desde a construção dos programas de nível operacional, passando pela implementação do DW, até a consulta as informações do *data warehouse* pode-se afirmar que as ferramentas CASE hoje disponíveis, são capazes de auxiliar em diversas tarefas reduzindo consideravelmente o esforço de desenvolvimento, passando conseqüentemente ao analista de negócios um trabalho mais voltado ao levantamento das necessidades de informação estratégica e gerencial da organização. Estas ferramentas também podem forçar a execução do projeto de DW de maneira lógica, cumprindo os passos necessários ao desenvolvimento de um SAD de comprovada qualidade

Como melhoria e sugestões para trabalhos futuros pode-se citar:

Construção de um modelo de *data warehouse* completo, para empresas de pequeno e médio porte, a fim de estimular a discussão sobre esta tecnologia e disponibilizar potencial competitivo para a comunidade empresarial.

Construir protótipo de *data warehouse* com diferentes ferramentas destinadas a este propósito, disponíveis no mercado, para comparação de pontos positivos e negativos.

Estudo de caso sobre os benefícios e prejuízos da aplicação desta tecnologia em diferentes organizações, acompanhando a evolução e medindo seu real valor na tomada de decisões.

7 BIBLIOGRAFIA

ARTHUR, Lowell Jay. Melhorando a Qualidade do Software. Rio de Janeiro, Infobook, 1994.

BARQUINI, Ramon, Planning and Designing the Warehouse, New Jersey, Prentice-Hall, 1996, 311 pg.

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.

BIGUS, Joseph P., Data Mining With Neural Networks, New York-US: Editora McGraw-Hill, 1996.

BISPO, Carlos A.F., Uma Análise da Nova Geração de Sistemas de Apoio à Decisão, São Carlos-SP, 1998.

CHEROBIN, Mauriciu A., Data Warehouse: Uma Ferramenta Para o Processo Decisório e Sistema de Informação, UNOESC, Videira-SC, 2000.

CÔRTEZ, Sérgio, Apostila MBA em e-Business Data-Warehousing / Data-Mining, FGV, Rio de Janeiro, 2001.

Revista Computerworld, Revista de computação, disponível em <http://www.computerworld.com.br> - Edição 221 - Agosto 1997, data do último acesso 10/07/2001.

FISCHER, Alan S., CASE: Utilização de Ferramentas para Desenvolvimento de Software, Rio de Janeiro, Campus, 1991.

FURLAN, José Davi, Reengenharia da Informação – Do Mito à Realidade, São Paulo-SP, Editora Makron Books, 1994.

GANE, Chris, CASE: O Relatório Gane, Rio de Janeiro, LTC - Livros Técnicos e Científicos Editora S.A, 1990.

- Gates, Bill, A empresa na velocidade do pensamento. Editora Schwarcz, 1999.
- HACKATHORN, Richard: Data Warehousing Energizes Your Enterprise. Datamation, New York, 1997.
- HARJINDER, G., RAO, P.C., The Official Guide to Data Warehousing, Que Corporation, 1996.
- INMON, William H., Building the Data Warehouse, John Wiley & Sons Inc., New York, 1996.
- INMON, William H., Como Construir o Data Warehouse, 2ª Edição, Rio de Janeiro: Editora Campus, 1997.
- INMON, William H. & RICHARD D. Hackathorn – Como Usar o Data Warehouse, Infobook, Rio de Janeiro, 1997.
- INMON, William H., WEICH, J.D., GLASSEY, Katherine L., Gerenciando Data Warehouse, São Paulo: Editora Makron Books, 1999.
- KHOSHAFIAN, Setrag. Banco de Dados Orientado a Objetos. São Paulo: IBPI Press, 1994
- KIMBALL, Ralph, Is E-R Modeling Hazardous to DSS, DBMS, 1995.
- KIMBALL, Ralph, The Data Warehouse Toolkit, John Wiley & Sons Inc., New York, 1996.
- KIMBALL, Ralph, Dimensional Modeling Manifesto, DBMS, 1997.
- KORTH, F.Henry, SILBERSCHATZ, Abraham. Sistemas de Bancos de Dados. São Paulo: McGraw-Hill, 1989.
- KORTH, F.Henry, Sistemas de Banco de Dados. São Paulo, Makron Books, 1993.
- MAFFEO, Bruno, Engenharia de Software e Especificação de Sistemas, Rio de Janeiro, Editora Campus, 1998.
- MARTIN, James, McCLURE, Carma, Técnicas Estruturadas e Case, São Paulo, Makron, McGraw-Hill, 1991.
- MARTIN, James, ODELL, J. James, Análise e Projeto Orientados a Objetos, São Paulo, Makron Books, 1995.
- McGUFF, Frank, Designing the Perfect Data Warehouse – Hitchhiker's Guide to Decision Support, disponível em <http://members.aol.com/fmcguff/dwmmodel/> data do último acesso 10/07/2001.
- MEYER, Bertrand, Object-Oriented Software Construction, C.A.R, Hoare, 1988.

- ONEIL, B., Oracle Data Warehousing, Indianapolis, Sams Publishing, 1997.
- ÖZSU, M. Tamer, VALDURIEZ, Patric, Principles of Distributed Database Systems, Editora Prentice-Hall do Brasil Ltda., 2ª Edição, Rio de Janeiro, 1999.
- PRESSMAN, Roger S., Engenharia de Software, São Paulo, Makron Books, 1995.
- REZENDE, Denis A., Engenharia de Software e Sistemas de Informação, Rio de Janeiro, Brasport, 1999.
- RUMBAUGH, James. BLAHA, Michael, Modelagem e Projetos Baseados em Objetos, Rio de Janeiro, Editora Campus, 1994.
- VALENTE, Daphnis Lopes, Estudo sobre Armazém de Dados, Porto Alegre, CPGCC da UFRGS, 1996, 56 pg.
- VALERIANO, Dalton L., Gerência em Projetos – Pesquisa, Desenvolvimento e Engenharia, São Paulo, Makron Books, 1998.
- WELDON, J.L., Warehouse Cornerstones, Revista Byte, v.22, nº 1, Janeiro 1997.
- WILLIAMS, J., Tools for Traveling Data. DBMS, Califórnia, 1997.
- WEBER, Kival C., ROCHA, Ana R.C. da, NASCIMENTO, Célia J. do, Qualidade e Produtividade em Software, 4ª Edição, Mackron Books, São Paulo, 2001.
- WURMAN, Richard S., Ansiedade de Informação, Como Transformar Informação em Compreensão, Cultura Editores, 1991.